HITACHI
Inspire the Next

スケーラブルデータベースサーバ

HiRDB Version 8 システム定義 (Windows(R)用)

文法書

3020-6-353-44

■ 対象製品

●適用 OS：Windows 2000，Windows Server 2003，Windows Server 2008，Windows XP，Windows Vista，Windows 7

P-2462-7184 HiRDB/Single Server Version 8 08-05，08-51※

P-2462-7384 HiRDB/Parallel Server Version 8 08-05，08-51※

P-2462-7H84 HiRDB Non Recover Front End Server Version 8 08-00

P-2462-7J84 HiRDB Advanced High Availability Version 8 08-00

P-2462-7K84 HiRDB Advanced Partitioning Option Version 8 08-00

●適用 OS：Windows 2000，Windows Server 2003

P-2462-7G84 HiRDB LDAP Option Version 8 08-02

●適用 OS：Windows Server 2003 x64 Editions，Windows Server 2008 R2，Windows Server 2008 (x64)，Windows XP x64 Edition，Windows Vista Ultimate (x64)，Windows Vista Business (x64)，Windows Vista Enterprise (x64)，Windows 7 Professional (x64)，Windows 7 Enterprise (x64)，Windows 7 Ultimate (x64)

P-2962-7184 HiRDB/Single Server Version 8(64) 08-05，08-51※

P-2962-7384 HiRDB/Parallel Server Version 8(64) 08-05，08-51※

P-2462-7P84 HiRDB Accelerator Version 8 08-03

●適用 OS：Windows XP x64 Edition，Windows Server 2003 x64 Editions，Windows Vista (x64)，Windows Server 2008 (x64)，Windows 7 (x64)

P-2962-1184 HiRDB/Run Time Version 8(64) 08-05，08-51※

P-2962-1284 HiRDB/Developer's Kit Version 8(64) 08-05，08-51※

●適用 OS：Windows 2000，Windows XP，Windows Server 2003，Windows Vista，Windows Server 2008，Windows 7

P-2662-1184 HiRDB/Run Time Version 8 08-05，08-51※

P-2662-1284 HiRDB/Developer's Kit Version 8 08-05，08-51※

注※　08-51 は，08-05 の修正版のバージョン・リビジョン番号です。

これらのプログラムプロダクトのほかにもこのマニュアルをご利用になれる場合があります。詳細は「リリースノート」でご確認ください。

■ 輸出時の注意

本製品を輸出される場合には、外国為替及び外国貿易法の規制並びに米国輸出管理規則など外国の輸出関連法規をご確認の上、必要な手続きをお取りください。

なお、不明な場合は、弊社担当営業にお問い合わせください。

■ 商標類

HITACHI，HiRDB，Cosminexus，DABroker，DBPARTNER，DocumentBroker，Groupmax，HA モニタ，HITSENSER，JP1，OpenTP1，OSAS，TPBroker，uCosminexus，VOS3/LS，XDM は，株式会社 日立製作所の商標または登録商標です。

ActiveX は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

AMD は，Advanced Micro Devices, Inc.の商標です。

IBM，AIX は，世界の多くの国で登録された International Business Machines Corporation の商標です。

IBM，DataStage, MetaBroker, MetaStage および QualityStage は，世界の多くの国で登録された International Business Machines Corporation の商標です。

IBM，DB2 は，世界の多くの国で登録された International Business Machines Corporation の商標です。

IBM，HACMP/6000 は，世界の多くの国で登録された International Business Machines Corporation の商標です。

IBM，OS/390 は，世界の多くの国で登録された International Business Machines Corporation の商標です。

Itanium は，アメリカ合衆国および / またはその他の国における Intel Corporation の商標です。
JBuilder は，Embarcadero Technologies, Inc.の米国およびその他の国における商標です。
Linux は，Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。
Microsoft および Visual Studio は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
Microsoft Access は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
Microsoft Office および Excel は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
Motif は，Open Software Foundation,Inc.の商標です。
MS-DOS は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
ODBC は，米国 Microsoft Corporation が提唱するデータベースアクセス機構です。
OLE は，米国 Microsoft Corporation が開発したソフトウェア名称です。
Oracle と Java は，Oracle Corporation 及びその子会社，関連会社の米国及びその他の国における登録商標です。
PowerBuilder は，Sybase,Inc.の登録商標です。
Red Hat は，米国およびその他の国で Red Hat, Inc. の登録商標もしくは商標です。
UNIX は，The Open Group の米国ならびに他の国における登録商標です。
Veritas 、 Veritas ロゴ は、Veritas Technologies LLC または関連会社の米国およびその他の国における商標または登録商標です。
Visual Basic は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
Visual C++は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
Windows は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
Windows NT は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
Windows Server は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
Windows Vista は，米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
その他記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

■ 発行

2017 年 4 月　3020-6-353-44

■ 著作権

All Rights Reserved. Copyright (C) 2006, 2017, Hitachi, Ltd.

変更内容

変更内容(3020-6-353-44)　HiRDB Version 8 08-05，08-51

追加・変更内容	関連するオペランド	変更個所
次のオペランドを追加しました。 • pd_rpc_bind_loopback_address	－	2.1，2.2.5

単なる誤字・脱字などはお断りなく訂正しました。

変更内容(3020-6-353-43)　HiRDB Version 8 08-05，08-51

追加・変更内容
リリースノートのマニュアル訂正を反映しました。

変更内容(3020-6-353-42)　HiRDB Version 8 08-05，08-51

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>[追加]
・HiRDB システム定義に指定している putenv 形式のオペランドをデフォルト値に変更する場合，オペランドを削除せずに明示的にデフォルト値を設定してください。</td><td>－</td></tr>
<tr><td>[追加]
<table>
<tr><th rowspan="2">オペランド名</th><th colspan="7">定義名</th><th rowspan="2">強制終了，異常終了後の変更可否</th><th rowspan="2">計画停止後の変更可否</th></tr>
<tr><th>SYS</th><th>UNT</th><th>SVR</th><th>SDS</th><th>FES</th><th>DS</th><th>BES</th></tr>
<tr><td>pd_pth_trace_max</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td></tr>
<tr><td>pd_rdarea_expand_format</td><td>○</td><td></td><td></td><td></td><td></td><td></td><td></td><td>○</td><td>○</td></tr>
</table></td><td>－</td></tr>
<tr><td>[訂正前]
<table>
<tr><th>オペランド名</th><th>＜省略＞</th><th>強制終了，異常終了後の変更可否</th><th>計画停止後の変更可否</th></tr>
<tr><td>SHMMAX</td><td>＜省略＞</td><td>○</td><td>○</td></tr>
</table>
[訂正後]
<table>
<tr><th>オペランド名</th><th>＜省略＞</th><th>強制終了，異常終了後の変更可否</th><th>計画停止後の変更可否</th></tr>
<tr><td>SHMMAX</td><td>＜省略＞</td><td>○※7</td><td>○※7</td></tr>
</table>
注※7
システム共通定義（pdsys）に pd_dbbuff_modify=Y を指定している場合，×（SHMMAX 指定値の変更はできない）となります。</td><td>－</td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>[訂正前]
<table>
<tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr>
<tr><td>55</td><td>〔set pd_watch_time = SQL最大実行時間〕</td><td>システム監視</td></tr>
<tr><td>197</td><td>〔set pd_ipc_inet_bufsize = サーバのユニット間通信で使用する送受信バッファ長〕</td><td>通信処理</td></tr>
<tr><td>198</td><td>〔set pd_tcp_inet_bufsize = HiRDBサーバが存在するホスト外のHiRDBクライアントとの通信で使用する送受信バッファ長〕</td><td>通信処理</td></tr>
</table>
[訂正後]
<table>
<tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr>
<tr><td>55</td><td>〔set pd_watch_time = 最大応答待ち時間〕</td><td>システム監視</td></tr>
<tr><td>200</td><td>〔set pd_ipc_inet_bufsize = サーバ間通信で使用する送受信バッファ長〕</td><td>通信処理</td></tr>
<tr><td>201</td><td>〔set pd_tcp_inet_bufsize = HiRDBクライアントとの通信で使用する送受信バッファ長〕</td><td>通信処理</td></tr>
</table></td><td>－</td></tr>
<tr><td>[追加]
<table>
<tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr>
<tr><td>110</td><td>〔set pd_pth_trace_max = 通信トレース格納最大数〕</td><td>トラブルシュート情報</td></tr>
<tr><td>121</td><td>〔set pd_rdarea_expand_format = Y | N〕</td><td>RDエリア</td></tr>
<tr><td>127</td><td>〔set pd_db_hold_action = dbhold | unitdown〕</td><td>RDエリア</td></tr>
</table></td><td>－</td></tr>
<tr><td>《指定値の目安》
[訂正前]
省略値＝a＋b×（c＋27）＋73＋i＋j＋k＋n
[訂正後]
省略値＝a＋b×（c＋30）＋70＋i＋j＋k＋n</td><td>pd_max_server_process</td></tr>
<tr><td>《指定値の目安》
[訂正前]
推奨値の目安＝a＋b×（c＋d×e＋f）＋d×g＋h＋i＋j＋k＋n
[訂正後]
推奨値の目安＝a＋b×（c＋d×e＋f＋3）＋d×g＋h＋i＋j＋k＋n</td><td>pd_max_server_process</td></tr>
<tr><td>《指定値の目安》
[訂正前]
<table>
<tr><th>変数</th><th>変数の説明</th></tr>
<tr><td>a</td><td>＜省略＞
・影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_server_processオペランドの値を代入します。</td></tr>
<tr><td>i</td><td>＜省略＞
・影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_serversオペランドの値を代入します。</td></tr>
</table>
[訂正後]</td><td>pd_max_server_process</td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>
<table>
<tr><th>変数</th><th>変数の説明</th></tr>
<tr><td>a</td><td>＜省略＞
・マルチフロントエンドサーバの場合，フロントエンドサーバごとに1を加算してください。
・影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_server_processオペランドの値を代入します。</td></tr>
<tr><td>i</td><td>＜省略＞
・影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_serversオペランドの値を加算します。</td></tr>
</table>
</td><td></td></tr>
<tr><td>
[訂正前]
35)　pd_rpc_bind_loopback_address = Y｜N
受信用ポートの生成時，ループバックアドレスで bind()するかどうかを指定します。
Y：ループバックアドレスで bind()します。
N：ループバックアドレスで bind()しません。
《前提条件》
このオペランドに Y を指定する場合は，次に示す条件をすべて満たしている必要があります。
• HiRDB/シングルサーバだけで HiRDB システムが構成されている※か，又はモニタモードで IP アドレスを引き継ぐ系切り替え機能を使用している
• pdunit オペランドの-x オプションとクライアント環境定義の PDHOST オペランドにループバックアドレスを指定している
注※
HiRDB/シングルサーバだけで HiRDB システムが構成されているとは，次に示す条件を満たすことをいいます。
• HiRDB クライアントと HiRDB サーバが同一マシンにある（HiRDB クライアントが別マシンにない）
《利点》
前記の前提条件を満たすシステム環境で，このオペランドに Y を指定すると，HiRDB サーバのプログラムを Windows ファイアウォールの例外リストに登録する必要がなくなります。
Windows ファイアウォールの例外リストの登録については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
[訂正後]
35)　pd_rpc_bind_loopback_address = Y｜N
受信用ポートの生成時，ループバックアドレスで bind()するかどうかを指定します。
Y：ループバックアドレスで bind()します。
N：ループバックアドレスで bind()しません。
《前提条件》
このオペランドに Y を指定する場合は，次に示す条件をすべて満たしている必要があります。
• HiRDB/シングルサーバで HiRDB システムが構成されている系切り替え機能を適用している場合，モニタモードで IP アドレスを引き継ぐ構成である
• HiRDB クライアントと HiRDB サーバが同一マシンにある（HiRDB クライアントが別マシンにない）
• pdunit オペランドの-x オプションとクライアント環境定義の PDHOST オペランドにループバックアドレスを指定している
</td><td>pd_rpc_bind_loopback_address</td></tr>
</table>

追加・変更内容	関連するオペランド
《利点》 前記の前提条件を満たすシステム環境で，このオペランドにYを指定すると，HiRDBサーバのプログラムをWindowsファイアウォールの例外リストに登録する必要がなくなります。 Windowsファイアウォールの例外リストの登録については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。	
［訂正前］ **50）　pd_max_list_users = 同時リスト所有可能ユーザ数** **～<符号なし整数>((0～32767))《0》** 同時にリストを所有できるユーザの最大数を指定します。 ［訂正後］ **50）　pd_max_list_users = リスト操作可能ユーザ数** **～<符号なし整数>((0～32767))《0》** リストを操作するすべてのユーザ数を指定します。	pd_max_list_users
［訂正前］ **55）　pd_watch_time = SQL最大実行時間** **～<符号なし整数>((0～65535))《0》（単位：秒）** このオペランドはHiRDB/パラレルサーバ限定のオペランドです。 HiRDBサーバプロセス上で実行されるSQLの最大実行時間を指定します。 このオペランドで指定した時間を過ぎてもSQLの実行が完了しない場合は，そのSQLの実行を中断することがあります。 《利点》 HiRDBクライアントがSQL実行を中断（クライアントプロセスの強制終了など）しても，HiRDBサーバがこのSQLの実行を中止しない場合，HiRDBサーバはこのSQLの実行を続けて長時間にわたって排他資源などを占有することがあります。このオペランドを指定すると，この占有時間を短くできます。 《指定値の目安》 次のうちの最も長い時間を指定してください。 • クライアント環境定義のPDCWAITTIMEオペランドで指定する時間 • クライアント環境定義のPDLCKWAITTIME オペランドで指定する時間 • pd_lck_wait_timeout オペランドで指定する時間 • 実行時間が最も掛かるSQLの処理時間 《注意事項》 • このオペランドに0を指定した場合は，SQL実行時間の監視をしません。 • クライアント環境定義のPDCWAITTIMEオペランドに0（時間監視しない）を指定したSQLについては，SQL実行時間の監視をしません。 • このオペランドにSQL実行時間より短い値を指定した場合，SQL実行中に処理が中断され，HiRDBクライアントにSQLエラーか，又はHiRDBサーバの異常終了が報告されることがあります。 • HiRDB/シングルサーバの場合にこのオペランドを指定しても，SQL最大実行時間を監視しません。ただし，このオペランドを指定した場合，その指定値がpd_lck_wait_timeout オペランドの省略値として使用されます。このため，HiRDB/シングルサーバの場合はこのオペランドを省略することをお勧めします。 ［訂正後］ **55）　pd_watch_time = 最大応答待ち時間**	pd_watch_time

追加・変更内容	関連するオペランド
～<符号なし整数>((0～65535))《0》(単位：秒) このオペランドは旧バージョンとの互換性保証のための指定です。したがって指定する必要はありません。 このオペランドは HiRDB/パラレルサーバ限定のオペランドです。 HiRDB サーバプロセスが他の HiRDB サーバプロセス（ディクショナリサーバ/バックエンドサーバ）から応答が返ってくるまでの最大待ち時間を指定します。 HiRDB サーバプロセスの中断を許したくない時間を指定しますが，指定した時間を過ぎても該当 HiRDB サーバプロセスの処理を確実に中断するわけではありません。 《利点》 HiRDB クライアントが SQL 実行を中断（クライアントプロセスの強制終了など），又はコマンド・ユティリティの実行を中断（強制停止等）しても，HiRDB サーバが当該 SQL やコマンド・ユティリティの実行を中止しない場合，HiRDB サーバはこの SQL やコマンド・ユティリティの実行を続けて長時間にわたって排他資源などを占有することがあります。このオペランドを指定すると，この占有時間を短くできます。 《指定値の目安》 次のうちの最も長い時間を指定してください。 • クライアント環境定義の PDCWAITTIME オペランドで指定する時間 • pd_lck_wait_timeout オペランドで指定する時間 • 実行時間が最も掛かる SQL の処理時間 • 実行時間が最もかかるコマンド・ユーティリティの時間値 《注意事項》 • このオペランドに 0 を指定した場合は，SQL 実行時間の監視をしません。 • SQL の最大実行時間を設定する場合は，このオペランドに 0 を設定し，クライアント環境変数 PDCWAITTIME で SQL 最大実行時間を設定することをお勧めします。クライアント環境定義 PDCWAITTIME については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。 • このオペランドに SQL 実行時間より短い値を指定した場合，SQL 実行中に処理が中断され，HiRDB クライアントに SQL エラーか，又は HiRDB サーバの異常終了が報告されることがあります。 • HiRDB/シングルサーバの場合にこのオペランドを指定しても，SQL 最大実行時間を監視しません。ただし，このオペランドを指定した場合，その指定値が pd_lck_wait_timeout オペランドの省略値として使用されます。このため，HiRDB/シングルサーバの場合はこのオペランドを省略することをお勧めします。	
[追加] **110)　pd_pth_trace_max = 通信トレース格納最大数** **～<符号なし整数>((1024～8388608))《1024》** トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。 ここでの指定値は，各ユニット制御情報定義の pd_pth_trace_max オペランドの省略時解釈値となります。 《指定値の目安》 通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。	pd_pth_trace_max

追加・変更内容	関連するオペランド
《注意事項》 通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。	
［追加］ 121） pd_rdarea_expand_format = Y｜N 以下の HiRDB ファイルシステム領域を通常ファイルで作成し，その領域を使用している RD エリアが RD エリアの自動増分機能を使用している場合，増分領域がスパースファイルになる可能性があります。 (i)-i オプション指定なしの pdfmkfs コマンドで作成した HiRDB ファイルシステム領域 (ii)-i オプションと-a オプションを指定した pdfmkfs コマンドで作成した HiRDB ファイルシステム領域 RD エリアを構成する HiRDB ファイルシステム領域がスパースファイルになると，グローバルバッファからのページ書き込み契機（デファードライト処理）で書き込みエラーが発生し，障害閉塞することがあります。 本オペランドの指定値に Y を指定すると, RD エリアの自動増分が発生したときに増分領域を初期化し，HiRDB ファイルシステム領域がスパースファイルになるのを防ぐことができます。 なお，初期化する対象は，ファイルの種類が通常ファイルで，-a オプションを指定した pdfmkfs コマンドで作成した HiRDB ファイルシステム領域です。 Y：増分領域の初期化を行います N：増分領域の初期化を行いません 《指定値の目安》 RD エリアの自動増分機能を使用するシステムでは，通常下記のリソース容量が満杯にならない運用（容量監視や定期的なデータベースの再編成の実施）を行ってください。この場合は本オペランドの指定値を N にすることを推奨します。 • RD エリア • HiRDB ファイルシステム領域 • HiRDB ファイルシステム領域を配置しているディスク 容量監視ができないシステムの場合は本オペランドの指定値を Y にしてください。詳細はマニュアル「HiRDB Version 8 運用ガイド」を参照してください。 《注意事項》 本オペランドの指定値を Y にする場合，以下の点に注意してください。 (a)自動増分が発生している RD エリアをアクセスする更新トランザクションの実行時間が延びるため，次の要因でトランザクションがキャンセルされることがあります。 • 排他待ち限界経過時間を超過 • HiRDB クライアントの最大待ち時間を超過 • ユティリティの最大実行時間を超過 《他パラメタとの関連》に示すパラメタについて，以下に示す RD エリアの構成情報と，HiRDB ファイルシステム領域を配置しているディスクの I/O 性能を考慮して，見積もりしてください※。 • ページサイズ • セグメントサイズ	pd_rdarea_expand_format

追加・変更内容	関連するオペランド
• 増分セグメント数 注※　増分領域の初期化にかかる処理時間(目安)[秒] • 自動増分に掛かる処理時間の目安 ページサイズ[byte]×(セグメントサイズ×増分セグメント数 + α) ÷HiRDB ファイルシステム領域を配置しているディスクの I/O 性能 [byte/秒] α：増分時に作成するディレクトリページ数 • データディクショナリ用 RD エリア・ユーザ用 RD エリア・レジストリ用 RD エリアの場合 ↑d÷b↑+↑d÷f↑ d：増分セグメント数 b：↓(P - 20)÷(↑d÷32↑×8 + 56)↓ f：↓(125×P)÷(16×b)↓×b P：ページサイズ S：セグメントサイズ • LOB 用 RD エリアの場合 ↑S÷64000↑×96 S：セグメントサイズ (b)本オペランドを指定した場合でも，増分領域以外は初期化を行いません。 このため，pdfmkfs コマンドに-i オプションを指定していない HiRDB ファイルシステム領域の場合，スパースファイルになります。このことから，HiRDB ファイルシステム領域を作成する際は，-i オプションの指定を推奨します。 《ほかのオペランドとの関連》 このオペランドは，次に示すオペランドと関連があります。 • PDCWAITTIME • pd_lck_wait_timeout • pd_utl_exec_time	
[追加] 127)　pd_db_hold_action = dbhold | unitdown RD エリアへのアクセス時に物理エラーが発生した場合（KFPH00306-E メッセージの閉塞理由に i/o error occurred，又は open error occurred を出力して閉塞するエラーの場合）の動作を指定します。 マスタディレクトリ用 RD エリアの場合は物理エラー検知時ユニットダウン機能を使用していなくても，障害閉塞しないでユニットダウンします。物理エラー検知時ユニットダウン機能については，マニュアル「HiRDB Version 8 システム運用 ガイド」を参照してください。 dbhold： 物理エラーが発生した場合，アクセス対象の RD エリアを閉塞します。 unitdown： 物理エラーが発生した場合，KFPH23047-I メッセージを出力して，物理エラー検知時ユニットダウン機能を適用します。 《適用基準》 通常，このオペランドを指定する必要はありません。 RD エリアの数が少なく，RD エリアが一つでも障害閉塞したら業務全体が停止するようなシステムの場合に，このオペランドの指定を検討してください。 unitdown を指定する場合は，pd_mode_conf オペランドに MANUAL2 を指定してください。	pd_db_hold_action

追加・変更内容	関連するオペランド
《注意事項》 • RD エリアに対するアクセスで物理エラーが発生し，KFPH00307-E メッセージを出力して RD エリアがコマンド閉塞した場合は，pd_db_hold_action オペランドに unitdown を指定していてもユニットダウンしません。 • unitdown を指定した場合に物理エラーが発生すると，次のケースでは処理対象の RD エリアが障害閉塞することがあります。 ・更新前ログ取得モード又はログレスモードで UAP やユティリティを実行している ・CREATE TABLE の RECOVERY オペランドで NO を指定してログレスモードにしたユーザ LOB 用 RD エリアに対して，UAP 又はユティリティを実行している 物理エラー検知時ユニットダウン機能を使用する場合は，できるだけこれらの運用は避けてください。もし，これらの運用が必要な場合は，RD エリアが閉塞しても最新の状態に回復できるよう，UAP 又はユティリティの実行前にバックアップを取得してください。バックアップの取得については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。 • データベース回復ユティリティ（pdrstr）での回復処理中は，unitdown を指定していてもユニットダウンしません。この場合は，データベース回復ユティリティ（pdrstr）を再度実行して回復してください。 《ほかのオペランドとの関連》 このオペランドは次に示すオペランドと関連があります。 • pd_mode_conf オペランド • pd_db_io_error_action オペランド • pd_db_access_error_action オペランド なお，pd_db_io_error_action オペランド，pd_db_access_error_action オペランド，及び pd_db_hold_action オペランドのうち，複数のオペランドで unitdown を指定した場合は，次の順序でオペランドの指定値が有効になります。 1. pd_db_io_error_action オペランド 2. pd_db_access_error_action オペランド 3. pd_db_hold_action オペランド そのため，RD エリアの入出力エラー，ファイルアクセスエラー，及び物理エラーのうち複数のエラーが同時に起こった場合，どのエラー要因によってユニットダウンしたのかは，この優先順位を参考に判断してください。また，出力されたメッセージを参照してください。	
［追加］ 《注意事項》 USE を指定，又は指定を省略した場合，PURGE TABLE 文を実行時にディクショナリ表（資源種別：3005 種別名：DICT）に対して一時的に EX モードで排他を掛けます。また，データディクショナリ用 RD エリア（資源種別：0001 種別名：RDAR）に対してトランザクションが終了するまで SU モードで排他を掛けます。 そのため，これらに排他を掛けるコマンド，又はユティリティと同時実行できないことがあります。PURGE TABLE 文と同時実行できないことがあるコマンド，又はユティリティと条件を次に示します。	pd_check_pending

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>
<table>
<tr><th>コマンドまたはユティリティ</th><th>条件</th></tr>
<tr><td>pdmod（データベース構成変更ユィリティ）</td><td>ディクショナリ表の属性定義変更の場合</td></tr>
<tr><td>pdcopy（データベース複写ユティリティ）</td><td>以下の条件を全て満たす場合
(a)-Mオプションにx，又はrを指定した場合
(b)複写対象にマスタディレクトリ用RDエリア，又はデータディクショナリ用RDエリアを含む場合</td></tr>
<tr><td>pdreginit（レジストリ機能初期設定ユティリティ）</td><td>-kオプションにallを指定した場合</td></tr>
</table>
そのため，参照制約，又は検査制約を使用しない場合は NOUSE を指定してください。USE を指定，又は指定を省略した場合は，PURGE TABLE 文と上記に示すコマンド，又はユティリティを同時実行しないようにしてください。
</td><td></td></tr>
<tr><td>[訂正前]
セキュリティ監査機能を使用する場合はこのオペランドを必ず指定してください。指定しないとセキュリティ監査機能を使用できません。
監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。
《注意事項》
• このオペランドを指定した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に監査証跡ファイル用の HiRDB ファイルシステム領域に対するアクセスエラーが発生すると，HiRDB を開始できません。
• 同一サーバマシン上で複数のユニットを稼働させるシステム構成の場合は，このオペランドでそれぞれの監査証跡ファイルを指定してください。システム共通定義の pd_aud_file_name オペランドの指定によって同一サーバマシン上の複数のユニットが同じ監査証跡ファイルを指定した場合，監査証跡を正しく取得できません。
[訂正後]
セキュリティ監査機能を使用する場合はこのオペランドを必ず指定してください。指定しないとセキュリティ監査機能を使用できません。
監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。
複数ユニットのシステム構成でセキュリティ監査機能を使用する場合，システム全体で監査証跡を取得することを推奨します。システム全体で監査証跡を取得するには，次のどちらかの指定をしてください。
• システム共通定義で pd_aud_file_name オペランドを指定する
• すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する
ただし，同一サーバマシンで複数のユニットを稼働させるシステム構成の場合は，すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する必要があります。
《注意事項》
• このオペランドを指定した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に監査証跡ファイル用の HiRDB ファイルシステム領域に対するアクセスエラーが発生すると，HiRDB を開始できません。
• システム共通定義の pd_aud_file_name オペランドの指定によって同一サーバマシン上の複数のユニットが同じ監査証跡ファイルを指定した場合，監査証跡を正しく取得できません。</td><td>pd_aud_file_name</td></tr>
<tr><td>[訂正前]</td><td>pd_aud_no_standby_file_opr</td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>
スワップ先にできる監査証跡ファイルがない場合の HiRDB の処理方式を指定します。
down：
スワップ先にできる監査証跡ファイルが残り一つになった場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を強制終了します。HiRDB を再開始した後に，データロード待ちの監査証跡ファイルを監査証跡表にデータロードしてください。その間，コマンド及びトランザクションの実行を一時的に抑止してください。
forcewrite：
スワップ先にできる監査証跡ファイルがない場合，データロード待ち（閉塞状態のファイルを除く）の監査証跡ファイルを強制的にスワップ先にして監査証跡の出力を続行します。このとき，最終更新日時が一番古いデータロード待ちの監査証跡ファイルをスワップ先にします。
また，全ファイルが閉塞状態の場合は監査証跡の出力を中止します。
[訂正後]
監査証跡の出力先が満杯になった場合の HiRDB の処理方式を指定します。
pd_aud_async_buff_size オペランドの値によって，以下に示す場合に，監査証跡の出力先が満杯になります。
<table>
<tr><th>pd_aud_async_buff_sizeオペランドの値</th><th>監査証跡の出力先が満杯になるケース</th></tr>
<tr><td>0（監査証跡の出力方式に同期出力を適用する場合）</td><td>・スワップ先にできる監査証跡ファイルがない場合</td></tr>
<tr><td>4096～6553600（監査証跡の出力方式に非同期出力を適用する場合）</td><td>・スワップ先にできる監査証跡ファイルがない場合
・監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちとなる場合</td></tr>
</table>
監査証跡の出力先が満杯になった場合，pd_aud_no_standby_file_opr オペランドの値によって，以下に示す HiRDB の処理を行います。
</td><td></td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>
<table>
<tr><th>pd_aud_no_standby_file_oprオペランドの値</th><th>スワップ先にできる監査証跡ファイルがない場合</th><th>監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちとなる場合</th></tr>
<tr><td>down</td><td>スワップ先にできる監査証跡ファイルが残り一つになった場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を強制終了します。downの指定が有効になるスワップの契機は，監査証跡ファイルの容量が一杯になった場合，及び現用ファイルに障害が発生した場合です。
このオペランドの指定によってHiRDBが強制終了した場合の対処については，マニュアル「HiRDB Version 8 システム運用ガイド」の「スワップ先にできる監査証跡ファイルがないためHiRDBが強制終了した場合」を参照してください。</td><td>監査証跡の出力処理が180秒以上滞った場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を強制終了します。その後の対処については，マニュアル「HiRDB Version 8 システム運用ガイド」の「監査証跡の非同期出力用バッファのすべての面が，バッファから監査証跡ファイルへの出力待ちとなった場合」に示す対処を行ってください。</td></tr>
<tr><td>forcewrite</td><td>スワップ先にできる監査証跡ファイルがない場合，データロード待ち（閉塞状態のファイルを除く）の監査証跡ファイルを強制的にスワップ先にして監査証跡の出力を続行します。このとき，最終更新日時が一番古いデータロード待ちの監査証跡ファイルをスワップ先にします。
ただし，pdaudswapコマンドを実行してスワップした場合，又は全ファイルが閉塞状態の場合は監査証跡の出力を中止します。</td><td>監査証跡の出力処理が180秒以上滞った場合，一番古いバッファに出力済みの監査証跡を破棄して，監査証跡の出力を続行します。</td></tr>
</table>
</td><td></td></tr>
<tr><td>[追加]
《注意事項》
• pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を小さく設定すると，監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちに陥り，監査証跡の出力処理が滞ることがあります。
単位時間あたりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を設定してください。</td><td>pd_aud_async_buff_size</td></tr>
<tr><td>[追加]
《注意事項》
• pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を小さく設定すると，監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちに陥り，監査証跡の出力処理が滞ることがあります。
単位時間あたりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を設定してください。</td><td>pd_aud_async_buff_count</td></tr>
<tr><td>[追加]</td><td>pd_aud_auto_loading</td></tr>
</table>

追加・変更内容	関連するオペランド
《留意点》 HiRDB/パラレルサーバにおいて，本オペランドに Y を指定して監査証跡ファイルから監査証跡表へデータロードする処理を自動で実行する場合，システムマネジャを配置したユニットに pd_aud_file_name オペランドで監査証跡ファイルのパス名を設定してください。 設定方法については，システム共通定義の pd_aud_file_name オペランドを参照してください。	
[訂正前] データ送信時に行うコネクション確立処理のリトライ間隔を指定します。 [訂正後] HiRDB のサーバ間通信で，データ送信時に行うコネクション確立処理のリトライ間隔を指定します。	pd_ipc_conn_interval
[訂正前] データ送信時に行うコネクション確立処理のリトライ回数を指定します。 ＜省略＞ 《注意事項》 ノンブロックモードのコネクション確立処理が実行される場合，このオペランドの指定は無効になります。 次に示す条件でノンブロックモードのコネクション確立処理が実行されます。 • pd_ipc_conn_nblock オペランドに Y を指定した場合 • クライアント環境定義の PDNBLOCKWAITTIME オペランドを指定した場合 [訂正後] HiRDB のサーバ間通信で，データ送信時に行うコネクション確立処理のリトライ回数を指定します。 ＜省略＞ 《注意事項》 ノンブロックモードのコネクション確立処理が実行される場合，このオペランドの指定は無効になります。 pd_ipc_conn_nblock オペランドに Y を指定すると，ノンブロックモードのコネクション確立処理が実行されます。	pd_ipc_conn_count
[訂正前] **197） pd_ipc_inet_bufsize = サーバのユニット間通信で使用する送受信バッファ長～＜符号なし整数＞((0～262144))《16384》（単位：バイト）** サーバのユニット間通信（TCP INET ドメイン）で使用する送受信バッファの最大値を 4096 の倍数で指定します。 《指定値の目安》 通常，このオペランドを指定する必要はありません。 《注意事項》 • 0 を指定した場合は OS の設定値を使用します。 • TCP のバッファサイズの上限値は各 OS で異なるので注意してください。 • 指定したバッファ長が無効になることがあります。この場合，Windows の設定値を使用します。 [訂正後] **200） pd_ipc_inet_bufsize = サーバ間通信で使用する送受信バッファ長**	pd_ipc_inet_bufsize

追加・変更内容	関連するオペランド
～<符号なし整数>((0～262144))《16384》（単位：バイト） サーバ間通信（TCP INET ドメイン）で使用する送受信バッファの最大値を 4096 の倍数で指定します。 《指定値の目安》 通常は 0 を指定して OS の設定値を適用してください。 《注意事項》 • 0 を指定した場合は OS の設定値を使用します。 • TCP のバッファサイズの上限値は各 OS で異なるので注意してください。 • システム定義 pd_utl_buff_size を使用するユティリティの通信処理では，指定したバッファ長を無効にし，pd_utl_buff_size 指定値を元に，最適な値を設定します。 《ほかのオペランドとの関連》 このオペランドは次に示すオペランドと関連があります。 • pd_utl_buff_size	
［訂正前］ 198） pd_tcp_inet_bufsize = HiRDB サーバが存在するホスト外の HiRDB クライアントとの通信で使用する送受信バッファ長 ～<符号なし整数>((0～262144))《0》（単位：バイト） HiRDB サーバがあるホスト以外のホスト上から接続した HiRDB クライアントとの通信（TCP INET ドメイン）で使用する送受信バッファの最大値を，4096 の倍数で指定します。 《指定値の目安》 通常，このオペランドを指定する必要はありません。次に示す現象が発生した場合に，このオペランドを指定します。 • HiRDB サーバと HiRDB クライアント間で送受信するデータの再送がある場合，このオペランドを指定すると再送を抑止できます。 • 受信側の処理で，データ到着速度が読み込み処理速度よりも遅い場合，このオペランドを指定すると，送信データ量が増えて受信時の到着待ちによる遅延を防げます。 このオペランドには，通信に使用するバッファサイズと同じ値を指定してください。 《注意事項》 • 0 を指定した場合は OS の設定値を使用します。 • TCP のバッファサイズの上限値は各 OS で異なるので注意してください。 • 指定したバッファ長が無効になることがあります。この場合，Windows の設定値を使用します。 ［訂正後］ 201） pd_tcp_inet_bufsize = HiRDB クライアントとの通信で使用する送受信バッファ長 ～<符号なし整数>((0～262144))《0》（単位：バイト） HiRDB クライアントとの通信（TCP INET ドメイン）で使用する送受信バッファの最大値を，4096 の倍数で指定します。 《指定値の目安》 通常，このオペランドを指定する必要はありません。次に示す現象が発生した場合に，このオペランドを指定します。	pd_tcp_inet_bufsize

追加・変更内容	関連するオペランド
• HiRDB サーバと HiRDB クライアント間で送受信するデータの再送がある場合，このオペランドを指定すると再送を抑止できます。 • 受信側の処理で，データ到着速度が読み込み処理速度よりも遅い場合，このオペランドを指定すると，送信データ量が増えて受信時の到着待ちによる遅延を防げます。 このオペランドには，通信に使用するバッファサイズと同じ値を指定してください。 《注意事項》 • 0 を指定した場合は OS の設定値を使用します。 • TCP のバッファサイズの上限値は各 OS で異なるので注意してください。	
[訂正前] 次に示すユティリティのプロセスが使用する通信用バッファ 1 面のサイズをキロバイト単位で指定します。 • データベース作成ユティリティ • データベース再編成ユティリティ（-g オプション指定時） • データベース回復ユティリティ • データベース複写ユティリティ [訂正後] 次に示すユティリティのプロセスが使用する通信用バッファ 1 面のサイズをキロバイト単位で指定します。 • データベース作成ユティリティ • データベース再編成ユティリティ（-g オプション指定時） • リバランスユティリティ • データベース回復ユティリティ • データベース複写ユティリティ	pd_utl_buff_size
[追加] 《利点》 ユティリティ処理を行うサーバプロセスと HiRDB サーバ間の通信処理で，データの再送が見られる場合に指定値を大きくして HiRDB サーバ間の通信回数を減らすことができます。	pd_utl_buff_size
[訂正前] グローバルバッファ用の共用メモリセグメントサイズの上限値をメガバイト単位で指定します。 《指定値の目安》 • グローバルバッファが使用する共用メモリの計算式で求めた値を指定してください。グローバルバッファを動的変更する場合は，追加予定のグローバルバッファのサイズを考慮してください。グローバルバッファが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。 • 次の場合，ページ固定に対応している Windows で共用メモリのメモリ所要量を見積もるときは, Windows の Large Page でのページサイズに切り上げた値を指定してください。なお，Windows がページ固定に対応しているかどうか，及び Large Page のページサイズについては，pdntenv -os コマンドで確認してください。 ・pd_dbbuff_attribute オペランドに fixed を指定	SHMMAX

追加・変更内容	関連するオペランド
・pd_max_resident_rdarea_no オペランドに 1 以上を指定 • HiRDB はこのオペランドの指定値を上限としてグローバルバッファ用の共用メモリセグメントを確保します。サーバマシン内の RD エリアに割り当てるグローバルバッファサイズの合計がこのオペランドの値を超える場合は，複数の共用メモリセグメントを割り当てます。複数の共用メモリセグメントにグローバルバッファが確保されると，共用メモリセグメントに対するアクセスオーバヘッドが増加してスループットが低下する原因になります。このため，できるだけ一つの共用メモリセグメント内にすべてのグローバルバッファを確保できるようにこのオペランドの値を見積もってください。 なお，一つの共用メモリセグメント内にすべてのグローバルバッファが確保されたかどうかは pdls -d mem コマンドで確認できます。1 ユニットに 1 サーバの場合は共用メモリセグメントの情報が二つ表示されます。したがって，1 ユニットに複数のサーバがある場合は，「セグメント情報数－1=サーバ数」となっていれば，一つの共用メモリセグメントに確保されています。 • 1 ユニット当たり最低二つの共用メモリセグメント（ユニットコントローラ，サーバ用及びグローバルバッファ用）を割り当てます。共用メモリセグメントは 16 個まで確保できます。 • グローバルバッファを動的変更すると，新規の共用メモリセグメントを確保し，動的変更したグローバルバッファをその共用メモリセグメントに割り当てます。確保できる共用メモリセグメント数の上限値は，pd_max_add_dbbuff_shm_no オペランドで指定します。 《注意事項》 • 共用メモリを HiRDB 運用ディレクトリ下のファイルに割り当てる場合，HiRDB 運用ディレクトリがあるディスクの空き容量が十分でないと，ディスク容量不足になります。 • 共用メモリをページファイルに割り当てる場合，ページファイル容量が十分でないと，ページファイル容量不足になります。 • 共用メモリの割り当て先は pdntenv コマンドで変更できます。pdntenv コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。 《ほかのオペランドとの関連》 このオペランドは次に示すオペランドと関連があります。 • pd_dbbuff_modify • pdbuffer • pd_max_add_dbbuff_no • pd_max_add_dbbuff_shm_no ［訂正後］ グローバルバッファ用の共用メモリセグメントサイズの上限値をメガバイト単位で指定します。 HiRDB はこのオペランドの指定値を上限としてグローバルバッファ用の共用メモリセグメントを確保します。サーバマシン内の RD エリアに割り当てるグローバルバッファサイズの合計がこのオペランドの値を超える場合は，複数の共用メモリセグメントを割り当てます。 HiRDB は開始時に 1 サーバ当たり最大 16 セグメントのグローバルバッファプール用共用メモリを確保します。グローバルバッファプールが使用している共用メモリセグメントの情報は pdls コマンド（-d mem 指定）によって確認できます。 《指定値の目安》 <32 ビットモード>	

追加・変更内容	関連するオペランド
• SHMMAX を特に指定する必要はありません。ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合には，200 を指定してください。 • KFPH23005-E メッセージ（エラーコード 20）が出力される場合，マニュアル「HiRDB Version 8 メッセージ」を参照して KFPH23005-E メッセージの対策に示される SHMMAX の値に変更してください。 <64 ビットモード> • グローバルバッファプール用共用メモリサイズが 16GB 以下の場合，SHMMAX を指定する必要はありません。 • グローバルバッファプール用共用メモリサイズが 16GB より大きい場合，SHMMAX の指定値に以下の式を満たす値を指定してください。 **（SHMMAX の指定値）×16 ＝（グローバルバッファプール用共用メモリサイズ）** 《注意事項》 • グローバルバッファ動的変更機能を使用している場合，SHMMAX の指定値の変更は正常停止後のみできます。 そのため，強制停止または計画停止後の再開始時に KFPH23005-E メッセージ（エラーコード 20）が発生し，SHMMAX の値の変更が必要な場合，マニュアル「HiRDB Version 8 メッセージ」の KFPH23005-E メッセージ（エラーコード 20）の表 項番 3 を参照して，SHMMAX の指定値を変更してください。 • 共用メモリは，共用メモリファイルとして HiRDB 運用ディレクトリがあるディスクに確保されます。このため，ディスクの空き容量が十分でないと，ディスク容量不足になります。 • 共用メモリをページファイルに割り当てる場合，ページファイル容量が十分でないと，ページファイル容量不足になります。 共用メモリを割り当てる方法については，マニュアル「HiRDB Version 8 コマンドリファレンス」の pdntenv コマンドを参照してください。 • 共用メモリのページ固定機能を使用する場合は，本オペランドにはラージページページ長の整数倍の値を指定してください。共用メモリのページ固定は以下の場合に使用されます。 ・pd_dbbuff_attribute オペランドに fixed を指定してください。 ・pd_max_resident_rdarea_no オペランドに 1 以上を指定ください。 指定値がページ長の整数倍でない場合は HiRDB がページ長の整数倍の値に補正します。 • 本オペランドを省略した場合，デフォルト値が仮定されます。また，本オペランドに指定可能範囲外の値を指定した場合，デフォルト値が仮定されたものとみなし，警告メッセージ KFPH23011-W を出力した後に HiRDB 開始処理を続行します。 • グローバルバッファプール用共用メモリが使用する管理領域は SHMMAX に指定したサイズで分割することができません。そのため，SHMMAX 指定値には上記管理領域のサイズより大きい値を指定してください。グローバルバッファプール用共用メモリが使用する管理領域サイズの詳細はマニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。 《ほかのオペランドとの関連》 このオペランドは次に示すオペランドと関連があります。 • pd_dbbuff_modify • pdbuffer • pd_max_add_dbbuff_no	

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>

- pd_max_add_dbbuff_shm_no
- pd_sysdef_default_option（32 ビットモードで v6compatible を指定している場合，このオペランドの省略値は 6 になります）

</td><td></td></tr>
<tr><td>

[訂正前]

<table>
<tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr>
<tr><td>11</td><td>〔set pd_watch_time = SQL最大実行時間〕※</td><td>システム監視</td></tr>
<tr><td>75</td><td>〔set pd_ipc_inet_bufsize = サーバのユニット間通信で使用する送受信バッファ長〕※</td><td>通信処理</td></tr>
<tr><td>76</td><td>〔set pd_tcp_inet_bufsize = HiRDBサーバが存在するホスト外のHiRDBクライアントとの通信で使用する送受信バッファ長〕※</td><td>通信処理</td></tr>
</table>

[訂正後]

<table>
<tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr>
<tr><td>11</td><td>〔set pd_watch_time = 最大応答待ち時間〕※</td><td>システム監視</td></tr>
<tr><td>76</td><td>〔set pd_ipc_inet_bufsize = サーバ間通信で使用する送受信バッファ長〕※</td><td>通信処理</td></tr>
<tr><td>77</td><td>〔set pd_tcp_inet_bufsize = HiRDBクライアントとの通信で使用する送受信バッファ長〕※</td><td>通信処理</td></tr>
</table>

</td><td>－</td></tr>
<tr><td>

[追加]

<table>
<tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr>
<tr><td>33</td><td>〔set pd_pth_trace_max = 通信トレース格納最大数〕</td><td>トラブルシュート情報</td></tr>
</table>

</td><td>－</td></tr>
<tr><td>

《指定値の目安》

[訂正前]

省略値＝a＋b×（c＋27）＋73＋i＋j＋k＋n

[訂正後]

省略値＝a＋b×（c＋30）＋70＋i＋j＋k＋n

</td><td>pd_max_server_process</td></tr>
<tr><td>

《指定値の目安》

[訂正前]

推奨値の目安＝a＋b×（c＋d×e＋f）＋d×g＋h＋i＋j＋k＋n

[訂正後]

推奨値の目安＝a＋b×（c＋d×e＋f＋3）＋d×g＋h＋i＋j＋k＋n

</td><td>pd_max_server_process</td></tr>
<tr><td>

[訂正前]

<table>
<tr><th>変数</th><th>変数の説明</th></tr>
<tr><td>a</td><td>＜省略＞
・影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_server_processオペランドの値を代入します。</td></tr>
<tr><td>i</td><td>＜省略＞
・影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_serversオペランドの値を代入します。</td></tr>
</table>

[訂正後]

</td><td>pd_max_server_process</td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>
<table>
<tr><th>変数</th><th>変数の説明</th></tr>
<tr><td>a</td><td>＜省略＞
・マルチフロントエンドサーバの場合，フロントエンドサーバごとに1を加算してください。
・影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_server_processオペランドの値を代入します。</td></tr>
<tr><td>i</td><td>＜省略＞
・影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_serversオペランドの値を加算します。</td></tr>
</table>
</td><td></td></tr>
<tr><td>
[訂正前]
11） pd_watch_time = SQL 最大実行時間
～＜符号なし整数＞((0～65535))（単位：秒）
このオペランドは HiRDB/パラレルサーバ限定のオペランドです。
HiRDB サーバプロセス上で実行される SQL の最大実行時間を指定します。
このオペランドで指定した時間を過ぎても SQL の実行が完了しない場合は，その SQL の実行を中断することがあります。
《利点》
HiRDB クライアントが SQL 実行を中断（クライアントプロセスの強制終了など）しても，HiRDB サーバがこの SQL の実行を中止しない場合，HiRDB サーバはこの SQL の実行を続けて長時間にわたって排他資源などを占有することがあります。このオペランドを指定すると，この占有時間を短くできます。
《指定値の目安》
次のうちの最も長い時間を指定してください。
• クライアント環境定義の PDCWAITTIME オペランドで指定する時間
• クライアント環境定義の PDLCKWAITTIME オペランドで指定する時間
• pd_lck_wait_timeout オペランドで指定する時間
• 実行時間が最も掛かる SQL の処理時間
《オペランドの省略値》
このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。
《注意事項》
• このオペランドに 0 を指定した場合は，SQL 実行時間の監視をしません。
• クライアント環境定義の PDCWAITTIME オペランドに 0（時間監視しない）を指定した SQL については，SQL 実行時間の監視をしません。
• このオペランドに SQL 実行時間より短い値を指定した場合，SQL 実行中に処理が中断され，HiRDB クライアントに SQL エラーか，又は HiRDB サーバの異常終了が報告されることがあります。
• HiRDB/シングルサーバの場合にこのオペランドを指定しても，SQL 最大実行時間を監視しません。ただし，このオペランドを指定した場合，その指定値が pd_lck_wait_timeout オペランドの省略値として使用されます。このため，HiRDB/シングルサーバの場合はこのオペランドを省略することをお勧めします。
[訂正後]
11） pd_watch_time = 最大応答待ち時間
～＜符号なし整数＞((0～65535))（単位：秒）
このオペランドは旧バージョンとの互換性保証のための指定です。したがって指定する必要はありません。
</td><td>pd_watch_time</td></tr>
</table>

追加・変更内容	関連するオペランド
このオペランドは HiRDB/パラレルサーバ限定のオペランドです。 HiRDB サーバプロセスが他の HiRDB サーバプロセス（ディクショナリサーバ/バックエンドサーバ）から応答が返ってくるまでの最大待ち時間を指定します。 HiRDB サーバプロセスの中断を許したくない時間を指定しますが，指定した時間を過ぎても該当 HiRDB サーバプロセスの処理を確実に中断するわけではありません。 《利点》 HiRDB クライアントが SQL 実行を中断（クライアントプロセスの強制終了など），又はコマンド・ユティリティの実行を中断（強制停止等）しても，HiRDB サーバが当該 SQL やコマンド・ユティリティの実行を中止しない場合，HiRDB サーバはこの SQL やコマンド・ユティリティの実行を続けて長時間にわたって排他資源などを占有することがあります。このオペランドを指定すると，この占有時間を短くできます。 《指定値の目安》 次のうちの最も長い時間を指定してください。 • クライアント環境定義の PDCWAITTIME オペランドで指定する時間 • pd_lck_wait_timeout オペランドで指定する時間 • 実行時間が最も掛かる SQL の処理時間 • 実行時間が最もかかるコマンド・ユーティリティの時間値 《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。 《注意事項》 • このオペランドに 0 を指定した場合は，SQL 実行時間の監視をしません。 • SQL の最大実行時間を設定する場合は，このオペランドに 0 を設定し，クライアント環境変数 PDCWAITTIME で SQL 最大実行時間を設定することをお勧めします。クライアント環境定義 PDCWAITTIME については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。 • このオペランドに SQL 実行時間より短い値を指定した場合，SQL 実行中に処理が中断され，HiRDB クライアントに SQL エラーか，又は HiRDB サーバの異常終了が報告されることがあります。 • HiRDB/シングルサーバの場合にこのオペランドを指定しても，SQL 最大実行時間を監視しません。ただし，このオペランドを指定した場合，その指定値が pd_lck_wait_timeout オペランドの省略値として使用されます。このため，HiRDB/シングルサーバの場合はこのオペランドを省略することをお勧めします。	
［追加］ **33)　pd_pth_trace_max = 通信トレース格納最大数** **～<符号なし整数>((1024～8388608))** トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。 《指定値の目安》 通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。 《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，1024 が仮定されます。	pd_pth_trace_max

追加・変更内容	関連するオペランド
《注意事項》 通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。	
［訂正前］ セキュリティ監査機能を使用する場合はこのオペランドを必ず指定してください。指定しないとセキュリティ監査機能を使用できません。 監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。 《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。 《注意事項》 • このオペランドを指定した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に監査証跡ファイル用の HiRDB ファイルシステム領域に対するアクセスエラーが発生すると，HiRDB を開始できません。 • 同一サーバマシン上で複数のユニットを稼働させるシステム構成の場合は，このオペランドでそれぞれの監査証跡ファイルを指定してください。システム共通定義の pd_aud_file_name オペランドの指定によって同一サーバマシン上の複数のユニットが同じ監査証跡ファイルを指定した場合，監査証跡を正しく取得できません。 ［訂正後］ セキュリティ監査機能を使用する場合はこのオペランドを必ず指定してください。指定しないとセキュリティ監査機能を使用できません。 監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。 複数ユニットのシステム構成でセキュリティ監査機能を使用する場合，システム全体で監査証跡を取得することを推奨します。システム全体で監査証跡を取得するには，次のどちらかの指定をしてください。 • システム共通定義で pd_aud_file_name オペランドを指定する • すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する ただし，同一サーバマシンで複数のユニットを稼働させるシステム構成の場合は，すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する必要があります。 《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。 《注意事項》 • このオペランドを指定した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に監査証跡ファイル用の HiRDB ファイルシステム領域に対するアクセスエラーが発生すると，HiRDB を開始できません。 • システム共通定義の pd_aud_file_name オペランドの指定によって同一サーバマシン上の複数のユニットが同じ監査証跡ファイルを指定した場合，監査証跡を正しく取得できません。	pd_aud_file_name
［追加］	pd_aud_async_buff_size

追加・変更内容	関連するオペランド
《注意事項》 • pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を小さく設定すると，監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちに陥り，監査証跡の出力処理が滞ることがあります。 単位時間あたりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を設定してください。	
［追加］ 《注意事項》 • pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を小さく設定すると，監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちに陥り，監査証跡の出力処理が滞ることがあります。 単位時間あたりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を設定してください。	pd_aud_async_buff_count
［訂正前］ **75） pd_ipc_inet_bufsize = サーバのユニット間通信で使用する送受信バッファ長～<符号なし整数>((0～262144))（単位：バイト）** サーバのユニット間通信（TCP INET ドメイン）で使用する送受信バッファの最大値を 4096 の倍数で指定します。 《指定値の目安》 通常，このオペランドを指定する必要はありません。 《注意事項》 • 0 を指定した場合は OS の設定値を使用します。 • TCP のバッファサイズの上限値は各 OS で異なるので注意してください。 • 指定したバッファ長が無効になることがあります。この場合，Windows の設定値を使用します。 《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，16384 が仮定されます。 ［訂正後］ **76） pd_ipc_inet_bufsize = サーバ間通信で使用する送受信バッファ長～<符号なし整数>((0～262144))（単位：バイト）** サーバ間通信（TCP INET ドメイン）で使用する送受信バッファの最大値を 4096 の倍数で指定します。 《指定値の目安》 通常は 0 を指定して OS の設定値を適用してください。 《注意事項》 • 0 を指定した場合は OS の設定値を使用します。 • TCP のバッファサイズの上限値は各 OS で異なるので注意してください。 • システム定義 pd_utl_buff_size を使用するユティリティの通信処理では，指定したバッファ長を無効にし，pd_utl_buff_size 指定値を元に，最適な値を設定します。	pd_ipc_inet_bufsize

追加・変更内容	関連するオペランド
《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。 システム共通定義の同じオペランドも省略すると，16384 が仮定されます。 《ほかのオペランドとの関連》 このオペランドは次に示すオペランドと関連があります。 • pd_utl_buff_size	
［訂正前］ **76） pd_tcp_inet_bufsize = HiRDB サーバが存在するホスト外の HiRDB クライアントとの通信で使用する送受信バッファ長** **～<符号なし整数>((0～262144))（単位：バイト）** HiRDB サーバがあるホスト以外のホスト上から接続した HiRDB クライアントとの通信（TCP INET ドメイン）で使用する送受信バッファの最大値を，4096 の倍数で指定します。 《指定値の目安》 通常，このオペランドを指定する必要はありません。次に示す現象が発生した場合に，このオペランドを指定します。 • HiRDB サーバと HiRDB クライアント間で送受信するデータの再送がある場合，このオペランドを指定すると再送を抑止できます。 • 受信側の処理で，データ到着速度が読み込み処理速度よりも遅い場合，このオペランドを指定すると，送信データ量が増えて受信時の到着待ちによる遅延を防げます。 このオペランドには，通信に使用するバッファサイズと同じ値を指定してください。 《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。 《注意事項》 • 0 を指定した場合は OS の設定値を使用します。 • TCP のバッファサイズの上限値は各 OS で異なるので注意してください。 • 指定したバッファ長が無効になることがあります。この場合，Windows の設定値を使用します。 ［訂正後］ **77） pd_tcp_inet_bufsize = HiRDB クライアントとの通信で使用する送受信バッファ長** **～<符号なし整数>((0～262144))（単位：バイト）** HiRDB クライアントとの通信（TCP INET ドメイン）で使用する送受信バッファの最大値を，4096 の倍数で指定します。 《指定値の目安》 通常，このオペランドを指定する必要はありません。次に示す現象が発生した場合に，このオペランドを指定します。 • HiRDB サーバと HiRDB クライアント間で送受信するデータの再送がある場合，このオペランドを指定すると再送を抑止できます。 • 受信側の処理で，データ到着速度が読み込み処理速度よりも遅い場合，このオペランドを指定すると，送信データ量が増えて受信時の到着待ちによる遅延を防げます。	pd_tcp_inet_bufsiz

追加・変更内容	関連するオペランド
このオペランドには，通信に使用するバッファサイズと同じ値を指定してください。 《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。 《注意事項》 • 0 を指定した場合は OS の設定値を使用します。 • TCP のバッファサイズの上限値は各 OS で異なるので注意してください。	
［訂正前］ • 32 ビットモードの場合：((6～2047)) • 64 ビットモードの場合：((6～4194304)) グローバルバッファ用の共用メモリセグメントサイズの上限値をメガバイト単位で指定します。 《指定値の目安》 • グローバルバッファが使用する共用メモリの計算式で求めた値を指定してください。グローバルバッファを動的変更する場合は，追加予定のグローバルバッファのサイズを考慮してください。グローバルバッファが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。 • 次の場合，ページ固定に対応している Windows で共用メモリのメモリ所要量を見積もるときは Windows の Large Page でのページサイズに切り上げた値を指定してください。 ・pd_dbbuff_attribute オペランドに fixed を指定 ・pd_max_resident_rdarea_no オペランドに 1 以上を指定 • HiRDB はこのオペランドの指定値を上限としてグローバルバッファ用の共用メモリセグメントを確保します。サーバマシン内の RD エリアに割り当てるグローバルバッファサイズの合計がこのオペランドの値を超える場合は，複数の共用メモリセグメントを割り当てます。複数の共用メモリセグメントにグローバルバッファが確保されると，共用メモリセグメントに対するアクセスオーバヘッドが増加してスループットが低下する原因になります。このため，できるだけ一つの共用メモリセグメント内にすべてのグローバルバッファを確保できるようにこのオペランドの値を見積もってください。 なお，一つの共用メモリセグメント内にすべてのグローバルバッファが確保されたかどうかは pdls -d mem コマンドで確認できます。1 ユニットに 1 サーバの場合は共用メモリセグメントの情報が二つ表示されます。したがって，1 ユニットに複数のサーバがある場合は，「セグメント情報数－1=サーバ数」となっていれば，一つの共用メモリセグメントに確保されています。 • 1 ユニット当たり最低二つの共用メモリセグメント（ユニットコントローラ，サーバ用及びグローバルバッファ用）を割り当てます。共用メモリセグメントは 16 個まで確保できます。 • グローバルバッファを動的変更すると，新規の共用メモリセグメントを確保し，動的変更したグローバルバッファをその共用メモリセグメントに割り当てます。確保できる共用メモリセグメント数の上限値は，pd_max_add_dbbuff_shm_no オペランドで指定します。 《オペランドの省略値》 このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次の値が仮定されます。	SHMMAX

追加・変更内容	関連するオペランド
• 32 ビットモードの場合：200 • 64 ビットモードの場合：1024 《注意事項》 • 共用メモリを HiRDB 運用ディレクトリ下のファイルに割り当てる場合，HiRDB 運用ディレクトリがあるディスクの空き容量が十分でないと，ディスク容量不足になります。 • 共用メモリをページファイルに割り当てる場合，ページファイル容量が十分でないと，ページファイル容量不足になります。 《ほかのオペランドとの関連》 このオペランドは次に示すオペランドと関連があります。 ・pd_dbbuff_modify ・pdbuffer ・pd_max_add_dbbuff_no ・pd_max_add_dbbuff_shm_no ［訂正後］ • 32 ビットモードの場合：((6～2047))《200》 • 64 ビットモードの場合：((6～4194304))《1024》 グローバルバッファ用の共用メモリセグメントサイズの上限値をメガバイト単位で指定します。 HiRDB はこのオペランドの指定値を上限としてグローバルバッファ用の共用メモリセグメントを確保します。サーバマシン内の RD エリアに割り当てるグローバルバッファサイズの合計がこのオペランドの値を超える場合は，複数の共用メモリセグメントを割り当てます。 HiRDB は開始時に 1 サーバ当たり最大 16 セグメントのグローバルバッファプール用共用メモリを確保します。グローバルバッファプールが使用している共用メモリセグメントの情報は pdls コマンド（-d mem 指定）によって確認できます。 《指定値の目安》 <32 ビットモード> • SHMMAX を特に指定する必要はありません。ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合には，200 を指定してください。 • KFPH23005-E メッセージ（エラーコード 20）が出力される場合，マニュアル「HiRDB Version 8 メッセージ」を参照して KFPH23005-E メッセージの対策に示される SHMMAX の値に変更してください。 <64 ビットモード> • グローバルバッファプール用共用メモリサイズが 16GB 以下の場合，SHMMAX を指定する必要はありません。 • グローバルバッファプール用共用メモリサイズが 16GB より大きい場合，SHMMAX の指定値に以下の式を満たす値を指定してください。 （SHMMAX の指定値）×16 ＝（グローバルバッファプール用共用メモリサイズ） 《注意事項》 • グローバルバッファ動的変更機能を使用している場合，SHMMAX の指定値の変更は正常停止後のみできます。 そのため，強制停止または計画停止後の再開始時に KFPH23005-E メッセージ（エラーコード 20）が発生し，SHMMAX の値の変更が必要な場合，マニュア	

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>
ル「HiRDB Version 8 メッセージ」の KFPH23005-E メッセージ（エラーコード 20）の表 項番 3 を参照して，SHMMAX の指定値を変更してください。
• 共用メモリは，共用メモリファイルとして HiRDB 運用ディレクトリがあるディスクに確保されます。このため，ディスクの空き容量が十分でないと，ディスク容量不足になります。
• 共用メモリをページファイルに割り当てる場合，ページファイル容量が十分でないと，ページファイル容量不足になります。
共用メモリを割り当てる方法については，マニュアル「HiRDB Version 8 コマンドリファレンス」の pdntenv コマンドを参照してください。
• 共用メモリのページ固定機能を使用する場合は，本オペランドにはラージページページ長の整数倍の値を指定してください。共用メモリのページ固定は以下の場合に使用されます。
・pd_dbbuff_attribute オペランドに fixed を指定してください。
・pd_max_resident_rdarea_no オペランドに 1 以上を指定ください。
指定値がページ長の整数倍でない場合は HiRDB がページ長の整数倍の値に補正します。
• 本オペランドに指定可能範囲外の値を指定した場合，本オペランドの省略値が仮定されたものとみなし，警告メッセージ KFPH23011-W を出力した後に HiRDB 開始処理を続行します。
• グローバルバッファプール用共用メモリが使用する管理領域は SHMMAX に指定したサイズで分割することができません。そのため，SHMMAX 指定値には上記管理領域のサイズより大きい値を指定してください。グローバルバッファプール用共用メモリが使用する管理領域サイズの詳細はマニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
《ほかのオペランドとの関連》
このオペランドは次に示すオペランドと関連があります。
・pd_dbbuff_modify
・pdbuffer
・pd_max_add_dbbuff_no
・pd_max_add_dbbuff_shm_no
・pd_sysdef_default_option（32 ビットモードで v6compatible を指定している場合，このオペランドの省略値は 6 になります）
</td><td></td></tr>
<tr><td>
［追加］
<table>
<tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr>
<tr><td>36</td><td>〔set pd_pth_trace_max = 通信トレース格納最大数〕</td><td>トラブルシュート情報</td></tr>
<tr><td>41</td><td>〔set pd_java_castoff = Y｜N〕</td><td>Java</td></tr>
</table>
</td><td>－</td></tr>
<tr><td>
［訂正前］
UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。
［訂正後］
UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，
</td><td>pd_spd_syncpoint_skip_limit</td></tr>
</table>

追加・変更内容	関連するオペランド
システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。 また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。 この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。	
《注意事項》 ［削除］ • HiRDB の開始（再開始も含みます）後，最初のシンクポイントダンプの有効化以降にこのオペランドの指定値が適用されます。	pd_spd_syncpoint_skip_limit
［訂正前］ **「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数** ［訂正後］ **「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数** 係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。	pd_lck_pool_size
［訂正前］ **「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数** ［訂正後］ **「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数** 係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。	pd_fes_lck_pool_size
［追加］ **36） pd_pth_trace_max = 通信トレース格納最大数** **～<符号なし整数>((1024～8388608))** トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。 《指定値の目安》 通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。 《オペランドの省略値》 このオペランドを省略すると，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。ユニット制御情報定義とシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。 《注意事項》 通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。	pd_pth_trace_max
［追加］ **41） pd_java_castoff = Y | N** HiRDB のサーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，バックエンドサーバ）プロセスで，Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。	pd_java_castoff

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>
Y：Java 仮想マシンを起動した場合，以下の契機でサーバプロセスを終了させます。

#	サーバの種類	プロセス名称	プロセス終了契機
1	シングルサーバ	pdsds	UAPの切り離し時
2	フロントエンドサーバ	pdfes	UAPの切り離し時
3	ディクショナリサーバ	pddic	トランザクション決着時，又はUAPの切り離し時
4	バックエンドサーバ	pdbes	トランザクション決着時，又はUAPの切り離し時

N：Java 仮想マシンを起動してもサーバプロセスが終了しない。

ここでの指定値は，各サーバ定義の pd_java_castoff オペランドの省略時の解釈値となります。

《指定値のチューニング方法》

Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。

Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に，pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。

#	懸案
1	Java仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。
2	Java仮想マシンを使用しないコネクションで実行するSQLでも，別のコネクションでJava仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多いSQLを実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。 なお，Java仮想マシンの機能についてはJava仮想マシンのドキュメントのオプションを参照してください。

《注意事項》

Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- pd_process_count
</td><td></td></tr>
<tr><td>
[追加]

番号	形式	オペランドの分類
32	[set pd_pth_trace_max = 通信トレース格納最大数]	トラブルシュート情報
38	[set pd_java_castoff = Y|N]	Java
</td><td>－</td></tr>
<tr><td>
[訂正前]

UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

[訂正後]

UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，
</td><td>pd_spd_syncpoint_skip_limit</td></tr>
</table>

追加・変更内容	関連するオペランド
システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。 また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。 この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。	
《注意事項》 [削除] • HiRDB の開始（再開始も含みます）後，最初のシンクポイントダンプの有効化以降にこのオペランドの指定値が適用されます。	pd_spd_syncpoint_skip_limit
[訂正前] **「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数** [訂正後] **「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数** 係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。	pd_lck_pool_size
[追加] **32)　pd_pth_trace_max = 通信トレース格納最大数** **～<符号なし整数>((1024～8388608))** トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。 《指定値の目安》 通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。 《オペランドの省略値》 このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。 《注意事項》 通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。	pd_pth_trace_max
[追加] **38)　pd_java_castoff = Y｜N** HiRDB のシングルサーバプロセスで，Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。 **Y**：Java 仮想マシンを起動した場合，UAP の切り離し時にサーバプロセスを終了します。 **N**：Java 仮想マシンを起動してもサーバプロセスが終了しません。 省略した場合は，サーバ共通定義の pd_java_castoff オペランドが解釈値となります。	pd_java_castoff

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>《指定値のチューニング方法》
Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。
Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に，pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。
<table><tr><th>#</th><th>懸案</th></tr><tr><td>1</td><td>Java仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。</td></tr><tr><td>2</td><td>Java仮想マシンを使用しないコネクションで実行するSQLでも，別のコネクションでJava仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多いSQLを実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。
なお，Java仮想マシンの機能についてはJava仮想マシンのドキュメントのオプションを参照してください。</td></tr></table>
《注意事項》
Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。
《ほかのオペランドとの関連》
このオペランドは次のオペランドと関連があります。
• pd_process_count</td><td></td></tr>
<tr><td>[追加]
<table><tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr><tr><td>26</td><td>〔set pd_pth_trace_max = 通信トレース格納最大数〕</td><td>トラブルシュート情報</td></tr><tr><td>29</td><td>〔set pd_java_castoff = Y｜N〕</td><td>Java</td></tr></table></td><td>－</td></tr>
<tr><td>[訂正前]
UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。
[訂正後]
UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。
また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。
この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。</td><td>pd_spd_syncpoint_skip_limit</td></tr>
<tr><td>《注意事項》
[削除]
• HiRDB の開始（再開始も含みます）後，最初のシンクポイントダンプの有効化以降にこのオペランドの指定値が適用されます。</td><td>pd_spd_syncpoint_skip_limit</td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>［訂正前］
「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数
［訂正後］
「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数
係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。</td><td>pd_fes_lck_pool_size</td></tr>
<tr><td>［追加］
26)　pd_pth_trace_max = 通信トレース格納最大数
～<符号なし整数>((1024～8388608))
トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。
《指定値の目安》
通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。
《オペランドの省略値》
このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。
《注意事項》
通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。</td><td>pd_pth_trace_max</td></tr>
<tr><td>［追加］
29)　pd_java_castoff = Y｜N
HiRDB のフロントエンドサーバプロセスで，Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。
Y：Java 仮想マシンを起動した場合，UAP の切り離し時にサーバプロセスを終了します。
N：Java 仮想マシンを起動してもサーバプロセスが終了しません。
省略した場合は，サーバ共通定義の pd_java_castoff オペランドが解釈値となります。
《指定値のチューニング方法》
Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。
Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に，pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。
<table>
<tr><th>#</th><th>懸案</th></tr>
<tr><td>1</td><td>Java仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。</td></tr>
<tr><td>2</td><td>Java仮想マシンを使用しないコネクションで実行するSQLでも，別のコネクションでJava仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多いSQLを実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。
なお，Java仮想マシンの機能についてはJava仮想マシンのドキュメントのオプションを参照してください。</td></tr>
</table></td><td>pd_java_castoff</td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>《注意事項》
Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。
《ほかのオペランドとの関連》
このオペランドは次のオペランドと関連があります。
• pd_process_count</td><td></td></tr>
<tr><td>《注意事項》
[訂正前]
Hub 最適化情報定義ファイルが複数ある場合の注意事項です。ここで指定した条件が一致するすべての Hub 最適化情報定義ファイルが適用されます。優先順位は-s の指定，-d 及び-v の指定，-d だけの指定の順になります。
[訂正後]
• -s(外部サーバ名)を指定する場合の注意事項
以下に示す文字を含む外部サーバ名は指定できません。指定した場合，表識別子に外部表を指定した SQL の動作を保証できません。
・半角空白
・半角ハイフン(-)
・¥
・@
・#
上記文字を含む外部サーバ名を CREATE SERVER に指定し，Hub の最適化を外部サーバごとに行う場合は，-s オプションに上記文字を含む外部サーバ名を指定した pdhubopt を省略し，%PDCONFPATH%下に「外部サーバ名.opt」のファイル名称で Hub 最適化情報定義ファイルを格納してください。
• Hub 最適化情報定義ファイルが複数ある場合の注意事項ここで指定した条件が一致するすべての Hub 最適化情報定義ファイルが適用されます。優先順位は-s の指定，-d 及び-v の指定，-d だけの指定の順になります。</td><td>pdhubopt</td></tr>
<tr><td>[追加]
<table><tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr><tr><td>24</td><td>〔set pd_pth_trace_max = 通信トレース格納最大数〕</td><td>トラブルシュート情報</td></tr><tr><td>28</td><td>〔set pd_java_castoff = Y｜N〕</td><td>Java</td></tr></table></td><td>－</td></tr>
<tr><td>[訂正前]
UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。
[訂正後]
UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。
また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。</td><td>pd_spd_syncpoint_skip_limit</td></tr>
</table>

追加・変更内容	関連するオペランド
この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。	
《注意事項》 ［削除］ • HiRDB の開始（再開始も含みます）後，最初のシンクポイントダンプの有効化以降にこのオペランドの指定値が適用されます。	pd_spd_syncpoint_skip_limit
［訂正前］ **「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数** ［訂正後］ **「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数** 係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。	pd_lck_pool_size
［追加］ **24)　pd_pth_trace_max = 通信トレース格納最大数** **～<符号なし整数>((1024～8388608))** トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。 《指定値の目安》 通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。 《オペランドの省略値》 このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。 《注意事項》 通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。	pd_pth_trace_max
［追加］ **28)　pd_java_castoff = Y｜N** HiRDB のディクショナリサーバプロセスで, Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。 **Y**：Java 仮想マシンを起動した場合，トランザクション決着時，又は UAP の切り離し時にサーバプロセスを終了します。 **N**：Java 仮想マシンを起動してもサーバプロセスが終了しません。 省略した場合は，サーバ共通定義の pd_java_castoff オペランドが解釈値となります。 《指定値のチューニング方法》 Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。 Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に，pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。	pd_java_castoff

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>
<table>
<tr><th>#</th><th>懸案</th></tr>
<tr><td>1</td><td>Java仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。</td></tr>
<tr><td>2</td><td>Java仮想マシンを使用しないコネクションで実行するSQLでも，別のコネクションでJava仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多いSQLを実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。
なお，Java仮想マシンの機能についてはJava仮想マシンのドキュメントのオプションを参照してください。</td></tr>
</table>
《注意事項》
Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。
《ほかのオペランドとの関連》
このオペランドは次のオペランドと関連があります。
• pd_process_count
</td><td></td></tr>
<tr><td>
[追加]
<table>
<tr><th>番号</th><th>形式</th><th>オペランドの分類</th></tr>
<tr><td>24</td><td>〔set pd_pth_trace_max = 通信トレース格納最大数〕</td><td>トラブルシュート情報</td></tr>
<tr><td>29</td><td>〔set pd_java_castoff = Y｜N〕</td><td>Java</td></tr>
</table>
</td><td>－</td></tr>
<tr><td>
[訂正前]
UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。
[訂正後]
UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。
また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。
この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。
</td><td>pd_spd_syncpoint_skip_limit</td></tr>
<tr><td>
《注意事項》
[削除]
• HiRDB の開始（再開始も含みます）後，最初のシンクポイントダンプの有効化以降にこのオペランドの指定値が適用されます。
</td><td>pd_spd_syncpoint_skip_limit</td></tr>
<tr><td>
[訂正前]
「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数
[訂正後]
「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数
係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。
</td><td>pd_lck_pool_size</td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>[追加]
24)　pd_pth_trace_max = 通信トレース格納最大数
～<符号なし整数>((1024～8388608))
トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。
《指定値の目安》
通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。
《オペランドの省略値》
このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。
《注意事項》
通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。</td><td>pd_pth_trace_max</td></tr>
<tr><td>[追加]
29)　pd_java_castoff = Y｜N
HiRDB のバックエンドサーバプロセスで, Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。
Y：Java 仮想マシンを起動した場合，トランザクション決着時，又は UAP の切り離し時にサーバプロセスを終了します。
N：Java 仮想マシンを起動してもサーバプロセスが終了しません。
省略した場合は，サーバ共通定義の pd_java_castoff オペランドが解釈値となります。
《指定値のチューニング方法》
Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。
Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に，pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。
<table><tr><th>#</th><th>懸案</th></tr><tr><td>1</td><td>Java仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。</td></tr><tr><td>2</td><td>Java仮想マシンを使用しないコネクションで実行するSQLでも，別のコネクションでJava仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多いSQLを実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。
なお，Java仮想マシンの機能についてはJava仮想マシンのドキュメントのオプションを参照してください。</td></tr></table>
《注意事項》
Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。
《ほかのオペランドとの関連》
このオペランドは次のオペランドと関連があります。</td><td>pd_java_castoff</td></tr>
</table>

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>• pd_process_count</td><td></td></tr>
<tr><td>(20) RD エリアに関するオペランド
[追加]
<table><tr><th>オペランドの内容</th><th>オペランド名</th></tr><tr><td>RDエリアの自動増分機能を適用している場合，増分領域を初期化するか否かを指定します。</td><td>pd_rdarea_expand_format</td></tr></table></td><td>－</td></tr>
<tr><td>(32) トラブルシュート情報に関するオペランド
[追加]
<table><tr><th>オペランドの内容</th><th>オペランド名</th></tr><tr><td>通信トレース格納最大数を指定します。</td><td>pd_pth_trace_max</td></tr></table></td><td>－</td></tr>
<tr><td>(19) PURGE TABLE
[訂正前]
<table><tr><th colspan="2">条件</th><th>排他要求数</th></tr><tr><td colspan="2">HiRDB/シングルサーバの場合</td><td>［前処理］※1＋対象RDエリア数※3＋3＋表の使用中セグメント数＋Σ（インデクスの使用中セグメント数＋2）※2＋Σ（↑HiRDBファイルにLOBデータを格納しているセグメント数÷64000↑）※4</td></tr><tr><td rowspan="2">HiRDB/パラレルサーバの場合</td><td>フロントエンドサーバ及びディクショナリサーバ</td><td>［前処理］※1</td></tr><tr><td>バックエンドサーバ</td><td>対象RDエリア数※3＋3＋表の使用中セグメント数＋Σ（インデクスの使用中セグメント数＋2）※2＋Σ（↑HiRDBファイルにLOBデータを格納しているセグメント数÷64000↑）※4</td></tr></table>
注※1
前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。
注※2
インデクスの数だけ繰り返し加算します。
注※3
対象となる表，インデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。
注※4
LOB 列と LOB 属性の LOB 用 RD エリアの HiRDB ファイル数だけ繰り返し加算します。
[訂正後]</td><td>－</td></tr>
</table>

追加・変更内容	関連するオペランド

条件	排他要求数
HiRDB/シングルサーバの場合	［前処理］※1＋対象RDエリア数※3＋3＋表の使用中セグメント数×2＋Σ（（インデクスの使用中セグメント数×2）＋2）※2＋ Σ（↑HiRDBファイルにLOBデータを格納しているセグメント数÷64000↑）※4 pd_check_pendingオペランドにUSEを指定，又は省略した場合に加算します。 ＋4 対象表が被参照表の場合にさらに加算します。 ＋3＋対象表の表格納用RDエリア数＋対象表を参照する参照表数×2＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5 ＋Σ（参照表の対象RDエリア数※3＋2）※5 対象表が分割表の場合の場合にさらに加算します。 ＋1＋対象表の表格納用RDエリア数 対象表を参照する参照表が分割表の場合にさらに加算します。 ＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5 対象表が参照表，又は検査制約を定義した表の場合にさらに加算します。 ＋1＋対象表に定義した参照制約の数 ＋1＋対象表に定義した検査制約の数

HiRDB/パラレルサーバの場合	フロントエンドサーバ	［前処理］※1
	ディクショナリサーバ	［前処理］※1 pd_check_pendingオペランドにUSEを指定，又は省略した場合に加算します。 ＋4 対象表が被参照表の場合にさらに加算します。 ＋3＋対象表の表格納用RDエリア数＋対象表を参照する参照表数×2＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5 対象表が分割表の場合の場合にさらに加算します。 ＋1＋対象表の表格納用RDエリア数 対象表を参照する参照表が分割表の場合にさらに加算します。 ＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5 対象表が参照表，又は検査制約を定義した表の場合にさらに加算します。 ＋1＋対象表に定義した参照制約の数 ＋1＋対象表に定義した検査制約の数
	バックエンドサーバ	対象RDエリア数※3＋3＋表の使用中セグメント数×2＋Σ（（インデクスの使用中セグメント数×2）＋2）※2＋ Σ（↑HiRDBファイルにLOBデータを格納しているセグメント数÷64000↑）※4 pd_check_pendingオペランドにUSEを指定，又は省略した場合，且つ対象表が被参照表の場合に加算します。 ＋Σ（参照表の対象RDエリア数※3＋2）※5

注※1

前処理の排他要求数については，「(25)　前処理の排他要求数」を参照してください。

注※2

インデクスの数だけ繰り返し加算します。

注※3

対象となる表，インデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

注※4

LOB 列と LOB 属性の LOB 用 RD エリアの HiRDB ファイル数だけ繰り返し加算します。

<table>
<tr><th>追加・変更内容</th><th>関連するオペランド</th></tr>
<tr><td>注※5
対象表を参照する参照表数だけ繰り返し加算します。</td><td></td></tr>
<tr><td>[追加]
<table><tr><th>オペランド名</th><th>文法のチェック</th><th>ファイルのチェック</th><th>アクセス権限のチェック</th><th>重複指定のチェック</th><th>ホスト名のチェック</th><th>サーバマシン間のチェック</th></tr><tr><td>pd_pth_trace_max</td><td>○</td><td>×</td><td>×</td><td>×</td><td>×</td><td>○</td></tr></table></td><td>－</td></tr>
<tr><td>[訂正前]
<table><tr><th>オペランド名</th><th>文法のチェック</th><th>ファイルのチェック</th><th>アクセス権限のチェック</th><th>重複指定のチェック</th><th>ホスト名のチェック</th><th>サーバマシン間のチェック</th></tr><tr><td>SHMMAX</td><td>○</td><td>×</td><td>×</td><td>×</td><td>×</td><td>○</td></tr><tr><td>TZ</td><td>○</td><td>×</td><td>×</td><td>×</td><td>×</td><td>○</td></tr></table>[訂正後]
<table><tr><th>オペランド名</th><th>文法のチェック</th><th>ファイルのチェック</th><th>アクセス権限のチェック</th><th>重複指定のチェック</th><th>ホスト名のチェック</th><th>サーバマシン間のチェック</th></tr><tr><td>SHMMAX</td><td>×</td><td>×</td><td>×</td><td>×</td><td>×</td><td>×</td></tr><tr><td>TZ</td><td>×</td><td>×</td><td>×</td><td>×</td><td>×</td><td>×</td></tr></table></td><td>－</td></tr>
</table>

変更内容(3020-6-353-40)　HiRDB Version 8 08-05

追加・変更内容	関連するオペランド
HiRDB が使用する共用メモリをメモリ上に固定できるようにしました。これによって，共用メモリのページングが防止できるため，アクセス性能が向上します。	pd_shmpool_attribute pd_dbbuff_attribute pd_max_resident_rdarea_no SHMMAX
表の空き領域の再利用機能を使用している場合に，監視不要なセグメント使用率通知メッセージの出力を抑止できるようにしました。	pd_rdarea_warning_point_msgout
バイナリデータの分岐行のページの LRU 管理を抑止できるようにしました。これによって，グローバルバッファにキャッシュされた内容がバイナリデータの読み込みによってメモリから追い出されるのを回避できます。	pd_dbbuff_lru_option pd_dbbuff_binary_data_lru
ユティリティで使用するファイルの入出力処理で使用するバッファサイズを指定できるようにしました。	pd_utl_file_buff_size
システムログファイルの空き容量不足を検知した場合，HiRDB が自動的にシステムログファイルを拡張できるようにしました（システムログファイルの自動拡張機能）。	pd_log_auto_expand_size
pd_large_file_use オペランドの省略値を N から Y に変更しました。	pd_large_file_use
データベース中でデータを呼び出すごとに一連の整数値を返す順序数生成子を追加しました（自動採番機能）。	pd_max_access_tables pd_max_access_tables_wrn_pnt pd_table_def_cache_size
HiRDB/シングルサーバの場合，HiRDB サーバに対する最大同時接続数を 3,000 まで拡大しました。これに伴い，関連するオペランドの指定値が変更になりました。	pd_max_users pd_max_commit_write_reclaim_no

追加・変更内容	関連するオペランド
	pdcltgrp -u pd_process_count
排他待ち限界経過時間をクライアント側でも指定できるようにしました。これに伴い，PDLCKWAITTIME についての説明を追加しました。	pd_watch_time pd_lck_wait_timeout pd_lck_deadlock_check
Windows Server 2008 のクラスタソフトウェア（MSFC）の説明を追加しました。	－
クライアント環境定義 PDLOCKLIMIT の最大排他資源要求数の上限値を 200,000,000 まで拡大しました。	PDLOCKLIMIT
ビュー定義，および WITH 句の導出問合せ中に，抽象データ型の指定，およびそれらのビューや問合せを指定できるようにしました (プラグインが提供する抽象データ型を含みます)。	pd_view_def_cache_size
次に示す定義系 SQL の排他資源数の見積もり式を追加しました。 • ALTER PROCEDURE • ALTER ROUTINE • ALTER TABLE • ALTER TRIGGER • COMMENT • CREATE AUDIT • DROP AUDIT • GRANT AUDIT • GRANT CONNECT • GRANT DBA • GRANT RDAREA • GRANT SCHEMA • GRANT アクセス権限 • REVOKE CONNECT • REVOKE DBA • REVOKE RDAREA • REVOKE SCHEMA • REVOKE アクセス権限	－

変更内容(3020-6-353-30)　HiRDB Version 8 08-04

追加・変更内容	関連するオペランド
DECIMAL 型の精度を拡張し，38 けたまで定義できるようになりました。また，精度 20 けた以上の DECIMAL 型の列にインデクスを定義できるようにしました。	pd_sql_dec_op_maxprec
IN 述語の右辺に指定する行値構成子の数(左辺の値と比較できる値の個数)を 255 から 30000 に拡張しました。	pd_apply_search_ats_num pd_hub_opt_in_value_num
SQLSTATE を細分化できるようにしました。	pd_standard_sqlstate
RD エリアの自動増分に次の機能を追加しました。 ・自動増分によって HiRDB ファイルシステム領域サイズの上限を超える場合, HiRDB ファイルシステム領域の上限を自動的に拡張するようにしました。	pdwork pd_rdarea_warning_point pd_rdarea_extension_timing

追加・変更内容	関連するオペランド
・HiRDB ファイルの使用率やエクステント数を監視して，警告メッセージを出力できるようにしました。 ・増分する契機を指定できるようにしました。	
syslogfile に出力する HiRDB の任意のメッセージの重要度を変更できるようにしました。これによって，syslogfile に出力するメッセージをエラーやワーニングだけで絞り込んでいる環境であっても，HiRDB のインフォメーションメッセージを JP1/AJS2 などでイベント監視できます。	pdmlgput
HiRDB の稼働プラットフォームに Windows Server 2008 を追加しました。	－
SQL 拡張最適化オプションに，値式に対する結合条件適用機能を追加しました。これによって，値式を含む結合条件しかない場合，アクセスパスが直積からネストループジョイン，ハッシュジョイン，またはマージジョインになり，SQL 実行の高速化が期待できます。	pd_additional_optimize_level
影響分散スタンバイレス型系切り替え機能を使用している場合，HA グループを複数定義できるようにしました。	pdhagroup
ビュー解析情報用バッファ長の見積もり式を追加しました。	pd_view_def_cache_size
SELECT 文，UPDATE 文，および DELETE 文に，アクセスする RD エリア名を指定できるようにしました。これによって，マルチフロントエンドサーバで表の横分割を行っている場合，複数の RD エリアに対して並列にアクセスできるようになり，サーバマシンの負荷が分散できるようになります。	pd_table_def_cache_size
インデクスの名称を変更できるようにしました。	－

はじめに

このマニュアルは，プログラムプロダクト スケーラブルデータベースサーバ HiRDB Version 8 のシステム定義について説明したものです。

■ 対象読者

HiRDB Version 8（以降，HiRDB と表記します）を使ってリレーショナルデータベースシステムを構築または運用する方々を対象にしています。

このマニュアルは次に示す知識があることを前提に説明しています。

- Windows のシステム管理の基礎的な知識
- SQL の基礎的な知識

また，このマニュアルは，次に示すマニュアルを前提としていますので，あらかじめお読みいただくことをお勧めします。

- 「HiRDB Version 8 システム導入・設計ガイド」
- 「HiRDB Version 8 システム運用ガイド」

■ 関連マニュアル

このマニュアルの関連マニュアルを次に示します。必要に応じてお読みください。

HiRDB（Windows 用マニュアル）

- HiRDB Version 8 解説（Windows(R)用）（3020-6-351）
- HiRDB Version 8 システム導入・設計ガイド（Windows(R)用）（3020-6-352）
- HiRDB Version 8 システム運用ガイド（Windows(R)用）（3020-6-354）
- HiRDB Version 8 コマンドリファレンス（Windows(R)用）（3020-6-355）
- HiRDB ファーストステップガイド（Windows(R)用）（3020-6-054）

HiRDB（UNIX 用マニュアル）

- HiRDB Version 8 解説（UNIX(R)用）（3000-6-351）
- HiRDB Version 8 システム導入・設計ガイド（UNIX(R)用）（3000-6-352）
- HiRDB Version 8 システム定義（UNIX(R)用）（3000-6-353）
- HiRDB Version 8 システム運用ガイド（UNIX(R)用）（3000-6-354）
- HiRDB Version 8 コマンドリファレンス（UNIX(R)用）（3000-6-355）
- インナレプリカ機能 HiRDB Staticizer Option Version 8（3000-6-363）
- HiRDB Version 8 ディザスタリカバリシステム　構築・運用ガイド（3000-6-364）
- HiRDB ファーストステップガイド（UNIX(R)用）（3000-6-254）

HiRDB（Windows，UNIX 共通マニュアル）

- HiRDB Version 8 UAP 開発ガイド（3020-6-356）
- HiRDB Version 8 SQL リファレンス（3020-6-357）
- HiRDB Version 8 メッセージ（3020-6-358）
- HiRDB Version 8 セキュリティガイド（3020-6-359）
- HiRDB Version 8 XDM/RD E2 接続機能（3020-6-365）
- HiRDB Version 8 バッチ高速化機能（3020-6-368）
- HiRDB データ連動機能 HiRDB Datareplicator Version 8（3020-6-360）

- HiRDB データ連動拡張機能 HiRDB Datareplicator Extension Version 8（3020-6-361）
- データベース抽出・反映サービス機能 HiRDB Dataextractor Version 8（3020-6-362）
- HiRDB 全文検索プラグイン HiRDB Text Search Plug-in Version 8（3020-6-375）
- HiRDB XML 拡張機能 HiRDB XML Extension Version 8（3020-6-376）

なお，本文中で使用している HiRDB Version 8 のマニュアル名は，（UNIX(R)用）または（Windows(R)用）を省略して表記しています。使用しているプラットフォームに応じて UNIX 用または Windows 用のマニュアルを参照してください。

関連製品

- HiRDB External Data Access Version 8（3020-6-366）
- Hitachi HA Toolkit（3000-9-115）

■ 利用者ごとの関連マニュアル

HiRDB のマニュアルをご利用になる場合，利用者ごとに次のようにお読みください。

また，より理解を深めるために，左側のマニュアルから順にお読みいただくことをお勧めします。

注※１　レプリケーション機能を使用してデータ連携をする場合にお読みください。
注※２　UNIX用マニュアルです。Windows用はありません。
注※３　インナレプリカ機能を使用する場合にお読みください。
注※４　ディザスタリカバリシステムを構築する場合にお読みください。
注※５　インメモリデータ処理によるバッチ高速化を行う場合にお読みください。
注※６　OLTPシステムと連携する場合は必ずお読みください。
注※７　XDM/RD E2 接続機能を使用して，XDM/RD E2のデータベースを操作する場合にお読みください。

■ このマニュアルでの表記

このマニュアルでは製品名称および名称について次のように表記しています。ただし，それぞれのプログラムについての表記が必要な場合はそのまま表記しています。

製品名称または名称	表記	
HiRDB/Single Server Version 8	HiRDB/シングルサーバ	HiRDB または HiRDB サーバ

<table>
<tr><th>製品名称または名称</th><th colspan="2">表記</th></tr>
<tr><td>HiRDB/Single Server Version 8(64)</td><td></td><td rowspan="3"></td></tr>
<tr><td>HiRDB/Parallel Server Version 8</td><td rowspan="2">HiRDB/パラレルサーバ</td></tr>
<tr><td>HiRDB/Parallel Server Version 8(64)</td></tr>
<tr><td>HiRDB/Developer's Kit Version 8</td><td rowspan="2">HiRDB/Developer's Kit</td><td rowspan="4">HiRDB クライアント</td></tr>
<tr><td>HiRDB/Developer's Kit Version 8(64)</td></tr>
<tr><td>HiRDB/Run Time Version 8</td><td rowspan="2">HiRDB/Run Time</td></tr>
<tr><td>HiRDB/Run Time Version 8(64)</td></tr>
<tr><td>HiRDB Datareplicator Version 8</td><td colspan="2">HiRDB Datareplicator</td></tr>
<tr><td>HiRDB Dataextractor Version 8</td><td colspan="2">HiRDB Dataextractor</td></tr>
<tr><td>HiRDB Text Search Plug-in Version 8</td><td colspan="2">HiRDB Text Search Plug-in</td></tr>
<tr><td>HiRDB XML Extension Version 8</td><td colspan="2">HiRDB XML Extension</td></tr>
<tr><td>HiRDB Spatial Search Plug-in Version 3</td><td colspan="2">HiRDB Spatial Search Plug-in</td></tr>
<tr><td>HiRDB Staticizer Option Version 8</td><td colspan="2">HiRDB Staticizer Option</td></tr>
<tr><td>HiRDB LDAP Option Version 8</td><td colspan="2">HiRDB LDAP Option</td></tr>
<tr><td>HiRDB Advanced Partitioning Option Version 8</td><td colspan="2">HiRDB Advanced Partitioning Option</td></tr>
<tr><td>HiRDB Advanced High Availability Version 8</td><td colspan="2">HiRDB Advanced High Availability</td></tr>
<tr><td>HiRDB Non Recover Front End Server Version 8</td><td colspan="2">HiRDB Non Recover FES</td></tr>
<tr><td>HiRDB Disaster Recovery Light Edition Version 8</td><td colspan="2">HiRDB Disaster Recovery Light Edition</td></tr>
<tr><td>HiRDB Accelerator Version 8</td><td colspan="2">HiRDB Accelerator</td></tr>
<tr><td>HiRDB External Data Access Version 8</td><td colspan="2">HiRDB External Data Access</td></tr>
<tr><td>HiRDB External Data Access Adapter Version 8</td><td colspan="2">HiRDB External Data Access Adapter</td></tr>
<tr><td>HiRDB Adapter for XML - Standard Edition</td><td colspan="2" rowspan="2">HiRDB Adapter for XML</td></tr>
<tr><td>HiRDB Adapter for XML - Enterprise Edition</td></tr>
<tr><td>HiRDB Control Manager</td><td colspan="2">HiRDB CM</td></tr>
<tr><td>HiRDB Control Manager Agent</td><td colspan="2">HiRDB CM Agent</td></tr>
<tr><td>Hitachi TrueCopy</td><td colspan="2" rowspan="4">TrueCopy</td></tr>
<tr><td>Hitachi TrueCopy basic</td></tr>
<tr><td>TrueCopy</td></tr>
<tr><td>TrueCopy remote replicator</td></tr>
<tr><td>JP1/Automatic Job Management System 2</td><td colspan="2">JP1/AJS2</td></tr>
<tr><td>JP1/Automatic Job Management System 2 - Scenario Operation</td><td colspan="2">JP1/AJS2-SO</td></tr>
</table>

製品名称または名称	表記	
JP1/Cm2/Extensible SNMP Agent	JP1/ESA	
JP1/Cm2/Extensible SNMP Agent for Mib Runtime		
JP1/Cm2/Network Node Manager	JP1/NNM	
JP1/Integrated Management - Manager	JP1/Integrated Management または JP1/IM	
JP1/Integrated Management - View		
JP1/Magnetic Tape Access	EasyMT	
EasyMT		
JP1/Magnetic Tape Library	MTguide	
JP1/NETM/Audit - Manager	JP1/NETM/Audit	
JP1/NETM/DM	JP1/NETM/DM	
JP1/NETM/DM Manager		
JP1/Performance Management	JP1/PFM	
JP1/Performance Management - Agent Option for HiRDB	JP1/PFM-Agent for HiRDB	
JP1/Performance Management - Agent Option for Platform	JP1/PFM-Agent for Platform	
JP1/Performance Management/SNMP System Observer	JP1/SSO	
JP1/VERITAS NetBackup BS v4.5	NetBackup	
JP1/VERITAS NetBackup v4.5		
JP1/VERITAS NetBackup BS V4.5 Agent for HiRDB License	JP1/VERITAS NetBackup Agent for HiRDB License	
JP1/VERITAS NetBackup V4.5 Agent for HiRDB License		
JP1/VERITAS NetBackup 5 Agent for HiRDB License		
OpenTP1/Server Base Enterprise Option	TP1/EE	
Virtual-storage Operating System 3/Forefront System Product	VOS3/FS	VOS3
Virtual-storage Operating System 3/Leading System Product	VOS3/LS	
Extensible Data Manager/Base Extended Version 2 XDM 基本プログラム XDM/BASE E2	XDM/BASE E2	
XDM/Data Communication and Control Manager 3 XDM データコミュニケーションマネジメントシステム XDM/DCCM3	XDM/DCCM3	
XDM/Relational Database リレーショナルデータベースシステム XDM/RD	XDM/RD	XDM/RD
XDM/Relational Database Extended Version 2 リレーショナルデータベースシステム XDM/RD E2	XDM/RD E2	
VOS3 Database Connection Server	DB コネクションサーバ	

製品名称または名称	表記	
BEA WebLogic Server	WebLogic Server	
DB2 Universal Database for OS/390 Version 6	DB2	
DNCWARE ClusterPerfect（Linux 版）	ClusterPerfect	
Microsoft(R) Office Excel	Microsoft Excel または Excel	
Microsoft(R) Visual C++(R)	Visual C++または C++言語	
Oracle8i	ORACLE	
Oracle9i		
Oracle 10g		
Sun Java™ System Directory Server	Sun Java System Directory Server またはディレクトリサーバ	
HP-UX 11i V2 (IPF)	HP-UX または HP-UX (IPF)	
HP-UX 11i V3 (IPF)		
AIX 5L V5.1	AIX 5L	AIX
AIX 5L V5.2		
AIX 5L V5.3		
AIX V6.1	AIX V6.1	
AIX V7.1	AIX V7.1	
Linux(R)	Linux	
Red Hat Linux	Red Hat Linux	Linux
Red Hat Enterprise Linux	Red Hat Enterprise Linux	
Red Hat Enterprise Linux AS 3 (IPF)	Linux (IPF)	
Red Hat Enterprise Linux AS 4 (IPF)		
Red Hat Enterprise Linux 5.1 Advanced Platform (Intel Itanium)		
Red Hat Enterprise Linux 5.1 (Intel Itanium)		
Red Hat Enterprise Linux 5.2 Advanced Platform (Intel Itanium)		
Red Hat Enterprise Linux 5.2 (Intel Itanium)		
Red Hat Enterprise Linux AS 3(AMD64 & Intel EM64T)	Linux (EM64T)	
Red Hat Enterprise Linux AS 4(AMD64 & Intel EM64T)		
Red Hat Enterprise Linux ES 4(AMD64 & Intel EM64T)		
Red Hat Enterprise Linux 5.1 Advanced Platform (AMD/Intel 64)		
Red Hat Enterprise Linux 5.1 (AMD/Intel 64)		

製品名称または名称	表記	
Red Hat Enterprise Linux 5.2 Advanced Platform (AMD/Intel 64)		
Red Hat Enterprise Linux 5.2 (AMD/Intel 64)		
Red Hat Enterprise Linux AS 4(AMD64 & Intel EM64T)	Linux AS 4	
Red Hat Enterprise Linux AS 4(x86)		
Red Hat Enterprise Linux ES 4(AMD64 & Intel EM64T)	Linux ES 4	
Red Hat Enterprise Linux ES 4(x86)		
Red Hat Enterprise Linux 5.1 Advanced Platform (x86)	Linux 5.1	Linux 5
Red Hat Enterprise Linux 5.1 (x86)		
Red Hat Enterprise Linux 5.1 Advanced Platform (AMD/Intel 64)		
Red Hat Enterprise Linux 5.1 (AMD/Intel 64)		
Red Hat Enterprise Linux 5.1 Advanced Platform (Intel Itanium)		
Red Hat Enterprise Linux ES 4(x86)		
Red Hat Enterprise Linux 5.2 Advanced Platform (x86)	Linux 5.2	
Red Hat Enterprise Linux 5.2 (x86)		
Red Hat Enterprise Linux 5.2 Advanced Platform (AMD/Intel 64)		
Red Hat Enterprise Linux 5.2 (AMD/Intel 64)		
Red Hat Enterprise Linux 5.2 Advanced Platform (Intel Itanium)		
Red Hat Enterprise Linux 5.2 (Intel Itanium)		
turbolinux 7 Server for AP8000	Linux for AP8000	
Microsoft(R) Windows NT(R) Workstation Operating System Version 4.0	Windows NT	
Microsoft(R) Windows NT(R) Server Network Operating System Version 4.0		
Microsoft(R) Windows(R) 2000 Professional Operating System	Windows 2000	
Microsoft(R) Windows(R) 2000 Server Operating System		
Microsoft(R) Windows(R) 2000 Datacenter Server Operating System		
Microsoft(R) Windows(R) 2000 Advanced Server Operating System		
Microsoft(R) Windows(R) 2000 Advanced Server Operating System	Windows 2000 Advanced Server	
Microsoft(R) Windows Server(R) 2003, Standard Edition	Windows Server 2003 Standard Edition	Windows Server 2003

製品名称または名称	表記	
Microsoft(R) Windows Server(R) 2003, Enterprise Edition	Windows Server 2003 Enterprise Edition	
Microsoft(R) Windows Server(R) 2003, Standard x64 Edition	Windows Server 2003 Standard x64 Edition	
Microsoft(R) Windows Server(R) 2003, Enterprise x64 Edition	Windows Server 2003 Enterprise x64 Edition	
Microsoft(R) Windows Server(R) 2003 R2, Standard Edition	Windows Server 2003 R2	
Microsoft(R) Windows Server(R) 2003 R2, Enterprise Edition		
Microsoft(R) Windows Server(R) 2003 R2, Standard x64 Edition		
Microsoft(R) Windows Server(R) 2003 R2, Enterprise x64 Edition		
Microsoft(R) Windows Server(R) 2003 R2, Standard x64 Edition	Windows Server 2003 R2 x64 Editions	
Microsoft(R) Windows Server(R) 2003 R2, Enterprise x64 Edition		
Microsoft(R) Windows Server(R) 2008 Standard	Windows Server 2008 Standard	Windows Server 2008
Microsoft(R) Windows Server(R) 2008 Enterprise	Windows Server 2008 Enterprise	
Microsoft(R) Windows Server(R) 2008 R2 Standard (x64)	Windows Server 2008 R2	
Microsoft(R) Windows Server(R) 2008 R2 Enterprise (x64)		
Microsoft(R) Windows Server(R) 2008 R2 Datacenter (x64)		
Microsoft(R) Windows Server(R) 2008 Standard (x64)	Windows Server 2008 (x64)	
Microsoft(R) Windows Server(R) 2008 Enterprise (x64)		
Microsoft(R) Windows Server(R) 2003, Standard x64 Edition	Windows Server 2003 x64 Editions	Windows (x64)
Microsoft(R) Windows Server(R) 2003, Enterprise x64 Edition		
Microsoft(R) Windows Server(R) 2003 R2, Standard x64 Edition		
Microsoft(R) Windows Server(R) 2003 R2, Enterprise x64 Edition		
Microsoft(R) Windows(R) XP Professional x64 Edition	Windows XP x64 Edition	
Microsoft(R) Windows Server(R) 2003, Enterprise x64 Edition	Windows Server 2003 (IPF)	Windows(IPF)
Microsoft(R) Windows(R) XP Professional x64 Edition	Windows XP x64 Edition	Windows XP
Microsoft(R) Windows(R) XP Professional Operating System	Windows XP Professional	
Microsoft(R) Windows(R) XP Home Edition Operating System	Windows XP Home Edition	

製品名称または名称	表記	
Microsoft(R) Windows Vista(R) Home Basic	Windows Vista Home Basic	Windows Vista
Microsoft(R) Windows Vista(R) Home Premium	Windows Vista Home Premium	
Microsoft(R) Windows Vista(R) Ultimate	Windows Vista Ultimate	
Microsoft(R) Windows Vista(R) Business	Windows Vista Business	
Microsoft(R) Windows Vista(R) Enterprise	Windows Vista Enterprise	
Microsoft(R) Windows Vista(R) Home Basic (x64)	Windows Vista (x64)	
Microsoft(R) Windows Vista(R) Home Premium (x64)		
Microsoft(R) Windows Vista(R) Ultimate (x64)		
Microsoft(R) Windows Vista(R) Business (x64)		
Microsoft(R) Windows Vista(R) Enterprise (x64)		
Microsoft(R) Windows Vista(R) Ultimate (x64)	Windows Vista Ultimate (x64)	
Microsoft(R) Windows Vista(R) Business (x64)	Windows Vista Business (x64)	
Microsoft(R) Windows Vista(R) Enterprise (x64)	Windows Vista Enterprise (x64)	
Microsoft(R) Windows(R) 7 Home Premium	Windows 7 Home Premium	Windows 7
Microsoft(R) Windows(R) 7 Professional	Windows 7 Professional	
Microsoft(R) Windows(R) 7 Enterprise	Windows 7 Enterprise	
Microsoft(R) Windows(R) 7 Ultimate	Windows 7 Ultimate	
Microsoft(R) Windows(R) 7 Home Premium (x64)	Windows 7 (x64)	
Microsoft(R) Windows(R) 7 Professional (x64)		
Microsoft(R) Windows(R) 7 Enterprise (x64)		
Microsoft(R) Windows(R) 7 Ultimate (x64)		
Microsoft(R) Windows(R) 7 Professional (x64)	Windows 7 Professional (x64)	
Microsoft(R) Windows(R) 7 Enterprise (x64)	Windows 7 Enterprise (x64)	
Microsoft(R) Windows(R) 7 Ultimate (x64)	Windows 7 Ultimate (x64)	
シングルサーバ	SDS	

製品名称または名称	表記
システムマネジャ	MGR
フロントエンドサーバ	FES
ディクショナリサーバ	DS
バックエンドサーバ	BES

- Windows Server 2003 および Windows Server 2008 を総称して Windows Server と表記します。また，Windows 2000，Windows XP，Windows Server，Windows Vista，および Windows 7 を総称して Windows と表記します。
- HiRDB 運用ディレクトリのパスを%PDDIR%と表記します。また，Windows のインストールディレクトリのパスを%windir%と表記します。
- TCP/IP が規定する hosts ファイルを hosts ファイルと表記します。hosts ファイルとは通常，%windir%¥system32¥drivers¥etc¥hosts のことです。

■ このマニュアルで使用する略語

このマニュアルで使用する英略語の一覧を次に示します。

英略語	英 字 の 表 記
ACK	Acknowledgement
ADM	Adaptable Data Manager
ADO	ActiveX Data Objects
ADT	Abstract Data Type
AP	Application Program
API	Application Programming Interface
ASN.1	Abstract Syntax Notation One
BES	Back End Server
BLOB	Binary Large Object
BMP	Basic Multilingual Plane
BOM	Byte Order Mark
CD-ROM	Compact Disc - Read Only Memory
CGI	Common Gateway Interface
CLOB	Character Large Object
CMT	Cassette Magnetic Tape
COBOL	Common Business Oriented Language
CORBA	Common ORB Architecture
CPU	Central Processing Unit
CSV	Comma Separated Values

英略語	英字の表記
DAO	Data Access Object
DAT	Digital Audio Taperecorder
DB	Database
DBM	Database Module
DBMS	Database Management System
DDL	Data Definition Language
DF for Windows NT	Distributing Facility for Windows NT
DF/UX	Distributing Facility / for UNIX
DIC	Dictionary Server
DLT	Digital Linear Tape
DML	Data Manipulate Language
DNS	Domain Name System
DOM	Document Object Model
DS	Dictionary Server
DTD	Document Type Definition
DTP	Distributed Transaction Processing
DWH	Data Warehouse
EUC	Extended UNIX Code
EX	Exclusive
FAT	File Allocation Table
FD	Floppy Disk
FES	Front End Server
FQDN	Fully Qualified Domain Name
FTP	File Transfer Protocol
GUI	Graphical User Interface
HBA	Host Bus Adapter
HD	Hard Disk
HTML	Hyper Text Markup Language
ID	Identification number
IP	Internet Protocol
IPF	Itanium(R) Processor Family

英略語	英字の表記
JAR	Java Archive File
Java VM	Java Virtual Machine
JDBC	Java Database Connectivity
JDK	Java Developer's Kit
JFS	Journaled File System
JFS2	Enhanced Journaled File System
JIS	Japanese Industrial Standard code
JP1	Job Management Partner 1
JRE	Java Runtime Environment
JTA	Java Transaction API
JTS	Java Transaction Service
KEIS	Kanji processing Extended Information System
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
LIP	Loop initialization process
LOB	Large Object
LRU	Least Recently Used
LTO	Linear Tape-Open
LU	Logical Unit
LUN	Logical Unit Number
LVM	Logical Volume Manager
MGR	System Manager
MIB	Management Information Base
MRCF	Multiple RAID Coupling Feature
MSCS	Microsoft Cluster Server
MSFC	Microsoft Failover Cluster
NAFO	Network Adapter Fail Over
NAPT	Network Address Port Translation
NAT	Network Address Translation
NIC	Network Interface Card
NIS	Network Information Service

英略語	英字の表記
NTFS	New Technology File System
ODBC	Open Database Connectivity
OLAP	Online Analytical Processing
OLE	Object Linking and Embedding
OLTP	On-Line Transaction Processing
OOCOBOL	Object Oriented COBOL
ORB	Object Request Broker
OS	Operating System
OSI	Open Systems Interconnection
OTS	Object Transaction Service
PC	Personal Computer
PDM II E2	Practical Data Manager II Extended Version 2
PIC	Plug-in Code
PNM	Public Network Management
POSIX	Portable Operating System Interface for UNIX
PP	Program Product
PR	Protected Retrieve
PU	Protected Update
RAID	Redundant Arrays of Inexpensive Disk
RD	Relational Database
RDB	Relational Database
RDB1	Relational Database Manager 1
RDB1 E2	Relational Database Manager 1 Extended Version 2
RDO	Remote Data Objects
RiSe	Real time SAN replication
RM	Resource Manager
RMM	Resource Manager Monitor
RPC	Remote Procedure Call
SAX	Simple API for XML
SDS	Single Database Server
SGML	Standard Generalized Markup Language

英略語	英字の表記
SJIS	Shift JIS
SNMP	Simple Network Management Protocol
SNTP	Simple Network Time Protocol
SQL	Structured Query Language
SQL/K	Structured Query Language / VOS K
SR	Shared Retrieve
SU	Shared Update
TCP/IP	Transmission Control Protocol / Internet Protocol
TM	Transaction Manager
TMS-4V/SP	Transaction Management System - 4V / System Product
UAP	User Application Program
UOC	User Own Coding
VOS1	Virtual-storage Operating System 1
VOS3	Virtual-storage Operating System 3
VOS K	Virtual-storage Operating System Kindness
WS	Workstation
WWW	World Wide Web
XDM/BASE E2	Extensible Data Manager / Base Extended Version 2
XDM/DF	Extensible Data Manager / Distributing Facility
XDM/DS	Extensible Data Manager / Data Spreader
XDM/RD E2	Extensible Data Manager / Relational Database Extended Version 2
XDM/SD E2	Extensible Data Manager / Structured Database Extended Version 2
XDM/XT	Extensible Data Manager / Data Extract
XFIT	Extended File Transmission program
XML	Extensible Markup Language

■ ログの表記

Windows のイベントビューアで表示されるアプリケーションログをイベントログと表記します。イベントログは，次の方法で参照できます。

〈手順〉

1. ［スタート］－［プログラム］－［管理ツール（共通）］－［イベントビューア］を選択します。
2. ［ログ］－［アプリケーション］を選択します。

アプリケーションログが表示されます。「ソース」の列が「HiRDBSingleServer」または「HiRDBParallelServer」になっているのが HiRDB が出力したメッセージです。

なお，セットアップ識別子を指定してインストールした場合は，「HiRDBSingleServer」または「HiRDBParallelServer」にセットアップ識別子が付いた名称となります。

■ Windows の操作説明で使う表記

Windows の操作説明で使う記号を次に示します。

記　号	意　味
［　］	ボタンやテキストボックスなど，画面に表示されている要素を示します。
［　］－［　］	画面に表示されるメニューやアイコンなどを選択する操作を示します。

Windows の用語「ディレクトリ」と「フォルダ」は，「ディレクトリ」に統一して表記しています。

■ 図中で使用する記号

このマニュアルの図中で使用する記号を，次のように定義します。

●ファイル　●ＣＭＴまたはＤＡＴ　●磁気テープ

●プログラム
またはサーバ　●ネットワーク（ＬＡＮ）

■ このマニュアルで使用する記号

形式および説明で使用する記号を次に示します。ここで説明する文法記述記号は，説明のための記号なので実際には記述しないでください。

記号	意味
〔　〕	この記号で囲まれている項目は省略できます。 （例）pdbuffer 〔-p〕 これは，pdbuffer と指定するか，または pdbuffer -p と指定することを示します。
…	この記号の直前の項目を繰り返して指定できます。 （例）pdbuffer -r RD エリア名 〔，RD エリア名〕… これは，-r オプションの RD エリア名を繰り返し指定できることを示します。
{{　}}	この記号で囲まれた複数の項目を一つの単位として，繰り返して指定できます。 （例）{{pdbuffer -a バッファ名}} これは，pdbuffer -a バッファ名 pdbuffer -a バッファ名 と指定できることを示します。

記号	意味
|	この記号で区切られた項目は選択できます。 （例）pdlogadfg -d sys | spd これは，-d オプションに sys か spd のどちらかを指定できることを示します。
{ }	この記号で囲まれている複数の項目のうちから，一つを選択できます。 （例）pdbuffer ［{-r RD エリア名 | -i 認可識別子．インデクス識別子 | -o}］ これは，-r RD エリア名，-i 認可識別子．インデクス識別子，-o の三つのオプションのうち，どれか一つを指定することを示します。
＿ （下線）	この記号で示す項目は，省略時の解釈値です。 （例）［ pd_rpc_trace = Y | N ］ これは，オペランドの指定を省略した場合，N が仮定されることを示します。
～	この記号の後にユーザ指定値の属性を示します。
《 》	ユーザが指定しなかった場合に仮定される値を示します。
〈 〉	ユーザ指定値の構文要素を示します。
(( ))	ユーザ指定値の指定範囲を示します。

■ このマニュアルで使用する構文要素記号

このマニュアルで使用する構文要素記号を次に示します。

構文要素記号	意味
〈英字〉	アルファベット（A～Z，a～z）と下線（_）
〈英字記号〉	アルファベット（A～Z，a～z）と#，@，¥
〈英数字〉	英字と数字（０～９）
〈英数字記号〉	英字記号と数字
〈符号なし整数〉	数字
〈符号なし 10 進数〉※1	数字（0～9 の並び）ピリオド（.）数字（0～9 の並び）
〈16 進数字〉	数字と（A～F，a～f）
〈識別子〉※2	先頭がアルファベットの英数字列
〈記号名称〉	先頭が英字記号の英数字記号列
〈文字列〉	任意の文字の配列
〈パス名〉※3	英数字，¥，丸括弧，空白，およびピリオド（.）
〈ホスト名〉※4	アルファベット（A～Z，a～z），数字，ピリオド（.），ハイフン（-），下線（_），および@で構成される文字列

注

すべて半角文字を使用してください。また，英字の大文字と小文字は区別されますが，一部区別されない名称もあります。

注※1

ピリオドの前の数字がすべて0の場合，ピリオドより前の0を省略できます。また，ピリオドの後ろの数字がすべて0の場合，ピリオド以降を省略できます。

例1：0.008 → .008

例2：15.000 → 15

注※2

RDエリア名を指定する場合：

・英数字記号，__（下線），または空白を使用してください。ただし，先頭は英字記号を使用してください。

・空白を使用する場合はRDエリア名を引用符（"）で囲んでください。

注※3

パス名は使用しているOSに依存します。丸括弧または空白を使用する場合は，パス名を引用符（"）で囲んでください。

注※4

ホスト名は次に示すどれかの形式で指定できます。

・ホスト名

・IPアドレス（XXX.XXX.XXX.XXX形式）

・FQDN

ただし，同じホスト名を異なる形式で指定しないでください。異なる形式で指定した場合，HiRDBは異なるホスト名として認識します。

また，ループバックアドレスを指定することもできます。

システム定義に指定するホスト名については，ホスト名の登録が必要になります。詳細については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「ホスト名の登録」を参照してください。

■ このマニュアルで使用する計算式の記号

このマニュアルで使用する計算式の記号の意味を次に示します。

記号	内容
↑　↑	計算結果の値の小数点以下を切り上げることを示します。 （例）↑34÷3↑の計算結果は12となります。
↓　↓	計算結果の値の小数点以下を切り捨てることを示します。 （例）↓34÷3↓の計算結果は11となります。
MAX	計算結果の最も大きい値を選ぶことを示します。 （例）MAX(3×6，4+7)の計算結果は18となります。
MIN	計算結果の最も小さい値を選ぶことを示します。 （例）MIN(3×6，4+7)の計算結果は11となります。

■ Windowsのパス名に関する注意

- パス名を絶対パスで指定する場合はドライブ名を指定してください。

（例）C:¥win32app¥hitachi¥hirdb_s¥spool¥tmp

- コマンドの引数，制御文ファイル，およびHiRDBシステム定義ファイル中に空白または丸括弧を含むパス名を指定する場合は，前後を引用符（”）で囲んでください。

（例）pdinit -d "C:¥Program Files(x86)¥hitachi¥hirdb_s¥conf¥mkinit"

ただし，バッチファイルもしくはコマンドプロンプト上で set コマンドを使用して環境変数を設定する場合，またはインストールディレクトリを指定する場合は引用符は不要です。引用符で囲むと，引用符も環境変数の値に含まれます。

（例）set PDCLTPATH=C:¥Program Files¥hitachi¥hirdb_s¥spool

- HiRDB はネットワークドライブのファイルを使用できないため，HiRDB のインストール，および環境構築はローカルドライブで行ってください。また，ユティリティの入出力ファイルなども，ローカルドライブ上のファイルを使用してください。
- パス名には，ショートパス名（例えば，C:¥PROGRA~1 など）は使用しないでください。

■ KB（キロバイト）などの単位表記について

1KB（キロバイト），1MB（メガバイト），1GB（ギガバイト），1TB（テラバイト）はそれぞれ 1,024 バイト，$1{,}024^2$ バイト，$1{,}024^3$ バイト，$1{,}024^4$ バイトです。

■ Version 7 と Version 8 の製品体系の違い

HiRDB Version 8 では，HiRDB Version 7 までオプション製品（HiRDB Object Option および HiRDB High Availability）で提供していた機能を HiRDB の標準機能としました。それに伴い，オプション製品が廃止になりました。

また，Version 8 以降，HiRDB External Data Access および HiRDB External Data Access Adapter は HiRDB シリーズではなく，Cosminexus Version 7 シリーズとなりました。

HiRDB Version 7 と Version 8 の製品体系の違いを次に示します。

注※　UNIX版でだけ使用できる製品です。

目次

1 概要 1

1.1 HiRDB/シングルサーバを利用される方へ 2

1.1.1 HiRDB システム定義の種類 2

1.1.2 HiRDB システム定義の作成方法 2

1.1.3 HiRDB システム定義ファイルの構成 4

1.1.4 各定義の関係 4

1.1.5 HiRDB システム定義（UAP 環境定義を除く）の変更手順 5

1.1.6 UAP 環境定義の追加又は変更手順 7

1.2 HiRDB/パラレルサーバを利用される方へ 8

1.2.1 HiRDB システム定義の種類 8

1.2.2 HiRDB システム定義の作成方法 9

1.2.3 HiRDB システム定義ファイルの構成 10

1.2.4 各定義の関係 12

1.2.5 HiRDB システム定義（UAP 環境定義を除く）の変更手順 13

1.2.6 UAP 環境定義の追加又は変更手順 15

1.2.7 影響分散スタンバイレス型系切り替え機能使用時の HiRDB システム定義の作成 16

1.3 HiRDB システム定義を記述するときの文法規則 17

1.4 オペランド一覧 19

1.4.1 指定値を変更できるオペランド 19

1.4.2 オペランドの優先順位について 32

1.5 バージョンによって省略値が異なるオペランド，及び指定不要になったオペランド 33

2 システム共通定義 37

2.1 オペランドの形式 38

2.2 オペランドの説明 51

2.2.1 システム構成に関するオペランド 51

2.2.2 同時実行最大数に関するオペランド 51

2.2.3 HiRDB の開始方法に関するオペランド 56

2.2.4 縮退起動に関するオペランド 58

2.2.5 HiRDB の処理方式に関するオペランド 60

2.2.6 全面回復処理に関するオペランド 73

2.2.7 トランザクション決着処理に関するオペランド 74

2.2.8 SQL の最適化に関するオペランド 77

2.2.9 絞込み検索に関するオペランド 83

2.2.10 システム監視に関するオペランド 85

2.2.11 SQL 実行時間警告出力機能に関するオペランド 96

2.2.12　拡張 SQL エラー情報出力機能に関するオペランド　98
2.2.13　SQL 予約語削除機能に関するオペランド　99
2.2.14　SQL からのコマンド実行に関するオペランド　100
2.2.15　SQLSTATE の細分化に関するオペランド　100
2.2.16　排他制御に関するオペランド　100
2.2.17　バッファに関するオペランド　107
2.2.18　共用メモリに関するオペランド　108
2.2.19　メッセージログファイルに関するオペランド　110
2.2.20　統計情報に関するオペランド　111
2.2.21　RPC トレース情報に関するオペランド　112
2.2.22　トラブルシュート情報に関するオペランド　113
2.2.23　RD エリアに関するオペランド　124
2.2.24　グローバルバッファに関するオペランド　135
2.2.25　インメモリデータ処理に関するオペランド　141
2.2.26　表又はインデクスの予約数に関するオペランド　142
2.2.27　参照制約及び検査制約に関するオペランド　144
2.2.28　HiRDB ファイルシステム領域に関するオペランド　145
2.2.29　再編成時期予測機能に関するオペランド　147
2.2.30　セキュリティに関するオペランド　147
2.2.31　系切り替え機能に関するオペランド　157
2.2.32　HiRDB Datareplicator に関するオペランド　163
2.2.33　JP1 との連携に関するオペランド　165
2.2.34　ディレクトリサーバ連携機能に関するオペランド　166
2.2.35　HiRDB External Data Access 機能に関するオペランド　167
2.2.36　OLTP に関するオペランド　167
2.2.37　バージョンアップに関するオペランド　167
2.2.38　通信処理に関するオペランド　168
2.2.39　Java™ に関するオペランド　181
2.2.40　外部 C ストアドルーチンに関するオペランド　184
2.2.41　文字コードに関するオペランド　184
2.2.42　日付・時刻に関するオペランド　185
2.2.43　ユニット構成に関するオペランド　186
2.2.44　サーバ構成に関するオペランド　194
2.2.45　グローバルバッファに関するオペランド　202
2.2.46　HA グループに関するオペランド　209
2.2.47　統計情報に関するオペランド　210
2.2.48　クライアントグループに関するオペランド　214
2.2.49　プラグインに関するオペランド　216
2.2.50　共用メモリに関するオペランド　216
2.2.51　日付・時刻に関するオペランド　217

2.2.52　メッセージの出力抑止機能に関するオペランド　217

3 ユニット制御情報定義　219

3.1　オペランドの形式　220

3.2　オペランドの説明　226

3.2.1　システム構成に関するオペランド　226

3.2.2　同時実行最大数に関するオペランド　226

3.2.3　HiRDB の処理方式に関するオペランド　228

3.2.4　全面回復処理に関するオペランド　232

3.2.5　システム監視に関するオペランド　232

3.2.6　SQL 実行時間警告出力機能に関するオペランド　235

3.2.7　拡張 SQL エラー情報出力機能に関するオペランド　236

3.2.8　排他制御に関するオペランド　237

3.2.9　共用メモリに関するオペランド　241

3.2.10　統計情報に関するオペランド　243

3.2.11　RPC トレース情報に関するオペランド　244

3.2.12　トラブルシュート情報に関するオペランド　245

3.2.13　グローバルバッファに関するオペランド　254

3.2.14　ユニット用ステータスファイルに関するオペランド　256

3.2.15　ユニット用ステータスファイル（障害発生時）に関するオペランド　257

3.2.16　セキュリティに関するオペランド　263

3.2.17　系切り替え機能に関するオペランド　267

3.2.18　HiRDB Datareplicator に関するオペランド　274

3.2.19　通信処理に関するオペランド　275

3.2.20　Java に関するオペランド　281

3.2.21　外部 C ストアドルーチンに関するオペランド　283

3.2.22　ワークファイル出力先ディレクトリの変更　283

3.2.23　共用メモリに関するオペランド　284

4 サーバ共通定義　287

4.1　オペランドの形式　288

4.2　オペランドの説明　292

4.2.1　プロセスに関するオペランド　292

4.2.2　作業表に関するオペランド　296

4.2.3　システム監視に関するオペランド　299

4.2.4　排他制御に関するオペランド　302

4.2.5　バッファに関するオペランド　308

4.2.6　共用メモリに関するオペランド　312

4.2.7　RPC トレース情報に関するオペランド　315

4.2.8　トラブルシュート情報に関するオペランド　316

4.2.9　グローバルバッファに関するオペランド　317
4.2.10　セキュリティに関するオペランド　319
4.2.11　Java に関するオペランド　319
4.2.12　システムログファイルに関するオペランド　321
4.2.13　シンクポイントダンプファイルに関するオペランド　325
4.2.14　サーバ用ステータスファイル（障害発生時）に関するオペランド　328
4.2.15　バックエンドサーバ接続保持機能に関するオペランド　333

5　シングルサーバ定義　335
5.1　オペランドの形式　336
5.2　オペランドの説明　341
5.2.1　プロセスに関するオペランド　341
5.2.2　作業表に関するオペランド　344
5.2.3　システム監視に関するオペランド　347
5.2.4　SQL 実行時間警告出力機能に関するオペランド　350
5.2.5　拡張 SQL エラー情報出力機能に関するオペランド　351
5.2.6　排他制御に関するオペランド　351
5.2.7　バッファに関するオペランド　356
5.2.8　共用メモリに関するオペランド　360
5.2.9　RPC トレース情報に関するオペランド　361
5.2.10　トラブルシュート情報に関するオペランド　362
5.2.11　グローバルバッファに関するオペランド　363
5.2.12　セキュリティに関するオペランド　365
5.2.13　プラグインインデクスの遅延一括作成に関するオペランド　366
5.2.14　Java に関するオペランド　366
5.2.15　システムログファイルに関するオペランド　367
5.2.16　シンクポイントダンプファイルに関するオペランド　373
5.2.17　サーバ用ステータスファイルに関するオペランド　376
5.2.18　サーバ用ステータスファイル（障害発生時）に関するオペランド　378
5.2.19　作業表用ファイルに関するオペランド　384
5.2.20　システムログファイルの構成に関するオペランド　385
5.2.21　シンクポイントダンプファイルの構成に関するオペランド　386
5.2.22　プラグインに関するオペランド　387

6　フロントエンドサーバ定義　389
6.1　オペランドの形式　390
6.2　オペランドの説明　395
6.2.1　プロセスに関するオペランド　395
6.2.2　SQL の最適化に関するオペランド　397
6.2.3　システム監視に関するオペランド　402

6.2.4　SQL 実行時間警告出力機能に関するオペランド　403
6.2.5　拡張 SQL エラー情報出力機能に関するオペランド　405
6.2.6　排他制御に関するオペランド　405
6.2.7　バッファに関するオペランド　408
6.2.8　RPC トレース情報に関するオペランド　412
6.2.9　トラブルシュート情報に関するオペランド　413
6.2.10　セキュリティに関するオペランド　414
6.2.11　Java に関するオペランド　415
6.2.12　システムログファイルに関するオペランド　416
6.2.13　シンクポイントダンプファイルに関するオペランド　422
6.2.14　サーバ用ステータスファイルに関するオペランド　425
6.2.15　サーバ用ステータスファイル（障害発生時）に関するオペランド　427
6.2.16　システムログファイルの構成に関するオペランド　433
6.2.17　シンクポイントダンプファイルの構成に関するオペランド　434
6.2.18　プラグインに関するオペランド　435
6.2.19　HiRDB External Data Access 機能に関するオペランド　435

7　ディクショナリサーバ定義　439
7.1　オペランドの形式　440
7.2　オペランドの説明　445
7.2.1　プロセスに関するオペランド　445
7.2.2　作業表に関するオペランド　448
7.2.3　システム監視に関するオペランド　451
7.2.4　排他制御に関するオペランド　454
7.2.5　バッファに関するオペランド　458
7.2.6　共用メモリに関するオペランド　459
7.2.7　RPC トレース情報に関するオペランド　460
7.2.8　トラブルシュート情報に関するオペランド　461
7.2.9　グローバルバッファに関するオペランド　462
7.2.10　Java に関するオペランド　464
7.2.11　システムログファイルに関するオペランド　465
7.2.12　シンクポイントダンプファイルに関するオペランド　471
7.2.13　サーバ用ステータスファイルに関するオペランド　474
7.2.14　サーバ用ステータスファイル（障害発生時）に関するオペランド　476
7.2.15　作業表用ファイルに関するオペランド　482
7.2.16　システムログファイルの構成に関するオペランド　483
7.2.17　シンクポイントダンプファイルの構成に関するオペランド　484
7.2.18　プラグインに関するオペランド　485

8 バックエンドサーバ定義 487
8.1 オペランドの形式 488
8.2 オペランドの説明 493
8.2.1 プロセスに関するオペランド 493
8.2.2 作業表に関するオペランド 497
8.2.3 システム監視に関するオペランド 500
8.2.4 排他制御に関するオペランド 502
8.2.5 バッファに関するオペランド 506
8.2.6 共用メモリに関するオペランド 507
8.2.7 RPC トレース情報に関するオペランド 508
8.2.8 トラブルシュート情報に関するオペランド 509
8.2.9 グローバルバッファに関するオペランド 510
8.2.10 プラグインインデクスの遅延一括作成に関するオペランド 512
8.2.11 Java に関するオペランド 512
8.2.12 HiRDB External Data Access 機能に関するオペランド 514
8.2.13 システムログファイルに関するオペランド 514
8.2.14 シンクポイントダンプファイルに関するオペランド 520
8.2.15 サーバ用ステータスファイルに関するオペランド 524
8.2.16 サーバ用ステータスファイル（障害発生時）に関するオペランド 525
8.2.17 バックエンドサーバ接続保持機能に関するオペランド 531
8.2.18 作業表用ファイルに関するオペランド 532
8.2.19 システムログファイルの構成に関するオペランド 533
8.2.20 シンクポイントダンプファイルの構成に関するオペランド 534
8.2.21 プラグインに関するオペランド 535
8.2.22 HiRDB External Data Access 機能（環境変数）に関するオペランド 535

9 UAP 環境定義 537
9.1 オペランドの形式 538
9.2 オペランドの説明 539

10 外部サーバ情報定義（外部サーバが HiRDB の場合） 543
10.1 オペランドの形式 544
10.2 オペランドの説明 548
10.2.1 外部サーバとの接続及び解除に関するオペランド 548
10.2.2 外部サーバエラー情報に関するオペランド 548
10.2.3 外部サーバインタフェーストレース情報に関するオペランド 549
10.2.4 FETCH に関するオペランド 550
10.2.5 外部サーバとの接続に関するオペランド 551
10.2.6 データベースの更新ログに関するオペランド 554

10.2.7　クライアントグループに関するオペランド　554
10.2.8　通信処理に関するオペランド　554
10.2.9　システム監視に関するオペランド　556
10.2.10　トラブルシュート情報に関するオペランド　558
10.2.11　排他制御に関するオペランド　562
10.2.12　インナレプリカ機能に関するオペランド　562
10.2.13　SQL の最適化に関するオペランド　562
10.2.14　拡張 SQL エラー情報出力機能に関するオペランド　564
10.2.15　バックエンドサーバ接続保持機能に関するオペランド　564
10.2.16　系切り替え機能に関するオペランド　565
10.3　注意事項　566

11 外部サーバ情報定義（外部サーバが XDM/RD E2 の場合）　567
11.1　オペランドの形式　568
11.2　オペランドの説明　570
11.2.1　外部サーバとの接続及び解除に関するオペランド　570
11.2.2　外部サーバエラー情報に関するオペランド　570
11.2.3　外部サーバインタフェーストレース情報に関するオペランド　571
11.2.4　外部サーバとの接続に関するオペランド　572
11.2.5　システム監視に関するオペランド　572
11.2.6　トラブルシュート情報に関するオペランド　573
11.2.7　データ転送に関するオペランド　574
11.3　外部サーバ（XDM/RD E2）に対して固定値が適用されるクライアント環境定義　575

12 外部サーバ情報定義（外部サーバが ORACLE の場合）　577
12.1　オペランドの形式　578
12.2　オペランドの説明　580
12.2.1　外部サーバとの接続及び解除に関するオペランド　580
12.2.2　外部サーバエラー情報に関するオペランド　580
12.2.3　外部サーバインタフェーストレース情報に関するオペランド　581
12.2.4　FETCH に関するオペランド　582
12.2.5　トラブルシュート情報に関するオペランド　582
12.2.6　排他制御に関するオペランド　582
12.2.7　DESCRIBE 文に関するオペランド　583
12.2.8　外部サーバとの接続に関するオペランド　583

13 Hub 最適化情報定義　585
13.1　オペランドの形式　586
13.2　オペランドの説明　588

13.3　注意事項 594
13.4　各外部サーバの推奨値 596

付録 599

付録 A　オペランドで指定する内容 600
付録 B　定義例 619
付録 B.1　HiRDB/シングルサーバの場合 619
付録 B.2　HiRDB/シングルサーバの場合：系切り替え機能使用時 621
付録 B.3　HiRDB/パラレルサーバの場合 623
付録 B.4　HiRDB/パラレルサーバの場合：スタンバイ型系切り替え機能使用時 631
付録 B.5　HiRDB/パラレルサーバの場合：1：1 スタンバイレス型系切り替え機能使用時 633
付録 B.6　HiRDB External Data Access 機能使用時 635
付録 C　オペランド指定値の見積もり式 639
付録 C.1　統計ログファイル（pd_stj_file_size）の見積もり式 639
付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式 643
付録 C.3　ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式 649
付録 C.4　表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式 655
付録 C.5　1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式 658
付録 C.6　ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式 659
付録 D　排他資源数の見積もり 662
付録 D.1　定義系 SQL 662
付録 D.2　操作系 SQL 697
付録 D.3　制御系 SQL 717
付録 D.4　ユティリティ及びコマンド 717
付録 E　pdconfchk コマンドでチェックできるオペランド 720
付録 F　影響分散スタンバイレス型系切り替え機能使用時に指定できるオペランド一覧（ユニット制御情報定義） 732

索引 733

1 概要

この章では，HiRDB システム定義の体系，及び文法規則について説明します。

1.1　HiRDB/シングルサーバを利用される方へ

HiRDB 管理者は, HiRDB の実行環境を設定するために HiRDB システム定義を作成します。ここで説明する項目は次のとおりです。

- HiRDB システム定義の種類
- HiRDB システム定義の作成方法
- HiRDB システム定義ファイルの構成
- 各定義の関係
- HiRDB システム定義（UAP 環境定義を除く）の変更手順
- UAP 環境定義の追加又は変更手順

1.1.1　HiRDB システム定義の種類

HiRDB システム定義は次の表に示す定義で構成されています。HiRDB 管理者は各定義のオペランドの指定値を決めてください。

表 1-1　HiRDB システム定義の種類（HiRDB/シングルサーバの場合）

定義の種類	内容
システム共通定義	HiRDB の構成及び共通情報を定義します。一つの HiRDB/シングルサーバに一つ必要です。システム共通定義のオペランドについては,「2．システム共通定義」を参照してください。
ユニット制御情報定義	ユニットの制御情報を定義します。一つのユニットに一つ必要です。ユニット制御情報定義のオペランドについては,「3．ユニット制御情報定義」を参照してください。
サーバ共通定義	シングルサーバ定義のオペランドの仮定値を定義します。必要に応じて定義してください。サーバ共通定義のオペランドについては,「4．サーバ共通定義」を参照してください。
シングルサーバ定義	シングルサーバの実行環境を定義します。一つのシングルサーバに一つ必要です。シングルサーバ定義のオペランドについては,「5．シングルサーバ定義」を参照してください。
UAP 環境定義	UAP の実行環境を定義します。必要に応じて定義してください。UAP 環境定義のオペランドについては,「9．UAP 環境定義」を参照してください。UAP 環境定義は最大 4096 個作成できます。
SQL 予約語定義	SQL の予約語を定義します。SQL 予約語削除機能使用時に必要です。SQL 予約語削除機能については，pd_delete_reserved_word_file オペランドを参照してください。

1.1.2　HiRDB システム定義の作成方法

HiRDB 管理者は次に示すどれかの方法で HiRDB システム定義を作成してください。

- 簡易セットアップツールで作成
- バッチファイル SPsetup.bat で作成
- メモ帳などで作成

ポイント

通常は，簡易セットアップツールを使用して HiRDB システム定義を作成してください。

作成した HiRDB システム定義は次の表に示すファイルに格納してください。なお，次の表に示すファイルを総称して **HiRDB システム定義ファイル**といいます。

表 1-2　HiRDB システム定義を格納するファイル（HiRDB/シングルサーバの場合）

定義の種類	格納ファイル名
システム共通定義	%PDDIR%¥conf¥pdsys
ユニット制御情報定義	%PDDIR%¥conf¥pdutsys
サーバ共通定義※1	%PDDIR%¥conf¥pdsvrc
シングルサーバ定義	%PDDIR%¥conf¥サーバ名※2
UAP 環境定義	%PDDIR%¥conf¥pduapenv¥任意の名称※3
SQL 予約語定義	%PDDIR%¥conf¥pdrsvwd¥任意の名称※3

注※1

バッチファイルで HiRDB システム定義を作成した場合は，サーバ共通定義は作成されません。サーバ共通定義が必要ならば，HiRDB 管理者がメモ帳などで作成してください。

注※2

システム共通定義の pdstart オペランドの-s オプションで指定するサーバ名と同じにしてください。

簡易セットアップツール又はバッチファイルで HiRDB システム定義を作成した場合は，ファイル名は sds01 となります。

注※3

- ファイル名称は先頭がアルファベットの英数字列（最大 8 文字）にしてください。大文字と小文字は同じ文字とします。例えば，A と a は同じとします。ファイル名 ABC と abc は同じファイル名となります。
- UAP 環境定義又は SQL 予約語定義を使用するユーザに対して，それぞれのファイルが格納されているディレクトリの読み込み権限（r）と実行権限（x）を与えてください。また，ファイルには読み込み権限（r）を与えてください。

注意事項

HiRDB システム定義ファイルのパーミッションは，ファイルの所有者（HiRDB 管理者）にだけ，読み込み権限及び書き込み権限を持たせるように設定，維持するようにしてください。

(1)　簡易セットアップツールで作成する場合

簡易セットアップツールで HiRDB の環境設定をする場合は，簡易セットアップツールで HiRDB システム定義を作成します。簡易セットアップツールで指定した情報を基にして，自動的に HiRDB システム定義が作成され，表 1-2 に示すファイルに格納されます。HiRDB 管理者は，必要に応じて簡易セットアップツールで定義内容（オペランドの指定値）を変更してください。なお，UAP 環境定義及び SQL 予約語定義は簡易セットアップツールでは作成できません。

(2)　バッチファイル SPsetup.bat で作成する場合

バッチファイルで HiRDB の環境設定をする場合は，HiRDB が自動的に HiRDB システム定義を作成します（表 1-2 に示すファイルに格納されます）。HiRDB 管理者は，必要に応じてメモ帳などのテキストエディタで定義内容（オペランドの指定値）を変更してください。

なお，UAP 環境定義及び SQL 予約語定義はバッチファイルで作成できません。

(3)　メモ帳などで作成する場合

コマンドで HiRDB の環境設定をする場合は，メモ帳などで HiRDB システム定義を作成します。HiRDB の実行環境に合わせて，HiRDB システム定義の各オペランドを指定してください。

なお，HiRDB システム定義を作成した後に，pdconfchk コマンドで HiRDB システム定義のオペランドの整合性をチェックしてください。チェック対象となるオペランドについては，「付録 E　pdconfchk コマンドでチェックできるオペランド」を参照してください。

1.1.3　HiRDB システム定義ファイルの構成

簡易セットアップツール又はバッチファイルで HiRDB システム定義を作成した場合，HiRDB は次の図に示すように HiRDB システム定義ファイルを作成します。メモ帳などで HiRDB システム定義を作成する場合も，次の次の図を参考にして作成してください。

図 1-1　HiRDB システム定義ファイルの構成（HiRDB/シングルサーバの場合）

1.1.4　各定義の関係

異なる定義で同じオペランドを指定した場合，HiRDB は次に示す優先順位に従って指定値を決定します。

1. シングルサーバ定義
2. サーバ共通定義
3. ユニット制御情報定義
4. システム共通定義

優先順位の低い定義で標準となる値を指定して，優先順位の高い定義で値を変更できます。

(1)　クライアント環境定義との関係

次の表に示す HiRDB システム定義のオペランドの値は，クライアント環境定義でクライアントごとに変更できます。

表 1-3　クライアント環境定義で指定値を変更できるオペランド（HiRDB/シングルサーバの場合）

HiRDB システム定義のオペランド	クライアント環境定義のオペランド	オペランドの内容
pd_additional_optimize_level	PDADDITIONALOPTLVL	SQL 拡張最適化オプションを指定します。
pd_cwaittime_wrn_pnt	PDCWAITTIMEWRNPNT	SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）に対する比率で指定します。
pd_delete_reserved_word_file	PDDELRSVWDFILE	SQL 予約語削除機能を使用する場合，SQL 予約語削除ファイルの名称を指定します。
pd_hash_table_size	PDHASHTBLSIZE	SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシュ表サイズを指定します。
pd_hashjoin_hashing_mode	PDHJHASHINGMODE	SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシング方式を指定します。
pd_optimize_level	PDSQLOPTLVL	SQL 最適化オプションを指定します。
pd_space_level	PDSPACELVL	空白変換レベルを指定します。
pd_uap_exerror_log_param_size	PDUAPEXERLOGPRMSZ	エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。
pd_uap_exerror_log_use	PDUAPEXERLOGUSE	拡張 SQL エラー情報出力機能を使用するかどうかを指定します。
pd_watch_pc_client_time	PDSWATCHTIME	Windows 対応の HiRDB クライアントからの要求待ち時間を指定します。

クライアントごとに値を変更する場合は，クライアント環境定義の該当するオペランドを指定してください。クライアント環境定義については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

1.1.5　HiRDB システム定義（UAP 環境定義を除く）の変更手順

(1)　HiRDB システム定義の変更手順

HiRDB システム定義の変更手順を次に示します。なお，%PDDIR%¥conf はユニット制御情報定義ファイルを格納しているディレクトリを意味しています。%PDCONFPATH%はそれ以外の HiRDB システム定義ファイルを格納しているディレクトリを意味しています。

〈手順〉

1. %PDDIR%¥conf 及び%PDCONFPATH%下にサブディレクトリを作成します。この例ではサブディレクトリとして work を作成します。

2. ユニット制御情報定義ファイルを%PDDIR%¥conf¥work 下にコピーします。そのほかの HiRDB システム定義ファイルを%PDCONFPATH%¥work 下にコピーします。
3. %PDDIR%¥conf¥work 及び%PDCONFPATH%¥work 下にコピーした HiRDB システム定義を変更します。
4. **pdconfchk -d work** コマンドで，%PDDIR%¥conf¥work 及び%PDCONFPATH%¥work 下の HiRDB システム定義の内容をチェックします。エラーがある場合は HiRDB システム定義を修正して，再度 pdconfchk コマンドを実行してください。
5. **pdstop** コマンドで HiRDB を正常終了します。
6. **pdlogunld** コマンドで，アンロード待ち状態のシステムログファイルをアンロードします。
7. 3 で変更した HiRDB システム定義ファイルを%PDDIR%¥conf 又は%PDCONFPATH%下にコピーして，HiRDB システム定義ファイルを置き換えます。
8. 次に示すオペランドの指定値を変更した場合は，**pdloginit** コマンドでシステムログファイルを初期化します。
   ・pd_log_dual
   ・pdstart
9. **pdstart** コマンドで HiRDB を正常開始します。

(2)　システム構成変更コマンドを使用した HiRDB システム定義の変更手順

システム構成変更コマンド（pdchgconf コマンド）を使用すると，HiRDB の稼働中に HiRDB システム定義を変更できるため，HiRDB を正常終了する必要はありません。ただし，このコマンドを使用する場合は HiRDB Advanced High Availability が必要になります。システム構成変更コマンドを使用した HiRDB システム定義の変更手順を次に示します。

〈手順〉

1. %PDDIR%¥conf¥chgconf ディレクトリを作成します。
2. 使用中の HiRDB システム定義ファイルを 1 で作成したディレクトリ下にコピーします。
3. %PDDIR%¥conf¥chgconf 下の HiRDB システム定義を変更します。
4. **pdconfchk -d chgconf** コマンドで，%PDDIR%¥conf¥chgconf 下の HiRDB システム定義のチェックをします。エラーがある場合は HiRDB システム定義を修正して，再度 pdconfchk コマンドを実行してください。
5. **pdchgconf** コマンドで，HiRDB システム定義を変更後の HiRDB システム定義に置き換えます。
   pdchgconf コマンドを実行すると，使用中（変更前）の HiRDB システム定義ファイルが%PDDIR%¥conf¥backconf 下に退避されます。そして，%PDDIR%¥conf¥chgconf 下の変更後の HiRDB システム定義ファイルが%PDDIR%¥conf 下にコピーされます。

注意事項

- pdchgconf コマンドの入力後，15 分以上トランザクション又はユティリティが動き続けた場合，pdchgconf コマンドが異常終了します。
- システム構成変更コマンドを使用した HiRDB システム定義の変更には制限事項があります。制限事項については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

(3)　注意事項

- 稼働中の HiRDB が使用している HiRDB システム定義は変更又は削除しないでください。変更又は削除した場合はその HiRDB の動作を保証できません。

- HiRDB が計画停止，強制終了，又は異常終了した場合，HiRDB システム定義のオペランドで変更できるものと変更できないものがあります。詳細については，「1.4　オペランド一覧」を参照するか，又は各定義文のオペランド一覧を参照してください。
- HiRDB システム定義を変更した後に，%PDDIR%¥conf 下のファイルのバックアップを取得してください。HiRDB 運用ディレクトリがあるディスクの障害などに備えて，HiRDB 運用ディレクトリ下のファイル（%PDDIR%¥conf 下のファイル）のバックアップを取得します。HiRDB 運用ディレクトリを回復するには，%PDDIR%¥conf 下のファイルのバックアップが必要になります。また，%PDCONFPATH%が HiRDB 運用ディレクトリ下にある場合は，同様にバックアップを取得してください。

(4)　HiRDB Datareplicator と連携している場合

次に示すオペランドを追加，変更，又は削除する場合は HiRDB Datareplicator を終了してください。そして，オペランドの追加，変更，又は削除後に HiRDB Datareplicator を開始してください。

- pd_log_dual
- pd_log_max_data_size
- pdlogadfg -d sys
- pdlogadpf -d sys

HiRDB Datareplicator を稼働したまま，これらのオペランドを追加，変更，又は削除した場合，HiRDB Datareplicator による抽出が失敗することがあります。

1.1.6　UAP 環境定義の追加又は変更手順

UAP 環境定義の追加又は変更手順を次に示します。

〈手順〉

1. 変更対象の UAP 環境定義を使用する UAP が実行中でないか確認し，実行中の場合は UAP が終了するまで待ちます。UAP の実行中に UAP 環境定義を追加又は変更すると，実行中の UAP は変更前の UAP 環境定義が適用されます。ただし，タイミングによっては変更後の UAP 環境定義が適用されることがあります。
2. UAP 環境定義を追加又は変更します。
3. 追加又は変更した UAP 環境定義を使用して UAP を実行します。

1.2　HiRDB/パラレルサーバを利用される方へ

HiRDB 管理者は HiRDB の実行環境を設定するために HiRDB システム定義を作成します。ここで説明する項目は次のとおりです。

- HiRDB システム定義の種類
- HiRDB システム定義の作成方法
- HiRDB システム定義ファイルの構成
- 各定義の関係
- HiRDB システム定義（UAP 環境定義を除く）の変更手順
- UAP 環境定義の追加又は変更手順
- 影響分散スタンバイレス型系切り替え機能使用時の HiRDB システム定義の作成

1.2.1　HiRDB システム定義の種類

HiRDB システム定義は次の表に示す定義で構成されています。HiRDB 管理者は各定義のオペランドの指定値を決めてください。

表 1-4　HiRDB システム定義の種類（HiRDB/パラレルサーバの場合）

定義の種類		内容
システム共通定義		HiRDB の構成及び共通情報を定義します。一つのユニットに一つ必要です。なお，**各ユニットのシステム共通定義の内容は同一にしてください。** システム共通定義のオペランドについては，「2．システム共通定義」を参照してください。
ユニット制御情報定義		ユニットの制御情報を定義します。一つのユニットに一つ必要です。ユニット制御情報定義のオペランドについては，「3．ユニット制御情報定義」を参照してください。
サーバ定義	サーバ共通定義	各サーバ定義のオペランドの仮定値を定義します。必要に応じて定義してください。 サーバ共通定義のオペランドについては，「4．サーバ共通定義」を参照してください。
	フロントエンドサーバ定義	フロントエンドサーバの実行環境を定義します。一つのフロントエンドサーバに一つ必要です。フロントエンドサーバ定義のオペランドについては，「6．フロントエンドサーバ定義」を参照してください。
	ディクショナリサーバ定義	ディクショナリサーバの実行環境を定義します。一つのディクショナリサーバに一つ必要です。ディクショナリサーバ定義のオペランドについては，「7．ディクショナリサーバ定義」を参照してください。
	バックエンドサーバ定義	バックエンドサーバの実行環境を定義します。一つのバックエンドサーバに一つ必要です。バックエンドサーバ定義のオペランドについては，「8．バックエンドサーバ定義」を参照してください。
UAP 環境定義		UAP の実行環境を定義します。必要に応じて定義してください。UAP 環境定義はフロントエンドサーバがあるユニットに作成します。マルチフロントエンドサーバの場合は，UAP 環境定義を適用したいフロントエンドサーバに定義してください。UAP 環境定義のオペランドについては，「9．UAP 環境定義」を参照してください。UAP 環境定義は最大 4096 個作成できます。
SQL 予約語定義		SQL 予約語削除機能使用時に必要です。SQL の予約語を定義します。フロントエンドサーバがあるユニットに定義します。マルチフロントエンドサーバの場合は，フロントエ

定義の種類	内容
	ンドサーバがある各ユニットに同一内容の SQL 予約語定義が必要です。SQL 予約語削除機能については，pd_delete_reserved_word_file オペランドを参照してください。
外部サーバ情報定義	外部サーバへの接続環境を定義します。接続する外部サーバごとに定義します。外部サーバ接続用のバックエンドサーバがあるユニットに定義します。HiRDB External Data Access 機能使用時に必要です。外部サーバ情報定義のオペランドについては，次に示す章を参照してください。 •「10．外部サーバ情報定義（外部サーバが HiRDB の場合）」 •「11．外部サーバ情報定義（外部サーバが XDM/RD E2 の場合）」 •「12．外部サーバ情報定義（外部サーバが ORACLE の場合）」
Hub 最適化情報定義	HiRDB External Data Access 機能の最適化情報を定義します。フロントエンドサーバがあるユニットに定義します。マルチフロントエンドサーバの場合は，各ユニットに同一内容の Hub 最適化情報定義が必要です。 HiRDB External Data Access 機能使用時に必要です。Hub 最適化情報定義のオペランドについては，「13．Hub 最適化情報定義」を参照してください。

1.2.2　HiRDB システム定義の作成方法

HiRDB 管理者は，次に示すどちらかの方法で HiRDB システム定義を作成してください。

- 簡易セットアップツールで作成
- メモ帳などで作成

ポイント

通常は，簡易セットアップツールを使用して HiRDB システム定義を作成してください。

作成した HiRDB システム定義は，次の表に示すファイルに格納してください。なお，次の表に示すファイルを総称して **HiRDB システム定義ファイル**といいます。

表 1-5　HiRDB システム定義を格納するファイル（HiRDB/パラレルサーバの場合）

定義の種類	格納ファイル名
システム共通定義	%PDDIR%¥conf¥pdsys
ユニット制御情報定義	%PDDIR%¥conf¥pdutsys
サーバ共通定義	%PDDIR%¥conf¥pdsvrc
フロントエンドサーバ定義	%PDDIR%¥conf¥サーバ名※1
ディクショナリサーバ定義	%PDDIR%¥conf¥サーバ名※1
バックエンドサーバ定義	%PDDIR%¥conf¥サーバ名※1
UAP 環境定義	%PDDIR%¥conf¥pduapenv¥任意の名称※2
SQL 予約語定義	%PDDIR%¥conf¥pdrsvwd¥任意の名称※2
外部サーバ情報定義	%PDDIR%¥conf¥外部サーバ名
Hub 最適化情報定義	%PDDIR%¥conf¥任意の名称

注※1

システム共通定義の pdstart オペランドの-s オプションで指定するサーバ名と同じにしてください。

注※2

- ファイル名称は先頭がアルファベットの英数字列（最大 8 文字）にしてください。大文字と小文字は同じ文字とします。例えば，A と a は同じとします。ファイル名 ABC と abc は同じファイル名となります。
- UAP 環境定義又は SQL 予約語定義を使用するユーザに対して，それぞれのファイルが格納されているディレクトリの読み込み権限（r）と実行権限（x）を与えてください。また，ファイルには読み込み権限（r）を与えてください。

！ 注意事項

HiRDB システム定義ファイルのパーミッションは，ファイルの所有者（HiRDB 管理者）にだけ，読み込み権限及び書き込み権限を持たせるように設定，維持するようにしてください。

(1)　簡易セットアップツールで作成する場合

簡易セットアップツールで HiRDB の環境設定をする場合は，簡易セットアップツールで HiRDB システム定義を作成します。簡易セットアップツールで指定した情報を基にして，自動的に HiRDB システム定義が作成され，表 1-5 に示すファイルに格納されます。HiRDB 管理者は，必要に応じて簡易セットアップツールで定義内容（オペランドの指定値）を変更してください。

なお，次に示す定義は簡易セットアップツールでは作成できません。

- UAP 環境定義
- SQL 予約語定義
- 外部サーバ情報定義
- Hub 最適化情報定義

(2)　メモ帳などで作成する場合

コマンドで HiRDB の環境設定をする場合は，メモ帳などで HiRDB システム定義を作成します。HiRDB の実行環境に合わせて，HiRDB システム定義の各オペランドを指定してください。

なお，HiRDB システム定義を作成した後に，pdconfchk コマンドで HiRDB システム定義のオペランドの整合性をチェックしてください。チェック対象となるオペランドについては，「付録 E　pdconfchk コマンドでチェックできるオペランド」を参照してください。

1.2.3　HiRDB システム定義ファイルの構成

簡易セットアップツールで HiRDB システム定義を作成した場合，HiRDB は次の図に示すように HiRDB システム定義ファイルを作成します。メモ帳などで HiRDB システム定義を作成する場合も，次の図を参考にして作成してください。

図 1-2　HiRDB システム定義ファイルの構成（HiRDB/パラレルサーバの場合）

注１　各サーバマシンのシステム共通定義は同一内容にしてください。
注２　各サーバマシンのSQL予約語定義は同一内容にしてください。

●HiRDB External Data Access 機能を使用している場合

HiRDB External Data Access 機能を使用している場合の HiRDB システム定義ファイルの構成例を次の図に示します。

図 1-3　HiRDB システム定義ファイルの構成例（HiRDB External Data Access 機能を使用している場合）

注１　各サーバマシンのシステム共通定義は同一内容にしてください。
注２　網掛けされているBESが外部サーバ接続用のバックエンドサーバです。
注３　各サーバマシンのSQL予約語定義は同一内容にしてください。

1.2.4　各定義の関係

異なる定義で同じオペランドを指定した場合，HiRDB は次に示す優先順位に従って指定値を決定します。

1. フロントエンドサーバ定義，ディクショナリサーバ定義，バックエンドサーバ定義
2. サーバ共通定義
3. ユニット制御情報定義
4. システム共通定義

優先順位の低い定義で標準となる値を指定して，優先順位の高い定義で値を変更できます。例えば，システム共通定義で標準となる値を指定し，各サーバ定義でサーバごとにその値を変更できます。

(1)　クライアント環境定義との関係

次の表に示す HiRDB システム定義のオペランドの値は，クライアント環境定義でクライアントごとに変更できます。

表 1-6　クライアント環境定義で指定値を変更できるオペランド（HiRDB/パラレルサーバの場合）

HiRDB システム定義のオペランド	クライアント環境定義のオペランド	オペランドの内容
pd_additional_optimize_level	PDADDITIONALOPTLVL	SQL 拡張最適化オプションを指定します。
pd_cwaittime_wrn_pnt	PDCWAITTIMEWRNPNT	SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）に対する比率で指定します。
pd_delete_reserved_word_file	PDDELRSVWDFILE	SQL 予約語削除機能を使用する場合，SQL 予約語削除ファイルの名称を指定します。
pd_ha_transaction	PDHATRNQUEUING	トランザクションキューイング機能を使用するかどうかを指定します。
pd_hash_table_size	PDHASHTBLSIZE	SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシュ表サイズを指定します。
pd_hashjoin_hashing_mode	PDHJHASHINGMODE	SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシング方式を指定します。
pd_optimize_level	PDSQLOPTLVL	SQL 最適化オプションを指定します。
pd_space_level	PDSPACELVL	空白変換レベルを指定します。
pd_uap_exerror_log_param_size	PDUAPEXERLOGPRMSZ	エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。
pd_uap_exerror_log_use	PDUAPEXERLOGUSE	拡張 SQL エラー情報出力機能を使用するかどうかを指定します。
pd_watch_pc_client_time	PDSWATCHTIME	Windows 対応の HiRDB クライアントからの要求待ち時間を指定します。

クライアントごとに値を変更する場合は，クライアント環境定義の該当するオペランドを指定してください。クライアント環境定義については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

1.2.5　HiRDB システム定義（UAP 環境定義を除く）の変更手順

(1)　HiRDB システム定義の変更手順

HiRDB システム定義の変更手順を次に示します。なお，%PDDIR%¥conf はユニット制御情報定義ファイルを格納しているディレクトリを意味しています。%PDCONFPATH%はそれ以外の HiRDB システム定義ファイルを格納しているディレクトリを意味しています。

〈手順〉

1. %PDDIR%¥conf 及び%PDCONFPATH%下にサブディレクトリを作成します。この例ではサブディレクトリとして work を作成します。
2. ユニット制御情報定義ファイルを%PDDIR%¥conf¥work 下にコピーします。そのほかの HiRDB システム定義ファイルを%PDCONFPATH%¥work 下にコピーします。
3. %PDDIR%¥conf¥work 及び%PDCONFPATH%¥work 下にコピーした HiRDB システム定義を変更します。
4. **pdconfchk -d work** コマンドで，%PDDIR%¥conf¥work 及び%PDCONFPATH%¥work 下の HiRDB システム定義の内容をチェックします。エラーがある場合は HiRDB システム定義を修正して，再度 pdconfchk コマンドを実行してください。
5. **pdstop** コマンドで HiRDB を正常終了します。
6. **pdlogunld** コマンドで，アンロード待ち状態のシステムログファイルをアンロードします。
7. 3 で変更した HiRDB システム定義ファイルを%PDDIR%¥conf 又は%PDCONFPATH%下にコピーして，HiRDB システム定義ファイルを置き換えます。
8. 次に示すオペランドの指定値を変更した場合は，**pdloginit** コマンドでシステムログファイルを初期化します。
   - pd_log_dual
   - pdstart
9. **pdstart** コマンドで HiRDB を正常開始します。

(2)　システム構成変更コマンドを使用した HiRDB システム定義の変更手順

システム構成変更コマンド（pdchgconf コマンド）を使用すると，HiRDB の稼働中に HiRDB システム定義を変更できるため，HiRDB を正常終了する必要はありません。ただし，このコマンドを使用する場合は HiRDB Advanced High Availability が必要になります。システム構成変更コマンドを使用した HiRDB システム定義の変更手順を次に示します。

〈手順〉

1. %PDDIR%¥conf¥chgconf ディレクトリを作成します。
2. 使用中の HiRDB システム定義ファイルを 1 で作成したディレクトリ下にコピーします。
3. %PDDIR%¥conf¥chgconf 下の HiRDB システム定義を変更します。
4. **pdconfchk -d chgconf** コマンドで，%PDDIR%¥conf¥chgconf 下の HiRDB システム定義のチェックをします。エラーがある場合は HiRDB システム定義を修正して，再度 pdconfchk コマンドを実行してください。
5. **pdchgconf** コマンドで，HiRDB システム定義を変更後の HiRDB システム定義に置き換えます。

   pdchgconf コマンドを実行すると，使用中（変更前）の HiRDB システム定義ファイルが%PDDIR%¥conf¥backconf 下に退避されます。そして，%PDDIR%¥conf¥chgconf 下の変更後の HiRDB システム定義ファイルが%PDDIR%¥conf 下にコピーされます。

注意事項

- pdchgconf コマンドの入力後，15 分以上トランザクション又はユティリティが動き続けた場合，pdchgconf コマンドが異常終了します。
- システム構成変更コマンドを使用した HiRDB システム定義の変更には制限事項があります。制限事項については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

(3)　注意事項

- システム共通定義を修正する場合はすべてのサーバマシンのシステム共通定義を同じように修正してください。
- 正常終了又は計画停止した場合に一部のユニットが異常終了したときは，次回開始前に HiRDB システム定義を変更しないでください。変更すると，HiRDB の開始に失敗します。失敗しなくても，開始後に正常に稼働できなくなります。
- HiRDB/パラレルサーバの場合はユニットごとに%PDDIR%¥conf 及び%PDCONFPATH%下にサブディレクトリを作成して，HiRDB システム定義の内容をチェックしてください。
- 稼働中の HiRDB が使用している HiRDB システム定義は変更又は削除しないでください。変更又は削除した場合はその HiRDB の動作を保証できません。
- HiRDB が計画停止，強制終了，又は異常終了した場合，HiRDB システム定義のオペランドで変更できるものと変更できないものがあります。詳細については，「1.4　オペランド一覧」を参照するか，又は各定義文のオペランド一覧を参照してください。
- HiRDB システム定義を変更した後に，%PDDIR%¥conf 下のファイルのバックアップを取得してください。HiRDB 運用ディレクトリがあるディスクの障害などに備えて，HiRDB 運用ディレクトリ下のファイル（%PDDIR%¥conf 下のファイル）のバックアップを取得します。HiRDB 運用ディレクトリを回復するには，%PDDIR%¥conf 下のファイルのバックアップが必要になります。また，%PDCONFPATH%が HiRDB 運用ディレクトリ下にある場合は，同様にバックアップを取得してください。
- 1：1 スタンバイレス型系切り替え機能使用時の注意事項です。正規 BES ユニットの HiRDB システム定義を変更する場合は，正規 BES ユニット及び代替 BES ユニットを pdstop -u コマンドで正常終了してから変更してください。HiRDB システム定義の変更後，正規 BES ユニットのユニット制御情報定義ファイル及びバックエンドサーバ定義ファイルを代替 BES ユニットにコピーしてください。詳細については，マニュアル「HiRDB Version 8 システム運用ガイド」の「HiRDB システム定義の作成〔系切り替え機能〕」を参照してください。

(4)　HiRDB Datareplicator と連携している場合

次に示すオペランドを追加，変更，又は削除する場合は HiRDB Datareplicator を終了してください。そして，オペランドの追加，変更，又は削除後に HiRDB Datareplicator を開始してください。

- pd_log_dual
- pd_log_max_data_size
- pdlogadfg -d sys
- pdlogadpf -d sys

HiRDB Datareplicator を稼働したまま，これらのオペランドを追加，変更，又は削除した場合，HiRDB Datareplicator による抽出が失敗することがあります。

1.2.6　UAP 環境定義の追加又は変更手順

UAP 環境定義の追加又は変更手順を次に示します。

〈手順〉

1. 変更対象の UAP 環境定義を使用する UAP が実行中でないか確認し，実行中の場合は UAP が終了するまで待ちます。UAP の実行中に UAP 環境定義を追加又は変更すると，実行中の UAP は変更

前の UAP 環境定義が適用されます。ただし，タイミングによっては変更後の UAP 環境定義が適用されることがあります。

2. UAP 環境定義を追加又は変更します。
3. 追加又は変更した UAP 環境定義を使用して UAP を実行します。

1.2.7　影響分散スタンバイレス型系切り替え機能使用時の HiRDB システム定義の作成

影響分散スタンバイレス型系切り替え機能使用時の HiRDB システム定義の作成方法と注意事項を説明します。

ポイント

ホスト BES が稼働するユニットとゲスト BES が稼働するユニットのバックエンドサーバの稼働環境を合わせる必要があります。

(1)　システム共通定義

全ユニットで同じシステム共通定義を使用します。作成したシステム共通定義をコピーしてください。

注意事項

バックエンドサーバ定義の省略値として指定するオペランドについては，システム共通定義に指定してください。例えば，pd_sql_object_cache_size オペランドが該当します。このオペランドをサーバ共通定義だけに指定することはできません。

(2)　ユニット制御情報定義

影響分散スタンバイレス型系切り替え機能を適用するユニットのユニット制御情報定義に指定できるオペランドは限定されています。ユニット制御情報定義に指定できるオペランドについては，「付録 F　影響分散スタンバイレス型系切り替え機能使用時に指定できるオペランド一覧（ユニット制御情報定義）」を参照してください。

ユニット制御情報定義に指定できないオペランドを指定する必要がある場合は，システム共通定義又はバックエンドサーバ定義に指定してください。

注意事項

次に示すオペランドについては，HA グループを構成するユニットのユニット制御情報定義で同じ指定をする必要があります。

- ユニット制御情報定義とバックエンドサーバ定義の両方に指定できるオペランド
- ユニット制御情報定義のオペランドでバックエンドサーバの動作に影響を与えるオペランド

(3)　サーバ共通定義

HA グループを構成するユニットは同じサーバ共通定義を使用します。作成したサーバ共通定義をコピーしてください。

(4)　バックエンドサーバ定義

HA グループを構成する各ユニットにホスト BES のバックエンドサーバ定義を定義します。作成したバックエンドサーバ定義をコピーしてください。

1.3　HiRDB システム定義を記述するときの文法規則

ここでは，HiRDB システム定義のオペランドを記述するときの文法規則について説明します。

(1)　オペランドの指定形式

HiRDB システム定義のオペランドは次に示す三つの形式で記述します。

set 形式

オペランドに値を設定します。指定例を次に示します。

set pd_max_users=15

コマンド形式

オペランドにオプション及びコマンド引数を設定します。指定例を次に示します。

pdlogadfg -d sys -g loggrp01 ONL

pdlogadfg：コマンド名

-d sys -g loggrp01：オプション

マイナス記号で始まる文字列です。フラグ引数を指定しない形式 1 と，フラグ引数を指定する形式 2 の二つがあります。

形式 1　－オプションフラグ

形式 2　－オプションフラグ　フラグ引数

オプションフラグ：マイナス記号に続く 1 文字の英字です。英大文字と英小文字は区別されます。

フラグ引数：オプションフラグに対する操作対象です。

ONL：コマンド引数

マイナス記号以外で始まる引数です。

putenv 形式

オペランドに環境変数と環境変数値を設定します。指定例を次に示します。

putenv SHMMAX 16

(2)　コメントの記述

各オペランドにコメントを記述できます。コメントを記述する場合は，コメントの先頭に"#"を記述してください。"#"を記述するとその行の終わりまでがコメントとしてみなされます。行の先頭に"#"を記述した場合は，1 行全体がコメント扱いになります。

入力例

```
set pd_max_users=15 # 最大同時接続数
pdlogadfg -d sys -g loggrp01 ONL
# システムログファイルのファイルグループの定義
```

(3)　行の継続

定義の 1 行の長さは最大 80 文字です。80 文字を超える指定の場合は，複数行に分けて記述してください。このとき，行の末尾には，継続記号である"¥"を記述します。

入力例

```
pdbuffer -a buffer ABC -n 160 -r rdareaA,rdareaB,……¥
rdareaZ
```

なお，コメントを記述した行には，継続行を続けられません。"#"を記述すると，その行の終わりまでがコメントとみなされるため，"¥"を記述しても行の継続はできません。

(4)　オペランドの重複指定について

1 定義文（1 ファイル）にオペランドを重複指定した場合，HiRDB は次に示す規則に従って処理をします。

- set 形式又は putenv 形式の場合は，重複指定したオペランドのうち一番最後に指定したオペランドの値が有効になります。
- コマンド形式の場合はオペランドによって異なります。各オペランドの説明を参照してください。

(5)　注意事項

- HiRDB システム定義に指定している putenv 形式のオペランドをデフォルト値に変更する場合，オペランドを削除せずに明示的にデフォルト値を設定してください。
- HiRDB システム定義ファイルには，.txt などの拡張子を付けないでください。
- パス名を絶対パスで指定するときには，必ずドライブ名を指定してください。
- （例）C:¥hirdb_s¥spool¥tmp
- HiRDB システム定義ファイル中に空白を含むパス名を指定する場合は，前後を引用符（”）で囲んでください。

  （例）pdinit -d "C:¥Program Files¥hitachi¥hirdb_s¥conf¥mkinit"

1.4　オペランド一覧

1.4.1　指定値を変更できるオペランド

HiRDB システム定義で定義するオペランド，及び各オペランドの再開始時の変更可否の一覧を次の表に示します。オペランド名はアルファベット順に記載しています。なお，次に示す定義のオペランドはここには記載していません。これらの定義のオペランドは HiRDB の再開始前（計画停止，強制終了，又は異常終了後）に変更できます。

- UAP 環境定義
- 外部サーバ情報定義（HiRDB External Data Access 機能）
- Hub 最適化情報定義（HiRDB External Data Access 機能）

表 1-7　HiRDB システム定義で定義するオペランド，及び各オペランドの再開始時の変更可否の一覧

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_additional_optimize_level ※6	○				○			○	○
pd_alv_port	○							○	○
pd_apply_search_ats_num	○							○	○
pd_assurance_index_no	○							○※4	○
pd_assurance_table_no	○							○※4	○
pd_audit	○	○						×	×
pd_aud_async_buff_count	○	○						○	○
pd_aud_async_buff_retry_intvl	○	○						○	○
pd_aud_async_buff_size	○	○						○	○
pd_aud_auto_loading	○							×	×
pd_aud_file_name	○	○						○	○
pd_aud_file_wrn_pnt	○							○	○
pd_aud_max_generation_num	○	○						○	○
pd_aud_max_generation_size	○	○						○	○
pd_aud_no_standby_file_opr	○							○	○
pd_aud_sql_data_size	○	○						○	○

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_aud_sql_source_size	○	○						○	○
pd_audit_def_buffer_size			○	○	○			○	○
pd_auth_cache_size			○	○	○			○※2，3	○※2
pd_auto_vrup	○							×	×
pd_bes_connection_hold			○				○	○	○
pd_bes_conn_hold_trn_interval			○				○	○	○
pd_bes_shmpool_size			○				○	○※2	○※2
pd_c_library_directory	○	○						○	○
pd_cancel_down_msgchange	○							○	○
pd_cancel_dump	○	○						○	○
pd_change_clt_ipaddr	○	○						○	○
pd_check_pending	○							○	○
pd_client_waittime_over_abort	○							○	○
pd_clt_waittime_over_dump_level	○							○	○
pd_cmdhold_precheck	○							○	○
pd_cmd_process_conwaittime	○							○	○
pd_command_deadlock_priority	○							○	○
pd_connect_errmsg_hide	○							○	○
pd_constraint_name	○							○	○
pd_cwaittime_report_dir	○	○						○	○
pd_cwaittime_report_size	○	○						○	○
pd_cwaittime_wrn_pnt※6	○			○	○			○	○
pd_db_access_error_action	○							○	○
pd_dbbuff_attribute	○	○						○	○
pd_dbbuff_binary_data_lru	○							○	○

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_dbbuff_lock_interval	○							○	○
pd_dbbuff_lock_release_detect	○							○	○
pd_dbbuff_lock_spn_count	○							○	○
pd_dbbuff_lru_option	○							○	○
pd_dbbuff_modify	○							×	×
pd_dbbuff_rate_updpage	○							○	○
pd_dbbuff_trace_level	○							○	○
pd_dbbuff_wait_interval	○	○						○	○
pd_dbbuff_wait_spn_count	○	○						○	○
pd_dbsync_altwrite_skip	○							○	○
pd_dbsync_lck_release_count			○	○		○	○	○	○
pd_dbsync_point	○							×	○
pd_db_io_error_action	○	○						○	○
pd_deadlock_priority_use	○							○	○
pd_debug_info_netstat	○							○	○
pd_dec_sign_normalize	○							×	○
pd_def_buf_control_area_assign	○							○	○
pd_delete_reserved_word_file※6	○							○	○
pd_deter_restart_on_stop_fail	○							○	○
pd_dfw_awt_process			○	○		○	○	○	○
pd_dfw_syncpoint_skip_limit			○	○		○	○	○	○
pd_dic_shmpool_size			○			○		○※2	○※2
pd_directory_server	○							○	○
pd_down_watch_proc	○	○						○	○
pd_dump_suppress_watch_time	○	○						○	○
pd_fes_lck_pool_partition			○		○			×	○

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_fes_lck_pool_size			○		○			○	○
pd_floatable_bes					○			○	○
pd_foreign_server_libpath							○	○	○
pd_ha	○							×	×
pd_ha_acttype		○						×	×
pd_ha_agent		○						×	×
pd_ha_ipaddr_inherit	○	○						×	×
pd_ha_max_act_guest_servers		○						×	×
pd_ha_max_server_process		○						×	○※2
pd_ha_mgr_rerun	○							○	○
pd_ha_process_count		○						○	○
pd_ha_resource_act_wait_time	○	○						○	○
pd_ha_server_process_standby		○						×	×
pd_ha_switch_timeout	○	○						○	○
pd_ha_transaction※6	○							○	○
pd_ha_trn_queuing_wait_time	○							○	○
pd_ha_trn_restart_retry_time	○							○	○
pd_ha_unit		○						×	×
pd_hash_table_size※6	○							×	×
pd_hashjoin_hashing_mode※6	○							○	○
pd_hostname		○						×	×
pd_host_watch_interval	○							○	○
pd_indexlock_mode	○							×	○
pd_ipc_conn_count	○							○	○
pd_ipc_conn_interval	○							○	○
pd_ipc_conn_nblock	○							○	○

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_ipc_conn_nblock_time	○							○	○
pd_ipc_inet_bufsize	○	○						○	○
pd_ipc_recv_count	○	○						○	○
pd_ipc_send_count	○	○						○	○
pd_ipc_send_retrycount	○	○						○	○
pd_ipc_send_retrysleeptime	○	○						○	○
pd_java_archive_directory	○	○						○	○
pd_java_classpath	○	○						○	○
pd_java_libpath	○	○						○	○
pd_java_option	○							○	○
pd_java_routine_stack_size	○							○	○
pd_java_runtimepath	○	○						○	○
pd_java_stdout_file	○	○	○	○	○	○	○	○	○
pd_jp1_event_level	○							○	○
pd_jp1_event_msg_out	○							○	○
pd_jp1_use	○							×	×
pd_key_resource_type	○							×	○
pd_large_file_use	○							×	○
pd_lck_deadlock_check	○	○						○	○
pd_lck_deadlock_check_interval	○	○						○	○
pd_lck_deadlock_info	○	○						○	○
pd_lck_hash_entry			○	○	○	○	○	○	○
pd_lck_pool_partition			○	○		○	○	×	○
pd_lck_pool_size			○	○		○	○	○※2	○※2
pd_lck_queue_limit	○							○	○
pd_lck_release_detect	○	○						○	○
pd_lck_release_detect_interval	○	○						○	○
pd_lck_until_disconnect_cnt			○	○		○	○	×	○

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_lck_wait_timeout	○	○						○	○
pd_leap_second	○							○	○
pd_list_initialize_timing	○							○	○
pd_lock_uncommited_delete_data	○							×	◎
pd_log_auto_expand_size			○	○	○	○	○	○	○
pd_log_auto_unload_path				○	○	○	○	×	×
pd_log_dual※5			○	○	○	○	○	×	×
pd_log_max_data_size※5			○	○	○	○	○	○※2，3	○※2
pd_log_rec_leng			○	○	○	○	○	○	○
pd_log_remain_space_check			○	○	○	○	○	○	○
pd_log_rerun_reserved_file_open			○	○	○	○	○	○	○
pd_log_rerun_swap			○	○	○	○	○	○	○
pd_log_rollback_buff_count			○	○	○	○	○	○※3	○
pd_log_rpl_no_standby_file_opr	○							○	○
pd_log_sdinterval			○	○	○	○	○	○	○
pd_log_singleoperation			○	○	○	○	○	×	×
pd_log_swap_timeout			○	○	○	○	○	○	○
pd_log_unload_check			○	○	○	○	○	○	○
pd_log_write_buff_count			○	○	○	○	○	○※3	○
pd_master_file_name	○							×	×
pd_max_access_tables	○							○※3	○
pd_max_access_tables_wrn_pnt	○							○	○
pd_max_add_dbbuff_no			○	○		○	○	×	○
pd_max_add_dbbuff_shm_no			○	○		○	○	×	○
pd_max_ard_process			○	○		○	○	○	○
pd_max_bes_process			○				○	×	○※2

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_max_commit_write_reclaim_no	○							○	○
pd_max_dic_process			○			○		×	○※2
pd_max_file_no	○							○※2，3	○
pd_max_file_no_wrn_pnt	○							○	○
pd_max_foreign_server	○							×	○
pd_max_list_users	○							○※3	○
pd_max_list_count	○							×	○
pd_max_list_users_wrn_pnt	○							○	○
pd_max_list_count_wrn_pnt	○							○	○
pd_max_open_holdable_cursors			○	○		○	○	×	×
pd_max_rdarea_no	○							×	○
pd_max_rdarea_no_wrn_pnt	○							○	○
pd_max_recover_process	○	○						○	○
pd_max_resident_rdarea_no	○							○	○
pd_max_resident_rdarea_shm_no	○							○	○
pd_max_server_process	○	○						○	○
pd_max_users	○							×	○
pd_max_users_wrn_pnt	○							○	○
pd_mlg_file_size	○							○	○
pd_mlg_msg_log_unit	○							×	×
pd_mlg_port	○							○	○
pd_mode_conf	○							○	○
pd_module_trace_max	○	○	○	○	○	○	○	○	○
pd_module_trace_timer_level	○	○	○	○	○	○	○	○	○
pd_name_fixed_port_lookup	○							○	○
pd_name_port	○							◎	◎

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_non_floatable_bes					○			○	○
pd_nowait_scan_option	○							○	○
pd_ntfs_cache_disable	○							○	○
pd_oltp_holdcr	○							○	○
pd_optimize_level※6	○				○			○	○
pd_overflow_suppress	○							○	○
pd_pageaccess_mode	○							○	○
pd_plugin_ixmk_dir				○			○	×	○
pd_process_count			○	○	○	○	○	○※2	○※2
pd_process_desktopheap_size	○							○	○
pd_process_terminator	○							○	○
pd_process_terminator_max	○							○	○
pd_pth_trace_max	○	○	○	○	○	○	○	○	○
pd_queue_watch_time	○							○	○
pd_queue_watch_timeover_action	○							○	○
pd_rdarea_expand_format	○							○	○
pd_rdarea_extension_timing	○							○	○
pd_rdarea_list_no_wrn_pnt	○							○	○
pd_rdarea_open_attribute	○							○	○
pd_rdarea_open_attribute_use	○							○	○
pd_rdarea_warning_point	○							○	○
pd_rdarea_warning_point_msgout	○							○	○
pd_redo_allpage_put	○							○	○
pd_reduced_check_time	○							○	○
pd_registered_port	○	○						×	◎
pd_registered_port_check	○	○						×	◎

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_registered_port_level	○	○						×	○
pd_registry_cache_size			○	○	○			○	○
pd_rorg_predict	○							○	○
pd_routine_def_cache_size			○	○	○			○	○
pd_rpc_bind_loopback_address	○							○	○
pd_rpc_trace	○	○	○	○	○	○	○	○	○
pd_rpc_trace_name	○	○	○	○	○	○	○	○	○
pd_rpc_trace_size	○	○	○	○	○	○	○	○	○
pd_rpl_func_control	○							○	○
pd_rpl_hdepath		○						×	×
pd_rpl_init_start	○							×	×
pd_rpl_reflect_mode	○							×	○
pd_scd_port	○							○	○
pd_sds_shmpool_size			○	○				○※2	○※2
pd_security_host_group	○							○	○
pd_server_cleanup_interval			○	○	○	○	○	○	○
pd_server_entry_queue	○	○						○	○
pd_service_port	○	○						◎	◎
pd_shared_rdarea_use	○							×	○
pd_shmpool_attribute	○	○						○	○
pd_space_level※6	○							○	○
pd_spd_assurance_count			○	○	○	○	○	×	×
pd_spd_assurance_msg			○	○	○	○	○	○	○
pd_spd_dual			○	○	○	○	○	×	×
pd_spd_max_data_size			○	○	○	○	○	×	×
pd_spd_reduced_mode			○	○	○	○	○	×	○
pd_spd_reserved_file_auto_open			○	○	○	○	○	×	○
pd_spd_syncpoint_skip_limit			○	○	○	○	○	○	○

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_spool_cleanup	○	○						○	○
pd_spool_cleanup_interval	○	○						○	○
pd_spool_cleanup_interval_level	○	○						○	○
pd_spool_cleanup_level	○	○						○	○
pd_sql_command_exec_users	○							○	○
pd_sql_dec_op_maxprec	○							○	○
pd_sql_object_cache_size	○		○	○	○	○	○	×	○
pd_sql_send_buff_size	○							○	○
pd_standard_sqlstate	○							○	○
pd_start_level	○							○	○
pd_start_skip_unit	○							○	○
pd_start_time_out	○							○	○
pd_statistics	○							○	○
pd_stj_file_size	○	○						○※3	○
pd_stj_buff_size	○	○						○※3	○
pd_sts_file_name_1～7				○	○	○	○	×	×
pd_sts_initial_error			○	○	○	○	○	○	○
pd_sts_last_active_file				○	○	○	○	○	○
pd_sts_last_active_side				○	○	○	○	○	○
pd_sts_singleoperation			○	○	○	○	○	○	○
pd_substr_length	○							○	○
pd_sysdef_default_option	○							×	○
pd_syssts_file_name_1～7		○						×	×
pd_syssts_initial_error		○						○	○
pd_syssts_last_active_file		○						○	○
pd_syssts_last_active_side		○						○	○
pd_syssts_singleoperation		○						○	○
pd_system_complete_wait_time	○							○	○

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_system_dbsync_point	○							×	○
pd_system_id	○							×	×
pd_table_def_cache_size			○	○	○			○※2，3	○※2
pd_tcp_inet_bufsize	○	○						○	○
pd_thdlock_pipe_retry_interval	○	○						○	○
pd_thdlock_retry_time	○	○						○	○
pd_thdlock_sleep_func	○							○	○
pd_thdlock_wakeup_lock	○	○						○	○
pd_thdspnlk_spn_count	○	○						○	○
pd_thread_max_stack_size	○							○	○
pd_tmp_directory		○						○	○
pd_trn_commit_optimize	○							○	○
pd_trn_port	○							○	○
pd_trn_rerun_branch_auto_decide	○							○	○
pd_trn_send_decision_interval	○							○	○
pd_trn_send_decision_intval_sec	○							○	○
pd_trn_send_decision_retry_time	○							○	○
pd_trn_watch_time	○							○	○
pd_type_def_cache_size			○	○	○			○	○
pd_uap_exerror_log_dir	○	○						○	○
pd_uap_exerror_log_param_size※6	○	○						○	○
pd_uap_exerror_log_size	○	○						○	○
pd_uap_exerror_log_use※6	○			○	○			○	○
pd_unit_id		○						×	×
pd_utl_buff_size	○							○※2	○※2

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pd_utl_exec_mode	○							×	○
pd_utl_exec_time	○							○	○
pd_utl_file_buff_size	○							○	○
pd_view_def_cache_size			○	○	○			○※2，3	○※2
pd_watch_pc_client_time※6			○	○	○			○	○
pd_watch_resource	○							○	○
pd_watch_time	○	○						○	○
pd_work_buff_expand_limit			○	○		○	○	○	○
pd_work_buff_mode			○	○		○	○	○	○
pd_work_buff_size			○	○		○	○	○	○
pd_work_table_option	○							○	○
pdaudload	○							○	○
pdbuffer	○							×	○
pdcltgrp	○							○	○
pdhagroup	○							×	×
pdhibegin	○							○	○
pdhubopt					○			×	○
pdlogadfg -d spd				○	○	○	○	○※1	○※1
pdlogadfg -d sys※5				○	○	○	○	○※1	○※1
pdlogadpf -d spd				○	○	○	○	○※1	○※1
pdlogadpf -d sys※5				○	○	○	○	○※1	○※1
pdmlgput	○							○	○
pdplgprm				○	○	○	○	○	○
pdplugin	○							×	×
pdstart	○							×	×
pdstbegin	○							○	○
pdunit	○							×	×
pdwork				○		○	○	×	×

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否
	SYS	UNT	SVR	SDS	FES	DS	BES		
pdwork_wrn_pnt	○							○	○
SHMMAX	○	○						○※7	○※7
TZ	○							○	○

（凡例）

○：指定値を変更できます。

×：指定値を変更できません。

◎：整合性を保って変更してください。

空白：該当しません。

SYS：システム共通定義

UNT：ユニット制御情報定義

SVR：サーバ共通定義

SDS：シングルサーバ定義

FES：フロントエンドサーバ定義

DS ：ディクショナリサーバ定義

BES：バックエンドサーバ定義

注※1

オペランドの追加だけができます。オペランドの削除，変更はできません。

注※2

オペランドの指定値を小さくすると，HiRDB を再開始できない場合があります。その場合は，指定値を変更前の値に戻して再開始してください。

注※3

オペランドの指定値を大きくすると，HiRDB を再開始できない場合があります。その場合は，指定値を変更前の値に戻して再開始してください。

注※4

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定した場合，HiRDB 再開始時に定義を変更しても，変更前の値で HiRDB を起動します。

注※5

HiRDB Datareplicator（抽出側）と連携している場合に注意が必要なオペランドです。このオペランドの指定を追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度停止してから再開始してください。HiRDB Datareplicator を稼働したまま，この定義を変更して HiRDB を再度開始すると，HiRDB Datareplicator による抽出が失敗することがあります。

注※6

次に示す HiRDB システム定義のオペランドの値は，クライアント環境定義でクライアントごとに変更できます。クライアントごとに値を変更する場合は，クライアント環境定義の該当するオペランドを指定してください。クライアント環境定義については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

HiRDB システム定義の オペランド	クライアント環境定義の オペランド
pd_additional_optimize_level	PDADDITIONALOPTLVL
pd_cwaittime_wrn_pnt	PDCWAITTIMEWRNPNT
pd_delete_reserved_word_file	PDDELRSVWDFILE
pd_ha_transaction	PDHATRNQUEUING
pd_hash_table_size	PDHASHTBLSIZE
pd_hashjoin_hashing_mode	PDHJHASHINGMODE
pd_optimize_level	PDSQLOPTLVL
pd_space_level	PDSPACELVL
pd_uap_exerror_log_param_size	PDUAPEXERLOGPRMSZ
pd_uap_exerror_log_use	PDUAPEXERLOGUSE
pd_watch_pc_client_time	PDSWATCHTIME

注※7

システム共通定義（pdsys）に pd_dbbuff_modify=Y を指定している場合，×（SHMMAX 指定値の変更はできない）となります。

1.4.2　オペランドの優先順位について

異なる定義で同じオペランドを指定した場合，HiRDB は次に示す優先順位に従って指定値を決定します。

1. シングルサーバ定義，フロントエンドサーバ定義，ディクショナリサーバ定義，バックエンドサーバ定義
2. サーバ共通定義
3. ユニット制御情報定義
4. システム共通定義

優先順位の低い定義で標準となる値を指定して，優先順位の高い定義で値を変更できます。

なお，すべての定義でオペランドを省略すると，優先順位の低い定義の省略値が仮定されます。

1.5　バージョンによって省略値が異なるオペランド，及び指定不要になったオペランド

(1)　バージョンによって省略値が異なるオペランド

バージョンによって省略値が異なるオペランドを次の表に示します。

バージョンアップした場合に，オペランドの省略値を変更したくないときは pd_sysdef_default_option オペランドに，v6compatible 又は v7compatible を指定してください。

表 1-8　バージョンによって省略値が異なるオペランド

オペランド名	recommendable （最新バージョンの省略値）	v7compatible （HiRDB Version 7 の省略値）	v6compatible （HiRDB Version 6 以前の省略値）
pd_additional_optimize_level	"COST_BASE_2"	"COST_BASE_2"	"NONE"
pd_assurance_index_no	500（HiRDB の正常開始時に自動的に増加しない）	• HiRDB/シングルサーバの場合 データディクショナリ表のインデクス数（124）＋ 50（HiRDB の正常開始時に自動的に増加する） • HiRDB/パラレルサーバの場合 50（HiRDB の正常開始時に自動的に増加する）	• HiRDB/シングルサーバの場合 データディクショナリ表のインデクス数（124）＋ 50（HiRDB の正常開始時に自動的に増加する） • HiRDB/パラレルサーバの場合 50（HiRDB の正常開始時に自動的に増加する）
pd_assurance_table_no	500（HiRDB の正常開始時に自動的に増加しない）	100（HiRDB の正常開始時に自動的に増加する）	0（HiRDB の正常開始時に自動的に増加しない）
pd_aud_async_buff_count	Max（1，ユニット内 HiRDB サーバ数×10）	3	3
pd_aud_async_buff_size	401408	4096	4096
pd_bes_shmpool_size	HiRDB による自動計算	HiRDB による自動計算	1024
pd_cancel_down_msgchange	Y	N	N
pd_change_clt_ipaddr	1	0	0
pd_dbbuff_lru_option	MIX	SEPARATE	SEPARATE
pd_dic_shmpool_size	HiRDB による自動計算	HiRDB による自動計算	1024
pd_large_file_use	• 08-05 以降のバージョン：Y • 08-04 以前のバージョン：N	N	N
pd_lck_hash_entry	0（HiRDB による自動計算）	0（HiRDB による自動計算）	11261

オペランド名	recommendable（最新バージョンの省略値）	v7compatible（HiRDB Version 7 の省略値）	v6compatible（HiRDB Version 6 以前の省略値）
pd_lck_pool_size	16000（64 ビットモードの場合は 32000）	16000（64 ビットモードの場合は 32000）	1024
pd_lck_wait_timeout	Max（180, pd_watch_time の値）	Max（180, pd_watch_time の値）	180
pd_log_max_data_size	400000	32000	32000
pd_log_sdinterval	• システムログ出力量：5000 • 経過時間：60	• システムログ出力量：1000 • 経過時間：60	• システムログ出力量：1000 • 経過時間：60
pd_log_write_buff_count	10	3	3
pd_max_access_tables	64	64	16
pd_max_commit_write_reclaim_no	10	0	0
pd_max_server_process	HiRDB による自動計算	HiRDB による自動計算	100
pd_ntfs_cache_disable	Y	N	N
pd_optimize_level	pd_optimize_level オペランドを参照してください。	pd_optimize_level オペランドを参照してください。	"SELECT_APSL"
pd_pageaccess_mode	SNAPSHOT	SNAPSHOT	NORMAL
pd_process_terminator	fixed	fixed	nonresident
pd_process_terminator_max	Max（3, ↑pd_max_users の値÷100↑）	↑pd_max_users の値÷100↑	↑pd_max_users の値÷100↑
pd_sds_shmpool_size	HiRDB による自動計算	HiRDB による自動計算	1024
pd_spool_cleanup_interval_level	• 日数：7 • 削除種別：all	• 日数：7 • 削除種別：dump	• 日数：7 • 削除種別：dump
pd_spool_cleanup_level	• 日数：7 • 削除種別：all	• 日数：7 • 削除種別：dump	• 日数：7 • 削除種別：dump
pd_thdlock_retry_time	1000	10000	10000
pd_trn_commit_optimize	ONEPHASE	ONEPHASE	NOUSE
pd_trn_send_decision_interval	pd_trn_send_decision_intval_sec オペランドの値	5	5
pd_work_buff_mode	pool	pool	each
pd_work_table_option	1	1	0
SHMMAX	200（32 ビットモードの場合だけ省略値が変更になります）	200（32 ビットモードの場合だけ省略値が変更になります）	6

(2)　指定不要になったオペランド

バージョンアップによって, 次に示すオペランドを指定する必要がなくなりました。バージョンアップした場合に，これらのオペランドを指定したままでもエラーにはなりません。

バージョン 08-00 から指定する必要がなくなったオペランド

- pd_dynamic_sql_object_cache

バージョン 07-00 から指定する必要がなくなったオペランド

- pd_multi_fes
- pd_redo_skip_inf

2 システム共通定義

この章では，システム共通定義の各オペランドの内容について説明します。

2.1　オペランドの形式

システム共通定義では HiRDB 全体の構成や，HiRDB のユニットに共通な情報を定義します。ここでは，システム共通定義の各オペランドの形式について説明します。なお，表中の「番号」は，「2.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　システム構成

番号	形式
1	set pd_system_id = HiRDB 識別子
2	〔set pd_name_port = HiRDB のポート番号〕
3	set pd_master_file_name = "マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称"

(2)　同時実行最大数

番号	形式
4	〔set pd_max_users = 最大同時接続数〕
5	〔set pd_max_server_process = 最大同時起動サーバプロセス数〕
6	〔set pd_max_access_tables = 同時アクセス可能実表数〕
7	〔set pd_utl_exec_mode = 0 ｜ 1 〕
8	〔set pd_max_commit_write_reclaim_no = pdreclaim コマンド（-p オプション）の最大同時実行数〕

(3)　HiRDB の開始方法

番号	形式
9	〔set pd_mode_conf = AUTO ｜ MANUAL1 ｜ MANUAL2〕
10	〔set pd_system_complete_wait_time = pdstart コマンド完了待ち時間〕
11	〔set pd_start_time_out = HiRDB 開始準備処理の最大待ち時間〕

(4)　縮退起動

番号	形式
12	〔set pd_start_level = 0 ｜ 1 〕
13	〔set pd_reduced_check_time = 縮退起動開始の連絡待ち時間〕
14	〔set pd_start_skip_unit = 開始しないユニットの名称〔，開始しないユニットの名称〕…〕

(5)　HiRDB の処理方式

番号	形式
15	〔set pd_dbsync_point = sync｜commit〕
16	〔set pd_system_dbsync_point = sync｜commit〕
17	〔set pd_dbsync_altwrite_skip = Y｜N 〕
18	〔set pd_overflow_suppress = Y｜N 〕
19	〔set pd_process_terminator = resident｜fixed｜nonresident〕
20	〔set pd_process_terminator_max = 後処理プロセスの最大常駐数〕
21	〔set pd_process_desktopheap_size = 1 プロセス当たりのデスクトップヒープ消費量〕
22	〔set pd_space_level = 0｜1｜3 〕
23	〔set pd_dec_sign_normalize = Y｜N 〕
24	〔set pd_sql_dec_op_maxprec = 演算結果が 29 けた以下の DECIMAL 型となる精度の最大値〕
25	〔set pd_server_entry_queue = spnfifo｜fifo｜loop〕
26	〔set pd_thdlock_sleep_func = 0｜1〕
27	〔set pd_thdlock_wakeup_lock = Y｜N〕
28	〔set pd_thdlock_pipe_retry_interval = スレッド間ロックの解放調査間隔〕
29	〔set pd_thdlock_retry_time = スレッド間ロックスリープ時間〕
30	〔set pd_thdspnlk_spn_count = スレッド間スピンロックのスピン回数〕
31	〔set pd_pageaccess_mode = SNAPSHOT ｜ NORMAL〕
32	〔set pd_cmdhold_precheck = Y｜N〕
33	〔set pd_db_io_error_action = dbhold｜unitdown〕
34	〔set pd_connect_errmsg_hide = Y｜N〕
35	〔set pd_rpc_bind_loopback_address = Y｜N〕
36	〔set pd_cancel_down_msgchange = Y｜N〕

(6)　全面回復処理

番号	形式
37	〔set pd_max_recover_process = 全面回復処理の並列実行プロセス数〕
38	〔set pd_redo_allpage_put = Y｜N〕

(7)　トランザクション決着処理

番号	形式
39	〔set pd_trn_rerun_branch_auto_decide = Y｜N〕

番号	形式
40	〔set pd_trn_send_decision_intval_sec = 秒単位でのトランザクション自動決着の送信リトライ時間間隔〕
41	〔set pd_trn_send_decision_interval = 分単位でのトランザクション自動決着の送信リトライ時間間隔〕
42	〔set pd_trn_send_decision_retry_time = トランザクション自動決着の最大待ち時間〕
43	〔set pd_trn_watch_time =トランザクション同期点処理時の最大通信待ち時間〕
44	〔set pd_trn_commit_optimize = ONEPHASE｜NOUSE〕

(8)　SQL の最適化

番号	形式
45	〔set pd_optimize_level = SQL 最適化オプション〔，SQL 最適化オプション〕…〕
46	〔set pd_additional_optimize_level = SQL 拡張最適化オプション〔，SQL 拡張最適化オプション〕…〕
47	〔set pd_hashjoin_hashing_mode = TYPE1｜TYPE2〕
48	〔set pd_hash_table_size = ハッシュ表サイズ〕
49	〔set pd_work_table_option = 作業表処理オプション〕

(9)　絞込み検索

番号	形式
50	〔set pd_max_list_users = 同時リスト所有可能ユーザ数〕
51	〔set pd_max_list_count = 1 ユーザ当たりのリスト作成数〕
52	〔set pd_list_initialize_timing = INITIAL｜DEFER｜STANDBY〕
53	〔set pd_apply_search_ats_num = サーチ条件 ATS を適用する絞り込み値の組み合わせ個数の上限〕

(10)　システム監視

番号	形式
54	〔et pd_cmd_process_conwaittime = 運用コマンド・ユティリティ実行におけるプロセス割り当て監視時間〕
55	〔set pd_utl_exec_time = ユティリティの実行監視時間〕
56	〔set pd_watch_time = 最大応答待ち時間〕
57	〔set pd_queue_watch_time = メッセージキュー監視時間〕
58	〔set pd_queue_watch_timeover_action = continue｜stop〕
59	〔set pd_down_watch_proc = サーバプロセスの異常終了回数の上限値〔，監視間隔〕〕
60	〔set pd_host_watch_interval = ホスト間監視時間間隔〕
61	〔set pd_watch_resource = MANUAL｜AUTO〕
62	〔set pd_max_users_wrn_pnt = HiRDB サーバへの接続数に関する警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕〕

番号	形式
63	〔set pd_max_access_tables_wrn_pnt = 同時アクセス可能実表数に関する警告メッセージの出力契機〕
64	〔set pd_max_rdarea_no_wrn_pnt = RD エリア数に関する警告メッセージの出力契機〕
65	〔set pd_max_file_no_wrn_pnt = HiRDB ファイル数に関する警告メッセージの出力契機〕
66	〔set pdwork_wrn_pnt = 作業表用ファイルに関する警告メッセージの出力契機〕
67	〔set pd_max_list_users_wrn_pnt = リスト作成ユーザ数に関する警告メッセージの出力契機〕
68	〔set pd_max_list_count_wrn_pnt = 1 ユーザ当たりのリスト作成数に関する警告メッセージの出力契機〕
69	〔set pd_rdarea_list_no_wrn_pnt = サーバ内のリスト作成数に関する警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕〕

(11)　SQL 実行時間警告出力機能

番号	形式
70	〔set pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜SQL 実行時間警告情報の出力条件（時間指定）〕
71	〔set pd_cwaittime_report_dir = SQL 実行時間警告情報ファイルの出力先ディレクトリ〕
72	〔set pd_cwaittime_report_size = SQL 実行時間警告情報ファイルの最大容量〕

(12)　拡張 SQL エラー情報出力機能

番号	形式
73	〔set pd_uap_exerror_log_use = YES｜NO〕
74	〔set pd_uap_exerror_log_dir = SQL エラーレポートファイルの格納ディレクトリ〕
75	〔set pd_uap_exerror_log_size = SQL エラーレポートファイルの最大サイズ〕
76	〔set pd_uap_exerror_log_param_size = エラーログファイル及び SQL エラーレポートファイルへ出力するパラメタ情報の最大データ長〕

(13)　SQL 予約語削除機能

番号	形式
77	〔set pd_delete_reserved_word_file = SQL 予約語削除ファイル名称 1〔，SQL 予約語削除ファイル名称 2〕…〕

(14)　SQL からのコマンド実行

番号	形式
78	〔set pd_sql_command_exec_users = 認可識別子〔，認可識別子〕…〕

(15)　SQLSTATE の細分化

番号	形式
79	〔set pd_standard_sqlstate = Y｜N〕

(16)　排他制御

番号	形式
80	〔set pd_lck_deadlock_info = Y｜N〕
81	〔set pd_lck_wait_timeout = 排他待ち限界経過時間〕
82	〔set pd_lck_release_detect =interval｜pipe〕
83	〔set pd_lck_release_detect_interval = 排他解除検知インターバル時間〕
84	〔set pd_nowait_scan_option = LOCK｜NOLOCK〕
85	〔set pd_lck_queue_limit = 排他待ちユーザ数警告メッセージ出力契機〕
86	〔set pd_deadlock_priority_use = Y｜N〕
87	〔set pd_command_deadlock_priority = 32｜64｜96｜120〕
88	〔set pd_key_resource_type = TYPE1｜TYPE2〕
89	〔set pd_indexlock_mode = {KEY｜NONE}〕
90	〔set pd_lock_uncommited_delete_data = WAIT｜NOWAIT〕
91	〔set pd_lck_deadlock_check = Y｜N〕
92	〔set pd_lck_deadlock_check_interval = デッドロック監視時間間隔〕

(17)　バッファ

番号	形式
93	〔set pd_sql_object_cache_size = SQL オブジェクト用バッファ長〕
94	〔set pd_def_buf_control_area_assign = INITIAL｜TRAN〕
95	〔set pd_thread_max_stack_size = 1 スレッドが使用する最大スタックサイズ〕

(18)　共用メモリ

番号	形式
96	〔set pd_shmpool_attribute = free｜fixed〕
97	〔set pd_dbbuff_attribute = free｜fixed〕

(19)　メッセージログファイル

番号	形式
98	〔set pd_mlg_msg_log_unit = manager｜local〕
99	〔set pd_mlg_file_size = メッセージログファイルの最大容量〕

(20)　統計情報

番号	形式
100	〔set pd_statistics = Y｜N〕
101	〔set pd_stj_file_size = 統計ログファイルの最大容量〕
102	〔set pd_stj_buff_size = 統計ログバッファ長〕

(21)　RPC トレース情報

番号	形式
103	〔set pd_rpc_trace = Y｜N〕
104	〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕
105	〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕

(22)　トラブルシュート情報

番号	形式
106	〔set pd_cancel_dump = put ｜ noput〕
107	〔set pd_client_waittime_over_abort = Y｜N〕
108	〔set pd_clt_waittime_over_dump_level = all｜shm_fesonly〕
109	〔set pd_dump_suppress_watch_time =トラブルシュート情報の出力抑止時間〕
110	〔set pd_debug_info_netstat = Y ｜ N〕
111	〔set pd_pth_trace_max = 通信トレース格納最大数〕
112	〔set pd_spool_cleanup_interval = トラブルシュート情報の削除処理間隔〕
113	〔set pd_spool_cleanup_interval_level = 日数〔，削除種別〕〕
114	〔set pd_spool_cleanup = normal｜force｜no〕
115	〔set pd_spool_cleanup_level = 日数〔，削除種別〕〕
116	〔set pd_module_trace_max = モジュールトレース格納最大数〕
117	〔set pd_module_trace_timer_level = モジュールトレース出力時刻取得方法〕

(23)　RD エリア

番号	形式
118	〔set pd_max_rdarea_no = RD エリアの最大数〕
119	〔set pd_max_file_no = RD エリアを構成する HiRDB ファイルの最大数〕
120	〔set pd_rdarea_warning_point = セグメント使用比率 1〔,セグメント使用比率 2〔,セグメント使用比率 3〕〕〕
121	〔set pd_rdarea_warning_point_msgout = Y｜N〕
122	〔set pd_rdarea_expand_format = Y｜N〕
123	〔set pd_rdarea_extension_timing = use｜nouse〕
124	〔set pd_rdarea_open_attribute_use = Y｜N〕
125	〔set pd_rdarea_open_attribute = INITIAL｜DEFER｜SCHEDULE〕
126	〔set pd_shared_rdarea_use = Y｜N〕
127	〔set pd_db_access_error_action = dbhold｜unitdown〕
128	〔set pd_db_hold_action = dbhold｜unitdown〕

(24)　グローバルバッファ

番号	形式
129	〔set pd_dbbuff_lru_option = SEPARATE｜MIX〕
130	〔set pd_dbbuff_binary_data_lru = Y｜N〕
131	〔set pd_dbbuff_modify = Y｜N〕
132	〔set pd_dbbuff_lock_release_detect = pipe｜interval｜switch〕
133	〔set pd_dbbuff_lock_spn_count = 排他獲得待ち処理中のスピン回数〕
134	〔set pd_dbbuff_lock_interval = 排他獲得待ち処理中のインターバル時間〕
135	〔set pd_dbbuff_wait_interval = グローバルバッファの占有状態の調査間隔〕
136	〔set pd_dbbuff_wait_spn_count = グローバルバッファの占有状態調査のスピンループ回数上限値〕
137	〔set pd_dbbuff_rate_updpage = デファードライトトリガの要求比率〕
138	〔set pd_dbbuff_trace_level = グローバルバッファ制御情報トレース取得レベル〕

(25)　インメモリデータ処理

番号	形式
139	〔set pd_max_resident_rdarea_no = インメモリ RD エリアの最大数〕
140	〔set pd_max_resident_rdarea_shm_no = インメモリデータバッファが使用する共用メモリセグメントの最大数〕

(26)　表又はインデクスの予約数

番号	形式
141	〔set pd_assurance_table_no = 表予約数の最低保証値〕
142	〔set pd_assurance_index_no = インデクス予約数の最低保証値〕

(27)　参照制約及び検査制約

番号	形式
143	〔set pd_constraint_name = LEADING｜TRAILING〕
144	〔set pd_check_pending = USE｜NOUSE〕

(28)　HiRDB ファイルシステム領域

番号	形式
145	〔set pd_large_file_use = Y｜N〕
146	〔set pd_ntfs_cache_disable = Y｜N〕

(29)　再編成時期予測機能

番号	形式
147	〔set pd_rorg_predict = Y｜N〕

(30)　セキュリティ

番号	形式
148	〔set pd_audit = Y｜N〕
149	〔set pd_aud_file_name = 監査証跡ファイル用の HiRDB ファイルシステム領域名〕
150	〔set pd_aud_max_generation_size = 1 監査証跡ファイルの最大容量〕
151	〔set pd_aud_max_generation_num = 監査証跡ファイルの最大数〕
152	〔set pd_aud_no_standby_file_opr = down｜forcewrite〕
153	〔set pd_aud_async_buff_size = 監査証跡の非同期出力時に使用するバッファ長〕
154	〔set pd_aud_async_buff_count = 監査証跡の非同期出力時に使用するバッファ面数〕
155	〔set pd_aud_async_buff_retry_intvl = 監査証跡の非同期出力時に使用するバッファの確保リトライ間隔〕
156	〔set pd_aud_sql_source_size = 監査証跡に出力する SQL 文のサイズ〕
157	〔set pd_aud_sql_data_size = 監査証跡に出力する SQL データのサイズ〕
158	〔set pd_aud_file_wrn_pnt = 警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕〕
159	〔set pd_aud_auto_loading = Y｜N〕

番号	形式
160	〔pdaudload〔-i インデクス作成方法〕 〔-l ログ取得方式〕 〔-n 〔一括出力用ローカルバッファ面数〕,, 〔ランダムアクセス用ローカルバッファ面数〕〕 〔-y〕 〔-X サーバ間通信の応答監視時間〕 〔-S ソート用バッファサイズ〕〕
161	〔set pd_security_host_group = ホスト名〔, ホスト名〕…〕

(31)　系切り替え機能

番号	形式
162	〔set pd_ha = use｜nouse〕
163	〔set pd_ha_ipaddr_inherit = Y｜N〕
164	〔set pd_ha_switch_timeout = Y｜N〕
165	〔set pd_ha_mgr_rerun = wait｜notwait〕
166	〔set pd_ha_transaction = error｜queuing〕
167	〔set pd_ha_trn_queuing_wait_time = トランザクションのキューイング待ち時間〕
168	〔set pd_ha_trn_restart_retry_time = トランザクション開始要求エラー時のリトライ時間の上限〕
169	〔set pd_ha_resource_act_wait_time = リソース活性化の最大待ち時間〕
170	〔set pd_deter_restart_on_stop_fail = Y｜N〕

(32)　HiRDB Datareplicator

番号	形式
171	〔set pd_rpl_init_start = Y｜N〕
172	〔set pd_rpl_reflect_mode = server｜uap〕
173	〔set pd_log_rpl_no_standby_file_opr = stop｜continue〕
174	〔set pd_rpl_func_control = BACKWARD_CUTOFF_UPDATE｜NONE〕

(33)　JP1 との連携

番号	形式
175	〔set pd_jp1_use = Y｜N〕
176	〔set pd_jp1_event_level = 1｜2〕
177	〔set pd_jp1_event_msg_out = Y｜N〕

(34)　ディレクトリサーバ連携機能

番号	形式
178	〔set pd_directory_server = sods〕

(35)　HiRDB External Data Access 機能

番号	形式
179	〔set pd_max_foreign_server = 外部サーバ数の最大値〕

(36)　OLTP

番号	形式
180	〔set pd_oltp_holdcr = use｜nouse〕

(37)　バージョンアップ

番号	形式
181	〔set pd_auto_vrup = Y｜N〕

(38)　通信処理

番号	形式
182	〔set pd_sysdef_default_option = recommendable｜v6compatible｜v7compatible〕
183	〔set pd_service_port = スケジューラプロセスのポート番号〕
184	〔set pd_name_fixed_port_lookup = Y｜N〕
185	〔set pd_scd_port = スケジューラプロセスのポート番号〕
186	〔set pd_trn_port = トランザクションサーバプロセスのポート番号〕
187	〔set pd_mlg_port = メッセージログサーバプロセスのポート番号〕
188	〔set pd_alv_port = ユニット監視プロセスのポート番号〕
189	〔set pd_change_clt_ipaddr = 0｜1 〕
190	〔set pd_registered_port = "ポート番号の予約範囲"〔,"ポート番号の予約範囲"〕…〕
191	〔set pd_registered_port_check = Y｜N｜C｜W〕
192	〔set pd_registered_port_level = 0｜1〕
193	〔set pd_ipc_send_retrycount = プロセス間通信の送信リトライ処理の連続繰り返し回数〕
194	〔set pd_ipc_send_retrysleeptime = プロセス間通信の送信リトライ処理のスリープ時間〕
195	〔set pd_ipc_send_count = サーバ間の送信処理のリトライ回数〕
196	〔set pd_ipc_recv_count = サーバ間の受信処理のリトライ回数〕

番号	形式
197	〔set pd_ipc_conn_nblock = Y｜N〕
198	〔set pd_ipc_conn_nblock_time = ノンブロックモードでのコネクション確立監視時間〕
199	〔set pd_ipc_conn_interval = コネクション確立処理のリトライ間隔〕
200	〔set pd_ipc_conn_count =コネクション確立処理のリトライ回数〕
201	〔set pd_ipc_inet_bufsize = サーバ間通信で使用する送受信バッファ長〕
202	〔set pd_tcp_inet_bufsize = HiRDB クライアントとの通信で使用する送受信バッファ長〕
203	〔set pd_utl_buff_size = ユティリティの通信用バッファ長〕
204	〔set pd_utl_file_buff_size = ユティリティのファイル用バッファ長〕
205	〔set pd_sql_send_buff_size = SQL 実行時のサーバ間通信用バッファ長〕

(39)　Java

番号	形式
206	〔set pd_java_option = "Java オプション"〔,"Java オプション"〕…〕
207	〔set pd_java_routine_stack_size = 外部 Java ルーチンが使用するスタック領域長〕
208	〔set pd_java_archive_directory = "JAR ファイル格納ディレクトリ"〕
209	〔set pd_java_classpath = "Java クラスパス"〕
210	〔set pd_java_runtimepath = "Java Runtime Environment のルートディレクトリ"〕
211	〔set pd_java_libpath = "Java 仮想マシンのライブラリディレクトリ"〕
212	〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕

(40)　外部 C ストアドルーチン

番号	形式
213	〔set pd_c_library_directory = "C ライブラリファイル格納ディレクトリ"〕

(41)　文字コード

番号	形式
214	〔set pd_substr_length = 3｜4｜5｜6〕

(42)　日付・時刻

番号	形式
215	〔set pd_leap_second = Y｜N〕

(43)　ユニット構成

番号	形式
216	{{pdunit -x ホスト名 -u ユニット識別子 〔-d "HiRDB 運用ディレクトリ名"〕 〔-c ホスト名〕 〔-p HiRDB のポート番号〕 〔-s スケジューラプロセスのポート番号〕 〔-t トランザクションサーバプロセスのポート番号〕 〔-m メッセージログサーバプロセスのポート番号〕 〔-a ユニット監視プロセスのポート番号〕}}

(44)　サーバ構成

番号	形式
217	{{pdstart -t サーバ種別 〔-s サーバ名〕 -x ホスト名｜-u ユニット識別子 〔-m ホスト名〔，ホスト名〕… 〔-n ホスト名〔，ホスト名〕…〕〕 〔-c サーバ名｜-g HA グループ識別子〕 〔-k stls〕}}

(45)　グローバルバッファ

番号	形式
218	〔{{pdbuffer -a グローバルバッファ名 {-r RD エリア名〔，RD エリア名〕…｜ -b RD エリア名〔，RD エリア名〕…｜ -o｜ -i 認可識別子．インデクス識別子} -n バッファ面数〔-l バッファサイズ〕 〔-m 同時実行最大プリフェッチ数〕 〔-p 一括入力最大ページ数〕 〔-w デファードライトトリガ時の更新ページ出力比率〕 〔-c〕 〔-y デファードライトトリガ契機の更新バッファ面数〕}}〕

(46)　HA グループ

番号	形式
219	〔pdhagroup -g HA グループ識別子 -u ユニット識別子〔，ユニット識別子〕…〕

(47)　統計情報

番号	形式
220	〔pdstbegin〔-k 統計情報種別〔，統計情報種別〕…〕 〔-m 時間間隔〕 〔{-x ホスト名｜-u ユニット識別子}〕 〔{-a｜-s サーバ名〔，サーバ名〕…}〕〕

(48)　クライアントグループ

番号	形式
221	〔pdhibegin -k 履歴情報種別〔，履歴情報種別〕…〕
222	{{{pdcltgrp -g クライアントグループ名称 -u グループごとの接続保証ユーザ数}}}

(49)　プラグイン

番号	形式
223	{{{pdplugin -n プラグイン名称}}}

(50)　共用メモリ

番号	形式
224	〔putenv SHMMAX 共用メモリセグメントサイズの上限値〕

(51)　日付・時刻

番号	形式
225	〔putenv TZ タイムゾーン〕

(52)　メッセージの出力抑止機能

番号	形式
226	〔pdmlgput -s 出力有無 {-c ALL｜〔-l メッセージの重要度〕 -m メッセージ ID〔，メッセージ ID〕…}〕

2.2　オペランドの説明

2.2.1　システム構成に関するオペランド

1）　pd_system_id = HiRDB 識別子

～<識別子>((4 文字))

HiRDB サーバの識別子を指定します。このオペランドは省略できません。必ず指定してください。

複数の HiRDB/シングルサーバを使用している場合は，システム間で HiRDB 識別子が一意になるようにしてください。

《注意事項》

一度設定した HiRDB 識別子を変更するには, データベース初期設定ユティリティで再度システムを構築し直す必要があります。したがって，後で変更が発生するような名称は避けてください。

2）　pd_name_port = HiRDB のポート番号

～<符号なし整数>((5001～65535))《20000》

HiRDB のポート番号を指定します。

《注意事項》

HiRDB のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pdunit オペランドの-p オプションと関連があります。

3）　pd_master_file_name = "マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称"

～<パス名>((167 文字以内))

マスタディレクトリ用 RD エリアを構成する先頭の HiRDB ファイル名称を, 絶対パス名で指定します。このオペランドは省略できません。必ず指定してください。

《注意事項》

- 一度設定したマスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を変更するには, データベース初期設定ユティリティで再度システムを構築し直す必要があります。したがって, 後で変更が発生するような名称は避けてください。
- HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが, HiRDB ファイル名は大文字と小文字の区別をします。

  C:¥rdarea¥mast¥mast01 の場合，C:¥rdarea¥mast については大文字と小文字の区別をしません。mast01 については大文字と小文字の区別をします。

2.2.2　同時実行最大数に関するオペランド

4）　pd_max_users = 最大同時接続数

～<符号なし整数>

- HiRDB/シングルサーバの場合：((1～3000))《10》
- HiRDB/パラレルサーバの場合：((1～2000))《10》

HiRDB サーバに対する最大同時接続数を指定します。HiRDB/パラレルサーバの場合は 1 フロントエンドサーバに対する最大同時接続数を指定します。HiRDB サーバへの接続要求数がこのオペランドの

値を超えると，HiRDB サーバはそれ以上の接続要求を受け付けません（接続要求はエラーになります）。なお，ここでいう接続とは，CONNECT 文による HiRDB サーバへの接続を意味しています。

《指定値の目安》

- OpenTP1 を使用する場合は，HiRDB サーバに接続する（XA 接続を含む）OpenTP1 のサーバプロセス数が接続数になります。
- DBPARTNER/Server を使用する場合は，DBPARTNER/Server の接続クライアント数が接続数になります。
- HiRDB SQL Executer を使用する場合は, HiRDB SQL Executer の接続クライアント数が接続数になります。
- 複数接続機能を使用する場合は，各 UAP の同時接続数の総和が接続数になります。
- HiRDB のコマンド及びユティリティの中には，内部的に HiRDB に接続するものがあるため，これらのコマンド及びユティリティの実行中は接続できるユーザ数が一時的に減少します。このことを考慮してオペランドの値を決めてください。コマンド及びユティリティの接続数については，マニュアル「HiRDB Version 8 コマンドリファレンス」の「コマンドの同時実行数」を参照してください。

《注意事項》

- pd_max_users オペランドの値を大きくした場合は，pd_max_server_process オペランドの値も大きくする必要があります。また，これに伴い HiRDB が使用する共用メモリ，及びポート数が増加します。pd_max_users オペランドの値を大きくした場合（ユーザ数が増えた場合）の注意事項については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。なお，pd_max_server_process オペランドを省略すると，このオペランドの値を HiRDB が自動計算します。
- マルチフロントエンドサーバの場合は，最大で pd_max_users オペランドの値×フロントエンドサーバ数の接続要求を処理できますが, pd_max_users オペランドの値×フロントエンドサーバ数の値が pd_max_users オペランドの指定可能上限値（2000）を超えないようにしてください。
- HiRDB サーバへの接続要求数がこのオペランドの値を超えると，接続要求がエラーとなり，プロセスが残ったままになることがあります。pdls -d prc コマンドでプロセスが残っていないか確認し，残っていた場合は pdcancel コマンドでそのプロセスを強制終了させてください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_hash_entry
- pd_lck_pool_size
- pd_lck_release_detect
- pd_max_bes_process
- pd_max_dic_process
- pd_max_server_process
- pd_process_count
- pd_process_terminator_max
- pdcltgrp

5）　pd_max_server_process = 最大同時起動サーバプロセス数

～<符号なし整数>((50～10000))

ユニット内で同時に起動するサーバのプロセス数の最大値を指定します。サーバのプロセス数には，システムサーバ，各サーバ，ユティリティなどのプロセス数を含めます。システムサーバとは HiRDB が内部的に使用するサーバのことです。

《指定値の目安》

- 通常はこのオペランドを省略してください。このオペランドを省略すると，次に示す計算式の値が仮定されます。なお，変数の説明中に出てくるオペランドの値を変更した場合は HiRDB が自動的に再計算します。また，最大同時起動サーバプロセス数が pd_max_server_process の最大値(10000)を超える場合は，各種定義値を小さくし，定義省略時の計算式が最大値内に収まるようにしてください。

  省略値＝ａ＋ｂ×（ｃ＋30）＋70＋ｉ＋ｊ＋ｋ＋ｎ

  ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合は 100 が仮定されます。

- このオペランドを指定する場合は次に示す計算式を参考にして求めてください。HiRDB/パラレルサーバの場合はユニットごとに求めてください。その結果のうち一番大きな値を目安としてください。

  推奨値の目安＝ａ＋ｂ×（ｃ＋ｄ×ｅ＋ｆ＋3）＋ｄ×ｇ＋ｈ＋ｉ＋ｊ＋ｋ＋ｎ

変数	変数の説明
a	**HiRDB/シングルサーバの場合** pd_max_users の値 **HiRDB/パラレルサーバの場合** ユニット内のサーバごとに次の計算式を実行して求めた値を合計してください。 バックエンドサーバの場合：pd_max_bes_process の値 ディクショナリサーバの場合：pd_max_dic_process の値 フロントエンドサーバの場合：pd_max_users の値 • pd_max_bes_process 又は pd_max_dic_process オペランドを省略している場合は pd_max_users の値で計算してください。 • ユニット内に複数のバックエンドサーバがある場合は，バックエンドサーバごとに計算してください。 • マルチフロントエンドサーバの場合，フロントエンドサーバごとに 1 を加算してください。 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_server_process オペランドの値を代入します。
b	**HiRDB/シングルサーバの場合** 1 **HiRDB/パラレルサーバの場合** ユニット内サーバ数（システム共通定義の pdstart オペランドで該当ユニットに割り当てたサーバ数） • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_servers オペランドの値（省略している場合はデフォルト値）を加算します。
c	全面回復処理の並列実行プロセス数（pd_max_recover_process オペランドの値）
d	同時実行するユティリティの最大数（このオペランドを省略した場合 2 が仮定されます）
e	ユティリティがサーバごとに起動するプロセス数（10 としてください）
f	HiRDB がサーバ制御のために起動するプロセス数（7 としてください）
g	ユティリティがユニット単位に起動するプロセス数（10 としてください）
h	HiRDB がユニット制御のために起動するプロセス数（50 としてください）

変数	変数の説明
i	HiRDB/シングルサーバの場合 1 HiRDB/パラレルサーバの場合 ユニット内バックエンドサーバ数 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_servers オペランドの値を加算します。
j	HiRDB/シングルサーバの場合 pd_max_ard_process オペランドの値 HiRDB/パラレルサーバの場合 ユニット内バックエンドサーバ数×pd_max_ard_process オペランドの値＋ユニット内ディクショナリサーバ数×pd_max_ard_process オペランドの値 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，次に示す値を加算してください。 ゲスト BES の pd_max_ard_process オペランドの最大値×pd_ha_max_act_guest_servers オペランドの値
k	pd_process_terminator オペランドの指定値によって値が変わります。 • resident の場合：1 • fixed の場合（省略値）：pd_process_terminator_max オペランドの値 • nonresident の場合：0
n	HiRDB/シングルサーバの場合 pd_dfw_awt_process オペランドの値 HiRDB/パラレルサーバの場合 バックエンドサーバ数×pd_dfw_awt_process オペランドの値＋ディクショナリサーバ数×pd_dfw_awt_process オペランドの値 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，次に示す値を加算してください。 ゲスト BES の pd_dfw_awt_process オペランドの最大値×pd_ha_max_act_guest_servers オペランドの値

《ほかのオペランドとの関連》

- このオペランドの指定値で，pd_process_count オペランドの上限値が制限されます。
- 各ユニットでサーバの構成が大きく異なるため，ユニットごとに値を調整する場合は，ユニット制御情報定義の pd_max_server_process オペランドを指定してください。

《注意事項》

- この指定値にはユニット内のサーバ，ユティリティなどのプロセス数が含まれます。したがって，この値が小さ過ぎると，次に示す現象が発生することがあります。
  ・ユニット又はサーバの開始処理がエラーになります。
  ・トランザクションの回復処理ができなくなります。
  ・HiRDB の計画停止ができなくなります。
- 実際に起動できるプロセス数は，システムのリソースなどに影響されるため，リソースの調整やサーバの配置変更などが必要となる場合があります。

6）　pd_max_access_tables = 同時アクセス可能実表数
～<符号なし整数>((4～32000))《64》

1 トランザクションで同時にアクセスできる表数と順序数生成子数の合計の最大値を指定します。同時にアクセスできる表数と順序数生成子数とは，1 トランザクション中の SQL 文に記述した表と順序数生成子の延べ数のことです。

《指定値の目安》

- 異なる SQL 文で同じ表，又は順序数生成子を指定している場合，それぞれに表，又は順序数生成子の指定回数を加えてください。
- 一つの SQL 文で同じ表，又は順序数生成子を複数回指定している場合，それぞれに表，又は順序数生成子の指定回数を加えてください。
- ディクショナリ表の参照権限を設定する場合は，同時アクセス可能実表数にアクセスするディクショナリ表の 5 倍の数を加算してください。ディクショナリ表の参照権限については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- HiRDB Datareplicator の反映表定義で load 文を指定した回数以上の値を指定してください。load 文の指定回数よりも小さい値を指定すると，KFPA11931-E エラーとなります。

7)　pd_utl_exec_mode = 0 | 1

ユティリティの最大同時実行数を拡張するかどうかを指定します。

0：

ユティリティの最大同時実行数を拡張しません。この場合，ユティリティの最大同時実行数は，HiRDB が決定します。

1：

ユティリティの最大同時実行数を拡張します。この場合，pd_max_users オペランドの値に応じてユティリティの最大同時実行数が拡張されます。

このオペランドの指定値によって決まるユティリティの最大同時実行数については，マニュアル「HiRDB Version 8 コマンドリファレンス」の「ユティリティの最大同時実行数」を参照してください。

《指定値の目安》

通常は 0 を指定しますが，次に示す場合は 1 を指定してください。

- 0 を指定したときの最大同時実行数では足りない場合
- 次に示す計算式を満たす場合

全バックエンドサーバ数（HiRDB/シングルサーバの場合は 1）×2×全ユティリティの同時実行数≧824

また，次に示す条件式をすべて満たす場合は，このオペランドに 1 を指定すると共用メモリの使用量を少なくできます。

- pd_max_users の値＜ 32
- pd_max_users の値＞ユティリティの最大同時実行数

1 を指定した場合，ユティリティ実行時に必要な共用メモリはユティリティ実行時に動的に確保されます。

8)　pd_max_commit_write_reclaim_no = pdreclaim コマンド（-p オプション）の最大同時実行数～＜符号なし整数＞

- HiRDB/シングルサーバの場合：((0～3000))《10》
- HiRDB/パラレルサーバの場合：((0～2000))《10》

pdreclaim コマンド（-p オプション）の最大同時実行数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ当たりの最大同時実行数を指定します。

0 を指定した場合，pdreclaim コマンド（-p オプション）はエラー終了します。また，pdreclaim コマンド（-p オプション）の同時実行数がこのオペランドの指定値を超えた場合，同時実行数の上限を超えた pdreclaim コマンド（-p オプション）はエラー終了します。

《指定値の目安》

pdreclaim コマンドに-p オプションを指定する場合は，コマンドを実行するサーバ当たりの同時実行数を指定します。pdreclaim コマンドに-p オプションを指定しない場合は 0 を指定します（0 を指定することで共用メモリ使用量を削減できます）。

《注意事項》

このオペランドの指定値を大きくすると，バックエンドサーバ，ディクショナリサーバ，シングルサーバが使用する共用メモリが増加するため，必要に応じてメモリ所要量を見直してください。共用メモリの見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は 0 になります。このため，pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定し，このオペランドを省略すると pdreclaim コマンド（-p オプション）を実行できません。

2.2.3　HiRDB の開始方法に関するオペランド

9)　pd_mode_conf = AUTO | MANUAL1 | MANUAL2

HiRDB の開始方法を指定します。

AUTO：

HiRDB を自動開始します。自動開始とは，OS を起動すると自動的に HiRDB も開始する開始方法です。

ただし, 次に示す場合は手動開始になります。手動開始とは pdstart コマンドを実行して HiRDB を開始する開始方法です。

- 正常終了後の正常開始（OS を再起動しない場合）
- 計画停止後の再開始（OS を再起動しない場合）
- 強制終了後の再開始

MANUAL1：

手動開始します。ただし，異常終了後の再開始は自動開始になります。

MANUAL2：

手動開始します。

《指定値の目安》

- HiRDB（ユニット）が異常終了した場合に HiRDB（ユニット）を自動的に再開始したい場合は，AUTO 又は MANUAL1 を指定します。
- 系切り替え機能を使用する場合は MANUAL1 又は MANUAL2 を指定します。MANUAL1 を指定するか，又は MANUAL2 を指定するかは，系切り替え機能の運用方法によって異なります。系切り替え機能の運用方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

- このオペランドの指定値と前回の終了モード（正常終了，強制終了，計画停止，又は異常終了）の組み合わせによって，次のように開始モードが決まります。

pd_mode_conf の値	前回の終了モード	開始モード	開始方法
AUTO	正常終了	正常開始	自動開始又は手動開始※1
	計画停止	再開始※2	
	強制終了	再開始※2	手動開始
	異常終了	再開始	自動開始
MANUAL1	正常終了	正常開始	手動開始
	計画停止	再開始※2	
	強制終了	再開始※2	
	異常終了	再開始	自動開始
MANUAL2	正常終了	正常開始	手動開始
	計画停止	再開始※2	
	強制終了	再開始※2	
	異常終了	再開始※2	

注※1

自動開始になるのは OS を起動するときだけです。正常終了後の正常開始又は計画停止後の再開始（OS を再起動しない場合）のときは手動開始になります。

注※2

pdstart コマンドの dbdestroy オプションで強制開始できます。ただし，強制開始すると，HiRDB はデータベースの内容を回復しません。したがって, HiRDB 管理者がデータベースを回復する必要があります。

《注意事項》

- HiRDB/パラレルサーバで自動開始（pd_mode_conf=AUTO）をする場合は，最初のユニットを開始してから 20 分以内に全ユニットを開始するようにしてください。20 分以内に全ユニットを開始しないと，HiRDB の開始処理を中止します。この 20 分という制限時間は pd_reduced_check_time オペランドで変更できます。

  なお，ユニットの異常終了，又は OS の異常終了後の再起動ではこのような制限時間はありません。

- pd_mode_conf に AUTO や MANUAL1 を指定した場合でも，HiRDB の開始処理中又は終了処理中に HiRDB が異常終了した場合，次回の開始は必ず手動開始となります。異常終了時のメッセージを確認して対処後，手動で HiRDB を開始してください。

10）　pd_system_complete_wait_time = pdstart コマンド完了待ち時間

～<符号なし整数>((610～3600))《610》（単位：秒）

このオペランドは, pdstart コマンドの完了待ち時間を長くする場合に指定します。HiRDB は, pdstart コマンドが入力されてから 610 秒（10 分 10 秒）を過ぎても開始処理が終了しないと，KFPS05078-I メッセージを出力して pdstart コマンドをエラーリターンしています。この 610 秒を長くする場合に指定します。

なお，pdstart コマンドがエラーリターンしても，開始処理は続行され HiRDB は開始します。したがって，次に示す場合にこのオペランドを指定します。

- KFPS05078-I メッセージを監視対象としている場合
- pdstart コマンドの正常終了後にほかの操作を自動的に行っている場合

《適用基準》

通常，このオペランドを指定する必要はありません。

HiRDB の開始処理時に KFPS05078-I メッセージが出力されて，pdstart コマンドがエラーリターンする場合に指定します。

11）　pd_start_time_out = HiRDB 開始準備処理の最大待ち時間

～<符号なし整数>((1～1440))《15》(単位：分)

pdstart コマンドの入力から HiRDB の開始準備処理の完了までの最大待ち時間を指定します。pdstart コマンドの入力からこのオペランドに指定した時間が経過しても HiRDB の開始準備処理が終了しない場合，pdstart コマンド入力画面に KFPS01861-E メッセージ（reason code = TIMEOUT）が出力されます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。一般的な環境では HiRDB の開始準備処理時間は長くても 3～6 分のため，最大待ち時間はこのオペランドの省略値の 15 分で十分です。サーバマシンの性能，ディスク容量，又はメモリ負荷などの要因によっては，HiRDB の開始準備処理に 15 分以上掛かることがまれにあります。この場合，pdstart コマンド入力画面に KFPS01861-E メッセージ（reason code = TIMEOUT）が出力されます。このとき，このオペランドに 15 より大きい値を指定してください。

参考

HiRDB の開始準備処理の完了待ち時間と HiRDB 開始完了待ち時間は，次に示すメッセージの出力時刻から算出できます。

2.2.4　縮退起動に関するオペランド

12）　pd_start_level = 0 | 1

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB の開始時に開始できないユニットがある場合，縮退起動をするかどうかを指定します。縮退起動については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

0：

縮退起動をしません。障害などで開始できないユニットが一つでもあると HiRDB を開始できません。

1：

縮退起動をします。障害などで開始できないユニットがあっても，HiRDB（それ以外のユニット）を開始できます。

13）　pd_reduced_check_time = 縮退起動開始の連絡待ち時間

～<符号なし整数>((300～1200))《1200》（単位：秒）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

各ユニットからの開始処理完了の連絡待ち時間を指定します。ここで指定した時間を過ぎても，ユニット開始処理完了の連絡がない場合は，そのユニットを除いて縮退起動をします。

《前提条件》

pd_start_level オペランドに 1 を指定する必要があります。

《指定値の目安》

pdstart コマンドを入力してから，KFPS01826-I メッセージ（システムマネジャを定義しているユニットに出力される KFPS01826-I メッセージ）が出力されるまでの時間を調べてください。このオペランドには，それ以上の値を指定してください。

《注意事項》

このオペランドで指定した連絡待ち時間が過ぎても，縮退起動ができる条件をすべて満たしていないと，縮退起動できません。このとき，HiRDB は異常終了します。縮退起動できる条件については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

14）　pd_start_skip_unit = 開始しないユニットの名称〔，開始しないユニットの名称〕…

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

障害などで開始できないユニットがあるため，縮退起動をする場合，そのユニット名称を指定します。ここで指定したユニットは起動されません。

《前提条件》

pd_start_level オペランドに 1 を指定する必要があります。

《利点》

HiRDB を開始するとき，HiRDB は各ユニットからの起動開始の連絡を最大 20 分待ちます。したがって，縮退起動をするとき，最大 20 分間の起動開始の連絡待ち時間が発生します。このオペランドに起動できないユニットを指定すると，このユニットの起動開始の連絡待ちをしません。したがって，最大 20 分間の起動開始の連絡待ち時間が解消されるため，縮退起動の時間を短縮できます。

《注意事項》

- 影響分散スタンバイレス型系切り替え機能を使用している場合，HA グループ内の全ユニットが起動していなくてもよいため，このオペランドの指定は無効になります。ただし，HA グループ内の全バックエンドサーバがどこかのユニットで起動していないと，HiRDB を開始できません。
- 回復不要 FES ユニットをこのオペランドに指定しても，回復不要 FES ユニットは縮退起動の対象になりません。

2.2.5　HiRDB の処理方式に関するオペランド

15)　pd_dbsync_point = sync | commit

データベースの更新内容をファイルに反映するタイミングを指定します。

sync：

シンクポイント時点でデータベースの更新内容をファイルに反映します。シンクポイント間で同一ページを更新するようなトランザクションが多く起動される場合に性能が向上します。COMMIT 文が発行されてもファイルに更新情報を反映しないため，入出力の負荷が軽減されます。ただし，commit を指定したときに比べて全面回復処理が遅くなります。

commit：

COMMIT 文発行時点でデータベースの更新内容をファイルに反映します。トランザクションの完了時点でデータベースの内容が保証されるため，シンクポイント時点からデータベースを回復する必要がなく，全面回復処理に要する時間が短縮できます。ただし，シンクポイント間で同一ページを更新するようなトランザクションが多く起動される場合は，sync を指定したときに比べて性能が低下します。

《備考》

LOB 用 RD エリアはこのオペランドの影響を受けません。ディレクトリ部は COMMIT 文発行時点で反映されます。データ部は LOB 用グローバルバッファを割り当てているかどうかによって処理が異なります。LOB 用グローバルバッファを割り当てていない場合は更新要求時にすぐに反映されます。LOB 用グローバルバッファを割り当てている場合は COMMIT 文発行時点で反映されます。ただし，グローバルバッファが満杯になったときはその時点で反映されます。

《ほかのオペランドとの関連》

このオペランドは pd_system_dbsync_point オペランドと関連があります。

16)　pd_system_dbsync_point = sync | commit

次に示す RD エリアの更新内容をファイルに反映するタイミングを指定します。

- マスタディレクトリ用 RD エリア
- データディレクトリ用 RD エリア
- データディクショナリ用 RD エリア
- データディクショナリ LOB 用 RD エリア
- レジストリ用 RD エリア
- レジストリ LOB 用 RD エリア

sync：

シンクポイント時点で前記の RD エリアの更新内容をファイルに反映します。COMMIT 文が発行されてもファイルに更新情報を反映しないため，定義系 SQL の処理性能が commit を指定したときに比べて若干向上します。ただし，全面回復処理が commit を指定したときに比べて遅くなります。

commit：

COMMIT 文発行時点で前記の RD エリアの更新内容をファイルに反映します。トランザクションの完了時点で前記の RD エリアの更新内容が保証されるため，前記の RD エリアをシンクポイント時点から回復する必要がなく，全面回復処理に要する時間が短縮できます。ただし，sync を指定したときに比べて，定義系 SQL の処理性能が若干低下します。

《ほかのオペランドとの関連》

このオペランドは pd_dbsync_point オペランドと関連があります。pd_dbsync_point オペランドとの関連を次に示します。

pd_dbsync_point オペランドの値	pd_system_dbsync_point オペランドの値	
	sync	commit（省略値）
sync（省略値）	全 RD エリアの更新内容をシンクポイント時点で反映します。	前記の RD エリアの更新内容は COMMIT 文発行時点で反映します。そのほかの RD エリアの更新内容はシンクポイント時点で反映します。
commit	全 RD エリアの更新内容を COMMIT 文発行時点で反映します。	

17）　pd_dbsync_altwrite_skip = Y | N

シンクポイント取得処理中に更新バッファに対して参照要求が発生した場合，その更新バッファの内容をデータベースに書き込む処理は，参照要求をしたトランザクションを実行するサーバプロセスが代行します。この代行処理をスキップするかどうかをこのオペランドで指定します。

このオペランドの指定値の比較を次に示します。

比較項目	pd_dbsync_altwrite_skip オペランドの値	
	Y	N（省略値）
シンクポイント取得処理中に，更新バッファに対して参照要求が発生した場合のデータベース書き込み処理方式	更新バッファの内容をデータベースに書き込む処理は，参照要求をしたトランザクションを実行するサーバプロセスが代行しません（代行処理をスキップします）。	更新バッファの内容をデータベースに書き込む処理は，参照要求をしたトランザクションを実行するサーバプロセスが代行します（代行処理をスキップしません）。
メリット	トランザクションを実行するサーバプロセスで代行処理を行わないため，シンクポイント取得処理中の参照トランザクションの性能が安定します。	参照トランザクションに対して負荷が分散される分，シンクポイント取得処理時間が短くなります。
デメリット	参照トランザクションに対して負荷が分散されない分，シンクポイント取得処理時間が長くなります。	シンクポイント取得処理中の参照トランザクションの性能が低下することがあります。

《指定値の目安》

統計解析ユティリティのグローバルバッファプールに関する統計情報を実行し，シンクポイント時の参照要求ヒットによるデータベース書き込み処理代行回数（ALTRW）を確認してください。この値が大きいと，シンクポイント取得処理中の参照トランザクションの性能が安定しません。安定させたい場合は Y を指定してください。ただし，Y を指定すると，シンクポイント取得処理時間が長くなるため，必要に応じてデファードライト処理の並列 WRITE 機能を使用し，書き込み処理に掛かる負荷を分散させてください。

18）　pd_overflow_suppress = Y | N

演算中のエラー抑止について指定します。エラー抑止の対象を次に示します。

- 演算途中でのオーバフロー
- 0 除算エラー

Y：

SQL 実行時の演算中に対象となるエラーが発生した場合，演算結果をナル値にして処理を続行します。

N：

SQL 実行時の演算中に対象となるエラーが発生した場合，エラーで処理を取り消します。

19）　pd_process_terminator = resident | fixed | nonresident

HiRDB のプロセスが異常終了した場合，HiRDB は後処理を実行するプロセス（これを**後処理プロセス**といいます）を起動して後処理をします。このオペランドでは，後処理プロセスを HiRDB の開始時に起動しておくかどうかを指定します。

resident：

HiRDB の開始時に後処理プロセスを一つ起動します。HiRDB/パラレルサーバの場合は，一つのユニットに一つの後処理プロセスを起動します。

複数のプロセスが同時に異常終了した場合，HiRDB が規定した数まで後処理プロセスを起動して後処理を並列実行します。メモリ不足などで後処理プロセスを新たに起動できない場合は，既に起動している後処理プロセスで後処理を順次実行します。

なお，メモリ不足などによって，HiRDB が規定した数まで後処理プロセスを起動できない場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）が異常終了することがあります。

fixed：

HiRDB の開始時に，pd_process_terminator_max オペランドで指定した数の後処理プロセスを起動します。HiRDB/パラレルサーバの場合は，ユニットごとに pd_process_terminator_max オペランドで指定した数の後処理プロセスを起動します。メモリ不足などで後処理プロセスを起動できない場合は HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を開始しません。

pd_process_terminator_max オペランドの値を超えるプロセスが同時に異常終了した場合，後処理プロセスを追加起動しません。この場合，既に起動している後処理プロセスで後処理を順次実行します。

nonresident：

HiRDB の開始時に後処理プロセスを起動しません。プロセスが異常終了するたびに後処理プロセスを起動します。

複数のプロセスが同時に異常終了した場合，複数の後処理プロセスを同時に起動して後処理を並列実行します。メモリ不足などで後処理プロセスを起動できない場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）が異常終了することがあります。

《指定値の目安》

- 信頼性を向上する場合は resident 又は fixed を指定します。後処理についての性能面は resident より fixed の方が優れていますが，fixed の方がより多くのメモリを必要とします。
- nonresident を指定すると，オンデマンドに後処理プロセスが起動されます。このため，メモリ不足が発生すると，後処理プロセスが起動できなくなります。また，プロセスの異常終了が重なった場合，複数の後処理プロセスが起動されるため，性能面でも好ましくありません。

《注意事項》

指定値を fixed に変更する場合は注意が必要です。HiRDB の開始時に後処理プロセスを起動するため，より多くのメモリが必要になります。メモリ不足などで後処理プロセスを起動できない場合は HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を開始しません。

20）　pd_process_terminator_max = 後処理プロセスの最大常駐数

～<符号なし整数>((1～100))《Max(3，↑pd_max_users の値÷100↑)》

pd_process_terminator オペランドを省略するか，又は fixed を指定した場合にこのオペランドを指定します。このオペランドには，HiRDB の開始時に起動する後処理プロセスの数を指定します。メモリ不足などで，ここで指定した数の後処理プロセスを起動できない場合は，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を開始しません。

《指定値の目安》

必要になる後処理プロセス数は，pd_max_users の値に比例します。このオペランドの値を決める場合はこのことを考慮してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は次の値になります。

↑pd_max_users の値÷100↑

21）　pd_process_desktopheap_size = 1 プロセス当たりのデスクトップヒープ消費量

～<符号なし整数>((50～10000))（単位：バイト）

- Windows 2000，Windows Server 2008，Windows XP，又は Windows Vista の場合：《5000》
- Windows Server 2003 の場合：《100》

HiRDB の 1 プロセス当たりのデスクトップヒープ消費量を指定します。

《指定値の目安》

- 通常は，このオペランドを指定する必要はありません。
- HiRDB のデスクトップヒープ消費量は，次の計算式から求めてください。

  HiRDB のデスクトップヒープ消費量＝ HiRDB が作成するデスクトップ数×デスクトップ 1 面のデスクトップヒープ消費量※1（単位：バイト）

  HiRDB が作成するデスクトップ数＝↑（pd_max_server_process オペランドの指定値＋ 50）÷デスクトップ 1 面のデスクトップヒープ消費量※1 ×pd_process_desktopheap_size オペランドの指定値↑

  デスクトップヒープの上限は，サーバマシン全体で 48 メガバイト※2 です。ほかに起動しているプロセスを含めたデスクトップヒープの合計が 48 メガバイト※2 を超える場合は，次に示す操作を実施してください。

  ・HiRDB のデスクトップヒープ消費量を計算し，pd_max_server_process オペランドの指定値を小さくしてから HiRDB を再起動する

  ・ほかに起動しているプロセスを停止する

  注※ 1　OS のレジストリに指定されている非対話型デスクトップヒープの使用量です。

  注※ 2　Windows 2000 でのデスクトップヒープの使用量です。ほかの OS の場合については，各 OS のマニュアルを参照してください。

22）　pd_space_level = 0 | 1 | 3

空白変換機能を使用する場合の空白変換レベルを指定します。空白変換機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

0：空白変換レベル 0 を指定します。

1：空白変換レベル 1 を指定します。

3：空白変換レベル 3 を指定します。

空白変換レベルの処理方式を次に示します。

レベル	説明
レベル 0	空白変換をしません。
レベル 1	操作系 SQL での定数，埋込み変数，又は？パラメタのデータの空白を次のように変換します。 • 文字列定数を各国文字列定数とみなした場合，半角空白 2 文字を全角空白 1 文字に変換します。なお，半角空白が 1 文字単独で現れる場合は変換しません。

レベル	説明
	• 文字列定数を混在文字列定数とみなした場合，全角空白 1 文字を半角空白 2 文字に変換します。 • 各国文字列型の列へのデータの格納時，及び各国文字列型の値式との比較時は，埋込み変数，又は？パラメタの半角空白 2 文字を全角空白 1 文字に変換します。なお，半角空白が 1 文字単独で現れる場合は変換しません。 • 混在文字列型の列へのデータの格納時，及び混在文字列型の値式との比較時は，埋込み変数，又は？パラメタの全角空白 1 文字を半角空白 2 文字に変換します。
レベル 3	空白変換レベル 1 の処理に加えて次の処理が加わります。 • 各国文字列型の値式のデータを検索するときに，全角空白 1 文字を半角空白 2 文字に変換します。

《注意事項》

- 空白変換レベルを変更した場合，変更の前後で UAP の結果が異なる場合があります。UAP の結果を同じにしたい場合には，空白変換レベルを変更しないでください。
- 空白変換レベル 3 を指定してソートをした場合，HiRDB はソートの結果に対して空白変換をするため，期待した結果を得られないことがあります。
- クラスタキーの列へデータを格納する場合，空白変換によってユニークエラーとなることがあります。この場合，空白変換をしないでデータを格納するか，又はデータベース再編成ユティリティで既存のデータベースの空白を統一してください。
- 各国文字列の空白変換は，先頭から 2 文字単位で変換します。
- 空白変換レベルに 1 又は 3 を指定したときの注意事項を次に示します。

  ハッシュ分割した表に対して，表分割ハッシュ関数を使用した UAP で格納先 RD エリアを求める場合は，表分割ハッシュ関数の引数に空白変換レベルを指定してください。指定しないと表分割ハッシュ関数の結果が不正になることがあります。

  キーレンジ分割（分割キーが各国文字データ又は混在文字データの場合）した表に対して，UAP でキーレンジ分割処理をしている場合は，分割キー値を空白変換関数で変換してください。変換しないとキーレンジ分割の結果が不正になることがあります。

  表分割ハッシュ関数，及び空白変換関数については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSPACELVL オペランドを指定してください。PDSPACELVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

23)　pd_dec_sign_normalize = Y | N

UAP から渡された符号付きパック形式データの符号部を正規化するかどうかを指定します（**DECIMAL 型の符号正規化機能**を使用するかどうかを指定します）。符号部を正規化するとは，符号付きパック形式データの符号部 A～F を C 又は D に変換することを意味しています。

DECIMAL 型の符号正規化機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

Y：

DECIMAL 型の符号正規化機能を使用します。符号付きパック形式データの符号部を正規化します。A～F を DECIMAL データの符号部とみなします。

N：

DECIMAL 型の符号正規化機能を使用しません。符号付きパック形式データの符号部を正規化しません。C，D，F を DECIMAL データの符号部とみなします。

《注意事項》

Y を指定した場合（DECIMAL 型の符号正規化機能を使用する場合）の注意事項を次に示します。

- ハッシュ分割した表に対して，表分割ハッシュ関数を使用した UAP で格納先 RD エリアを求める場合は，表分割ハッシュ関数の引数に「DECIMAL 型の符号正規化機能あり」を指定してください。指定しないと表分割ハッシュ関数の結果が不正になることがあります。
- キーレンジ分割（分割キーが DECIMAL 型のデータの場合）した表に対して，UAP でキーレンジ分割処理をしている場合は，分割キー値を DECIMAL 型符号正規化関数で変換してください。変換しないとキーレンジ分割の結果が不正になることがあります。

表分割ハッシュ関数，及び DECIMAL 型符号正規化関数については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

24)　pd_sql_dec_op_maxprec = 演算結果が 29 けた以下の DECIMAL 型となる精度の最大値

～<符号なし整数>((29 | 38))

次に示す最大値を指定します。

- 演算結果のデータ型が DECIMAL 型となる精度の最大値
  - ・29 けた以下のデータだけに対する四則演算
  - ・29 けた以下のデータだけに対する集合関数 AVG/SUM
  - ・29 けた以下のデータだけに対するスカラ関数 DECIMAL/VALUE/GREATEST/LEAST
  - ・引数が FLOAT 型でかつ，精度を省略したスカラ関数 DECIMAL
  - ・対応する THEN 句及び ELSE 句の値が 29 けた以下のデータだけの CASE 式
  - ・抽象データ型のパラメタを含み，戻り値のデータ型が 29 けた以下であるユーザ定義関数だけが”候補となる関数”となる関数の呼び出し
- 集合演算の結果列のデータ型が DECIMAL 型となる場合，集合演算の導出表に対応する列が 29 けた以下のデータだけのときの精度の最大値

《指定値の目安》

- 08-04 以降のバージョンを新たに導入する場合
  38 を指定することをお勧めします。
- 上記以外の場合
  このオペランドを省略するか，又は 29 を指定することをお勧めします。

《注意事項》

- このオペランドの値を変更する場合，変更直後に次の処理を行ってください。
  < 08-04 以降のバージョンで定義したビュー表の場合>
  次に示す DECIMAL 型の演算を指定して定義したビュー表は，すべて再定義してください。
  - ・29 けた以下のデータだけに対する四則演算
  - ・29 けた以下のデータだけに対する集合関数 AVG/SUM
  - ・29 けた以下のデータだけに対するスカラ関数 MOD
  - ・引数が FLOAT 型でかつ精度を省略したスカラ関数 DECIMAL

  < 08-04 より前のバージョンで定義したビュー表の場合>
  定義されているビュー表をすべて再定義してください。

- このオペランドの指定値を変更する場合，次のことに注意してください。

  ・このオペランドの指定値を 29 から 38 に変更した場合

  メモリ所要量が増加したり，演算のオーバヘッドが増加したりするデメリットがあります。DECIMAL 型の演算を 29 けた以下の範囲で使用している場合は，このオペランドの値を 29 から 38 に変更しないことをお勧めします。

  ・このオペランドの指定値を 38 から 29 に変更した場合

  38 のときに動作していた 29 けた以下の DECIMAL 型の演算がオーバフローすることがあります。

25)　pd_server_entry_queue = spnfifo | fifo | loop

UAP の多重実行時に HiRDB サーバプロセスで処理が競合すると，処理要求を一時的にキューイングすることがあります。このときの HiRDB の処理方式を指定します。なお，ここでいう処理の競合とは，HiRDB サーバプロセス上でトランザクションが稼働するときに，複数のプロセスが表や RD エリアなどの内部資源に対して行う排他制御の競合を意味しています。内部資源に対して排他制御を行えるプロセスは常に一つだけです。また，以下の説明でいうスピンとは，排他制御の実行権を得るための処理を意味しています。ほかのプロセスが排他制御の実行権を解放した場合，スピン中のプロセスが排他制御の実行権を得る可能性が高くなります。

spnfifo：

先に発生した処理要求を優先的に処理します。ただし，キューに登録される前に一定回数のスピンをするため，完全に順序が守られるわけではありません。これは，バージョン 06-00 以前の処理方式です。

fifo：

先に発生した処理要求を spnfifo 指定時より優先的に処理します。キューに登録される前に一定回数のスピンをしないため，spnfifo 指定より処理順序は守られます。また，CPU の負荷を軽減する効果があります。

loop：

すべての処理要求を均等な優先度で処理します。キューに登録されている間は高速なスピンをします。loop を指定すると，UAP の多重実行時のレスポンスが向上することがあります。しかし，そのほかの指定より CPU に負荷が掛かります。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

UAP の多重実行時の処理性能が向上しない場合に指定値を変更してください。性能が向上することがあります。

26)　pd_thdlock_sleep_func = 0 | 1

共用メモリなどの共用資源を使用するときのロック獲得処理などで使用するプロセスの待機方法を指定します。このオペランドと pd_thdlock_retry_time オペランドの指定値の対応関係を次に示します。

pd_thdlock_sleep_func オペランドの値	pd_thdlock_retry_time オペランドの値	
	1～10000	10001～1000000
0	select()又は Sleep()を使用して指定したスレッド間ロックのスリープ時間だけ，各プロセスが待機します。	
1	sched_yield()又は SwitchToThread()を使用して，OS によってプロセスの割り当てを決定します	select()又は Sleep()を使用して指定したスレッド間ロックのスリープ時間だけ，各プロセスが待機します。

pd_thdlock_sleep_func オペランドの値	pd_thdlock_retry_time オペランドの値	
	1～10000	10001～1000000
	(pd_thdlock_retry_time オペランドの値は無効となります)。※	

注※　プロセスが待機しないため，CPU 使用率が高騰します。

《指定値の目安》

このオペランドを指定する必要はありません。このオペランドに 0 を指定するか，又はこのオペランドを省略してください。

《注意事項》

- このオペランドの指定値を 0 から 1 に変更すると，CPU 使用率が高騰します。
- スレッドロック確保処理は，一度のリトライで確保できるとは限りません。確保できるまでリトライを繰り返します。

  このオペランドに 1 を指定してリトライした場合，スリープ処理がないため，いったん CPU 使用率が 100%になると，多重に実行されるロック確保処理が大きなオーバヘッドとなります。そのため，CPU 使用率が 100%になる前と同等のトランザクション件数が投入されている場合でも，CPU 使用率が 100%の状態が長期間継続する場合があります。

《ほかのオペランドとの関連》

pd_thdlock_retry_time オペランドの指定値が 10000 より大きい場合，このオペランドは無効となります。

27)　pd_thdlock_wakeup_lock = Y | N

スレッド間ロックの解放通知方式を指定します。このオペランドに Y を指定することで，解放通知が確実に送信されることを保証します。

Y：

スレッド間ロックの解放通知をするときに，これとは別の一時的なロックを新たに取得します。

N：

スレッド間ロックの解放通知をするときに，これとは別の一時的なロックを新たに取得しません。

これは，バージョン 06-02 以前の HiRDB の処理方式です。

《指定値の目安》

このオペランドには Y を指定してください。

《注意事項》

N を指定した場合，ほかのトランザクションに比べて，実行時間が長いトランザクションが発生し，UAP の多重実行時のレスポンスが低下することがあります。

28)　pd_thdlock_pipe_retry_interval = スレッド間ロックの解放調査間隔～<符号なし整数>((0～2147483647))《1000000》(単位：マイクロ秒)

スレッド間ロックの解放調査を行う間隔をマイクロ秒単位で指定します。

《指定値の目安》

次の条件をすべて満たす場合にデフォルト値以上の値にすると，CPU 使用率が下がることがあります。

- pd_thdlock_wakeup_lock = Y を指定している
- UAP の多重実行時の処理性能が向上しない

- CPU 使用率が高い

ただし，ほかのトランザクションに比べて，実行時間が長いトランザクションが発生するようになります。

上記に該当しない場合は，このオペランドを指定しないでください。

《注意事項》

このオペランドにデフォルト値未満の値を指定すると，解放調査処理を短時間で繰り返すため，CPU 使用率が高騰します。

29)　pd_thdlock_retry_time = スレッド間ロックスリープ時間

～<符号なし整数>((1～1000000))《1000》（単位：マイクロ秒）

スレッド間ロックのスリープ時間をマイクロ秒単位で指定します。次に示す条件をすべて満たす場合にこのオペランドを指定すると，CPU 使用率が下がることがあります。

- CPU 使用率が非常に高い
- 性能が低下してでも CPU 使用率を下げたい
- pd_thdlock_sleep_func オペランドに 0 を指定している

上記に該当しない場合，このオペランドは指定しないでください。pd_thdlock_sleep_func と pd_thdlock_retry_time オペランドの指定値の組み合わせによる HiRDB の処理を次に示します。

pd_thdlock_sleep_func オペランドの値	pd_thdlock_retry_time オペランドの値	
	1～10000	10001～1000000
0	select()又は Sleep()を使用して指定したスレッド間ロックのスリープ時間だけ，各プロセスが待機します。	
1	sched_yield()又は SwitchToThread()を使用して，OS によってプロセスの割り当てを決定します（pd_thdlock_retry_time オペランドの値は無効となります）。※	select()又は Sleep()を使用して指定したスレッド間ロックのスリープ時間だけ，各プロセスが待機します。

注※　プロセスが待機しないため，CPU 使用率が高騰します。

《指定値の目安》

- このオペランドを指定する場合，最初は 10000 を指定してください。
- 10000 を指定して CPU 使用率が高くなり過ぎる場合は，指定値を大きくしてください。

《注意事項》

- 指定値を小さくしても，性能が変わらない場合があります。
- 1000 以上の値を指定すると，性能が低下することがあります。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，このオペランドの省略値は 10000 になります。

30)　pd_thdspnlk_spn_count = スレッド間スピンロックのスピン回数

～<符号なし整数>((0～2147483647))《512》

スレッド間スピンロックのスピン回数を指定します。次に示す条件をすべて満たす場合にこのオペランドを指定すると，性能が向上することがあります。それ以外の場合は指定する必要はありません。

- CPU 使用率に余裕がある

- CPU 使用率を上げてでも性能を良くしたい

《指定値の目安》

- このオペランドを指定する場合は省略値（512）より大きい値を指定してください。
- 指定値は OS の種類，プロセッサの種類，マシンの性能，ディスクの性能，UAP の内容，UAP の多重実行数などに依存するため，明確な目安はありません。指定値の変更と性能測定を行って適切な値を求めてください。

《注意事項》

- このオペランドの値が大き過ぎると，CPU 使用率が高くなって OS がスローダウンするなどの弊害が発生することがあります。その場合はこのオペランドの値を小さくしてください。
- このオペランドの値を大きくしても，性能が向上しないこともあります。

31）　pd_pageaccess_mode = SNAPSHOT ｜ NORMAL

データベース検索時のページアクセス方式を指定します。

SNAPSHOT：

ページアクセス方式を**スナップショット方式**にします。グローバルバッファへの初回アクセス時，探索条件に一致する行をプロセス固有メモリにコピーします。2 回目の検索要求時にはプロセス固有メモリを参照して検索結果を返します。スナップショット方式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

NORMAL：

ページアクセス方式を通常方式にします。検索要求ごとにグローバルバッファをアクセスします。

《指定値の目安》

グループ分け高速化機能など，性能向上を目的とした機能を使用できない場合に，スナップショット方式の適用を検討してください。通常の検索 SQL では，指定された探索条件に一致する行の数とほぼ同じ回数分だけグローバルバッファへのアクセスを行っています。このため，検索 SQL の実行が重なるとグローバルバッファへのアクセスが集中して，期待した性能が得られないことがあります。このような場合にスナップショット方式を適用すると，検索 SQL のグローバルバッファへのアクセス回数を削減できるため，検索性能が向上することがあります。ただし，スナップショット方式を適用した場合，HiRDB が使用するプロセス固有メモリが増加します。スナップショット方式を適用した場合のプロセス固有メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

32）　pd_cmdhold_precheck = Y｜N

RD エリアの閉塞状態のチェックを RD エリアの排他処理の前に行うかどうかを指定します。

Y：

RD エリアの閉塞状態のチェックを RD エリアの排他処理の前に行います。

N：

RD エリアの閉塞状態のチェックを RD エリアの排他処理の前に行いません。排他取得後に行います。

チェック対象になる閉塞の種類を次に示します。

- コマンド閉塞
- 参照可能閉塞
- 参照可能バックアップ閉塞

《指定値の目安》

通常はこのオペランドを省略するか，又は Y を指定してください。Y を指定したときと，N を指定したときの違いを次に示します。

項目	Y を指定した場合	N の場合
HiRDB の処理	UAP 又はコマンド※の実行時，RD エリアの排他を取得する前に，アクセスする可能性があるすべての RD エリアの閉塞状態をチェックします。例えば，RD エリア 1～4 に横分割されている表にアクセスする場合，RD エリア 1～4 の閉塞状態をチェックします。ただし，キーレンジ分割，又は FIX ハッシュ分割で条件を指定して，UAP 側でアクセスする可能性がある RD エリアを絞り込んでいる場合，アクセスする可能性がない RD エリアが閉塞していてもエラーにはなりません。	UAP 又はコマンド※の実行時，RD エリアの排他を取得した後に，アクセスする可能性があるすべての RD エリアの閉塞状態をチェックします。例えば，RD エリア 1～4 に横分割されている表にアクセスするとします。インデクスを使用してアクセスする RD エリアを絞り込んだ結果，アクセス対象 RD エリアが RD エリア 1 の場合，RD エリア 1 に対してだけ閉塞状態をチェックします。これは HiRDB Version 5.0 以前の処理方式です。
閉塞中の RD エリアに UAP がアクセスした場合	RD エリアの排他を取得する前に閉塞チェックをするため，RD エリアが閉塞していることを，N を指定したときよりも早く検知できます。	RD エリアの排他を取得した後に閉塞チェックをするため，閉塞中の RD エリアに UAP がアクセスした場合，RD エリアの排他によるタイムアウトエラー（KFPA11770-E）が発生することがあります。 また，アクセス対象 RD エリアがデータロード又は再編成で閉塞中の場合，データロード又は再編成処理の終了後に UAP が閉塞エラー（KFPA11920-E）になることがあります。
非横分割インデクスを使用してアクセスする RD エリアを絞り込む場合	表が横分割されていてインデクスが横分割されていない場合に注意が必要です。非横分割インデクスを使用してアクセス対象 RD エリアを絞り込む場合，アクセス対象外の RD エリアが閉塞していても，閉塞エラー（KFPA11920-E）になります。HiRDB の処理で説明した例の場合，RD エリア 1～4 のどれかが閉塞していると，UAP が閉塞エラー（KFPA11920-E）になります。	非横分割インデクスを使用してアクセス対象 RD エリアを絞り込む場合，アクセス対象外の RD エリアが閉塞していても，UAP 又はコマンドを実行できます。HiRDB の処理で説明した例の場合，RD エリア 2～4 が閉塞していても，UAP を実行できます。

注※　RD エリアが閉塞状態では実行できない UAP 及びコマンドが対象になります。

33)　pd_db_io_error_action = dbhold | unitdown

RD エリア（マスタディレクトリ用 RD エリアを除く）の入出力エラーが発生したときの HiRDB の処理を指定します。なお，マスタディレクトリ用 RD エリアに入出力エラーが発生した場合，このオペランドの指定に関係なく常に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。RD エリアの入出力エラーが発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

ここでいう入出力エラーとは，HiRDB がファイルを操作したときに HiRDB が判別できない理由でファイル操作に失敗したエラーを意味し，HiRDB ファイルシステムに対するアクセス要求から返されるエラーコードに-1544 が出力されるエラーのことです。

dbhold：

RD エリアの入出力エラーが発生した場合，その RD エリアを障害閉塞します。

unitdown：

RD エリアの入出力エラーが発生した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。ただし，異常終了後に再度入出力エラーが発生した場合，その RD エリアを障害閉塞します。再び，unitdown の指定を有効にするには，次に示すどちらかの方法を実行してください。

- HiRDB を正常開始する
- システム構成変更コマンド（pdchgconf コマンド）を実行する

《指定値の目安》

マニュアル「HiRDB Version 8 システム運用ガイド」の「RD エリアの入出力エラーが発生したときの対処方法」を参照して，このオペランドの指定値を決めてください。

《注意事項》

- unitdown を指定したときに入出力エラーが発生すると HiRDB が異常終了するため，次の場合に，処理対象の RD エリアが障害閉塞することがあります。

  ・更新前ログ取得モード又はログレスモードで UAP やユティリティを実行している場合

  ・CREATE TABLE の RECOVERY オペランドで NO を指定してログレスモードにしたユーザ LOB 用 RD エリアに対して，UAP 又はユティリティを実行している場合
- unitdown を指定する場合は，データベースの更新ログを取得する運用をしてください。データベースの更新ログを取得しない運用をする場合は，UAP 又はユティリティの実行前にバックアップを取得して，RD エリアの障害閉塞を回復できるようにしてください。バックアップの取得については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- 開始処理中又は終了処理中の入出力エラーについては，unitdown を指定しても HiRDB を異常終了しません。
- データベース回復ユティリティ（pdrstr）での回復処理中は，unitdown を指定していても HiRDB を異常終了しません。この場合は，データベース回復ユティリティ（pdrstr）を再度実行して回復してください。

34）　pd_connect_errmsg_hide = Y | N

CONNECT 失敗時に出力されるメッセージで，エラーの要因を隠すかどうかを指定します。

Y：CONNECT 失敗時にエラーの要因を隠します。

N：CONNECT 失敗時にエラーの要因を隠しません。

このオペランドの指定値によって CONNECT 失敗時に出力されるメッセージが変わることがあります。詳細を次に示します。

エラー要因	出力されるメッセージ	
	Y を指定した場合	N（省略値）を指定した場合
認可識別子の不正（指定されたユーザは存在しない）	KFPA19632-E	KFPA11561-E
パスワードの不正（指定されたパスワードが一致しない）	KFPA19632-E	KFPA11560-E

35）　pd_rpc_bind_loopback_address = Y | N

受信用ポートの生成時，ループバックアドレスで bind()するかどうかを指定します。

Y：ループバックアドレスで bind()します。

N：ループバックアドレスで bind()しません。

《前提条件》

このオペランドに Y を指定する場合は，次に示す条件をすべて満たしている必要があります。

- HiRDB/シングルサーバで HiRDB システムが構成されている
- 系切り替え機能を適用している場合，モニタモードで IP アドレスを引き継ぐ構成である
- HiRDB クライアントと HiRDB サーバが同一マシンにある（HiRDB クライアントが別マシンにない）
- pdunit オペランドの-x オプションとクライアント環境定義の PDHOST オペランドにループバックアドレスを指定している

《利点》

前記の前提条件を満たすシステム環境で，このオペランドに Y を指定すると，HiRDB サーバのプログラムを Windows ファイアウォールの例外リストに登録する必要がなくなります。

Windows ファイアウォールの例外リストの登録については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

36）　pd_cancel_down_msgchange = Y | N

サーバプロセスの強制終了が発生した場合に，出力されるエラーメッセージを変更するかどうかを指定します。

Y：

エラーメッセージを警告メッセージに変更します。エラーメッセージを変更する機能を**トランザクションキャンセル時のプロセスダウンメッセージ変更機能**といいます。

N：

エラーメッセージを変更しません。

このオペランドの指定値と出力されるエラーメッセージの関係を次に示します。

条件	出力されるメッセージ	
	Y（省略値）を指定した場合	N を指定した場合
次に示す原因によってサーバプロセスの強制終了が発生した場合※ • ユーザ操作による意図的な強制終了 • タイムアウトによる強制終了 • クライアント側の障害による強制終了	• KFPS01852-W • KFPO00115-W	• KFPS01820-E • KFPO00105-E
上記以外の原因によってサーバプロセスの強制終了が発生した場合	• KFPS01820-E • KFPO00105-E	
サーバプロセスの強制終了の原因を HiRDB が判断できない場合		

注※

メッセージの変更が発生する原因はほかにもあります。詳細については，マニュアル「HiRDB Version 8 システム運用ガイド」の「トランザクションキャンセル時のプロセスダウンメッセージ変更機能」を参照してください。

《利点》

このオペランドに Y を指定すると，サーバプロセスの強制終了が発生した原因で出力されるメッセージを切り分けられます。

《備考》

KFPS01820-E 及び KFPO00105-E メッセージが出力された場合，その原因が HiRDB の不正検知によるものか，又はトランザクションキャンセルなどのユーザ操作による意図的なものかをメッセージ ID から判断できません。判断するには，各メッセージに出力されるプロセス ID を比較する必要があります。

JP1 を使用してメッセージを監視している場合，複数のメッセージ情報を比較できないため，KFPS01820-E 及び KFPO00105-E メッセージが出力されたときに対処が難しくなることがあります。このオペランドに Y を指定すると，出力されるメッセージがエラー原因によって切り分けられるため，対処が特定しやすくなります。したがって，JP1 を使用してメッセージを監視している場合，このオペランドに Y を指定することをお勧めします。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，このオペランドの省略値は N になります。

2.2.6　全面回復処理に関するオペランド

37)　pd_max_recover_process = 全面回復処理の並列実行プロセス数

～<符号なし整数>((1～10))《3》

全面回復処理をするときの回復処理プロセス（REDO プロセス）数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（ディクショナリサーバ又はバックエンドサーバ）当たりの回復処理プロセス（REDO プロセス）数になります。

《前提条件》

rawI/O 機能を使用する必要があります。

《指定値の目安》

- RD エリアを定義した論理ボリュームの数が 3 以上のとき（1 サーバ当たり）：3
- RD エリアを定義した論理ボリュームの数が 3 未満のとき（1 サーバ当たり）：論理ボリューム数
- このオペランドの値を大きくすると，全面回復処理時の入出力並列度が増すため，回復処理時間を短縮できます。ただし，「このオペランドの値×サーバ数」分のプロセスが起動されるため，前記の指定値の目安と HiRDB のリソースを考慮して指定値を決めてください。

38)　pd_redo_allpage_put = Y | N

全面回復処理時に，シンクポイント以降に更新されたページをデータベースに書き出すかどうかを指定します。

Y：

全面回復処理時に，シンクポイント以降に更新されたすべてのページを書き出します。

N：

全面回復処理時に，障害発生時にデータベースに書き出していなかったページだけを書き出します。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。

LVM（論理ボリューム・マネジャ）のミラーリング機能の使用によってデータベースをミラー化する場合，OS 又はマシンの障害発生時，及びユニット障害による系切り替え発生時に，ミラー正副不一致状態になることがあります。その場合は，次の方法で正副ボリュームを一致させてください。

1. LVM の機能を使用して正副ボリュームを一致させてから HiRDB を再開始してください。

2.1 の方法で正副ボリュームを一致させる運用ができない場合，又は正副ボリュームを一致させる処理に時間が掛かり，系切り替え時間などのシステム要件を満たせない場合は，このオペランドに Y を指定してください。障害発生後の HiRDB 再開始での全面回復処理で，シンクポイント以降に更新されたすべてのページをデータベースに書き出します。これによって，HiRDB が正副ボリュームを一致させてミラー正副不一致状態を解消します。

ミラー正副不一致状態が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- このオペランドに Y を指定した場合, 全面回復処理時にすべてのページをデータベースに書き出すためのオーバヘッドが掛かります。
- このオペランドを指定した場合，全面回復処理終了時に KFPH24004-I メッセージを出力します。このオペランドを省略した場合は出力しません。

2.2.7　トランザクション決着処理に関するオペランド

39)　pd_trn_rerun_branch_auto_decide = Y | N

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

トランザクションのコミットメント制御で一相目のプリペア処理が完了する前にユニットが異常終了した場合，ユニットの再開始時にこのトランザクションからブランチした未決着状態のトランザクションを自動決着するかどうかを指定します。

Y：

未決着状態のトランザクションを自動決着します。

N：

未決着状態のトランザクションを自動決着しません。この場合，HiRDB 管理者が未決着状態のトランザクションを決着させる必要があります。未決着状態のトランザクションを決着させる方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

Y を指定した場合，又はこのオペランドを省略した場合の注意事項を次に示します。

- 縮退起動機能を使用している場合，縮退起動時に開始していなかったユニットのトランザクションは自動決着の対象とはなりません。したがって，開始していなかったユニットを障害復旧後などに再開始した場合は，未決着状態のトランザクションがないかどうかを確認してください。未決着状態のトランザクションがある場合は，未決着状態のトランザクションを決着させてください。未決着状態のトランザクションの確認方法，及び未決着状態のトランザクションを決着させる方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- ブランチ元のトランザクションがあるサーバのシステムログ量が増加します。
- 未決着状態のトランザクションの決着処理を実行している間，ブランチ元のトランザクションがあるサーバのシステムログが上書き禁止状態になります。

40)　pd_trn_send_decision_intval_sec = 秒単位でのトランザクション自動決着の送信リトライ時間間隔

～<符号なし整数>～((0～65535))《15》（単位：秒）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

ブランチしたトランザクションに自動決着指示を送信した場合，通信障害などによって送信が失敗したときに決着指示を再送するまでの時間を秒単位で指定します。

《前提条件》

pd_trn_rerun_branch_auto_decide オペランドに Y を指定，又は指定を省略している必要があります。

《注意事項》

0 を指定した場合は決着指示を連続して再送するため，通信負荷が高くなります。

《ほかのオペランドとの関連》

このオペランドは，pd_trn_send_decision_interval オペランドと次に示す関係があります。

- 再送するまでの時間を秒単位で指定する場合は pd_trn_send_decision_intval_sec オペランドを，分単位で指定する場合は pd_trn_send_decision_interval オペランドを指定します。
- pd_trn_send_decision_intval_sec オペランドと pd_trn_send_decision_interval オペランドを両方とも指定すると，pd_trn_send_decision_intval_sec オペランドの指定値が優先されます。
- pd_trn_send_decision_intval_sec オペランドと pd_trn_send_decision_interval オペランドを両方とも指定しないと，pd_sysdef_default_option オペランドの指定値によって次の値が仮定されます。

pd_sysdef_default_option オペランドの指定値	仮定値
recommendable	pd_trn_send_decision_intval_sec オペランドの省略値（15 秒）
v6compatible 又は v7compatible	pd_trn_send_decision_interval オペランドの省略値（5 分）

41）　pd_trn_send_decision_interval = 分単位でのトランザクション自動決着の送信リトライ時間間隔～<符号なし整数>((0～65535))（単位：分）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

ブランチしたトランザクションに自動決着指示を送信した場合，通信障害などによって送信が失敗したときに決着指示を再送するまでの時間を分単位で指定します。通常は pd_trn_send_decision_intval_sec オペランドを指定し，このオペランドは省略してください。

《前提条件》

pd_trn_rerun_branch_auto_decide オペランドに Y を指定，又は指定を省略している必要があります。

《注意事項》

0 を指定した場合は決着指示を連続して再送するため，通信負荷が高くなります。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は 5 になります。

また，このオペランドは pd_trn_send_decision_intval_sec オペランドと次に示す関係があります。

- 再送するまでの時間を秒単位で指定する場合は pd_trn_send_decision_intval_sec オペランドを，分単位で指定する場合は pd_trn_send_decision_interval オペランドを指定します。
- pd_trn_send_decision_intval_sec オペランドと pd_trn_send_decision_interval オペランドを両方とも指定すると，pd_trn_send_decision_intval_sec オペランドの指定値が優先されます。

- pd_trn_send_decision_intval_sec オペランドと pd_trn_send_decision_interval オペランドを両方とも指定しないと，pd_sysdef_default_option オペランドの指定値によって次の値が仮定されます。

pd_sysdef_default_option オペランドの指定値	仮定値
recommendable	pd_trn_send_decision_intval_sec オペランドの省略値（15 秒）
v6compatible 又は v7compatible	pd_trn_send_decision_interval オペランドの省略値（5 分）

42）　pd_trn_send_decision_retry_time = トランザクション自動決着の最大待ち時間～<符号なし整数>((0～65535))《360》（単位：分）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

ブランチしたトランザクションに自動決着指示を送信してから決着完了の応答があるまでの最大待ち時間を指定します。ここで指定した時間を過ぎても決着完了の応答がない場合，通信障害などの回復不能な障害が発生したとみなし，ブランチしたトランザクションへの決着指示を中止して，ブランチ元のトランザクションを決着させます。

なお，このオペランドに 0 を指定した場合は時間監視をしません。

《前提条件》

pd_trn_rerun_branch_auto_decide オペランドに Y を指定，又は指定を省略している必要があります。

43）　pd_trn_watch_time = トランザクション同期点処理時の最大通信待ち時間～<符号なし整数>((0，300～65535))《3600》（単位：秒）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB サーバプロセス上で実行されるトランザクションの同期点処理で，トランザクションブランチ間で行う通信（プリペア及びコミットの要求又は完了通信）の受信待ち時間の最大値を指定します。指定時間を過ぎても要求又は完了通信がない場合，該当するトランザクションブランチが 2 相コミットの 1 相目完了前であればロールバックします。1 相目完了後であれば，指示された処理を実行してトランザクションを決着します。トランザクション間で行う通信については，マニュアル「HiRDB Version 8 解説」の「コミットとロールバック」を参照してください。

《利点》

HiRDB クライアントがトランザクション決着指示を中断（クライアントプロセスの強制終了など）しても，HiRDB サーバがこのトランザクションの実行を中止しないと，トランザクションを実行し続けます。このため，データベースの排他資源などを長時間占有することがあります。このオペランドを指定すると，排他資源などの占有時間を短縮できます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。次に示す場合にこのオペランドを指定してください。

- コミット時に KFPA11989-E 又は KFPA11722-E メッセージが出力される場合
- 該当トランザクションでデータベースの更新件数が少ないのに COMMIT 文の処理時間が掛かる場合

《オペランドの規則》

- 0 を指定すると，プリペア及びコミットの要求又は完了を受信するまで要求又は完了通信を待ち続けます。

- 1～299 を指定した場合は 300 に切り上げられます。

《注意事項》

2 相コミットの 2 相目で行うコミットの要求又は完了通信に対しては，pd_dbsync_point=commit を指定したときだけ，このオペランドの指定値が有効になります。

44)　pd_trn_commit_optimize = ONEPHASE | NOUSE

このオペランドは，HiRDB/パラレルサーバ限定のオペランドです。

HiRDB/パラレルサーバのコミットメント制御で一相コミットを使用するかどうかを指定します。一相コミットについては，マニュアル「HiRDB Version 8 解説」を参照してください。

ONEPHASE：

トランザクション内の更新ブランチ数が一つの場合（一つのトランザクションが更新するサーバ数が一つの場合），コミットメント制御に**一相コミット**を使用します。なお，コミットメント制御で一相コミットを行うことを**一相最適化**といいます。

NOUSE：

コミットメント制御に**二相コミット**を使用します。一相コミットは使用しません。

《注意事項》

- OLTP システムから二相コミットの指示がある場合は，このオペランドの指定は無効になります。
- pd_sysdef_default_option オペランドに v6compatible を指定している場合，このオペランドの省略値は NOUSE になります。recommendable，又は v7compatible を指定している場合は，ONEPHASE が省略値になります。
- 回復不要 FES を使用している場合，このオペランドの指定よりも回復不要 FES の制限事項が優先されます。制限事項を次に示します。

  ・フロントエンドサーバ上に出力されるログを抑止します。

  ・回復不要 FES を適用したフロントエンドサーバでは，X/Open XA インタフェースを使用して接続する UAP を実行できません。

2.2.8　SQL の最適化に関するオペランド

45)　pd_optimize_level = SQL 最適化オプション〔，SQL 最適化オプション〕…

～＜識別子又は符号なし整数＞

SQL 最適化オプションを指定します。SQL 最適化オプションの機能を次に示します。SQL 最適化オプションの各機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

SQL 最適化オプションの機能	識別子	符号なし整数	S	P
ネストループジョイン強制	"FORCE_NEST_JOIN"	4	○	○
複数の SQL オブジェクト作成	"SELECT_APSL"	10	○	○
フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）	"FLTS_INC_DATA_BES"	16	×	○
ネストループジョイン優先	"PRIOR_NEST_JOIN"	32	○	○
フロータブルサーバ候補数の拡大	"FLTS_MAX_NUMBER"	64	×	○

SQL 最適化オプションの機能	識別子	符号なし整数	S	P
OR の複数インデクス利用の優先	"PRIOR_OR_INDEXES"	128	○	○
自バックエンドサーバでのグループ化，ORDER BY，DISTINCT 集合関数処理	"SORT_DATA_BES"	256	×	○
AND の複数インデクス利用の抑止	"DETER_AND_INDEXES"	512	○	○
グループ分け高速化処理	"RAPID_GROUPING"	1024	○	○
フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）	"FLTS_ONLY_DATA_BES"	2048	×	○
データ収集用サーバの分離機能	"FLTS_SEPARATE_COLLECT_SVR"	2064	×	○
インデクス利用の抑止（テーブルスキャン強制）	"FORCE_TABLE_SCAN"	4096	○	○
複数インデクス利用の強制	"FORCE_PLURAL_INDEXES"	32768	○	○
更新 SQL の作業表作成抑止	"DETER_WORK_TABLE_FOR_UPDATE"	131072	○	○
探索高速化条件の導出	"DERIVATIVE_COND"	262144	○	○
スカラ演算を含むキー条件の適用	"APPLY_ENHANCED_KEY_COND"	524288	○	○
プラグイン提供関数からの一括取得機能	"PICKUP_MULTIPLE_ROWS_PLUGIN"	1048576	○	○
導出表の条件繰り込み機能	"MOVE_UP_DERIVED_COND"	2097152	○	○

（凡例）

Ｓ：HiRDB/シングルサーバ

Ｐ：HiRDB/パラレルサーバ

○：指定が有効になります。

×：指定しても無効になります。

《オペランドの指定方法》

適用する SQL 最適化オプションを選択して，その識別子又は符号なし整数を指定します。識別子で指定する方法と，符号なし整数（計算値）で指定する方法の二つの方法がありますが，通常は識別子で指定してください。

- **識別子で指定する場合**

  「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用する場合は，次のように指定します。

  pd_optimize_level = "FORCE_NEST_JOIN","SELECT_APSL"

- **符号なし整数で指定する場合**

  「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用する場合は，次のように指定します。

  pd_optimize_level = 4,10

●HiRDB Version 5.0 以前からバージョンアップした場合

HiRDB Version 5.0 以前の合計値指定も有効です。最適化オプションを変更する必要がない場合は，HiRDB Version 6 以降にバージョンアップしたときにこのオペランドの指定値を変更する必要はありません。また，最適化オプションを追加する場合は次のように指定できます。

（例）

HiRDB Version 5.0 で「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用していて，今回「OR の複数インデクス利用の優先」を追加します。

pd_optimize_level = 14,128

ただし，この指定方法はどの機能を適用しているか分かりにくいため，識別子指定に変更することをお勧めします。

《オペランドの規則》

- 識別子と符号なし整数の混在指定はできません。

●識別子指定の場合

- SQL 最適化オプションを引用符（"）で囲んでください。
- ここで説明した SQL 最適化オプションを使用しない場合は"NONE"を指定してください。ただし，NONE と NONE 以外の識別子を指定した場合，NONE の指定は無効になります。
- 識別子は大文字でも小文字でも指定できます。
- 同じ識別子を 2 回以上指定しても，1 回指定した場合と同じになります。

●符号なし整数指定の場合

- ここで説明した SQL 最適化オプションを使用しない場合は 0 を指定してください。ただし，0 と 0 以外の符号なし整数を指定した場合，0 の指定は無効になります。
- 同じ符号なし整数を 2 回以上指定しても，1 回指定した場合と同じになります。

《指定値の目安》

指定値の目安については，マニュアル「HiRDB Version 8 UAP 開発ガイド」の「SQL 最適化オプション」を参照してください。

《オペランドの省略値》

このオペランドを省略すると次に示す値が仮定されます。

- **HiRDB/シングルサーバの場合**

  "PRIOR_NEST_JOIN"，"PRIOR_OR_INDEXES"，"DETER_AND_INDEXES"，"RAPID_GROUPING"，"DETER_WORK_TABLE_FOR_UPDATE"，"APPLY_ENHANCED_KEY_COND"

- **HiRDB/パラレルサーバの場合**

  "PRIOR_NEST_JOIN"，"PRIOR_OR_INDEXES"，"SORT_DATA_BES"，"DETER_AND_INDEXES"，"RAPID_GROUPING"，"DETER_WORK_TABLE_FOR_UPDATE"，"APPLY_ENHANCED_KEY_COND"

ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合は"SELECT_APSL"が仮定されます。

《注意事項》

- SQL 文中に SQL 最適化指定を指定している場合は，このオペランドの指定よりも SQL 最適化指定が優先されます。SQL 最適化指定については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。
- ストアドルーチン及びトリガ中の SQL 文（CREATE PROCEDURE, CREATE TYPE, ALTER PROCEDURE，CREATE TRIGGER，ALTER ROUTINE，又は ALTER TRIGGER）に SQL 最適化オプションを指定している場合は，このオペランドの指定よりも SQL 文中の SQL 最適化オプションが優先されます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSQLOPTLVL オペランドを指定してください。PDSQLOPTLVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

- フロントエンドサーバ定義に pd_floatable_bes，又は pd_non_floatable_bes オペランドを指定している場合には，「フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）」及び「フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）」の指定は無効になります。
- pd_indexlock_mode オペランドに KEY を指定している場合，「更新 SQL の作業表作成抑止」の指定は無効になります。

46）　pd_additional_optimize_level = SQL 拡張最適化オプション〔，SQL 拡張最適化オプション〕…～<識別子又は符号なし整数>《"COST_BASE_2"》

SQL 拡張最適化オプションを指定します。SQL 拡張最適化オプションの機能を次に示します。SQL 拡張最適化オプションの各機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

SQL 拡張最適化オプションの機能	識別子	符号なし整数
コストベース最適化モード 2 の適用	"COST_BASE_2"	1
ハッシュジョイン，副問合せのハッシュ実行※1	"APPLY_HASH_JOIN"	2
値式に対する結合条件適用機能	"APPLY_JOIN_COND_FOR_VALUE_EXP"	32
ジョインを含む SQL 文の外部サーバ実行の抑止※1※2	"DETER_JOIN_SQL"	67108864
直積を含む SQL 文の外部サーバ実行の強制※1※2	"FORCE_CROSS_JOIN_SQL"	134217728
無条件に生成する，外部サーバで実行できる探索高速化条件の導出の抑止※1※2	"DETER_FSVR_DERIVATIVE_COND"	1073741824

注※1　これらの項目は「コストベース最適化モード 2 の適用」を指定した場合に有効になります。

注※2　これらの項目は外部表を検索する場合に有効になります。それ以外の場合は無効になります。

《オペランドの指定方法》

適用する SQL 拡張最適化オプションを選択して，その識別子又は符号なし整数を指定します。通常は識別子で指定してください。

- **識別子で指定する場合**

  「コストベース最適化モード 2 の適用」及び「ハッシュジョイン，副問合せのハッシュ実行」を適用する場合は，次のように指定します。

  pd_additional_optimize_level = "COST_BASE_2","APPLY_HASH_JOIN"

- **符号なし整数で指定する場合**

  「コストベース最適化モード 2 の適用」及び「ハッシュジョイン，副問合せのハッシュ実行」を適用する場合は，次のように指定します。

  pd_additional_optimize_level = 1,2

《オペランドの規則》

- 識別子と符号なし整数の混在指定はできません。

●識別子指定の場合

- SQL 拡張最適化オプションを引用符（"）で囲んでください。
- ここで説明した SQL 拡張最適化オプションを使用しない場合は"NONE"を指定してください。ただし，NONE と NONE 以外の識別子を指定した場合，NONE の指定は無効になります。
- 識別子は大文字でも小文字でも指定できます。
- 同じ識別子を 2 回以上指定しても，1 回指定した場合と同じになります。

●符号なし整数指定の場合

- ここで説明した SQL 拡張最適化オプションを使用しない場合は 0 を指定してください。ただし，0 と 0 以外の符号なし整数を指定した場合，0 の指定は無効になります。
- 同じ符号なし整数を 2 回以上指定しても，1 回指定した場合と同じになります。

《指定値の目安》

指定値の目安については，マニュアル「HiRDB Version 8 UAP 開発ガイド」の「SQL 拡張最適化オプション」を参照してください。

《注意事項》

- SQL 文中に SQL 最適化指定を指定している場合は，このオペランドの指定よりも SQL 最適化指定が優先されます。SQL 最適化指定については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。
- ストアドルーチン及びトリガ中の SQL 文（CREATE PROCEDURE，CREATE TYPE，CREATE TRIGGER，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER）に SQL 拡張最適化オプションを指定している場合は，このオペランドの指定よりも SQL 文中の SQL 拡張最適化オプションが優先されます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDADDITIONALOPTLVL オペランドを指定してください。
PDADDITIONALOPTLVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

47）　pd_hashjoin_hashing_mode = TYPE1 | TYPE2

SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を指定した場合のハッシング方式を指定します。

TYPE1：バージョン 07-02 より前のハッシング方式です。

TYPE2：TYPE1 に比べて，均等にハッシングされます。

《指定値の目安》

- 通常は TYPE2 を指定してください。ただし，結合条件に指定した列のデータによっては均等にならないこともあるので，その場合は TYPE1 を指定してください。
- 07-02 以降にバージョンアップしたユーザで，TYPE1 を指定して期待した性能が得られなかった場合は，TYPE2 を指定してください。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDHJHASHINGMODE オペランドを指定してください。
PDHJHASHINGMODE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

48)　pd_hash_table_size = ハッシュ表サイズ

～<符号なし整数>《256》(単位：キロバイト)

- 32 ビットモードの場合：((128～524288))
- 64 ビットモードの場合：((128～2097152))

SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を指定した場合のハッシュ表サイズを指定します。

《指定値の目安》

このオペランドに指定するハッシュ表サイズについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《オペランドの規則》

このオペランドには 128 の倍数を指定してください。128 の倍数を指定しなかった場合は自動的に 128 の倍数に切り上げられます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDHASHTBLSIZE オペランドを指定してください。PDHASHTBLSIZE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

49)　pd_work_table_option = 作業表処理オプション

～<符号なし整数>《1》

作業表を使用する SQL を実行するときの HiRDB の処理方式を指定します。このオペランドの指定値は次に示す計算式で求めてください。

作業表処理オプション＝ａ＋ｂ

作業表の処理区分	変数名	値	HiRDB の処理方式
AND の複数インデクス利用時の排他取得方式※1	a	0	アクセスパスとして AND の複数インデクス利用が選択されると，探索条件中で探索に使用するインデクスが定義された列を含む述語のうち一つでも真となるデータがある場合，そのデータに対して共用モード（PR）で排他が掛かります。 このため，排他オプション※2 に WITH EXCLUSIVE LOCK を指定しても，排他モード（EX）ではなく，共用モード（PR）で排他が掛かります。
		1	アクセスパスとして AND の複数インデクス利用が選択されると，探索処理中には，探索条件の中で探索に使用するインデクスが定義された列を含む述語のうち一つでも真となるデータがある場合，そのデータに対して指定したモードで排他が掛かります。ただし，1 件目のデータの探索完了までは，探索に使用したインデクスが定義された列に対する述語を論理演算し，偽となったデータについては排他が解除されます。
作業表用バッファの自動増分時のメッセージ出力抑止※3	b	0	作業表用バッファの自動増分時，KFPH29008-I メッセージを出力します。このメッセージは各サーバプロセスの初回の作業表用バッファの自動増分時に出力されます。
		8	作業表用バッファの自動増分時，KFPH29008-I メッセージを出力しません。

注※1

AND の複数インデクス利用をするときの HiRDB の処理方式を指定します。

AND の複数インデクス利用とは，探索条件に AND で結ばれた条件が複数あり，それぞれの列に異なるインデクスが定義してある場合（例えば，SELECT ROW FROM T1 WHERE C1 = 100

AND C2 = 200)，それぞれのインデクスを使って条件を満たす行の作業表を作成し，これらの積集合を求める方式です。

注※2

SQL に排他オプションを指定していない場合は，通常 WITH SHARE LOCK が仮定されます。ただし，次に示す指定によっては仮定される排他オプションが変わるため，注意してください。

- カーソルへの FOR UPDATE 句の指定
- クライアント環境定義の PDISLLVL オペランドの指定
- 手続き，関数，又はトリガの CREATE PROCEDURE，CREATE TRIGGER，CREATE TYPE，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER のデータ保証レベルの指定

注※3

pd_work_buff_expand_limit オペランドを指定していることが前提条件となります。作業表用バッファの自動増分については，pd_work_buff_expand_limit オペランドを参照してください。

《指定値の目安》

作業表の処理区分	値	指定値の目安
AND の複数インデクス利用時の排他取得方式	0	0 を指定すると排他を解除する処理が実行されない分，SQL の処理時間が短縮されます。ただし，複数のユーザが同時に同一表を，AND の複数インデクス利用で更新をするとデッドロックになることがあります。デッドロックを避けるには 1 を指定してください。
	1	1 を指定すると排他を解除する処理が実行される分，SQL の処理時間が増加します。しかし，最終的な排他範囲は狭くなるので，更新 UAP の同時実行性が向上します。
作業表用バッファの増分確保メッセージの出力抑止	0	0 を指定すると，作業表用バッファの増分確保がされたかどうかを監視できます。
	8	8 を指定すると，メッセージの出力量を削減できます。

《注意事項》

作業表の処理区分に AND の複数インデクス利用時の排他取得方式を選択したときの注意事項を次に示します。

- 手続き，関数，及びトリガに対しては，ストアドルーチン及びトリガ生成時（CREATE PROCEDURE，CREATE TRIGGER，CREATE TYPE，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER の実行時）の AND の複数インデクス利用時の排他取得方式が適用されます。ストアドルーチン実行時（CALL 文の実行時）及びトリガ実行時の AND の複数インデクス利用時の排他取得方式は適用されません。
- 次に示すオペランド又はオプションで AND の複数インデクス利用が抑止される場合，このオペランドの指定は無効になります。

  ・pd_optimize_level オペランド

  ・クライアント環境定義の PDSQLOPTLVL オペランド

  ・CREATE PROCEDURE，CREATE TRIGGER，CREATE TYPE，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER の SQL 最適化オプション

2.2.9　絞込み検索に関するオペランド

50)　pd_max_list_users = リスト操作可能ユーザ数

～<符号なし整数>((0～32767))《0》

リストを操作するすべてのユーザ数を指定します。

《ほかのオペランドとの関連》

このオペランドを指定する場合，同時に pd_max_list_count オペランドも指定してください。

《注意事項》

- このオペランドに必要以上に大きい値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。サーバが使用する共用メモリについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- このオペランド及び pd_max_list_count オペランドの両方に最大値（32767）を指定すると，OS が確保できる共用メモリの上限値を超えるおそれがあります。

51）　pd_max_list_count = 1 ユーザ当たりのリスト作成数

～<符号なし整数>((0～32767))《0》

1 ユーザが作成できるリストの最大数を指定します。

《ほかのオペランドとの関連》

このオペランドを指定する場合，同時に pd_max_list_users オペランドも指定してください。

《注意事項》

- このオペランドに必要以上に大きい値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。サーバが使用する共用メモリについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- このオペランド及び pd_max_list_users オペランドの両方に最大値（32767）を指定すると，OS が確保できる共用メモリの上限値を超えるおそれがあります。

52）　pd_list_initialize_timing = INITIAL | DEFER | STANDBY

リストの初期化（削除）時期を指定します。通常，HiRDB を開始（再開始を含む）するときにリストが初期化されます。このオペランドでリストの初期化時期を変更できます。リストの作成数が多い場合に，このオペランドの指定値の変更を検討してください。

リストの初期化時期の変更については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

INITIAL：

HiRDB を開始するときにリストを初期化します。全リストの初期化が終了しないと HiRDB を開始しません。

DEFER：

HiRDB を開始するときにリストを初期化しません。ASSIGN LIST 文を実行するときにリストを初期化します。そのため，ASSIGN LIST 文を実行するときに初期化する分のオーバヘッドが掛かります。次に示す場合に DEFER 指定を検討してください。

- HiRDB の開始時間を短縮したい
- 系の切り替え時間を短縮したい（ユーザサーバホットスタンバイ又は高速系切り替え機能を使用している）

STANDBY：

高速系切り替え機能を使用するときに有効になる指定です。高速系切り替え機能を使用しないときに STANDBY を指定しても INITIAL が仮定されます。

待機系 HiRDB を開始するときにリストを初期化します。系の切り替え時にはリストを初期化しません。また，系切り替え後の ASSIGN LIST 文の実行時にもリストを初期化しません。ただし，実行系と待機系の両方に同じリストを作成しておく必要があります。

53)　pd_apply_search_ats_num = サーチ条件 ATS を適用する絞り込み値の組み合わせ個数の上限

～<符号なし整数>((255～30000))《255》

インデクスを用いた検索で，IN 述語及び限定述語（= ANY（表副問合せ），= SOME（表副問合せ））を指定する場合，サーチ条件 ATS 又は RANGES を適用する絞り込み値の組み合わせ個数の上限を指定します。

《指定値の目安》

- バージョン 08-04 以降に HiRDB を初めて導入する場合

  指定値を，30000 にすることをお勧めします。

- バージョン 08-03 以前の HiRDB をバージョンアップする場合

  バージョンアップをする前と同じ状態でお使いいただくために，指定値を 255 にすることをお勧めします。

  ただし，IN 述語及び限定述語（= ANY（表副問合せ），= SOME（表副問合せ））を指定し，サーチ条件の絞り込み値の組み合わせ個数が 255 を超えて RANGE，又は RANGES になっている SQL 文では，このオプションの指定値を変更すれば性能が向上することがあります。サーチ条件種別については，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。

2.2.10　システム監視に関するオペランド

54)　pd_cmd_process_conwaittime = 運用コマンド・ユティリティ実行におけるプロセス割り当て監視時間

～<符号なし整数>

・HiRDB/シングルサーバの場合：((0,16～4000))<<60>>(単位:秒)

・HiRDB/パラレルサーバの場合：((0,16～4000))<<0>>(単位:秒)

運用コマンド・ユティリティ実行時に，プロセスを割り当てるまでの監視時間を指定します。

本オペランドに 0 を指定した場合，プロセスを割り当てるまでの時間を監視しません。

1～15 を指定した場合，プロセスを割り当てるまでの監視時間を 16 秒とします。

《指定値の目安》

- HiRDB/シングルサーバの場合

  通常，このオペランドは指定する必要はありません。

- HiRDB/パラレルサーバの場合

  通常，このオペランドに 60 を指定してください。

《注意事項》

- HiRDB/パラレルサーバの場合

  プロセスを割り当てるまでの時間監視を有効にする場合，HiRDB システムを停止してから本オペランドの値を指定してください。

《本オペランドの適用対象となる運用コマンド・ユティリティ》

- pdclose（RD エリアのクローズ）
- pddbls（RD エリアの状態表示）
- pdhold（RD エリアの閉塞）
- pdopen（RD エリアのオープン）
- pdrels（RD エリアの閉塞解除）

- pdcopy（データベース複写ユティリティ）
- pdload（データベース作成ユティリティ）
- pdrorg（データベース再編成ユティリティ）
- pdreclaim（空きページ解放ユティリティ）
- pdpgbfon（グローバルバッファ常駐化ユティリティ）
- pdrstr（データベース回復ユティリティ）

55）　pd_utl_exec_time = ユティリティの実行監視時間
～<符号なし整数>((0～35791394))《0》（単位：分）

次に示すユティリティの実行時間を監視する場合，その監視時間を分単位で指定します。

- データベース作成ユティリティ（pdload コマンド）
- データベース再編成ユティリティ（pdrorg コマンド）
- 空きページ解放ユティリティ（pdreclaim コマンド）
- グローバルバッファ常駐化ユティリティ（pdpgbfon コマンド）

このオペランドで指定した監視時間を超えてもユティリティが終了しない場合，実行中のユティリティを強制終了して，無応答原因の調査用に次に示す障害情報を出力します。

取得する障害情報	出力先
• コアファイル • .deb ファイル	ユティリティが動作したサーバマシンの%PDDIR%¥spool¥save 下に出力されます。
• pdls -d rpc -a コマンドの実行結果 • pdls -d lck コマンドの実行結果	ユティリティのコマンドを入力したサーバマシンの，pd_tmp_directory オペランドで指定したディレクトリ下に出力されます。pd_tmp_directory オペランドの指定を省略した場合は，システム環境変数の TMP に指定したディレクトリ下に出力されます。システム環境変数の TMP の指定も省略した場合は，%PDDIR%¥tmp 下に出力されます。 ただし，HiRDB 管理者がコマンドを実行した場合は，%PDDIR%¥spool¥save 下にも出力されます。ファイル名称は次のとおりです。 コマンド名 YYYYMMDDHHMMSSpid.txt • pid はプロセス ID です。 • pdreclaim コマンド又は pdpgbfon コマンドを実行した場合，内部的に HiRDB が pdrorg コマンドを実行するため，ファイル名に付けられるコマンド名が pdrorg になります。

《利点》

夜間バッチジョブなどでユティリティ実行中に何らかの障害（通信障害又はディスク障害など）が発生し，ユティリティが無応答状態になっても，後続のジョブの停滞を防止できます。

《指定値の目安》

- このオペランドは無応答障害の対処を目的とし，長大トランザクションの監視を目的としていません。したがって，このオペランドの指定値にはユティリティ実行時間の実績値の最大値に余裕を持たせた値を指定してください。例えば，データベース作成ユティリティの最大実行時間が 60 分程度で，データベース再編成ユティリティの最大実行時間が 90 分程度の場合，pd_utl_exec_time=120 と余裕を持った値を指定してください。これは，通常 90 分で終了する処理が 30 分過ぎても応答が返ってこないのは無応答障害のおそれがあると判断しています。

- このオペランドを省略するか，又は 0 を指定した場合，次に示すような障害が発生するとユティリティが無応答状態になることがあります。したがって，0 以外の値を指定してください。
  ・サーバ間の通信障害（一時的な障害を含む）
  ・ディスク障害などによるプロセスの沈み込み

《オペランドの規則》

- このオペランドを省略，又は 0 を指定した場合は，ユティリティの実行時間を監視しません。
- 各ユティリティの option 制御文の exectime オペランドに監視時間を指定した場合は，exectime オペランドの指定が優先されます。

56）pd_watch_time = 最大応答待ち時間

～<符号なし整数>((0～65535))《0》（単位：秒）

このオペランドは旧バージョンとの互換性保証のための指定です。したがって指定する必要はありません。

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB サーバプロセスが他の HiRDB サーバプロセス（ディクショナリサーバ/バックエンドサーバ）から応答が返ってくるまでの最大待ち時間を指定します。

HiRDB サーバプロセスの中断を許したくない時間を指定しますが，指定した時間を過ぎても該当 HiRDB サーバプロセスの処理を確実に中断するわけではありません。

《利点》

HiRDB クライアントが SQL 実行を中断（クライアントプロセスの強制終了など），又はコマンド・ユティリティの実行を中断（強制停止等）しても，HiRDB サーバが当該 SQL やコマンド・ユティリティの実行を中止しない場合，HiRDB サーバはこの SQL やコマンド・ユティリティの実行を続けて長時間にわたって排他資源などを占有することがあります。このオペランドを指定すると，この占有時間を短くできます。

《指定値の目安》

次のうちの最も長い時間を指定してください。

- クライアント環境定義の PDCWAITTIME オペランドで指定する時間
- pd_lck_wait_timeout オペランドで指定する時間
- 実行時間が最も掛かる SQL の処理時間
- 実行時間が最もかかるコマンド・ユーティリティの時間値

《注意事項》

- このオペランドに 0 を指定した場合は，SQL 実行時間の監視をしません。
- SQL の最大実行時間を設定する場合は，このオペランドに 0 を設定し，クライアント環境変数 PDCWAITTIME で SQL 最大実行時間を設定することをお勧めします。クライアント環境定義 PDCWAITTIME については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
- このオペランドに SQL 実行時間より短い値を指定した場合，SQL 実行中に処理が中断され，HiRDB クライアントに SQL エラーか，又は HiRDB サーバの異常終了が報告されることがあります。
- HiRDB/シングルサーバの場合にこのオペランドを指定しても，SQL 最大実行時間を監視しません。ただし，このオペランドを指定した場合，その指定値が pd_lck_wait_timeout オペランドの省略値として使用されます。このため，HiRDB/シングルサーバの場合はこのオペランドを省略することをお勧めします。

57)　pd_queue_watch_time = メッセージキュー監視時間

～<符号なし整数>((0～3600))《600》(単位：秒)

このオペランドは，HiRDB のサーバプロセスの沈み込みを防止することを目的としています。サーバプロセスの沈み込みについては，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

参考

HiRDB サーバプロセス数は次に示すオペランドで制限されています。

- pd_max_server_process
  ユニット内で稼働するサーバ数が多い場合は，このオペランドの指定値を詳細に見積もってください。また，スタンバイレス型系切り替え機能を使用する場合は，系切り替えが発生したときを想定した見積もりが必要になります。
- pd_max_bes_process
  マルチフロントエンドサーバ，又は 1：1 スタンバイレス型系切り替え機能を使用する場合は，このオペランドの指定値を詳細に見積もってください。
- pd_max_dic_process
  マルチフロントエンドサーバを使用する場合は，このオペランドの指定値を詳細に見積もってください。
- pd_ha_max_server_process
  影響分散スタンバイレス型系切り替え機能を使用する場合は，このオペランドの指定値を詳細に見積もってください。
- pd_max_users
  同時接続数が多い場合は，適切な値を指定する必要があります。

HiRDB では，サーバプロセスの割り当て処理でメッセージキューを使用しています。サーバプロセスの沈み込みが発生すると，メッセージキューからメッセージを取り出せなくなります。メッセージキュー中のメッセージが，このオペランドに指定した時間（**メッセージキュー監視時間**）を過ぎても取り出されない場合，警告メッセージ又はエラーメッセージ（KFPS00888-W 又は KFPS00889-E）を出力します。この機能を**メッセージキュー監視機能**といいます。このメッセージが出力されると，サーバプロセスが沈み込んでいるおそれがあります。

なお，このオペランドに 0 を指定した場合はメッセージキューを監視しません。

メッセージキュー監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの規則》

メッセージキューの監視は 10 秒単位で行われるため，このオペランドには 10 秒単位の時間（100 秒，110 秒など）を指定してください。10 秒単位で指定しない場合，下一けたを切り上げます。例えば，105 秒と指定した場合，110 秒が仮定されます。

《ほかのオペランドとの関連》

このオペランドは pd_queue_watch_timeover_action オペランドと関連があります。

58)　pd_queue_watch_timeover_action = continue | stop

メッセージキュー監視時間を超えても，メッセージキュー中のメッセージが取り出されない場合の HiRDB の処理を指定します。

continue：

警告メッセージ（KFPS00888-W）を出力します。

stop：

警告メッセージ及びエラーメッセージ（KFPS00888-W 及び KFPS00889-E）を出力して，該当するメッセージキューがあるユニットは異常終了します。

《指定値の目安》

- 通常は stop を指定します（又はこのオペランドを指定しません）。メッセージキュー監視時間を超えても，メッセージキュー中のメッセージが取り出されない場合，HiRDB のサーバプロセスが沈み込んでいるおそれがあります。サーバプロセスの沈み込みが発生したユニットを再開始すると，サーバプロセスの沈み込みを解決できることがあります。
- HiRDB を停止したくない場合は continue を指定します。この場合，サーバプロセスの沈み込みが発生したサーバではトランザクションを実行できません。それ以外のサーバではトランザクションを実行できます。サーバプロセスの沈み込みを解決するには，サーバプロセスの沈み込みが発生したサーバで実行中のトランザクションを pdcancel コマンドなどで終了させてください。その後，サーバプロセスの沈み込み要因を特定して対策してください。また，トランザクションが発生していない場合には，サーバプロセスの沈み込みが発生したサーバを pdkill コマンドなどで終了させた後，沈み込み要因を特定して対策してください。サーバプロセスの沈み込み要因及び対策については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

59）　pd_down_watch_proc = サーバプロセスの異常終了回数の上限値〔，監視間隔〕

このオペランドは，HiRDB のサーバプロセスの異常終了回数を監視するオペランドです。PDCWAITTIME オーバ，又はアボートによって異常終了するプロセスを監視対象とします。

サーバプロセスの異常終了が多発すると，新たなサービスを受け付けられないことがあります。しかし，サーバプロセスの異常終了では HiRDB は異常終了しないため，実質オンライン停止状態になります。このオペランドを指定すると，HiRDB の再開始によってこの状態を抜け出せます。

サーバプロセスの異常終了回数の上限値：～<符号なし整数>((0～65535))《0》

サーバプロセスの異常終了回数がこのオペランドに指定した値を超えた場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を異常終了させます。これを**プロセスの異常終了回数監視機能**といいます。プロセスの異常終了回数監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

HiRDB/シングルサーバの場合は，シングルサーバプロセスの異常終了回数がカウントされます。HiRDB/パラレルサーバの場合は，ユニット内のフロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバプロセスの異常終了回数の合計値がカウントされます。

なお，0 を指定すると，サーバプロセスの異常終了回数を監視しません。

監視間隔：～<符号なし整数>((10～3600))《600》（単位：秒）

サーバプロセスの異常終了回数を監視する時間を秒単位で指定します。

例えば，100 を指定すると，100 秒間隔でサーバプロセスの異常終了回数を監視します。

《利点》

- HiRDB の再開始によってメモリ及びリソース状態がリフレッシュされて処理効率が向上します。
- サーバプロセスの異常終了が多発した場合，HiRDB が異常終了するため，すぐに系を切り替えられます。

《注意事項》

- サーバプロセスが異常終了すると，KFPS01820-E メッセージが出力されます。pdcancel コマンドでサーバプロセスが異常終了したときにも KFPS01820-E メッセージが出力されますが，これは異常終了回数のカウントに含まれません。
- 相互系切り替え構成の場合，系が切り替わると同一サーバマシンで複数の HiRDB を稼働するため，逆にトラフィックが上昇して効果が得られないことがあります。このオペランドを指定する場合は，異常終了した系で HiRDB を再開始する運用をお勧めします。

《オペランドの規則》

サーバプロセスの異常終了回数の上限値を指定しないで監視間隔だけの指定はできません。

《備考》

- プロセスの異常終了回数監視機能によって HiRDB が異常終了する場合，KFPS01821-E 及び KFPS00729-E メッセージを出力します。
- サーバプロセスが異常終了する要因と異常終了回数のカウント対象を次に示します。

サーバプロセスが異常終了する要因	異常終了回数のカウント対象			
	シングルサーバプロセス	フロントエンドサーバプロセス	ディクショナリサーバプロセス	バックエンドサーバプロセス
クライアント環境定義の PDCWAITTIME オペランドの値を超えた	○	○	×※1	×※1
pdcancel コマンド	×	×※2	×	×
内部強制終了（HiRDB が内部的に SIGKILL を発行してプロセスを終了させる場合）	○※3	○※3	×※1	×※1
アボート	○	○	○	○
XA 接続の UAP でロールバックが発生した	○	○	×	×
上記以外のプロセス異常終了	○	○	○	○

（凡例）

○：プロセスの異常終了回数にカウントされます。

×：プロセスの異常終了回数にカウントされません。

注※1

トランザクションブランチが発生した状態で障害を検知した場合，同一トランザクションブランチから発生したフロントエンドサーバプロセスの異常終了がカウントの対象になります。

注※2

pdcancel コマンドでバックエンドサーバプロセス又はディクショナリサーバプロセスを強制終了すると，フロントエンドサーバプロセスを内部的に強制終了します。この場合，フロントエンドサーバプロセスの異常終了がカウントされることがあります。

注※3

OLTP システムによるグローバルトランザクションが発生した状態で障害を検知した場合，同一グローバルトランザクションから発生したシングルサーバプロセス又はフロントエンドサーバプロセスの異常終了がカウントの対象になります。

60）　pd_host_watch_interval = ホスト間監視時間間隔

～<符号なし整数>((1～180))《10》（単位：秒）

ほかのホスト（サーバマシン）の稼働状態を監視するために，問い合わせをする時間間隔を秒単位で指定します。

《注意事項》

- このオペランドは，HiRDB/シングルサーバの場合，及び HiRDB/パラレルサーバでも 1 ユニットの場合は，無効になります。
- この値が大き過ぎると，ほかのホストの異常を検出しにくくなります。

《ほかのオペランドとの関連》

このオペランドは pd_ipc_conn_nblock_time オペランドと関連があります。

61）　pd_watch_resource = MANUAL | AUTO

リソースの使用率が 80％以上になった場合に，警告メッセージを出力するかどうかを指定します。

MANUAL：警告メッセージを出力しません。

AUTO：警告メッセージを出力します。

監視対象のリソースを次に示します。

監視対象のリソース	出力されるメッセージ
pd_max_users オペランドで指定した最大同時接続数	KFPS05123-W
pd_max_access_tables オペランドで指定した同時アクセス可能実表数	
pd_max_rdarea_no オペランドで指定した RD エリアの最大数	
pd_max_file_no オペランドで指定した RD エリアを構成する HiRDB ファイルの最大数	
pdwork オペランドで指定した作業表用ファイル用の HiRDB ファイルシステム領域	
pd_max_list_users で指定したリスト作成ユーザ数	
pd_max_list_count で指定した 1 ユーザ当たりのリスト作成数	
スワップ先にできない監査証跡ファイル数	
サーバ内のリスト作成数	KFPH22023-W

《ほかのオペランドとの関連》

このオペランドに AUTO を指定すると，次に示すオペランドの省略値に 80 が設定されます。したがって，AUTO を指定すると，次に示すオペランドを各々指定する必要がなくなります。ただし，各オペランドの値を 80 から変更したい場合は，次に示す各オペランドで変更してください。

- pd_max_users_wrn_pnt※
- pd_max_access_tables_wrn_pnt
- pd_max_rdarea_no_wrn_pnt
- pd_max_file_no_wrn_pnt
- pdwork_wrn_pnt
- pd_max_list_users_wrn_pnt
- pd_max_list_count_wrn_pnt
- pd_aud_file_wrn_pnt※
- pd_rdarea_list_no_wrn_pnt※

注※　警告メッセージの出力済み状態のリセット契機には 50 が設定されます。

《留意点》

バージョンアップ時に，監視できるリソースの対象が増えることがあります（前記以外のオペランドが増えることがあります）。したがって，常に現状と同じ監視体制にしておきたい場合は，このオペランドを指定しないで，前記のオペランドで項目ごとにリソース監視を実施してください。

62)　pd_max_users_wrn_pnt = HiRDB サーバへの接続数に関する警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕

HiRDB サーバへの接続数に関する警告メッセージの出力契機：～<符号なし整数>((0～100))《0 又は 80》（単位：%）

HiRDB サーバへの接続数がある一定の比率（pd_max_users オペランドで指定した最大同時接続数に対する比率）以上になったとき，警告メッセージを出力する比率を%単位で指定します。例えば，pd_max_users オペランドに 200 を指定し，このオペランドに 90 を指定すると，HiRDB サーバへの接続数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

- このオペランドに 0 を指定すると，警告メッセージを出力しません。
- pd_max_users オペランドの指定値が 9 以下の場合，このオペランドの指定は無効になります。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

警告メッセージの出力済み状態のリセット契機：～<符号なし整数>((0～99))（単位：%）

警告メッセージの出力済み状態のリセット契機を指定します。警告メッセージ（KFPS05123-W）が出力されると，HiRDB は警告メッセージ出力済み状態となります。警告メッセージ出力済み状態となると，HiRDB サーバへの接続数がもう一度警告メッセージの出力契機の値以上になっても警告メッセージが出力されません。しかし，ここで指定する警告メッセージの出力済み状態のリセット契機を HiRDB サーバへの接続数が下回ると，警告メッセージの出力済み状態が解除されます。

例えば，pd_max_users_wrn_pnt=90,70 と指定すると，HiRDB サーバへの接続数が最大同時接続数の 90%以上になると警告メッセージが出力されます。この後，HiRDB サーバへの接続数が最大同時接続数の 70%を下回るまで警告メッセージは出力されません。一度 70%を下回り，再度 90%以上になったときに警告メッセージが出力されます。

《注意事項》

- この指定を省略すると，「警告メッセージの出力契機－30」の値が仮定されます。その値がマイナスになる場合は，0 が仮定されます。
- 警告メッセージの出力契機より大きい値を指定すると，その指定は無効になり，警告メッセージの出力契機の指定値と同じ値が仮定されます。

63)　pd_max_access_tables_wrn_pnt = 同時アクセス可能実表数に関する警告メッセージの出力契機～<符号なし整数>((0～100))《0 又は 80》（単位：%）

同時にアクセスする表数と順序数生成子数の合計が，ある一定の比率（pd_max_access_tables オペランドで指定した同時アクセス可能実表数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_access_tables オペランドに 200 を指定し，このオペランドに 90 を指定すると，同時にアクセスする表数と順序数生成子数の合計が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

64）pd_max_rdarea_no_wrn_pnt = RD エリア数に関する警告メッセージの出力契機

～<符号なし整数>((0～100))《0 又は 80》（単位：%）

RD エリア数が，ある一定の比率（pd_max_rdarea_no オペランドで指定した RD エリアの最大数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_rdarea_no オペランドに 200 を指定し，このオペランドに 90 を指定すると，RD エリア数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

65）pd_max_file_no_wrn_pnt = HiRDB ファイル数に関する警告メッセージの出力契機

～<符号なし整数>((0～100))《0 又は 80》（単位：%）

RD エリアを構成する HiRDB ファイル数が，ある一定の比率（pd_max_file_no オペランドで指定した RD エリアを構成する HiRDB ファイルの最大数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_file_no オペランドに 200 を指定し，このオペランドに 90 を指定すると，RD エリアを構成する HiRDB ファイル数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

66）pdwork_wrn_pnt = 作業表用ファイルに関する警告メッセージの出力契機

～<符号なし整数>((0～100))《0 又は 80》（単位：%）

作業表用ファイル用の HiRDB ファイルシステム領域（pdwork オペランドで指定した作業表用ファイル用の HiRDB ファイルシステム領域）の使用率※が，ある一定の比率以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，このオペランドに 90 を指定すると，作

業表用ファイル用の HiRDB ファイルシステム領域の使用率が 90%以上になると警告メッセージ KFPS05123-W が出力されます。

注※

pdfmkfs コマンドで HiRDB ファイルシステム領域を作成するときに指定する次の項目の使用率を示しています。次に示す項目ごとにそれぞれ警告メッセージが出力されます。

- 容量に対する使用率
- 最大ファイル数に対する使用率
- 最大増分回数に対する使用率

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

67）pd_max_list_users_wrn_pnt = リスト作成ユーザ数に関する警告メッセージの出力契機

～<符号なし整数>((0～100))《0 又は 80》（単位：%）

リストを使用するユーザ数がある一定の比率（pd_max_list_users オペランドで指定したリスト作成ユーザ数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_list_users オペランドに 200 を指定し，このオペランドに 90 を指定すると，リストを使用するユーザ数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

- このオペランドに 0 を指定すると，警告メッセージを出力しません。
- 警告メッセージは 1 回しか出力されません。ただし，警告メッセージ出力後に対応するサーバを開始し直すと，再度使用率のチェックが実行され，警告メッセージが出力されます。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

68）pd_max_list_count_wrn_pnt = 1 ユーザ当たりのリスト作成数に関する警告メッセージの出力契機

～<符号なし整数>((0～100))《0 又は 80》（単位：%）

1 ユーザ当たりのリスト作成数がある一定の比率（pd_max_list_count オペランドで指定した 1 ユーザ当たりのリスト作成数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_list_count オペランドに 200 を指定し，このオペランドに 90 を指定すると，あるユーザのリスト作成数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

- このオペランドに 0 を指定すると，警告メッセージを出力しません。
- 警告メッセージは 1 回しか出力されません。ただし，警告メッセージ出力後に対応するサーバを開始し直すと，再度使用率のチェックが実行され，警告メッセージが出力されます。
- 複数のユーザが同時にリストを作成している場合，タイミングによっては一つのサーバで警告メッセージが 2 回以上出力されることがあります。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

69）　pd_rdarea_list_no_wrn_pnt = サーバ内のリスト作成数に関する警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕

サーバ内のリスト作成数に関する警告メッセージの出力契機：～<符号なし整数>((0～100))《0 又は 80》（単位：%）

サーバ内のリスト作成数がリスト作成可能数に対してある一定の比率以上になったとき，警告メッセージを出力する比率を%単位で指定します。例えば，サーバ内のリスト作成可能数が 1000 の場合，このオペランドに 90 を指定すると，リスト作成数が 900 以上になると警告メッセージ KFPH22023-W が出力されます。

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

警告メッセージの出力済み状態のリセット契機：～<符号なし整数>((0～99))（単位：%）

警告メッセージの出力済み状態のリセット契機を指定します。

警告メッセージ（KFPH22023-W）が出力されると，HiRDB は警告メッセージ出力済み状態となります。警告メッセージ出力済み状態となると，リスト作成数がもう一度警告メッセージの出力契機の値以上になっても警告メッセージが出力されません。しかし，ここで指定する警告メッセージの出力済み状態のリセット契機をリスト作成数が下回ると，警告メッセージの出力済み状態が解除されます。

例えば，pd_rdarea_list_no_wrn_pnt=90,70 と指定すると，リスト作成数がリスト作成可能数の 90%以上になると警告メッセージが出力されます。この後，リスト作成数がリスト作成可能数の 70%を下回るまで警告メッセージは出力されません。一度 70%を下回り，再度 90%以上になったときに警告メッセージが出力されます。

《注意事項》

- この指定を省略すると，「警告メッセージの出力契機－30」の値が仮定されます。その値がマイナスになる場合は，0 が仮定されます。

- 警告メッセージの出力契機より大きい値を指定すると，その指定は無効になり，警告メッセージの出力契機の指定値と同じ値が仮定されます。

2.2.11　SQL 実行時間警告出力機能に関するオペランド

70）　pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜SQL 実行時間警告情報の出力条件（時間指定）

SQL 実行時間警告出力機能を使用する場合にこのオペランドを指定します。このオペランドの指定方法は次に示す二つの方法があります。

- 比率で指定する方法
- 時間で指定する方法

SQL 実行時間警告出力機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

SQL 実行時間警告情報の出力条件（比率指定）：～<符号なし整数>((0～99))又は<符号なし 10 進数>((0～99.999999))《0》（単位：%）

SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）に対する比率で指定します。SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間がこのオペランドの指定値によって設定される時間以上になった場合，次に示す警告情報を出力します。これを SQL 実行時間警告出力機能といいます。

- SQL 実行時間警告情報ファイル
- 警告メッセージ（KFPA20009-W）

《オペランドの指定方法》

PDCWAITTIME オペランドの値に対する比率（%単位）で指定します。例えば，PDCWAITTIME オペランドに 100（秒）を指定し，このオペランドに 90（%）を指定すると，SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間が 90 秒以上 100 秒未満の場合に警告情報が出力されます。

（例）

PDCWAITTIME = 100

pd_cwaittime_wrn_pnt = 90

SQL 実行時間警告情報の出力条件（時間指定）：～<符号なし 10 進数>((0～65534.999999))sec《0》（単位：秒）

SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を時間で指定します。SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間がこのオペランドの指定値によって設定される時間以上になった場合，次に示す警告情報を出力します。これを SQL 実行時間警告出力機能といいます。

- SQL 実行時間警告情報ファイル
- 警告メッセージ（KFPA20009-W）

《オペランドの指定方法》

出力契機とする時間を秒単位で指定します（小数点以下 6 けたまで指定できます）。指定値に「sec」を付けてください。

（例）

pd_cwaittime_wrn_pnt = 0.001sec

これ以降の説明は，比率指定及び時間指定の両方共通の説明です。

《オペランドの規則》

このオペランドに 0 又は 0sec を指定すると，警告情報を出力しません（SQL 実行時間警告出力機能を使用しません）。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDCWAITTIMEWRNPNT オペランドを指定してください。
PDCWAITTIMEWRNPNT オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_cwaittime_report_dir
- pd_cwaittime_report_size

71）　pd_cwaittime_report_dir = SQL 実行時間警告情報ファイルの出力先ディレクトリ～<パス名>

SQL 実行時間警告情報ファイルを出力する場合に，出力先のディレクトリを絶対パス名で指定します。ここで指定したディレクトリ下に SQL 実行時間警告情報ファイルが二つ（pdcwwrn1 及び pdcwwrn2）作成されます。

なお，このオペランドを省略すると，SQL 実行時間警告情報ファイルを出力しません。ただし，警告メッセージ（KFPA20009-W）は出力されます。

HiRDB/パラレルサーバの場合，警告対象となった SQL を発行する UAP が接続するフロントエンドサーバがあるサーバマシンに SQL 実行時間警告情報ファイルを出力します。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

72）　pd_cwaittime_report_size = SQL 実行時間警告情報ファイルの最大容量～<符号なし整数>((2048～2147473627))《100000》（単位：バイト）

SQL 実行時間警告情報ファイルの最大容量を指定します。このオペランドに指定した値は SQL 実行時間警告情報ファイル一つ分の容量です。SQL 実行時間警告情報ファイルは二つ作成されるため，指定値に注意してください。例えば，10,000 を指定すると，ディレクトリ下に最大 10,000 バイトのファイルが二つ作成されます。

《指定値の目安》

次に示す計算式を目安にこのオペランドの指定値を決めてください。

{1280 ＋ SQL 文のサイズ（バイト）}　×ファイルに格納する警告情報数

SQL 文に注釈（コメント）や SQL の最適化を指定している場合は，注釈及び SQL の最適化の指定サイズ（バイト）も，SQL 文のサイズに含めてください。

《備考》

- SQL 実行時間警告情報ファイルへの出力容量がこのオペランドの値を超えると，出力先をもう片方のファイルに切り替えます。これを繰り返して二つのファイルを交互に使用します。このとき，切り替え先の古い情報は削除されます。
- 1 回で出力する SQL 実行時間警告情報の大きさがファイル容量を超えた場合，SQL 実行時間警告情報をすべて出力しません。ファイル容量までの SQL 実行時間警告情報を出力します。この場合，SQL 実行時間警告情報の最後に#を付けます。

2.2.12　拡張 SQL エラー情報出力機能に関するオペランド

73)　pd_uap_exerror_log_use = YES | NO

拡張 SQL エラー情報出力機能を使用するかどうかを指定します。拡張 SQL エラー情報出力機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

YES：

拡張 SQL エラー情報出力機能を使用します。エラーログファイル及び SQL エラーレポートファイルに SQL エラー情報を出力します。

NO：

拡張 SQL エラー情報出力機能を使用しません。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDUAPEXERLOGUSE オペランドを指定してください。このオペランドと，クライアント環境定義 PDUAPEXERLOGUSE の両方を指定した場合は，クライアント環境定義 PDUAPEXERLOGUSE が優先されます。

PDUAPEXERLOGUSE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

74)　pd_uap_exerror_log_dir = SQL エラーレポートファイルの格納ディレクトリ

～< 255 文字以内のパス名>

SQL エラーレポートファイルの格納ディレクトリを絶対パス名で指定します。

指定したディレクトリに，二つの SQL エラーレポートファイルが作成されます。ファイル名は pduaperrlog1 と pduaperrlog2 です。

このオペランドを省略した場合は，SQL エラー情報が SQL エラーレポートファイルに出力されません。

拡張 SQL エラー情報出力機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《注意事項》

指定したパス名のアルファベットの大文字と小文字を同一文字として扱います。

75)　pd_uap_exerror_log_size = SQL エラーレポートファイルの最大サイズ

～<符号なし整数>((2048～2147483647))《1000000》（単位：バイト）

SQL エラーレポートファイルの最大サイズを指定します。このオペランドで指定した値は，二つ作成される SQL エラーレポートファイルのうち一つ分の容量になります。SQL エラーレポートファイルへの出力容量がこのオペランドの値を超えると，出力先をもう片方のファイルに切り替えます。これを繰り返して二つのファイルを交互に使用します。1 回の処理で出力する SQL エラー情報のサイズがこのオペランドの値を超える場合，先頭からこの指定値－1 バイトまで（このオペランドに 1000000 を指定した場合 999999）の SQL エラー情報に#（シャープ 1 文字）を付けて出力します。

《指定値の目安》

このオペランドの指定値は，残しておきたい SQL エラー情報件数を考慮して決定してください。計算式を次に示します。

（A+B）×保持したい件数

- A = 1100+SQL 文のサイズ（単位：バイト）

  パラメタ情報の出力サイズを除いた SQL エラー情報の 1 件当たりのサイズです。SQL 文に注釈（コメント）や SQL 最適化指定を記述している場合は，注釈（コメント）及び SQL 最適化指

定分のサイズも，SQL 文のサイズに含めます。注釈及び SQL 最適化指定については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。

- B ＝（↑pd_uap_exerror_log_param_size オペランドの値÷16↑＋1）×89×パラメタの個数
  パラメタ情報の出力サイズです。

76）　pd_uap_exerror_log_param_size ＝ エラーログファイル及び SQL エラーレポートファイルへ出力するパラメタ情報の最大データ長

～＜符号なし整数＞((0～32008))《0》（単位：バイト）

エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。

- 1 以上を指定した場合
  エラーログファイル及び SQL エラーレポートファイルにパラメタ情報を出力します。
- 0 を指定した場合
  エラーログファイル及び SQL エラーレポートファイルにパラメタ情報を出力しません。

可変長文字列型，BLOB 型，及び BINARY 型の場合，データ長の領域もこの指定値に含まれます。

エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の大きさがこのオペランドの指定値を超えた場合，指定値の長さまでパラメタ情報を出力します。この場合，超過分は切り捨てられます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDUAPEXERLOGPRMSZ オペランドを指定してください。このオペランドと，クライアント環境定義 PDUAPEXERLOGPRMSZ の両方を指定した場合は，クライアント環境定義 PDUAPEXERLOGPRMSZ が優先されます。

PDUAPEXERLOGPRMSZ オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

2.2.13　SQL 予約語削除機能に関するオペランド

77）　pd_delete_reserved_word_file ＝ SQL 予約語削除ファイル名称 1〔，SQL 予約語削除ファイル名称 2〕…

～＜ 8 文字以下の識別子＞

SQL 予約語削除機能を使用する場合に，このオペランドを指定します。このオペランドには SQL 予約語削除ファイルの名称を指定します。SQL 予約語削除ファイルは，%PDDIR%¥conf¥pdrsvwd 下に作成してください。SQL 予約語削除機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《オペランドの規則》

- SQL 予約語削除ファイルの名称は，先頭がアルファベットの英数字（最大 8 文字）の任意の名称にしてください。大文字と小文字は同一文字として扱います。
- SQL 予約語削除ファイル名称は最大 16 個指定できます。

《SQL 予約語削除ファイルの形式》

予約語〔,予約語…〕

- 予約語を複数指定する場合は，コンマ（,）で区切って指定してください。また，コンマの前後に空白やタブを挿入しないでください。

- 定義の 1 行の長さは最大 80 文字です。80 文字を超える指定の場合は，改行してください。ただし，予約語（キーワード）の途中では改行しないでください。
- 予約語は，大文字でも小文字でも指定できます。
- SQL の予約語については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。

《注意事項》

予約語を削除した場合，その予約語を使う SQL の機能が使用できなくなります。予約語削除機能で削除できる予約語については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。

《クライアント環境定義との関連》

UAP 実行時に使用する SQL 予約語削除ファイルは，クライアント環境定義 PDDELRSVWDFILE で指定してください。PDDELRSVWDFILE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

2.2.14　SQL からのコマンド実行に関するオペランド

78）　pd_sql_command_exec_users = 認可識別子〔，認可識別子〕…

～＜ 8 文字以下の記号名称＞

SQL の CALL COMMAND 文でコマンド及びユティリティを実行できる認可識別子を指定します。認可識別子を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されます。引用符で囲まない場合は，すべて大文字として扱われます。

《注意事項》

認可識別子に引用符を含める場合は，指定する認可識別子中の引用符の前に「¥」を記述してください。

2.2.15　SQLSTATE の細分化に関するオペランド

79）　pd_standard_sqlstate = Y｜N

SQLSTATE の値を詳細に出力するかどうかを指定します。

Y：

SQLSTATE の値を詳細に出力します。

N：

SQLSTATE の値を詳細に出力しません。

出力される SQLSTATE の値については，マニュアル「HiRDB Version 8 メッセージ」を参照してください。

2.2.16　排他制御に関するオペランド

80）　pd_lck_deadlock_info = Y｜N

デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報を出力するかどうかを指定します。デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報は，%PDDIR%¥spool¥pdlckinf 下に出力されます。デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

Y：デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報を出力します。

N：デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報を出力しません。

《ほかのオペランドとの関連》

- pd_lck_deadlock_check オペランドに N を指定した場合は，このオペランドには常に N が仮定されます。
- このオペランドは pd_lck_deadlock_check オペランドと関連があります。

81）　pd_lck_wait_timeout = 排他待ち限界経過時間

～<符号なし整数>((0～65535))《Max(180，pd_watch_time の値)》（単位：秒）

排他待ち時間を監視する最大時間を秒単位で指定します。排他待ち時間とは，排他要求が待ち状態になってから解除されるまでの時間のことです。

指定した時間内に待ち状態が解除されない場合，SQL はエラーリターンします。0 を指定した場合，排他待ち時間を監視しないで，待ち状態が解除されるまで待ち続けます。

《注意事項》

このオペランドを省略すると，pd_watch_time オペランドの値が省略値として仮定されることがあります。pd_watch_time オペランドは HiRDB/シングルサーバに指定しても無効になりますが，pd_lck_wait_timeout オペランドの省略値に使用されることがあります。このため，HiRDB/シングルサーバの場合は pd_watch_time オペランドを省略することをお勧めします。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDLCKWAITTIME を指定してください。PDLCKWAITTIME については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_sysdef_default_option オペランドに v6compatible を指定している場合，このオペランドの省略値は 180 になります。
- このオペランドは，次に示すオペランドと関連があります。
  - ・pd_lck_deadlock_check
  - ・pd_lck_deadlock_check_interval

82）　pd_lck_release_detect = interval | pipe

このオペランドでは，HiRDB が排他の解除を検知する方法（資源を占有している処理が，資源の占有を解除したかどうかを知る方法）について指定します。

interval：一定間隔で排他管理用領域を参照して排他解除状態を調べます。

pipe：pipe ファイル（プロセス間通信のパイプ）を使用して排他解除通知を受け取ります。

省略値は HiRDB/シングルサーバの場合は pipe が，HiRDB/パラレルサーバの場合は interval が仮定されます。

《指定値の目安》

このオペランドの指定値を変更するときの目安を次に示します。

指定値	HiRDB の処理方式	利点・適用基準
interval	共用メモリにある排他管理用領域を参照して，排他が解除されたかどうかを検知します。HiRDB は排他管理用領域を一定間隔（一定のインターバル）で参照しています。インターバル時間は，pd_lck_release_detect_interval オペランドで指定します。	資源を占有している処理が排他を解除しても，次の排他管理用領域の参照時までそれを検知できません。このため，1 トランザクションの処理時間が短い UAP では，排他待ち時間が長くなることがあります。ただし，この待ちでは CPU に負荷を掛けたり，ファイルをオープンしたりしません。 pd_lck_release_detect_interval オペランドに小さい値を指定すると，CPU 利用率が上がり過ぎてス

指定値	HiRDB の処理方式	利点・適用基準
		ループットが伸びないことがあります。低速な CPU を使用している場合に，CPU 負荷を軽減する効果があります。
pipe	セマフォを使用して排他が解除されたかどうかを検知します。資源を占有している処理から排他待ちをしている処理に排他解除が通知されます。排他占有プロセスが排他を解除した時点で，排他待ちプロセスは排他の解除を検知できます。	• 1 トランザクションの処理時間が pd_lck_release_detect_interval オペランドの値以下の場合は，スループットが向上します。 • 排他待ちが多発した場合，CPU に負荷が掛かります。 • 高速な CPU を使用している場合は，スループットを向上させる効果があります。

83)　pd_lck_release_detect_interval = 排他解除検知インターバル時間
～<符号なし整数>((1～1000))《10》(単位：ミリ秒)

排他管理用領域の参照は一定のインターバルをおいて行われます。このオペランドにはそのインターバル時間を指定します。

・このオペランドに 49 以下を指定した場合

インターバル時間はこのオペランドで指定した値から始まり，最大 50 ミリ秒になります。

・このオペランドに 50 以上を指定した場合

インターバル時間は 50 ミリ秒から始まり，最大このオペランドで指定した値になります。

《前提条件》

pd_lck_release_detect オペランドに interval を指定している必要があります。

なお，HiRDB/パラレルサーバの場合は interval が省略値です。

《指定値の目安》

- 基本的にはこのオペランドを省略してください。
  短時間の間に排他が多発し，排他獲得待ちが発生しているため，トランザクション処理性能が大幅に低下している場合に指定値の変更を検討してください。
- 指定値を小さくすると，排他待ちが多発したときに CPU に負荷が掛かります。
- 指定値を大きくすると，排他待ち時間が長くなることがあります。
- 統計解析ユティリティのシステムの稼働に関する統計情報の「排他待ち時間（WAIT TIME)」を参考にして指定値を決定してください。統計情報に出力された排他待ち時間がこのオペランドの値に比べて短い場合は，このオペランドの値を小さくしてください。

84)　pd_nowait_scan_option = LOCK | NOLOCK

WITHOUT LOCK NOWAIT 指定の検索を実行したときの処理方式を指定します。

LOCK：

WITHOUT LOCK NOWAIT 指定の検索を実行した場合，行単位で排他を確保します。このとき，ほかのトランザクションが更新前か更新後の行だけを参照できます。ほかのトランザクションが更新中の場合は，その行の更新処理が終了するまで待機します。

NOLOCK：

WITHOUT LOCK NOWAIT 指定の検索を実行した場合，排他を確保しません。このとき，ほかのトランザクションが更新中の行も参照できます。ただし，更新中の行を参照した場合，検索対象の行を受け取れないことがあります。

《注意事項》

このオペランドに LOCK を指定して，非 FIX 表の UPDATE 及び WITHOUT LOCK NOWAIT 指定の検索を実行すると，排他資源数が 1 トランザクション当たり 1 個増加します。また，非 FIX 表に対する UPDATE 文による更新で出力されるデータ 1 件当たりの基本行ログ量が 628 バイト増加します。

85）　pd_lck_queue_limit = 排他待ちユーザ数警告メッセージ出力契機

～<符号なし整数>((0～500))《10》

同一リソースへの排他処理で，排他待ちユーザ数が多数発生したとき，警告メッセージを出力する境界となる排他待ちユーザ数を指定します。

同一リソースでの排他待ちユーザ数が指定値に達すると，警告メッセージ（KFPS00446-W）を出力します。メッセージの出力先は，メッセージログファイル及びイベントログです。

指定値が 0 の時は，警告メッセージを出力しません。

86）　pd_deadlock_priority_use = Y | N

デッドロックの優先順位を制御するかどうかを指定します。

Y：

デッドロック発生時，どのプログラムを優先的に実行させるかを制御します。デッドロックが発生した場合，優先順位の一番低いプログラムの SQL がエラーとなります。優先順位が等しい場合は，トランザクションを実行した時刻が一番遅いプログラムの SQL がエラーとなります。デッドロックの優先順位は，クライアント環境定義の PDDLKPRIO オペランドで指定してください。クライアント環境定義の PDDLKPRIO オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

N：

実行したトランザクションの時刻が一番遅いプログラムの SQL がエラーとなります。

87）　pd_command_deadlock_priority = 32 | 64 | 96 | 120

コマンドのデッドロックプライオリティ値を指定します。

このオペランドの指定値が有効になる運用コマンドを次に示します。

- pdhold -b（バックアップ閉塞）

コマンドのデッドロックプライオリティ値を変更する方法についてはマニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《前提条件》

pd_deadlock_priority_use オペランドに Y が指定されている必要があります。

《指定値の目安》

デッドロック発生時，業務プログラム側でリトライ可能なシステムの場合，運用コマンドの優先度を高くして業務プログラム側をエラーにできます。指定する値が小さいほど排他時の優先度は高くなります。

88）　pd_key_resource_type = TYPE1 | TYPE2

インデクスキー値の排他資源の作成方法を指定します。インデクスキー値の排他資源の作成方法については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

TYPE1：インデクスキー値の排他資源をビットシフトしながら排他的論理和で作成します。

TYPE2：インデクスキー値の排他資源をバイトオーダの排他的論理和で作成します。

《指定値の目安》

- 通常は TYPE1 を指定します。キー長が 10 バイトを超える場合，インデクスキー値の排他資源でシノニムによる排他競合が発生することがあります。TYPE1 を指定すると，これを防止できます。
- TYPE2 は，HiRDB Version 5.0 05-02 又は HiRDB Version 4.0 04-05 より前の処理方式です。バージョンアップしたユーザで，以前の処理方式でインデクスキー値の排他資源でシノニムによる排他競合が発生していなかった場合は，TYPE2 を指定してください。
- インデクスの構成列に文字列データを含む場合には TYPE1 を指定します。
- インデクスの構成列に数値データを含む場合には TYPE2 を指定します。

89）　pd_indexlock_mode = {KEY | NONE}

B-tree インデクスの排他制御方式を指定します。

KEY：インデクスキー値で排他をします。

NONE：インデクスキー値で排他をしません（**インデクスキー値無排他**を実施します）。

インデクスキー値無排他については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは pd_lock_uncommited_delete_data オペランドと関連があります。詳細については，pd_lock_uncommited_delete_data オペランドの説明を参照してください。

90）　pd_lock_uncommited_delete_data = WAIT | NOWAIT

コミットしていない削除データの排他待ち制御方式を指定します。

WAIT：

DELETE 文で削除した行，又は UPDATE 文で更新した更新前のインデクスキーに対し，検索時に削除又は更新したトランザクションが決着するまで排他待ちし，トランザクション決着後のデータを基に検索結果を返す方式を指定します。また，**インデクスキー値排他**と同じ方式で一意性制約が保証されます。インデクスキー値排他については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

NOWAIT：

DELETE 文で削除した行，又は UPDATE 文で更新した更新前のインデクスキーに対し，検索時に削除又は更新したトランザクションの決着を待たないで検索結果を返す方式を指定します。

《注意事項》

次の場合には，このオペランドに WAIT を指定しても無効になります。

- ディクショナリ表を検索又は更新する場合
- WITHOUT ROLLBACK オプションを指定した表にインデクスを定義しないで検索又は更新する場合

このオペランドに WAIT を指定すると，インデクスが定義された表の表格納 RD エリアを構成する HiRDB ファイルの使用可能サイズが次のように制限されます。

表格納 RD エリアのページ長（バイト）	pd_lock_uncommited_delete_data の指定値	
	WAIT	NOWAIT
4096	32GB	64GB
6144	48GB	

表格納 RD エリアのページ長（バイト）	pd_lock_uncommited_delete_data の指定値	
	WAIT	NOWAIT
上記以外	64GB	

HiRDB ファイルのサイズが既に使用可能サイズを超えている場合，又は使用可能サイズ以上のサイズのファイルを使用したい場合には，表格納 RD エリアのページサイズを 8192 バイト以上に変更してください。ページサイズを変更しないでそのまま運用すると，次の場合にエラーとなり，警告メッセージ KFPA19176-E が出力されます。

- 行の追加で使用可能サイズを超える領域を割り当てたとき
- 使用可能サイズを超える領域のページに格納されている行に対して更新又は削除の操作をしたとき

《クライアント環境定義との関連》

クライアント環境定義の PDLOCKSKIP オペランドに YES を指定している場合は，このオペランドに WAIT を指定しても無効となり，無排他条件判定の処理が優先されます。そのため，削除中の行では排他待ちをしません。ただし，削除した行のインデクスは残存エントリとして残ります。

《ほかのオペランドとの関連》

このオペランドの指定値と pd_indexlock_mode オペランドの指定値には，指定できない組み合わせがあります。指定できない組み合わせを次に示します。

pd_lock_uncommited_delete_data の指定値	pd_indexlock_mode の指定値	
	NONE	KEY
WAIT	○	×※1
NOWAIT	○	○
NOWAIT（過去に WAIT 指定履歴あり）	○	×※2

（凡例）

○：指定できる組み合わせです。

×：指定できない組み合わせです。

注※1

指定した場合，HiRDB 起動時にエラー（KFPS01857-E）となります。

注※2

このオペランドに WAIT を指定して HiRDB を開始したことがあると，このオペランドに NOWAIT を指定して pd_indexlock_mode オペランドに KEY を指定した場合，DB アクセス時に不正な検索結果が出力されます。データベース初期設定ユティリティ（pdinit）を実行して HiRDB を初期化してから再度オペランドを指定してください。

91）　pd_lck_deadlock_check = Y | N

デッドロックの発生を監視するかどうかを指定します。

Y：デッドロックの発生を監視します。

N：デッドロックの発生を監視しません。

《指定時の目安》

デッドロックが発生しない業務システムでは，このオペランドに N を指定すると SQL の実行性能が改善することがあります。特に，デッドロックの検出方法がインターバル監視方式の場合，排他

制御用プールパーティション数が増加するとデッドロックを検出するたびに排他制御の性能が劣化することがあるため，デッドロックが発生しない業務システムを構築した上でこのオペランドにNを指定することをお勧めします。

デッドロックが発生する業務システムでは，このオペランドにYを指定してください。Nを指定すると，デッドロックが発生した場合，pd_lck_timeoutオペランドに指定した時間が経過するまでSQLが終了しません。また，HiRDBがデッドロック情報を出力しないため，デッドロックが発生した要因が分からなくなるおそれがあります。

《注意事項》

このオペランドにNを指定すると，デッドロックが発生したトランザクションはエラーとなりません。この場合，トランザクションは次の要因でキャンセルされます。

- 排他待ち時間を監視する最大時間が経過し，SQLがエラーリターンする
- HiRDBクライアントの最大待ち時間が経過し，要求がUAPにエラーリターンする

《クライアント環境定義との関連》

このオペランドは次に示すクライアント環境定義と関連があります。

- PDCWAITTIME
- PDLCKWAITTIME

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_deadlock_info
- pd_lck_pool_partition
- pd_fes_lck_pool_partition
- pd_lck_deadlock_check_interval
- pd_lck_wait_timeout

92)　pd_lck_deadlock_check_interval = デッドロック監視時間間隔

～<符号なし整数>((1～2000000000))（単位：ミリ秒）《1000》

インターバル監視方式でデッドロックの発生を監視する場合の，監視を実施する間隔を指定します。

《前提条件》

このオペランドの前提条件は，サーバ種別によって異なります。

シングルサーバ，バックエンドサーバ，又はディクショナリサーバの場合は，次に示す条件をすべて満たしている必要があります。

- pd_lck_deadlock_checkオペランドにYを指定している
- pd_lck_pool_partitionオペランドに2以上の値を指定している

フロントエンドサーバの場合は，次に示す条件をすべて満たしている必要があります。

- pd_lck_deadlock_checkオペランドにYを指定している
- pd_fes_lck_pool_partitionオペランドに2以上の値を指定している

《指定時の目安》

デッドロックが発生してから検出するまでの時間を短くしたい場合は，このオペランドの指定値を小さくしてください。

《注意事項》

- このオペランドの指定値を小さくし過ぎると，システムのパフォーマンスが低下する場合があります。
- このオペランドの指定値を大きくし過ぎると，次の要因でトランザクションがキャンセルされることがあります。
  - ・排他待ち時間を監視する最大時間が経過し，SQL がエラーリターンする
  - ・HiRDB クライアントの最大待ち時間が経過し，要求が UAP にエラーリターンする

《クライアント環境定義との関連》

このオペランドは次に示すクライアント環境定義と関連があります。

- PDCWAITTIME

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_deadlock_check
- pd_lck_pool_partition
- pd_fes_lck_pool_partition
- pd_lck_wait_timeout

2.2.17　バッファに関するオペランド

93)　pd_sql_object_cache_size = SQL オブジェクト用バッファ長

～<符号なし整数>《(pd_max_users の値＋ 3) ×22》(単位：キロバイト)

- 32 ビットモードの場合：((22～256000))
- 64 ビットモードの場合：((22～2000000))

SQL オブジェクトを格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- SQL オブジェクトは，ユーザごとにトランザクションが終了するまでバッファに保存されます。したがって，同時に実行するトランザクションの SQL オブジェクトがバッファに格納できる大きさが必要です。
- 静的 SQL の SQL オブジェクトはトランザクション終了後もバッファに空きがなくなるまで保存して，同一 UAP を実行する複数のユーザで共用することで，SQL 解析処理の削減を図ります。バッファを有効に利用するためには，使用頻度の高い UAP の SQL オブジェクトがバッファに常駐するようにバッファを確保してください。
- バッファ長の見積もり時は，まず UAP で発行する SQL 文の SQL オブジェクト長から UAP 実行に必要なバッファ長を見積もります。次に，同時に実行される UAP と同時実行ユーザ数を考慮して必要なバッファ長を算出してください。
- 1SQL 文の SQL オブジェクト長の見積もり方法については，「付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

SQL オブジェクト用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

94)　pd_def_buf_control_area_assign = INITIAL | TRAN

表定義情報用バッファ，ビュー解析情報用バッファ，ユーザ定義型情報用バッファ，及びルーチン定義情報用バッファの管理領域を HiRDB 開始時に一括して確保するか，又はトランザクション開始時に確保するかを指定します。

INITIAL：

各種バッファの管理領域を HiRDB 開始時に一括して確保します。この管理領域は HiRDB を終了するまで解放されません。

TRAN：

各種バッファの管理領域をトランザクション開始時に確保します。この管理領域はトランザクション終了時に解放されます。

《指定値の目安》

性能を重視する場合は INITIAL を指定してください。トランザクション性能が向上します。ただし，TRAN 指定時よりも共用メモリを多く必要とするため，メモリ所要量を見直してください。共用メモリの見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

95)　pd_thread_max_stack_size = 1 スレッドが使用する最大スタックサイズ

～<符号なし整数>((0～OS が許容する最大スタックサイズ（最大 2047)))《0》（単位：メガバイト）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

1 スレッドが使用するスタックサイズの最大値を指定します。0 を指定するか，又はこのオペランドを省略した場合, 1 スレッドが使用するスタックサイズを制限しません。HiRDB は擬似スレッド制御をしているため，独自に確保したメモリをスタックとして割り当てています。

《利点》

UAP の不良，又は UAP からの複雑な SQL の要求などによって，1 スレッドが多量のシステムリソースを使用することがあります。これが原因で OS がハングアップすることがあります。このオペランドを指定すると，1 スレッドが使用するスタックサイズを制限できるため，OS のハングアップが防止できます。

《注意事項》

このオペランドの指定値が，OS が許容する最大スタックサイズ以下であることを確認してください。

2.2.18　共用メモリに関するオペランド

96)　pd_shmpool_attribute = free | fixed

HiRDB のユニットコントローラが使用する共用メモリをメモリ上に固定するかどうかを指定します。なお，Windows のバージョンによって，サポートの有無が異なります。Windows の前提条件については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「システム設計」の「共用メモリのページ固定」を参照してください。サポートされていない場合は，このオペランドに fixed を指定しても無効になります。

free：

メモリ上に固定しません。実メモリ量によっては，別の共用メモリのページングが発生し，性能に影響する場合があります。

fixed：

メモリ上に固定します。

《利点》

HiRDB が使用する共用メモリをメモリ上に固定（fixed 指定）すると，共用メモリのページングを防止できます。このため，共用メモリに対するアクセス性能が向上します。

《指定値の目安》

算出した共用メモリサイズとサーバマシンの実メモリから，メモリを固定するかどうかを決定してください。HiRDB がメモリを固定できるサイズに上限があるため，メモリを固定するとほかのメモリのページングが多発することがあります。したがって，実メモリや全使用量に対する共用メモリプールの占める割合をよく検討する必要があります。

ユニットコントローラが使用する共用メモリの計算式，及び各サーバが使用する共用メモリについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《注意事項》

ページ固定に対応している Windows で fixed を指定する場合の注意事項を次に示します。

- pdntenv コマンドを使用して，共用メモリをページングファイルに割り当てる必要があります。pdntenv コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。
- 共用メモリは Windows の Large Page でのページサイズに切り上げて取得するため，メモリ所要量を見積もるときは注意してください。Windows の Large Page のページサイズについては，pdntenv -os コマンドで確認してください。

97）pd_dbbuff_attribute = free | fixed

グローバルバッファが使用する共用メモリを実メモリ上に固定するかどうかを指定します。なお，Windows のバージョンによって，サポートの有無が異なります。Windows の前提条件については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「システム設計」の「共用メモリのページ固定」を参照してください。サポートされていない場合は，このオペランドに fixed を指定しても無効になります。

free：

グローバルバッファが使用する共用メモリを実メモリにページ固定しません。

fixed：

グローバルバッファが使用する共用メモリを実メモリにページ固定します。

《利点》

グローバルバッファが使用する共用メモリを実メモリにページ固定（fixed 指定）すると，共用メモリのページングを防止できます。このため，共用メモリに対するアクセス性能が向上します。

《指定値の目安》

- 算出した共用メモリサイズとサーバマシンの実メモリから，メモリを固定するかどうかを決定してください。実メモリに比べて多大なメモリをページ固定すると，ページングの多発や仮想メモリ不足の原因になります。したがって，実メモリや全仮想メモリに対する共用メモリプールの占める割合を検討する必要があります。グローバルバッファが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- ページ固定は，実メモリサイズからページ固定する共用メモリサイズを減算したサイズが，スワップ領域サイズからページ固定する共用メモリサイズを減算したサイズの 1/2 以上にならないことを目安にしてください。

《ほかの機能との関連》

- fixed を指定した場合，動的変更したグローバルバッファが使用する共用メモリも実メモリ上に固定されます。そのため，実メモリサイズを十分考慮してグローバルバッファを追加又は変更してください。

《注意事項》

ページ固定に対応している Windows で fixed を指定する場合の注意事項を次に示します。

- pdntenv コマンドを使用して，共用メモリをページングファイルに割り当てる必要があります。pdntenv コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。
- 共用メモリは Windows の Large Page でのページサイズに切り上げて取得するため，メモリ所要量を見積もるときは注意してください。Windows の Large Page のページサイズについては，pdntenv -os コマンドで確認してください。
- SHMMAX オペランドに Windows の Large Page のページサイズに切り上げた値を指定してください。
- HiRDB 起動時にグローバルバッファプールが使用する共用メモリがメモリ上に固定できない場合，KFPH23045-W メッセージを出力します。その場合，fixed を指定しても共用メモリの一部又はすべてがメモリ上にページ固定されないで，処理を続行します。

2.2.19　メッセージログファイルに関するオペランド

98)　pd_mlg_msg_log_unit = manager | local

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

メッセージログの出力先ユニットを指定します。

manager：

システムマネジャユニットのメッセージログファイル及びイベントログに，メッセージログを出力します。メッセージログファイルはシステムマネジャのサーバマシンの%PDDIR%¥spool 下に作成されます。

local：

メッセージ出力元のユニットのメッセージログファイル及びイベントログに，メッセージログを出力します。メッセージログファイルは，各サーバマシンの%PDDIR%¥spool 下に作成されます。

《指定値の目安》

- システムマネジャユニットでメッセージログを一元管理したい場合は，このオペランドに manager を指定してください。
- システムマネジャユニットが障害などで停止している場合，メッセージログはメッセージ発行元のユニットのイベントログに一定時間遅延して出力されたり，又はメッセージ自体が出力されなかったりします。このオペランドに local を指定すると，メッセージを遅延することなくメッセージ出力元のユニットに出力できます。メッセージログの出力分散化については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

このオペランドの値とメッセージログの出力先の関係を次に示します。

pd_mlg_msg_log_unit の値	条件	メッセージログの出力先
manager	通常時	システムマネジャがあるサーバマシンのメッセージログファイル及びイベントログに出力されます。

pd_mlg_msg_log_unit の値	条件	メッセージログの出力先
	システムマネジャユニットの障害時又は通信障害時	各サーバマシンのイベントログに出力されます。出力されないメッセージログがあります。また，メッセージログが遅れて出力されることがあります。
local	通常時	各サーバマシンのメッセージログファイル及びイベントログに出力されます。
	システムマネジャユニットの障害時又は通信障害時	

《ほかのオペランドとの関連》

このオペランドに local を指定した場合，メッセージログファイルの容量を見積もり直し，その結果を pd_mlg_file_size オペランドに指定してください。

99）　pd_mlg_file_size = メッセージログファイルの最大容量

～<符号なし整数>((64～1000000))《1024》（単位：キロバイト）

メッセージログファイルの最大容量をキロバイト単位で指定します。

メッセージログファイルは二つあり，メッセージの量がここで指定した最大容量に達した場合，メッセージログファイルをスワップします。

pd_mlg_msg_log_unit オペランドに local を指定した場合，各サーバマシンにここで指定した容量のメッセージログファイルが作成されます。

2.2.20　統計情報に関するオペランド

100）　pd_statistics = Y | N

HiRDB の開始時から統計ログを取得するかどうかを指定します。

Y：HiRDB の開始時から統計ログを取得します。

N：HiRDB の開始時から統計ログを取得しません。

Y を指定した場合，取得する統計ログはユニット全体のシステムの稼働に関する統計情報だけです。サーバ別の情報，又はシステムの稼働に関する統計情報以外の情報が必要なときは，HiRDB の開始完了後に一度 pdstend コマンドで統計情報取得を停止し，改めて pdstbegin コマンドで統計情報取得を開始する必要があります。

なお，N を指定した場合でも，pdstbegin コマンドを入力すれば，HiRDB の稼働中に統計ログを取得できます。

《注意事項》

次に示す場合は統計ログを取得できません。

- pdstbegin オペランドを指定している場合
- 1：1 スタンバイレス型系切り替え機能を使用していて，代替 BES ユニットに系を切り替える場合
- 影響分散スタンバイレス型系切り替え機能を適用しているユニットの場合
- ユニット内に開始しているサーバが存在しない場合

101）　pd_stj_file_size = 統計ログファイルの最大容量

～<符号なし整数>((64～1000000))《1024》（単位：キロバイト）

統計ログファイルの最大容量をキロバイト単位で指定します。統計ログファイルは二つあり，統計ログの出力量がここで指定した最大容量に達したときに統計ログファイルをスワップします。

《指定値の目安》

- このオペランドの指定値の見積もり方法については，「付録 C.1　統計ログファイル（pd_stj_file_size）の見積もり式」を参照してください。
- pd_stj_buff_size オペランドとの間に次に示す関係が保たれるように指定してください。
  pd_stj_file_size≧pd_stj_buff_size×2

102）　pd_stj_buff_size = 統計ログバッファ長

～<符号なし整数>((32～512))《32》（単位：キロバイト）

統計ログバッファの大きさを指定します。

《指定値の目安》

次に示す統計情報を出力しない場合は，仮定値である 32 で問題ありません。

- SQL オブジェクト実行に関する統計情報
- SQL オブジェクト転送に関する統計情報

これらの統計情報を出力する場合は，次に示す計算式で求めた値に 32 を加算して指定してください。ただし，512 を超えた場合は，512 を指定してください。

（a÷1024）×（0.03÷b）

a：統計ログ出力量（バイト）

「付録 C.1　統計ログファイル（pd_stj_file_size）の見積もり式」を参照してください。

b：統計情報の出力時間（秒）

2.2.21　RPC トレース情報に関するオペランド

103）　pd_rpc_trace = Y｜N

RPC トレース情報を取得するかどうかを指定します。RPC トレース情報には, HiRDB の保守情報が出力されます。

通常，このオペランドは省略してください。

Y：RPC トレースを取得します。

N：RPC トレースを取得しません。

《注意事項》

このオペランドに Y を指定すると，通信性能が低下します。

104）　pd_rpc_trace_name = "RPC トレースを取得するファイル名称"

～< 254 文字以内のパス名>《%PDDIR%¥spool¥rpctr》

RPC トレースファイルの名称を絶対パス名で指定します。RPC トレースファイルとして，指定したファイル名の後に 1，2，l（エル）を付けた三つのファイルが作成されます。

《注意事項》

- このオペランドで指定したディレクトリ下に，最大「pd_rpc_trace_size の値× 2」の大きさのファイルが作成されます。ファイル容量に注意してください。

105）　pd_rpc_trace_size = RPC トレースを取得するファイルの容量

～<符号なし整数>((1024～2147483648))《4096》（単位：バイト）

RPC トレースファイルの容量をバイト単位で指定します。

このオペランド指定値は，"RPC トレースファイル"1 及び"RPC トレースファイル"2 のそれぞれの容量となります。

《指定値の目安》

RPC トレースは，すべてのプロセス間の通信電文が一つのファイルに時系列に蓄積されます。その量がこの指定値を超えると，もう一方のファイルにトレースデータの出力を切り替えます。古いトレースは上書きされて消去されます。このため，指定値が小さいと十分なトレースを取得できなくなり，トラブルシュートが困難になることがあります。したがって，このオペランドには 1,000,000 以上の値を指定してください。

《注意事項》

このオペランドで指定した値は RPC トレースファイル l（エル）の容量に影響を与えません。RPC トレースファイル l（エル）の容量は 0 バイト固定です。

2.2.22　トラブルシュート情報に関するオペランド

106）pd_cancel_dump = put | noput

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

次に示す場合にトラブルシュート情報を取得するかどうかを指定します。

- クライアント環境定義の PDCWAITTIME オペランドで指定した監視時間内に SQL が終了しなかった場合
- pdcancel コマンドで実行中の UAP を中断させた場合

取得されるトラブルシュート情報については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

put：

トラブルシュート情報を取得します。トラブルシュート情報が%PDDIR%¥spool 下のファイルに出力されるため，ファイルシステムが圧迫されるおそれがあります。

なお，出力されたトラブルシュート情報は次に示すタイミングで HiRDB が自動的に削除します。

- HiRDB が稼働中の場合は 24 時間ごと（削除間隔を pd_spool_cleanup_interval オペランドで変更できます）
- HiRDB を開始したとき（削除するかどうかを pd_spool_cleanup オペランドで変更できます）

HiRDB 管理者がトラブルシュート情報を削除する場合は，pdcspool コマンドを実行してください。

noput：

トラブルシュート情報を取得しません。トラブルシュート情報が出力されないため，ファイルシステムへの負荷を軽減できます。通常の運用で UAP の中断が頻繁に発生し，かつ，その原因を調査する必要がない場合に指定してください。

異常終了時に出力される障害情報については，pd_dump_suppress_watch_time オペランドの「表 2-1　異常終了時に出力される障害情報」を参照してください。

107）pd_client_waittime_over_abort = Y | N

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

トランザクション実行中にクライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）を超えた場合，次に示すトラブルシュート情報を取得するかどうかを指定します。

- 共用メモリダンプ（1 回だけ取得します）
- 簡易ダンプ
- ユニット内の詳細情報

トラブルシュート情報は%PDDIR%¥spool¥save 下に取得されます。

Y：

トラブルシュート情報を取得します。HiRDB 開始後の最初のクライアントの最大待ち時間オーバが起きた場合だけ，共用メモリダンプを取得します。HiRDB/パラレルサーバの場合，接続したフロントエンドサーバプロセスと関連するバックエンドサーバプロセス，ディクショナリサーバプロセスのトラブルシュート情報を取得します。HiRDB/シングルサーバの場合，シングルサーバプロセスのトラブルシュート情報を取得します。

N：

トラブルシュート情報を取得しません。HiRDB/パラレルサーバの場合，接続したフロントエンドサーバプロセスのトラブルシュート情報を取得しません。関連するバックエンドサーバプロセス，ディクショナリサーバプロセスはロールバックし，トラブルシュート情報は取得しません。HiRDB/シングルサーバの場合，シングルサーバプロセスのトラブルシュート情報を取得しません。

《注意事項》

- トラブルシュート情報を放置しておくと，ファイル容量を圧迫する原因になります。出力されたトラブルシュート情報は次に示すタイミングで HiRDB が自動的に削除します。

  ・HiRDB が稼働中の場合は 24 時間ごと（削除間隔を pd_spool_cleanup_interval オペランドで変更できます）

  ・HiRDB を開始したとき（削除するかどうかを pd_spool_cleanup オペランドで変更できます）

  HiRDB 管理者がトラブルシュート情報を削除する場合は，pdcspool コマンドを実行してください。

- クライアントの最大待ち時間を超えた場合でも，クライアントが接続しているシングルサーバプロセス又はフロントエンドサーバプロセス がクリティカル状態のときは，トラブルシュート情報が取得できないことがあります。プロセスがクリティカル状態かどうかは pdls -d rpc コマンドで確認できます。

《ほかのオペランドとの関連》

HiRDB/パラレルサーバの場合，pd_clt_waittime_over_dump_level オペランドの指定によって，共用メモリダンプを出力するユニットを制限し，共用メモリダンプの出力量を削減できます。

108）　pd_clt_waittime_over_dump_level = all | shm_fesonly

このオペランドは，HiRDB/パラレルサーバに関するオペランドで，トラブルシュート情報の出力量を削減するためのオペランドです。

pd_client_waittime_over_abort オペランドに Y（省略値）を指定している場合，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）を超えると，次に示すトラブルシュート情報が取得されます。

- 共用メモリダンプ（1 回だけ取得します）
- 簡易ダンプ
- ユニット内の詳細情報

このうち，共用メモリダンプは出力サイズが大きいため，共用メモリダンプの取得処理の負荷によって，マシンの沈み込みやスローダウンによる系切り替えが発生する可能性があります。

このオペランドに shm_fesonly を指定すると，共用メモリダンプを出力するユニットを制限できるため，共用メモリダンプの出力量を削減できます。

all：

共用メモリダンプを出力するユニットを制限しません（共用メモリダンプの出力量を削減しません）。

shm_fesonly：

共用メモリダンプを出力するユニットを制限します（共用メモリダンプの出力量を削減します）。フロントエンドサーバを定義したユニットだけが共用メモリダンプの出力対象になります。

このオペランドの指定値によるトラブルシュート情報の出力有無を次に示します。

トラブルシュート情報の種類		クライアントが接続したFES ユニット	トランザクションがアクセスしたユニット	左記以外のユニット
共用メモリダンプ	このオペランドに all を指定した場合	○	○	○
	このオペランドに shm_fesonly を指定した場合	○	×	×
簡易ダンプ		○	○	×
ユニット内の詳細情報		○	○	○

（凡例）

○：トラブルシュート情報を取得します。

×：トラブルシュート情報を取得しません。

注　簡易ダンプ及びユニット内の詳細情報は，このオペランドの指定に関係なく取得されます。

《適用基準》

通常，このオペランドを指定する必要はありません。

ユニット数が多く，1 台のサーバマシンに複数のユニットを配置している場合，共用メモリダンプの取得処理が 1 台のサーバマシン上でユニット数分，同時に実行されることがあります。

このような場合，共用メモリダンプの取得処理の負荷によってマシンの沈み込みや，スローダウンによる系切り替えが発生する可能性があるため，shm_fesonly を指定して，マシンの負荷を下げてください。

shm_fesonly を指定しても，必要最低限のトラブルシュート情報は取得されます。

109）　pd_dump_suppress_watch_time =トラブルシュート情報の出力抑止時間～<符号なし整数>((0～3600))《0》（単位：秒）

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

次に示す現象が発生した場合に出力されるトラブルシュート情報（%PDDIR%¥spool の下のファイル）の再出力を抑止する時間を秒単位で指定します。

- PDCWAITTIME タイムオーバが発生した場合
- pdcancel コマンドで実行中の UAP を中断させた場合
- プロセスが異常終了した場合

トラブルシュート情報を出力してから，このオペランドで指定した時間を経過するまでトラブルシュート情報を出力しません。例えば，このオペランドに 60 を指定した場合，トラブルシュート情報を出力した後，60 秒間はトラブルシュート情報を出力しません。

なお，このオペランドに 0 を指定した場合は，トラブルシュート情報の出力を抑止しません。

《利点》

HiRDB サーバのプロセス数が複数あると，タイムアウトなどでサーバプロセスの異常終了が連続して発生することがあります。サーバプロセスの異常終了が連続して発生すると，core 及び簡易ダンプなどのトラブルシュート情報を繰り返し出力するため，HiRDB 運用ディレクトリがあるディスクの容量を圧迫します。HiRDB 運用ディレクトリがあるディスクの容量が不足すると HiRDB が異常終了することがあるため，このオペランドを指定してディスク容量の圧迫を防いでください。

《注意事項》

- 内部矛盾による異常終了，外部からのシグナル受信処理などは，このオペランドの指定値に関係なくトラブルシュート情報を出力します。
- 異常終了時に出力される障害情報を次の表に示します。

表 2-1　異常終了時に出力される障害情報

異常終了の要因		障害情報	pd_dump_suppress_watch_time の値			
			0		0 以外	
			pd_cancel_dump の値			
			put	noput	put	noput
PDSWAITTIME オーバ		退避コアファイル	×	×	×	×
		デバック情報ファイル	×	×	×	×
		簡易ダンプファイル	×	×	×	×
		KFPA20009-W メッセージ	×	×	×	×
		SQL 実行時間警告情報ファイル	×	×	×	×
PDSWATCHTIME オーバ		退避コアファイル	×	×	×	×
		デバック情報ファイル	×	×	×	×
		簡易ダンプファイル	×	×	×	×
		KFPA20009-W メッセージ	×	×	×	×
		SQL 実行時間警告情報ファイル	×	×	×	×
PDCWAITTIME オーバ	pd_client_waittime_over_abort=Y	退避コアファイル	○	○	●	●
		デバック情報ファイル	○	○	●	●
		デバック情報ファイル 2	○	○	●	●
		簡易ダンプファイル	○	○	●	●
		KFPA20009-W メッセージ	○	○	●	●
		SQL 実行時間警告情報ファイル	○	○	●	●
		共用メモリダンプファイル	▲	▲	▲	▲
	pd_client_waittime	退避コアファイル	×	×	×	×

異常終了の要因		障害情報	pd_dump_suppress_watch_time の値			
			0		0 以外	
			pd_cancel_dump の値			
			put	noput	put	noput
	_over_abort=N	デバック情報ファイル	○	×	●	×
		デバック情報ファイル 2	×	×	×	×
		簡易ダンプファイル	○	×	●	×
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
		共用メモリダンプファイル	×	×	×	×
pdcancel コマンド		退避コアファイル	×	×	×	×
		デバック情報ファイル	○	×	●	×
		簡易ダンプファイル	○	×	●	×
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
内部 kill9※1		退避コアファイル	×	×	×	×
		デバック情報ファイル	○	×	●	×
		簡易ダンプファイル	○	×	●	×
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
内部 kill3※2		退避コアファイル	○	○	●	●
		デバック情報ファイル	○	○	●	●
		簡易ダンプファイル	○	○	●	●
		KFPA20009-W メッセージ	○	○	●	●
		SQL 実行時間警告情報ファイル	○	○	●	●
Abort※3		退避コアファイル	○	○	●	●
		デバック情報ファイル	○	○	●	●
		簡易ダンプファイル	○	○	●	●
		KFPA20009-W メッセージ	○	○	●	●

異常終了の要因	障害情報	pd_dump_suppress_watch_time の値			
		0		0 以外	
		pd_cancel_dump の値			
		put	noput	put	noput
	SQL 実行時間警告情報ファイル	○	○	●	●
	アボート情報ファイル	○	○	●	●
上記以外※4	退避コアファイル	△	△	△	△
	デバック情報ファイル	○	○	○	○
	簡易ダンプファイル	△	△	△	△
	KFPA20009-W メッセージ	△	△	△	△
	SQL 実行時間警告情報ファイル	△	△	△	△

（凡例）

○：障害情報を出力します。pd_dump_suppress_watch_time オペランドの指定は無効になります。

×：障害情報を出力しません。

●：障害情報を出力します。pd_dump_suppress_watch_time オペランドの指定が有効になります。

△：プロセスの終了形態によっては障害情報が出力されないことがあります。

▲：ユニットの開始後，1 回目のダンプ出力時に出力されます。pd_clt_waittime_over_dump_level オペランドに shm_fesonly を指定すると，共用メモリダンプを出力するユニットを制限できます。

注※ 1

OpenTP1 の UAP 異常終了処理など，内部的に SIGKILL を発行したときのことです。
PDCWAITTIME オーバ，pdcancel コマンドによる異常終了を含みません。

注※ 2

異常検知時など, 内部的に SIGQUIT を発行したときのことです。PDCWAITTIME オーバ, pdcancel コマンドによる異常終了を含みません。

注※ 3

HiRDB が矛盾を検知して abort() したときのことです。

注※ 4

SIGSEGV，SIGBUS，外部からのシグナル受信，exit，そのほかの予期しない障害などのことです。

110）　pd_debug_info_netstat = Y ｜ N

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

HiRDB プロセスの異常終了，又は HiRDB（ユニット）が異常終了したときに取得するトラブルシュート情報中に，ネットワーク情報を取得するかどうかを指定します。

Y：取得します。

N：取得しません。

ネットワーク情報に関するトラブルシュート情報は，%PDDIR%¥spool¥save 下の**サーバ名 n .deb** ファイルに出力されます。n は 1～3 の通番です。ファイル名の例を次に示します。

（例）

sds1.deb

fes2.deb

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- N を指定すると，HiRDB プロセス又は HiRDB（ユニット）の異常終了時のネットワーク情報取得によるネットワーク負荷を軽減できますが，異常終了の原因が判明できないことがあります。

111）　pd_pth_trace_max = 通信トレース格納最大数

～<符号なし整数>((1024～8388608))《1024》

トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。

ここでの指定値は，各ユニット制御情報定義の pd_pth_trace_max オペランドの省略時解釈値となります。

《指定値の目安》

通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。

《注意事項》

通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

112）　pd_spool_cleanup_interval = トラブルシュート情報の削除処理間隔

～<符号なし整数>((0～744))《24》（単位：時間）

このオペランドは，出力したトラブルシュート情報及び作業用一時ファイルを削除するためのオペランドです。トラブルシュート情報及び作業用一時ファイルを残しておくと HiRDB 運用ディレクトリがあるディスクの容量を圧迫する原因になります。HiRDB 運用ディレクトリがあるディスクの容量が不足すると HiRDB が異常終了することがあるため，HiRDB は次に示すファイルを定期的に削除します。

- トラブルシュート情報ファイル（%PDDIR%¥spool 下のファイル）
- 作業用一時ファイル（%PDDIR%¥tmp 下のファイル）
- pd_tmp_directory オペランドに指定したディレクトリ下のファイル

このオペランドでは，その削除間隔を時間単位で指定します。例えば，このオペランドに 48 を指定すると，48 時間ごとにこれらのファイルを削除します。通常は（このオペランドを省略すると），24 時間ごとにファイルを削除します。

なお，HiRDB の正常開始時点が時間のカウント開始時点になります。HiRDB を正常終了すると，時間のカウントも止まります。そして，次の正常開始時に時間のカウントが 0 に戻ります。

削除対象のファイルは，次で説明している pd_spool_cleanup_interval_level オペランドで指定します。

《オペランドの規則》

0 を指定すると，ファイルの削除処理を実行しません。

《指定値の目安》

このオペランドに 24，48，72，…を指定すると，決まった時間にファイルの削除処理が実行されます。削除処理はシステムに負荷が掛かる時間帯を外して実施するようにしてください。

《注意事項》

HiRDB が計画停止，強制終了，及び異常終了して HiRDB が停止している間は，時間がカウントされます。ただし，HiRDB の停止中に削除時間になった場合は，ファイルを削除しません。次の削除時間になるまでファイルを削除しません。ファイルを削除してから HiRDB を再開始する場合は，pdcspool コマンドを実行してください。

《備考》

pd_spool_cleanup_interval オペランドと pd_spool_cleanup オペランドの違いについて説明します。

- pd_spool_cleanup_interval オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup オペランドの指定を検討してください。

113）　pd_spool_cleanup_interval_level = 日数〔，削除種別〕

このオペランドは，出力したトラブルシュート情報及び作業用一時ファイルを削除するためのオペランドです。トラブルシュート情報ファイル及び作業用一時ファイルの定期削除の削除条件を指定します。

日数：～<符号なし整数>((1～24855))《7》（単位：日）

ここで指定した日よりも前に作成したトラブルシュート情報ファイルを削除します。例えば，3 と指定した場合，3 日以内（3 日×24 時間＝ 72 時間以内）に作成したトラブルシュート情報ファイルだけを残し，残りをすべて削除します。

削除種別：<文字列>《all》

削除するトラブルシュート情報ファイルの種類を指定します。

all：すべてのファイルを削除対象にします。

dump：HiRDB が内部取得するファイルだけを削除対象にします。

削除対象のファイルを次に示します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
デッドロック・タイムアウト情報	pdlckinf	○	×	排他制御でエラーが発生したときに出力します。
アクセスパス情報	pdsqldump	○	×	アクセスパス表示ユティリティ使用時に出力します。
退避コアファイルなど	save	○	○	プロセスが異常終了したときに出力します。
共用メモリダンプファイル	pdshmdump	○	○	プロセス又はユニットが異常終了したときに出力します。
簡易ダンプファイル	pdsysdump	○	○	なし。
	pdsdsdump	○	○	HiRDB/パラレルサーバには存在しません。
	pdfesdump pddicdump	○	○	HiRDB/シングルサーバには存在しません。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
	pdbesdump			
システムログファイルの状態情報ファイル	pdjnlinf	○	×	¥pdjnlinf¥errinf 下のファイルは削除しません。

（凡例）

○：削除対象のファイルです。

×：削除対象外のファイルです。

注　ディレクトリ名は%PDDIR%¥spool 下のディレクトリ名です。

作業用一時ファイルについては，削除種別の指定に関係なく次に示すファイル以外を削除します。（　）内は%PDDIR%¥tmp 下のディレクトリ名です。

- HiRDB が起動するプロセスのカレントワーキングディレクトリ（home）
- 共用メモリ情報ファイル（pdommenv）
- pdbufls コマンドの差分情報ファイル（ファイル名称が"CMb"で始まるファイル）

《前提条件》

pd_spool_cleanup_interval オペランドに 0 以外を指定している必要があります。

《指定値の目安》

コマンド（ユティリティを含む）の実行時間より多い日数を指定してください。例えば，バックアップを取得する pdcopy コマンドの実行に 24 時間（1 日）必要な場合は，日数に 2 以上の値を指定してください。コマンドの実行時間より多い日数を指定しないと，コマンドが使用中の作業用一時ファイルも削除されるため，コマンドが正常に動作しないことがあります。

《オペランドの規則》

日数を指定しないで削除種別だけの指定はできません。

《注意事項》

pd_tmp_directory オペランドを指定しないで環境変数 TMP だけを指定している場合，コマンド又はユティリティが使用する作業用一時ファイルの出力先は環境変数 TMP に指定したディレクトリになります。環境変数 TMP に指定したディレクトリに出力された作業用一時ファイルは定期削除の対象外になるため，エクスプローラなどで削除してください。

《備考》

pd_spool_cleanup_interval_level オペランドと pd_spool_cleanup_level オペランドの違いについて説明します。

- pd_spool_cleanup_interval_level オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup_level オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval_level オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup_level オペランドの指定を検討してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの削除種別の省略値は dump になります。

114）　pd_spool_cleanup = normal | force | no

このオペランドは，出力したトラブルシュート情報を削除するためのオペランドです。

HiRDB が出力したトラブルシュート情報ファイル（%PDDIR%¥spool 下のファイル）を HiRDB 開始時に削除するかどうかを指定します。削除対象のファイルは，次で説明している pd_spool_cleanup_level オペランドで指定します。

normal：HiRDB の正常開始及び計画停止後の再開始時に削除します。

force：HiRDB の開始モードに関係なく HiRDB の開始時に削除します。

no：削除しません。

《指定値の目安》

トラブルシュート情報ファイルがディスク容量を圧迫する場合に normal 又は force を指定してください。

《備考》

pd_spool_cleanup_interval オペランドと pd_spool_cleanup オペランドの違いについて説明します。

- pd_spool_cleanup_interval オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup オペランドの指定を検討してください。

115）　pd_spool_cleanup_level = 日数〔，削除種別〕

このオペランドは，出力したトラブルシュート情報を削除するためのオペランドです。HiRDB 開始時のトラブルシュート情報ファイルの削除条件を指定します。

日数：～<符号なし整数>((0～24855))《7》（単位：日）

ここで指定した日よりも前に作成したトラブルシュート情報ファイルを削除します。例えば，3 と指定した場合，3 日以内（3 日×24 時間＝ 72 時間以内）に作成したトラブルシュート情報ファイルだけを残し，残りをすべて削除します。

0 を指定した場合はすべてのファイルを削除します。

削除種別：<文字列>《all》

削除するトラブルシュート情報ファイルの種類を指定します。

all：すべてのファイルを削除対象にします。

dump：HiRDB が内部取得するファイルだけを削除対象にします。

削除対象のファイルを次に示します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
デッドロック・タイムアウト情報	pdlckinf	○	×	排他制御でエラーが発生したときに出力します。
アクセスパス情報	pdsqldump	○	×	アクセスパス表示ユティリティ使用時に出力します。
退避コアファイルなど	save	○	○	プロセスが異常終了したときに出力します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
共用メモリダンプファイル	pdshmdump	○	○	プロセス又はユニットが異常終了したときに出力します。
簡易ダンプファイル	pdsysdump	○	○	なし。
	pdsdsdump	○	○	HiRDB/パラレルサーバには存在しません。
	pdfesdump pddicdump pdbesdump	○	○	HiRDB/シングルサーバには存在しません。
システムログファイルの状態情報ファイル	pdjnlinf	○	×	¥pdjnlinf¥errinf 下のファイルは削除しません。

（凡例）

○：削除対象のファイルです。

×：削除対象外のファイルです。

注　ディレクトリ名は%PDDIR%¥spool 下のディレクトリ名です。

《前提条件》

pd_spool_cleanup オペランドで normal 又は force（省略値）を指定している必要があります。

《オペランドの規則》

日数を指定しないで削除種別だけを指定できません。

《備考》

pd_spool_cleanup_interval_level オペランドと pd_spool_cleanup_level オペランドの違いについて説明します。

- pd_spool_cleanup_interval_level オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup_level オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval_level オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup_level オペランドの指定を検討してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの削除種別の省略値は dump になります。

116）　pd_module_trace_max = モジュールトレース格納最大数

～<符号なし整数>((126～16383))《126》

HiRDB のプロセスは，実行した関数やマクロの履歴をプロセス固有メモリ中に記録しています。この履歴をモジュールトレースといい，このオペランドではモジュールトレースのレコード数を指定します。この履歴の内容は core ファイルに取り込まれ，プロセス異常終了時に出力されます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《注意事項》

各プロセスでは，以下のサイズのプロセス固有メモリが確保されます。

32 ビットモードの場合：64 ＋ 48×pd_module_trace_max オペランドの値（バイト）

64 ビットモードの場合：64 ＋ 64×pd_module_trace_max オペランドの値（バイト）

117）　pd_module_trace_timer_level = 0 | 10 | 20

モジュールトレースに出力する時刻の取得方法を指定します。このオペランドの指定値の意味を次に示します。

指定値	時刻取得方法
0	すべてのモジュールトレース出力箇所で，秒単位で取得します。
10	すべてのモジュールトレース出力箇所のうち，入出力処理前後など性能調査で問題となりやすい箇所だけマイクロ秒単位で取得し，そのほかの箇所では秒単位で取得します。
20	すべてのモジュールトレース出力箇所で，マイクロ秒単位で取得します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《注意事項》

このオペランドに 0 以外の値を指定すると，時刻をマイクロ秒で取得する関数を発行するため，性能が低下することがあります。

2.2.23　RD エリアに関するオペランド

118）　pd_max_rdarea_no = RD エリアの最大数

～<符号なし整数>((5～8388592))《200》

RD エリアの最大数を指定します。RD エリアの合計数がこのオペランドの値を超えている場合，HiRDB を正常開始できません。なお，ここでいう RD エリアには，マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアも含まれます。

《指定値の目安》

- RD エリアの合計数以上の値を指定してください（ある程度余裕をもたせてください）。RD エリアの合計数は pddbls コマンドで確認できます。なお，RD エリアを追加する予定がある場合は，その数を加算した値を指定してください。
- HiRDB/パラレルサーバの場合，このオペランドの値はバックエンドサーバごとに適用されます。例えば，このオペランドに 100 を指定した場合，各バックエンドサーバに最大 100 個の RD エリアを作成できます。したがって，RD エリア数が最も多いバックエンドサーバを目安にしてこのオペランドを指定してください。
- 共用 RD エリアを使用している場合，参照専用バックエンドサーバで使用する共用 RD エリアの数も加算してください。

《注意事項》

- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

119）　pd_max_file_no = RD エリアを構成する HiRDB ファイルの最大数

～<符号なし整数>((5～134217728))《400》

RD エリアを構成する HiRDB ファイルの最大数を指定します。HiRDB ファイルの合計数がこのオペランドの値を超えている場合，HiRDB を正常開始できません。なお，ここでいう RD エリアを構成する HiRDB ファイルには，マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアの HiRDB ファイルも含まれます。

《指定値の目安》

- RD エリアを構成する HiRDB ファイルの合計数以上の値を指定してください（ある程度余裕をもたせてください）。各 HiRDB ファイルシステム領域内の HiRDB ファイルの数は pdfstatfs コマンドで確認できます。なお，HiRDB ファイルを追加する予定がある場合は，その数を加算した値を指定してください。RD エリアの追加，RD エリアの再初期化，及び RD エリアの拡張時に HiRDB ファイルを追加します。
- HiRDB/パラレルサーバの場合，このオペランドの値はバックエンドサーバごとに適用されます。例えば，このオペランドに 100 を指定した場合，各バックエンドサーバに最大 100 個の HiRDB ファイルを作成できます。したがって, HiRDB ファイル数が最も多いバックエンドサーバを目安にしてこのオペランドを指定してください。
- 共用 RD エリアを使用している場合, 参照専用バックエンドサーバで使用する共用 RD エリアを構成する HiRDB ファイルの数も加算してください。

《注意事項》

- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

120）　pd_rdarea_warning_point = セグメント使用比率 1〔，セグメント使用比率 2〔，セグメント使用比率 3〕〕

～<符号なし整数>((0～100))（単位：%）

RD エリアのセグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I），又は増分する HiRDB ファイルの領域使用率通知メッセージ（KFPH22037-W）の出力契機を指定します。

RD エリアの自動増分を適用しない場合

セグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I）が出力されます。

このオペランドの指定の有無，及び対象となる RD エリア種別ごとのセグメント使用率通知メッセージの内容を次に示します。

対象となる RD エリア	pd_rdarea_warning_point 省略時	pd_rdarea_warning_point 指定時
• マスタディレクトリ用 RD エリア • データディレクトリ用 RD エリア	メッセージを出力しません。	対象となる RD エリアの全体のセグメントのうち，指定された使用率のセグメントの使用を開始しました。
• データディクショナリ用 RD エリア • ユーザ用 RD エリア	対象となる RD エリアの最終ファイルのセグメントのうち，相対位置で 80，90，100%のどれかのセグメントの使用を開始しました。	対象となる RD エリアの全体のセグメントのうち，指定された使用率のセグメントの使用を開始しました。
• LOB 用 RD エリア	対象となる RD エリアの最終ファイルのセグメントのうち，相対位置で 80，90，100%のどれかのセグメントの使用を開始しました。	対象となる RD エリアの全体のセグメントのうち，指定された使用率のセグメントの使用を開始しました。

RD エリアの自動増分を適用する場合

自動増分する HiRDB ファイルの領域使用率通知メッセージ（KFPH22037-W）が出力されます。

RD エリアの自動増分を適用する場合，このオペランドの指定の有無と出力するメッセージの内容を次に示します。

オペランドの指定値	pd_rdarea_warning_point 省略時	pd_rdarea_warning_point 指定時
メッセージの出力条件	増分した HiRDB ファイルの領域は，使用率 80，90，100%のどれかの比率以上に領域を割り当てました。	増分した HiRDB ファイルの領域は，指定された使用率以上に領域を割り当てました。

自動増分する HiRDB ファイルの領域使用率は，増分できる最大容量に対する使用率です。次の計算式で求めます。

自動増分する HiRDB ファイルの領域使用率（単位：%）＝ A÷B×100

- pdfmkfs -e コマンド指定の HiRDB ファイルシステム領域，又は pdfmkfs -a 指定のダイレクトディスクアクセスをする HiRDB ファイルシステム領域の場合

  A：自動増分する HiRDB ファイルの容量

  B：自動増分できる最大容量＝ min｛(A ＋ C)，64GB｝

  C：自動増分する HiRDB ファイルがある HiRDB ファイルシステム領域の空き容量

- pdfmkfs -a コマンド指定の OS のファイルシステム上の HiRDB ファイルシステム領域の場合

  A：自動増分する HiRDB ファイルの容量

  B：自動増分できる最大容量＝ min｛(A ＋ C)，64GB｝

  C：自動増分する HiRDB ファイルがある HiRDB ファイルシステム領域の空き容量＋自動増分する HiRDB ファイルがある HiRDB ファイルシステム領域が存在するディスクの空き容量

また，自動増分した HiRDB ファイルの使用エクステントが，20，22，24 以上のどれかになった場合，自動増分する HiRDB ファイルの使用エクステント数の通知メッセージ（KFPH22038-W）が表示されます。

このオペランドの指定例を次に示します。

メッセージの出力条件	オペランドの指定値
対象となる RD エリアの RD エリア全体のセグメントのうち，80，90，100%分の使用を開始した，それぞれの時点でメッセージを出力する場合	pd_rdarea_warning_point = 80,90,100
対象となる RD エリアの RD エリア全体のセグメントのうち，50，90%分の使用を開始した，それぞれの時点でメッセージを出力する場合	pd_rdarea_warning_point = 50,90
セグメント使用率通知メッセージを出力しない場合	pd_rdarea_warning_point = 0

《オペランドの規則》

- この指定値は，三つまで指定できます。
- 同じ値を複数指定した場合には，その値に対して 1 回だけメッセージが出力されます。
- セグメント使用比率 1～3 に 0 だけを指定した場合には，セグメント使用率通知メッセージが出力されません。
- 0 と 0 以外の数値が混在している場合には，指定された 0 以外の数値をメッセージの出力契機とします。

《注意事項》

このオペランドの指定は，データベース初期設定ユティリティ（pdinit コマンド），データベース回復ユティリティ（pdrstr コマンド）に対しては無効になります。このオペランドの指定の有無に関係なく，未指定時のメッセージの内容になります。

121）　pd_rdarea_warning_point_msgout = Y｜N

RD エリアのセグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I）を出力するかどうかを指定します。

Y：pd_rdarea_warning_point の指定値に従って，RD エリアのセグメント使用率通知メッセージを出力します。

N：RD エリアのセグメント使用率通知メッセージを出力しません。

《指定値の目安》

RD エリアのセグメント使用率通知メッセージの指定値の目安については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「空き領域の再利用機能」の「環境設定」を参照してください。

《注意事項》

次の場合には，このオペランドにNを指定してもセグメント使用率通知メッセージを出力します。

- データベース作成ユティリティ（pdload），又はデータベース再編成ユティリティ（pdrorg）でデータを格納する場合
- 空き領域の再利用機能を適用していない表にデータを格納する場合
- インデクスにデータを格納する場合
- 空き領域の再利用機能を適用している表の構成列に可変長の列を含み，データ長を長くする更新を頻繁に行う場合
- 表の定義数が pd_assurance_table_no オペランドの指定値を超えたため，空き領域の再利用機能が使用できなくなっている場合

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_rdarea_warning_point
- pd_assurance_table_no

122）　pd_rdarea_expand_format = Y｜N

以下の HiRDB ファイルシステム領域を通常ファイルで作成し，その領域を使用している RD エリアが RD エリアの自動増分機能を使用している場合，増分領域がスパースファイルになる可能性があります。

(i)-i オプション指定なしの pdfmkfs コマンドで作成した HiRDB ファイルシステム領域

(ii)-i オプションと-a オプションを指定した pdfmkfs コマンドで作成した HiRDB ファイルシステム領域

RD エリアを構成する HiRDB ファイルシステム領域がスパースファイルになると，グローバルバッファからのページ書き込み契機（デファードライト処理）で書き込みエラーが発生し，障害閉塞することがあります。

本オペランドの指定値に Y を指定すると，RD エリアの自動増分が発生したときに増分領域を初期化し，HiRDB ファイルシステム領域がスパースファイルになるのを防ぐことができます。

なお，初期化する対象は，ファイルの種類が通常ファイルで，-a オプションを指定した pdfmkfs コマンドで作成した HiRDB ファイルシステム領域です。

Y：増分領域の初期化を行います

N：増分領域の初期化を行いません

《指定値の目安》

RD エリアの自動増分機能を使用するシステムでは，通常下記のリソース容量が満杯にならない運用（容量監視や定期的なデータベースの再編成の実施）を行ってください。この場合は本オペランドの指定値を N にすることを推奨します。

- RD エリア
- HiRDB ファイルシステム領域
- HiRDB ファイルシステム領域を配置しているディスク

容量監視ができないシステムの場合は本オペランドの指定値を Y にしてください。詳細はマニュアル「HiRDB Version 8 運用ガイド」を参照してください。

《注意事項》

本オペランドの指定値を Y にする場合，以下の点に注意してください。

(a)自動増分が発生している RD エリアをアクセスする更新トランザクションの実行時間が延びるため，次の要因でトランザクションがキャンセルされることがあります。

- 排他待ち限界経過時間を超過
- HiRDB クライアントの最大待ち時間を超過
- ユティリティの最大実行時間を超過

《他パラメタとの関連》に示すパラメタについて，以下に示す RD エリアの構成情報と，HiRDB ファイルシステム領域を配置しているディスクの I/O 性能を考慮して，見積もりしてください※。

- ページサイズ
- セグメントサイズ
- 増分セグメント数

注※　増分領域の初期化にかかる処理時間(目安)[秒]

- 自動増分に掛かる処理時間の目安
  ページサイズ[byte]×(セグメントサイズ×増分セグメント数 ＋ α)
  ÷HiRDB ファイルシステム領域を配置しているディスクの I/O 性能 [byte/秒]
  α：増分時に作成するディレクトリページ数
- データディクショナリ用 RD エリア・ユーザ用 RD エリア・レジストリ用 RD エリアの場合
  ↑d÷b↑＋↑d÷f↑
  d：増分セグメント数
  b：↓(P - 20)÷(↑d÷32↑×8 ＋ 56)↓
  f：↓(125×P)÷(16×b)↓×b
  P：ページサイズ
  S：セグメントサイズ
- LOB 用 RD エリアの場合
  ↑S÷64000↑×96
  S：セグメントサイズ

(b)本オペランドを指定した場合でも，増分領域以外は初期化を行いません。

このため，pdfmkfs コマンドに-i オプションを指定していない HiRDB ファイルシステム領域の場合，スパースファイルになります。このことから，HiRDB ファイルシステム領域を作成する際は，-i オプションの指定を推奨します。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- PDCWAITTIME
- pd_lck_wait_timeout
- pd_utl_exec_time

123）pd_rdarea_extension_timing = use | nouse

RD エリアの自動増分機能を適用している場合，どのタイミング（契機）で RD エリアを自動増分するかを指定します。

use：RD エリア内の空きセグメント数が 1 回の自動増分で増分するセグメント数以下になったときに自動増分します。例えば，1 回の自動増分セグメント数を 50 セグメントとしている場合，空きセグメント数が 50 セグメント以下になったときに自動増分します。

nouse：RD エリア内に空きセグメントがなく，新しいセグメントを確保できないときに自動増分します。

《指定値の目安》

それぞれの指定値のメリットとデメリットを次の表に示します。

項目	オペランドの指定値	
	use	nouse
メリット	自動増分ができない場合※でも，データ追加によるページ不足エラーになるまで，空きセグメント（増分セグメント数分）を利用できます。この間に，RD エリアの状態を解析し，使用状況に応じて再編成や RD エリアの拡張などの対処ができるため，業務への影響を最小限にできます。	空きセグメントがなくなってから，自動増分を行うため，HiRDB ファイルシステム領域内に複数 HiRDB ファイルを作成する場合，格納効率が良くなります。
デメリット	HiRDB ファイルシステム領域内に複数 HiRDB ファイルを作成する場合，格納効率が悪くなることがあります。	自動増分ができない場合※，空きセグメントがない状態のため，データ追加によるページ不足エラーで業務が停止するおそれがあります。

注※　次のような場合，自動増分できません。

- HiRDB ファイルシステム領域内で HiRDB ファイルを拡張する方式（pdfmkfs -e コマンド）の場合，HiRDB ファイルシステム領域に空きがないとき。
- HiRDB ファイルシステム領域を自動的に拡張する方式（pdfmkfs -a コマンド）の場合，ディスクに空きがない，又は HiRDB ファイルの最大サイズ（64GB）に達しているとき。
- RD エリアがバックアップ閉塞状態のため，自動増分が抑止されるとき。

このオペランドの指定値は，業務への影響を最小限にできる use をお勧めします。

124）pd_rdarea_open_attribute_use = Y | N

RD エリアのオープン契機に DEFER 属性又は SCHEDULE 属性を使用するかどうかを指定します。

Y：DEFER 属性又は SCHEDULE 属性を使用します。

N：DEFER 属性又は SCHEDULE 属性を使用しません。

このオペランドを省略するか，又は N を指定した場合，RD エリアのオープン契機は常に INITIAL 属性になります。したがって，次に示すオペランド又はユティリティで RD エリアのオープン契機を DEFER 属性又は SCHEDULE 属性にしても無効になります。

- pd_rdarea_open_attribute オペランド
- データベース初期設定ユティリティ

- データベース構成変更ユティリティ

《注意事項》

- Y を指定すると，HiRDB が必要とする共用メモリ量が増加するため，共用メモリが不足して HiRDB を開始できないことがあります。
- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用する場合はこのオペランドに Y が仮定されます。このため，サーバが使用する共用メモリが増加するので共用メモリを見積もり直してください。サーバが使用する共用メモリの見積もり式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

125）pd_rdarea_open_attribute = INITIAL | DEFER | SCHEDULE

RD エリアのオープン契機の標準値を指定します。

データベース初期設定ユティリティ又はデータベース構成変更ユティリティで open attribute の指定がない RD エリアは，このオペランドに指定した属性が仮定されます。ただし，システム用 RD エリアは，常に INITIAL が仮定されます。

《指定値の目安》

- 同じ HiRDB ファイルシステム領域の RD エリアには，同じ属性を指定してください。異なる属性を指定すると，期待どおりの効果が得られない場合があります。
- 各属性の RD エリアのオープン／クローズ契機，及び長所／短所を次に示します。

属性	初期状態	オープン契機	クローズ契機	長所	短所
INITIAL	オープン状態	• HiRDB 開始時 • pdopen コマンド実行時	pdclose コマンド実行時	初回 SQL から高速実行	システムの開始に時間が掛かる
DEFER	クローズ状態	• RD エリアの初回アクセス時 • pdopen コマンド実行時	pdclose コマンド実行時	• システムの開始が高速 • 初回アクセス以後は通常 SQL も高速	各 RD エリアの初回アクセスに時間が掛かる
SCHEDULE	クローズ状態	• トランザクション内での RD エリア初回アクセス時 • pdopen コマンド実行時	• トランザクション終了時 • pdclose コマンド実行時	• システムの開始が高速 • ファイルオープンの集中を回避	• トランザクションごとに RD エリアの初回アクセスが高負荷となる

- 各属性に適する運用形態を次に示します。

属性	適する運用形態
INITIAL	システム開始時に HiRDB ファイルシステム領域をオープンして，RD エリア情報をメモリ上に常駐します。RD エリアの初回アクセス時にも，そのプロセス上でオープンします。ただし，この場合には RD エリア情報の再作成をしないため，初回 SQL から高速な運用ができます。 システム開始時の RD エリア初期状態はオープン状態であり，以降は障害閉塞への遷移を除いて，運用コマンドの入力がないかぎり RD エリアの状態は遷移しません。 **特殊な運用形態を用いない場合は，この属性を推奨します。** この属性のとき，クローズ状態の RD エリアに対してはアクセスできません。

属性	適する運用形態
DEFER	システム開始時に HiRDB ファイルシステム領域のオープンをしないで，RD エリアに対する初回アクセス時にオープンし，RD エリア情報をメモリ上に常駐します。2 回目以降のアクセスでは，HiRDB ファイルシステム領域のオープン以降の処理をしないため，高速な運用ができます。 システム開始時の RD エリア初期状態はクローズ状態であり，各 RD エリアに対する初回アクセス時に該当する RD エリアをオープン状態にします。以降は障害閉塞への遷移を除いて，運用コマンドの入力がないかぎり RD エリアの状態は遷移しません。 **多数の HiRDB ファイルシステム領域に対するオープンが重なるケースを回避したい場合や，HiRDB の開始時間を短縮したい場合に，この属性を指定します。** HiRDB を再開始する場合は，回復処理時に回復対象 RD エリアをオープンします。 この属性を指定した場合，クローズ状態の RD エリアに対してもアクセスできます。
SCHEDULE	システム開始時に HiRDB ファイルシステム領域のオープンをしないで，システム開始後，各トランザクション内での RD エリアに対する初回アクセス時にオープンし，RD エリア情報をメモリ上に常駐します。トランザクションの終了時に，そのトランザクション内でオープンした HiRDB ファイルシステム領域をクローズします。以降もトランザクションが変わると RD エリアに対する初回アクセス時にオープン以降の処理をするため，トランザクションに掛かる負荷は増加します。 システム開始時の RD エリアの初期状態はクローズ状態であり，アクセスのあった RD エリアのトランザクション処理中だけオープン状態とします。トランザクション終了時に，トランザクション内でオープン状態としたすべての RD エリアをクローズ状態にします。 pdopen コマンドを入力すると，次回閉塞クローズ状態になるまでの間オープン状態を継続できます。そのほかの運用コマンドを用いて，RD エリアのステータスを任意に遷移させることもできます。障害を検知したときは障害閉塞となります。 **多数の HiRDB ファイルシステム領域のオープンが重なることを回避したい場合や，システムの開始時間を短縮したい場合に，この属性を指定します。** HiRDB を再開始する場合は，回復処理時に回復対象 RD エリアをオープンして，回復処理の終了後にクローズします。 この属性を指定した場合，クローズ状態の RD エリアに対してもアクセスできます。

《注意事項》

- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用する場合の注事項を次に示します。

使用する機能	注意事項
高速系切り替え機能	高速系切り替え機能の対象になる待機系ユニットは，待機状態のときに RD エリアをオープンしていません。また，系の切り替え時間を最小限に抑えるため，系切り替えの発生時に全面回復で必要な RD エリアだけをオープンして，そのほかの RD エリアはオープンしません。したがって，待機系の RD エリアのオープン契機は INITIAL 属性になりません。INITIAL 属性の RD エリアは DEFER 属性になります。
1:1 スタンバイレス型系切り替え機能	系の切り替え時間を最小限に抑えるため，系切り替えの発生時に全面回復で必要な RD エリアだけをオープンして，そのほかの RD エリアはオープンしません。したがって，正規 BES 又は代替部の RD エリアのオープン契機は次に示すようになります。 • 系切り替えが発生した場合，代替部の RD エリアのオープン契機は SCHEDULE 属性になります。 • 障害が回復して正規 BES に系を切り戻した場合，正規 BES 下の INITIAL 又は DEFER 属性の RD エリアのオープン契機は DEFER 属性になります。SCHEDULE 属性の RD エリアは SCHEDULE 属性のままです。
影響分散スタンバイレス型系切り替え機能	系の切り替え時間を最小限に抑えるため，系切り替えによる再開始時には全面回復で必要な RD エリアだけをオープンして，そのほかの RD エリアはオープンしません。したがって，

使用する機能	注意事項
	系切り替えが発生した場合，ゲスト BES 下の INITIAL 属性の RD エリアのオープン契機は DEFER 属性になります。

- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能と RD エリアのオープン契機の関係を次に示します。

<table>
<tr><th colspan="3" rowspan="4">条件</th><th colspan="4">pd_rdarea_open_attribute_use の指定値</th></tr>
<tr><th rowspan="3">N</th><th colspan="3">Y</th></tr>
<tr><th colspan="3">RD エリアのオープン契機</th></tr>
<tr><th>INITIAL</th><th>DEFER</th><th>SCHEDULE</th></tr>
<tr><td colspan="3">系切り替え機能を使用しない※1</td><td rowspan="3">INITIAL</td><td rowspan="3">INITIAL</td><td rowspan="3">DEFER</td><td rowspan="3">SCHEDULE</td></tr>
<tr><td rowspan="2">スタンバイ型系切り替え機能※1</td><td colspan="2">高速系切り替え機能を使用しない</td></tr>
<tr><td rowspan="2">高速系切り替え機能</td><td>実行系</td></tr>
<tr><td>待機系</td><td>DEFER※2</td><td>DEFER</td><td>DEFER</td><td>SCHEDULE</td></tr>
<tr><td rowspan="4">1：1 スタンバイレス型系切り替え機能※1</td><td rowspan="2">受け入れ部（代替 BES ユニットの代替部以外）</td><td>実行系</td><td>INITIAL
※2※3</td><td>INITIAL
※3</td><td>DEFER</td><td>SCHEDULE</td></tr>
<tr><td>待機系</td><td>DEFER※2</td><td>DEFER</td><td>DEFER</td><td>SCHEDULE</td></tr>
<tr><td rowspan="2">代替部</td><td>実行系</td><td rowspan="2">SCHEDULE
※2</td><td rowspan="2">SCHEDULE</td><td rowspan="2">SCHEDULE</td><td rowspan="2">SCHEDULE</td></tr>
<tr><td>待機系</td></tr>
<tr><td rowspan="2">影響分散スタンバイレス型系切り替え機能※1</td><td colspan="2">正常開始又は再開始</td><td>INITIAL</td><td>INITIAL</td><td>DEFER</td><td>SCHEDULE</td></tr>
<tr><td colspan="2">系切り替えによる再開始</td><td>DEFER※2</td><td>DEFER</td><td>DEFER</td><td>SCHEDULE</td></tr>
</table>

注※1　システム用 RD エリアのオープン契機は INITIAL になります。

注※2　pd_rdarea_open_attribute_use オペランドに Y が指定されているとみなします。

注※3　再開始時は DEFER になります。

126）pd_shared_rdarea_use = Y | N

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

共用 RD エリアを使用するかどうかを指定します。

Y：共用 RD エリアを使用します。

N：共用 RD エリアを使用しません。

《注意事項》

- 共用 RD エリアを定義しているのに，このオペランドを省略した場合（又はこのオペランドに N を指定した場合），HiRDB を正常開始できません。
- このオペランドに Y を指定すると，pd_dbsync_point オペランドに commit を指定したときと同じ共用メモリブロックを確保します。

127）pd_db_access_error_action = dbhold | unitdown

RD エリアへのアクセス時にファイルアクセスエラーが発生した場合の動作を指定します。

RD エリアにアクセスする際，アクセス対象の RD エリアを構成する HiRDB ファイルシステム領域にあるファイルを HiRDB 以外のプロセスが占有排他でオープンしていると，ファイルアクセスエラーとなります※1（エラーコード-1540※2）。

注※1　Windows の COPY コマンドで HiRDB ファイルシステム領域をコピーしているときに RD エリアにアクセスすると，ファイルアクセスエラーが発生ことがあります。

注※2　HiRDB ファイルシステム領域のアクセス権限の設定を誤った場合にも返却されます。

dbhold：

ファイルアクセスエラーが発生した場合，アクセス対象の RD エリアを閉塞します。

unitdown：

ファイルアクセスエラーが発生した場合，KFPH23040-I メッセージを出力して，ファイルアクセスエラー検知時ユニットダウン機能を適用します。ファイルアクセスエラー検知時ユニットダウン機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《適用基準》

通常，このオペランドを指定する必要はありません。

HiRDB の稼働中に HiRDB 以外のプロセスが RD エリアを構成する HiRDB ファイルシステム領域にアクセスする運用を行っていて，そのプロセスが HiRDB ファイルシステム領域を占有排他する可能性がある場合に，このオペランドの指定を検討してください。

《指定値の目安》

ファイルアクセスエラー検知時ユニットダウン機能を使用して RD エリアの閉塞とそれに伴う回復作業を回避したい場合，unitdown を指定してください。

《注意事項》

- unitdown を指定したときにファイルアクセスエラーが発生すると，次の場合に処理対象の RD エリアが障害閉塞することがあります。

  ・更新前ログ取得モード又はログレスモードで UAP やユティリティを実行している場合

  ・CREATE TABLE の RECOVERY オペランドで NO を指定してログレスモードにしたユーザ LOB 用 RD エリアに対して，UAP 又はユティリティを実行している場合

- unitdown を指定する場合は，データベースの更新ログを取得する運用をしてください。データベースの更新ログを取得しない運用をする場合は，UAP 又はユティリティの実行前にバックアップを取得して，RD エリアの障害閉塞を回復できるようにしてください。バックアップの取得については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

- マスタディレクトリ用 RD エリアへのアクセス時にファイルアクセスエラーが発生した場合，このオペランドの指定値に関係なく常にユニットダウンします。ただし，ユニットダウンする際には KFPH23040-I メッセージが出力されます。

- データベース回復ユティリティ（pdrstr）での回復処理中は，unitdown を指定していても HiRDB を異常終了しません。この場合は，データベース回復ユティリティ（pdrstr）を再度実行して回復してください。

128）pd_db_hold_action = dbhold | unitdown

RD エリアへのアクセス時に物理エラーが発生した場合（KFPH00306-E メッセージの閉塞理由に i/o error occurred，又は open error occurred を出力して閉塞するエラーの場合）の動作を指定します。

マスタディレクトリ用 RD エリアの場合は物理エラー検知時ユニットダウン機能を使用していなくても，障害閉塞しないでユニットダウンします。物理エラー検知時ユニットダウン機能については，マニュアル「HiRDB Version 8 システム運用 ガイド」を参照してください。

dbhold：

物理エラーが発生した場合，アクセス対象の RD エリアを閉塞します。

unitdown：

物理エラーが発生した場合，KFPH23047-I メッセージを出力して，物理エラー検知時ユニットダウン機能を適用します。

《適用基準》

通常，このオペランドを指定する必要はありません。

RD エリアの数が少なく，RD エリアが一つでも障害閉塞したら業務全体が停止するようなシステムの場合に，このオペランドの指定を検討してください。unitdown を指定する場合は，pd_mode_conf オペランドに MANUAL2 を指定してください。

《注意事項》

- RD エリアに対するアクセスで物理エラーが発生し，KFPH00307-E メッセージを出力して RD エリアがコマンド閉塞した場合は，pd_db_hold_action オペランドに unitdown を指定していてもユニットダウンしません。
- unitdown を指定した場合に物理エラーが発生すると，次のケースでは処理対象の RD エリアが障害閉塞することがあります。

  ・更新前ログ取得モード又はログレスモードで UAP やユティリティを実行している

  ・CREATE TABLE の RECOVERY オペランドで NO を指定してログレスモードにしたユーザ LOB 用 RD エリアに対して，UAP 又はユティリティを実行している

  物理エラー検知時ユニットダウン機能を使用する場合は，できるだけこれらの運用は避けてください。もし，これらの運用が必要な場合は，RD エリアが閉塞しても最新の状態に回復できるよう，UAP 又はユティリティの実行前にバックアップを取得してください。バックアップの取得については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- データベース回復ユティリティ（pdrstr）での回復処理中は，unitdown を指定していてもユニットダウンしません。この場合は，データベース回復ユティリティ（pdrstr）を再度実行して回復してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_mode_conf オペランド
- pd_db_io_error_action オペランド
- pd_db_access_error_action オペランド

なお，pd_db_io_error_action オペランド，pd_db_access_error_action オペランド，及び pd_db_hold_action オペランドのうち，複数のオペランドで unitdown を指定した場合は，次の順序でオペランドの指定値が有効になります。

1. pd_db_io_error_action オペランド
2. pd_db_access_error_action オペランド
3. pd_db_hold_action オペランド

そのため，RD エリアの入出力エラー，ファイルアクセスエラー，及び物理エラーのうち複数のエラーが同時に起こった場合，どのエラー要因によってユニットダウンしたのかは，この優先順位を参考に判断してください。また，出力されたメッセージを参照してください。

2.2.24　グローバルバッファに関するオペランド

129）　pd_dbbuff_lru_option = SEPARATE | MIX

グローバルバッファの LRU 管理方式を指定します。ただし，次のどちらかに該当する場合は無条件に MIX を仮定します。

- pd_dbsync_point に commit を指定している場合
- pd_dbbuff_binary_data_lru に N を指定している場合

SEPARATE：

参照バッファ，及び更新バッファをそれぞれ独立した LRU で管理します。グローバルバッファの不足時には，グローバルバッファ内のアクセスした参照バッファの中で，最も古いバッファがメモリから追い出されます。オンライン業務のように 1 トランザクション当たりの参照，更新件数が比較的少ない場合に指定します。

MIX：

グローバルバッファを一括した LRU で管理します。グローバルバッファの不足時には，グローバルバッファ内のアクセスしたバッファで，最も古いバッファがメモリから追い出されます。オンライン業務とバッチ業務など，大量検索，大量更新が共存する場合に指定します。

グローバルバッファの LRU 管理方式については，マニュアル「HiRDB Version 8 解説」を参照してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は SEPARATE になります。

130）　pd_dbbuff_binary_data_lru = Y | N

UAP がアクセスするバイナリデータに対してグローバルバッファの LRU 管理を行うかどうかを指定します。UAP がアクセスするバイナリデータの LRU 管理抑止設定については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

Y：

バイナリデータが格納された分岐行ページに LRU 管理を適用して，グローバルバッファ上にキャッシュします。

N：

バイナリデータが格納された分岐行ページに LRU 管理を適用しないで，アクセス頻度に関係なく最も古くアクセスしたページとしてグローバルバッファ上にキャッシュします。

ただし，基本行ページは LRU 管理を適用してグローバルバッファ上にキャッシュします。

グローバルバッファを使用してサイズが大きいバイナリデータに大量にアクセスする UAP を実行する場合，バイナリデータをグローバルバッファにキャッシュするために，グローバルバッファにキャッシュされた直近の内容がメモリから追い出され，一時的に性能を低下させてしまうことがあります。その場合，サイズが大きいバイナリデータのアクセス頻度が低いのであれば，LRU 管理を抑止することによって性能低下を回避できます。

それぞれの指定値による LRU 管理，メリット及びデメリットを次の表に示します。

項目	オペランドの指定値	
	Y	N
LRU 管理	すべてのグローバルバッファを LRU で管理し，グローバルバッファが不足したときは，グローバル	バイナリデータが格納された分岐行ページのグローバルバッファの LRU 管理を抑止し，最も古くアクセスしたページとしてキャッシュします。グ

項目	オペランドの指定値	
	Y	N
	バッファプール内で最も古くアクセスしたグローバルバッファをメモリから追い出します。	ローバルバッファが不足したときは，バイナリデータが格納された分岐行ページのグローバルバッファをメモリから追い出します。
メリット	アクセスデータの種類に関係なく，グローバルバッファを均等に使用するため，UAP の性能が偏りません。	サイズが大きいバイナリデータに連続してアクセスしても，グローバルバッファからバイナリデータを優先的に追い出すため，バイナリデータ以外の検索性能を維持できます。
デメリット	サイズが大きいバイナリデータに連続してアクセスするとグローバルバッファから直近にアクセスしたデータが追い出されるため，バイナリデータ以外の検索性能が低下します。	• サイズが大きいバイナリデータにアクセスする UAP は，バッファヒット率が低くなります。それによって，入出力回数が増加し，レスポンス性能が低下する可能性があります。 • バイナリデータの更新時に，バイナリデータの分岐行の追い出しが多発することでシステムログ量が増加します。

《指定値の目安》

- 通常はこのオペランドを指定しないでください。
- バイナリデータにアクセスしないで，蓄積するだけのシステムの場合は，N を指定してください。バイナリデータに頻繁にアクセスするシステムの場合，このオペランドに N を指定するとデメリットの影響が大きいので適用するときは十分検討してください。
- BINARY 型や BINARY 型の属性を含む抽象データ型，及び XML 型などのサイズの大きなバイナリデータを含む表が存在するシステムの場合，サイズの大きなバイナリデータにほとんどアクセスしないときは，このオペランドに N を指定してください。バイナリデータが格納されたページをアクセス頻度に関係なく優先的にグローバルバッファから追い出し対象とするため，バイナリデータ以外のデータ格納ページがグローバルバッファから追い出される頻度を少なくできます。

《クライアント環境定義との関連》

クライアント環境定義の PDDBBUFLRU に NO を指定した場合は，このオペランドの指定に関係なく，指定した UAP でアクセスするすべてのデータの LRU 管理を抑止します。

《ほかのオペランドとの関連》

pd_dbbuff_lru_option オペランドに SEPARATE を指定しても，このオペランドで N を指定すると，pd_dbbuff_lru_option オペランドに MIX を仮定します。

131）　pd_dbbuff_modify = Y | N

HiRDB の稼働中に pdbufmod コマンドでグローバルバッファを動的変更するかどうかを指定します。グローバルバッファの動的変更については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

Y：グローバルバッファを動的変更します。

N：グローバルバッファを動的変更しません。

《前提条件》

このオペランドに Y を指定する場合（グローバルバッファを動的変更する場合）は HiRDB Advanced High Availability が必要になります。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_no
- pd_max_add_dbbuff_shm_no

132）pd_dbbuff_lock_release_detect = pipe | interval | switch

グローバルバッファへのアクセス時，グローバルバッファの排他獲得処理が発生します。この排他獲得処理で処理待ちになったプロセスが，グローバルバッファの排他が解除されたことを検知する処理方式をこのオペランドで指定します。このオペランドは同一グローバルバッファに対するアクセス頻度が高いときの処理性能に影響します。このオペランドの各指定値の処理方式と特徴を次に示します。

指定値	HiRDB の処理方式と特徴	特徴	
		レスポンスタイム	CPU 使用率
pipe	排他解除プロセスから排他獲得待ちプロセスに，パイプファイルを使用して排他の解除を通知します。 レスポンスタイムがジョブ多重度の影響を受けません。そのため，interval 又は switch を指定したときに比べてレスポンスタイムが一定しています。	長い	低い
interval	排他が掛けられているかを排他獲得待ちプロセスが定期的にチェックします。 レスポンスタイムがジョブ多重度の影響を受けます。そのため，ジョブ多重度が大きくなれば，レスポンスタイムも長くなります。	短い	高い
switch	基本的な処理方式は interval と同じです。 interval を指定したときに比べて，CPU を効果的に利用するため，CPU 使用率が interval を指定したときに比べて低くなります。 ただし，interval を指定したときに比べてチューニングが難しくなります。	短い	中

注

レスポンスタイム及び CPU 使用率についてはあくまで一般的な傾向です。実行環境や pd_dbbuff_lock_interval 又は pd_dbbuff_lock_spn_count オペランドの値によって変わることがあります。

《指定値の目安》

- UAP の多重実行性能が単体性能に比べて著しく低い場合に switch を指定すると，性能が向上することがあります。
- interval 又は switch のどちらを指定した方がよい（性能が向上する）かは，OS の種類，マシン性能，UAP の処理内容，及び UAP の多重実行数などによって変わります。ただし，一般的には switch を指定した方が比較的安定した性能が得られます。

《ほかのオペランドとの関連》

このオペランドに interval 又は switch を指定した場合は，次のオペランドの指定を検討してください。

- pd_dbbuff_lock_spn_count
- pd_dbbuff_lock_interval

133）　pd_dbbuff_lock_spn_count = 排他獲得待ち処理中のスピン回数

～<符号なし整数>((0～2147483646))《100》

pd_dbbuff_lock_release_detect オペランドに interval 又は switch を指定した場合の排他獲得待ち処理中のスピン回数を指定します。

グローバルバッファ用の排他獲得処理の概要を次に示します。このオペランドには 1 の繰り返し回数を指定します。

1. グローバルバッファ用の排他が解除されている場合は排他を獲得します。獲得できた場合は処理を終了しますが，獲得できない場合はこのオペランドに指定した回数まで同処理を繰り返します。
2. 1.で獲得できなかった場合，pd_dbbuff_lock_interval オペランドに指定した時間だけスリープ（一定時間処理を待つこと）します。
3. 1.に戻ります。

《前提条件》

pd_dbbuff_lock_release_detect オペランドに interval 又は switch を指定している必要があります。

《指定値の目安》

マニュアル「HiRDB Version 8 システム運用ガイド」の「グローバルバッファの排他処理の排他競合待ち発生率」を参照してオペランドの値を決定してください。

《ほかのオペランドとの関連》

このオペランドを指定した場合は，pd_dbbuff_lock_interval オペランドの指定を検討してください。

134）　pd_dbbuff_lock_interval = 排他獲得待ち処理中のインターバル時間

～<符号なし整数>((0～2147483647))《1》（単位：ミリ秒）

pd_dbbuff_lock_release_detect オペランドに interval 又は switch を指定した場合の排他獲得待ち処理中のインターバル時間を指定します。

グローバルバッファ用の排他獲得処理の概要を次に示します。このオペランドには 2 のスリープ時間を指定します。

1. グローバルバッファ用の排他が解除されている場合は排他を獲得します。獲得できた場合は処理を終了しますが，獲得できない場合は pd_dbbuff_lock_spn_count オペランドに指定した回数まで同処理を繰り返します。
2. 1.で獲得できなかった場合，このオペランドに指定した時間だけスリープ（一定時間処理を待つこと）します。
3. 1.に戻ります。

《前提条件》

pd_dbbuff_lock_release_detect オペランドに interval 又は switch を指定している必要があります。

《指定値の目安》

マニュアル「HiRDB Version 8 システム運用ガイド」の「グローバルバッファの排他処理の排他競合待ち発生率」を参照してオペランドの値を決定してください。

135）　pd_dbbuff_wait_interval = グローバルバッファの占有状態の調査間隔

～<符号なし整数>((0～2147483647))（単位：ミリ秒）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

グローバルバッファの占有状態を調査する間隔を指定します。また，このオペランドを指定すると，グローバルバッファの占有状態を調査する方式が変わります。

このオペランドを指定していない場合

次に示す処理方式になります。

注※
HiRDBサーバ内部で使用するタイマの精度がプラットフォームに依存するため，実際には約10ミリ秒，約16ミリ秒，約55ミリ秒のいずれかとなります。

このオペランドを指定している場合

次に示す処理方式になります。

《指定値の目安》

次に示す条件をすべて満たす場合にこのオペランドを指定してください。性能が向上することがあります。なお，このオペランドを指定する場合は通常 1 を指定します。

- グローバルバッファの排他待ちが発生している（pdbufls コマンドの実行結果の WAITL で確認できます）
- CPU 使用率を高くしても性能を良くしたい

このオペランドに 1 を指定したため，CPU 使用率が高くなり過ぎた場合は，値を大きくしてください。また，このオペランドに 1 を指定しても CPU 使用率に余裕がある場合は，pd_dbbuff_wait_spn_count オペランドの値を大きくしてください。性能が向上することがあります。

136）　pd_dbbuff_wait_spn_count = グローバルバッファの占有状態調査のスピンループ回数上限値

～<符号なし整数>((0～2147483646))《0》

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

グローバルバッファの占有状態の調査処理中に発生する，インターバルループ内のスピンループ回数の上限値を指定します。詳細については，pd_dbbuff_wait_interval オペランドの説明を参照してください。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。pd_dbbuff_wait_interval オペランドに 1 を指定している場合にこのオペランドを指定してください。

137）　pd_dbbuff_rate_updpage = デファードライトトリガの要求比率

～<符号なし整数>((1～100))（単位：%）

デファードライトトリガのトリガ契機を更新バッファ比率で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。デファードライトの書き込み処理がシンクポイントダンプ取得間隔時間内で終了しない場合など，更新バッファ数を減らし，更新バッファヒット率を多少抑えてでも書き込み時間を短縮したいときにこのオペランドを指定します。指定する場合は，50%（HiRDB が設定する初期値）を目安にし，マニュアル「HiRDB Version 8 システム運用ガイド」の「デファードライト処理のチューニング」及び「デファードライト処理適用時のシンクポイント処理時間のチューニング」を参照してオペランドの値を決定してください。

《ほかのオペランドとの関連》

このオペランドは，pdbuffer オペランドの-y オプションと次に示す関係があります。

- pd_dbbuff_rate_updpage オペランドの指定値は全グローバルバッファに適用されます。
- pdbuffer オペランドの-y オプションの指定値はグローバルバッファごとに適用されます。
- pdbuffer オペランドの-y オプションの指定値が優先されます。
- pdbuffer オペランドの-y オプションを指定しない場合，デファードライトトリガ契機の更新バッファ面数は，このオペランドの指定によって次のように異なります。

pd_dbbuff_rate_updpage オペランドの指定	デファードライトトリガ契機の更新バッファ面数
あり	グローバルバッファ面数×pd_dbbuff_rate_updpage オペランドの指定値
なし	HiRDB が自動計算します。

138）　pd_dbbuff_trace_level = グローバルバッファ制御情報トレース取得レベル

～<符号なし整数>((0～2147483647))《0》

グローバルバッファ制御情報トレースの取得レベルを符号なし整数で指定します。取得レベルは，グローバルバッファ制御情報トレース取得機能の各機能に対応したレベル値を合計して算出します。0 を指定した場合，グローバルバッファ制御情報トレースを取得しません。グローバルバッファ制御情報トレース取得機能とレベル値の対応を次に示します。グローバルバッファ制御情報トレース取得機能の詳細については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

グローバルバッファ制御情報トレース取得機能	レベル値
シンクポイント出力同期制御情報取得機能	1

《指定値の目安》

通常は，このオペランドを指定する必要はありません。障害調査などで保守員にこのオペランドの指定を依頼された場合は，保守員の指示に従って指定してください。

2.2.25　インメモリデータ処理に関するオペランド

インメモリデータ処理については，マニュアル「HiRDB Version 8 バッチ高速化機能」を参照してください。

139)　pd_max_resident_rdarea_no = インメモリ RD エリアの最大数

～<符号なし整数>((0～8388592))《0》

インメモリ RD エリアの最大数を指定します。このオペランドの指定値を超えた数の RD エリアをインメモリ化できません。

《前提条件》

このオペランドを指定する場合は HiRDB Accelerator が必要になります。

《指定値の目安》

- インメモリデータ処理を行う場合は，このオペランドに 1 以上を指定してください。0 を指定すると，インメモリデータ処理を実行できません。
- HiRDB/パラレルサーバの場合，このオペランドの指定値は 1 バックエンドサーバ当たりの最大数になります。

《注意事項》

- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- ページ固定に対応している Windows で，このオペランドに 1 以上を指定した場合の注意事項を次に示します。

  ・インメモリデータバッファが使用する共用メモリを実メモリ上に固定(pdmemdb -k stay -p fixed 指定)する場合，pdntenv コマンドを使用して共用メモリをページングファイルに割り当てる必要があります。pdntenv コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。

  ・インメモリデータバッファが使用する共用メモリを実メモリ上に固定(pdmemdb -k stay -p fixed 指定)する場合, 共用メモリは Windows の Large Page でのページサイズに切り上げて取得するため，メモリ所要量を見積もるときは注意してください。Windows の Large Page のページサイズについては，pdntenv -os コマンドで確認してください。なお，インメモリデータ処理で必要な共用メモリの見積もり式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「HiRDB のメモリ所要量」を参照してください。

・SHMMAX オペランドに Windows の Large Page のページサイズに切り上げた値を指定してください。

140）　pd_max_resident_rdarea_shm_no = インメモリデータバッファが使用する共用メモリセグメントの最大数

～<符号なし整数>((1～2147450879))《pd_max_resident_rdarea_no の値×1.5》

インメモリデータバッファが使用する共用メモリセグメントの最大数を指定します。このオペランドの指定値を超えた数の共用メモリセグメントを使用することはできません。

《前提条件》

- このオペランドを指定する場合は HiRDB Accelerator が必要になります。
- pd_max_resident_rdarea_no オペランドが指定されている必要があります。

《指定値の目安》

- インメモリデータ処理を行う場合はこのオペランドの指定を検討してください。マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「インメモリデータバッファが使用する共用メモリセグメント数」を参照してから，このオペランドの指定値を決定してください。
- HiRDB/パラレルサーバの場合，このオペランドの指定値は 1 バックエンドサーバ当たりの最大数になります。

《注意事項》

- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- このオペランドに指定するのはあくまで HiRDB が管理する上限値です。OS が管理する上限値とは異なります。例えば，このオペランドに 200 を指定しても，OS の上限値が 100 の場合，確保できる共用メモリセグメントの最大数は 100 になります。

2.2.26　表又はインデクスの予約数に関するオペランド

141）　pd_assurance_table_no = 表予約数の最低保証値

～<符号なし整数>((0～4194294500))《500》

空き領域の再利用機能を使用する場合にこのオペランドを指定するかどうかを検討してください。空き領域の再利用機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。このオペランドには表予約数の最低保証値を指定します。

現在定義済みの表数が表の予約数以下の場合は，表管理情報をメモリ上に常駐できるため，空き領域の再利用機能を使用できます。ただし，現在定義済みの表数が表予約数を超えると，空き領域の再利用機能を使用できなくなります。なお，HiRDB/パラレルサーバの場合は，表予約数がバックエンドサーバごとに適用されます。例えば，表予約数が 500 の場合，各バックエンドサーバの表予約数が 500 になります。

《指定値の目安》

- pd_sysdef_default_option オペランドを省略，又は recommendable を指定している場合

  現在の空き領域の再利用機能を使用した表数＋次回の HiRDB 開始から終了までの間に定義予定の空き領域の再利用機能を使用した表数

- pd_sysdef_default_option オペランドに v6compatible，又は v7compatible を指定している場合

  表予約数は空き領域の再利用機能を指定した表の数に応じて，HiRDB の正常開始時に自動的に増加します。したがって，通常はこのオペランドを指定する必要はありません。ただし，HiRDB

の開始から終了までの間に空き領域の再利用機能を使用した表を 101 個以上追加する場合は，次の二つのうち大きい方の値を，表予約数としてこのオペランドに指定してください。

・現在の空き領域の再利用機能を使用した表数＋次回の HiRDB 開始から終了までの間に定義予定の空き領域の再利用機能を使用した表数

・現在の空き領域の再利用機能を使用した表数＋ 100

《注意事項》

- HiRDB/パラレルサーバの場合は，バックエンドサーバごとに表予約数を計算して最大値をこのオペランドに指定してください。
- 横分割表の場合は 1 分割当たり 1 表として計算してください。
- 指定した値が不要に大きいと共用メモリ不足で HiRDB（HiRDB/パラレルサーバの場合はバックエンドサーバ）を開始できないことがあります。共用メモリの算出式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- pd_sysdef_default_option オペランドに v6compatible，又は v7compatible を指定している場合に，pd_bes_shmpool_size オペランド，又は pd_sds_shmpool_size オペランドを省略して共用メモリの自動計算機能を使用するときは，必ずこのオペランドにシステム開始時点の空き領域の再利用機能を使用した表数＋ 100 以上の値を指定してください。

《ほかのオペランドとの関連》

このオペランドの省略値は，pd_sysdef_default_option オペランドの指定によって次のように異なります。

- recommendable（省略値）：500
- v6compatible の場合：0
- v7compatible の場合：100

142）　pd_assurance_index_no ＝インデクス予約数の最低保証値

～＜符号なし整数＞((50～4194294500))《500》

インデクス予約数の最低保証値を指定します。インデクス予約数がこのオペランドの値以下の場合は，インデクス管理情報をメモリ上に常駐できるため，次に示すメリットがあります。

- インデクス情報をメモリ上に常駐し，性能を向上できる
- インデクスに関する統計情報を取得できる
- メッセージの重複出力を抑止できる

インデクス予約数がこのオペランドの値を超える場合は，これらのメリットがなくなることがあります。

なお，HiRDB/パラレルサーバの場合は，このオペランドの値はバックエンドサーバごとに適用されます。このオペランドに 600 を指定した場合，各バックエンドサーバのインデクス予約数の最低保証値が 600 になります。

《指定値の目安》

次に示す計算式の値をこのオペランドに指定してください。

現在のインデクス予約数＋次回 HiRDB 開始時までに追加予定のインデクス予約数

インデクス予約数の計算方法を次に示します。

- 1 インデクスを 1 インデクス予約数と数えてください。
- インデクスを横分割している場合は，1 分割当たり 1 インデクス予約数として数えてください。例えば，3 分割している場合は，インデクス予約数は 3 となります。

- HiRDB/パラレルサーバの場合は，バックエンドサーバごとにインデクス予約数を計算し，最大値をこのオペランドに指定してください。

なお，pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，インデクス予約数を HiRDB が自動計算するため，通常このオペランドを指定する必要はありません。ただし，HiRDB の開始から終了までの間に，HiRDB 開始時点のインデクス数の 1.2 倍を超える数のインデクスを定義する場合は，前記の指定値の目安に従ってこのオペランドを指定する必要があります。HiRDB/シングルサーバの場合は，データディクショナリ表のインデクス予約数（124）もカウント対象になります。求めた値に 124 を加算してください。

《注意事項》

- 指定した値が不要に大きいと共用メモリ不足で HiRDB（HiRDB/パラレルサーバの場合はバックエンドサーバ）を開始できないことがあります。

  共用メモリの算出式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

- pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合に，pd_bes_shmpool_size オペランド，又は pd_sds_shmpool_size オペランドを省略して共用メモリの自動計算機能を使用するときは，必ずこのオペランドにシステム開始時点のインデクス数の 1.2 倍を超える値を指定してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドの値によって，このオペランドの省略値が次のように変わります。

HiRDB のサーバ種別	pd_sysdef_default_option オペランドの値	省略値
HiRDB/シングルサーバ	recommendable（省略値）	500
	v6compatible，又は v7compatible	データディクショナリ表のインデクス数＋ 50
HiRDB/パラレルサーバ	recommendable（省略値）	500
	v6compatible，又は v7compatible	50

2.2.27　参照制約及び検査制約に関するオペランド

143）　pd_constraint_name = LEADING | TRAILING

参照制約又は検査制約で，制約名の定義を制約定義の前後どちらにするかを指定します。

LEADING：制約名定義を制約定義の前に指定します（標準 SQL 仕様）。

TRAILING：制約名定義を制約定義の後に指定します（XDM/RD 互換仕様）。

144）　pd_check_pending = USE | NOUSE

参照制約又は検査制約で，検査保留状態（整合性が保証できなくなったデータの操作を制限している状態）を使用するかどうかを指定します。

USE：検査保留状態を使用します。

NOUSE：検査保留状態を使用しません。

《指定値の目安》

このオペランドに NOUSE を指定すると，データの整合性が保証されないことがあるため，通常は USE を指定してください。データの整合性よりも処理性能を優先する場合は，NOUSE を指定してください。

《注意事項》

USE を指定，又は指定を省略した場合，PURGE TABLE 文を実行時にディクショナリ表（資源種別：3005 種別名：DICT）に対して一時的に EX モードで排他を掛けます。また，データディクショナリ用 RD エリア（資源種別：0001 種別名：RDAR）に対してトランザクションが終了するまで SU モードで排他を掛けます。

そのため，これらに排他を掛けるコマンド，又はユティリティと同時実行できないことがあります。PURGE TABLE 文と同時実行できないことがあるコマンド，又はユティリティと条件を次に示します。

コマンドまたはユティリティ	条件
pdmod（データベース構成変更ユィリティ）	ディクショナリ表の属性定義変更の場合
pdcopy（データベース複写ユティリティ）	以下の条件をすべて満たす場合 (a)-M オプションに x，又は r を指定した場合 (b)複写対象にマスタディレクトリ用 RD エリア，又はデータディクショナリ用 RD エリアを含む場合
pdreginit（レジストリ機能初期設定ユティリティ）	-k オプションに all を指定した場合

そのため，参照制約，又は検査制約を使用しない場合は NOUSE を指定してください。USE を指定，又は指定を省略した場合は，PURGE TABLE 文と上記に示すコマンド，又はユティリティを同時実行しないようにしてください。

2.2.28　HiRDB ファイルシステム領域に関するオペランド

145）　pd_large_file_use = Y | N

2048 メガバイト以上の HiRDB ファイルシステム領域を使用するかどうかを指定します。

Y：

2048 メガバイト以上の HiRDB ファイルシステム領域を使用する場合に指定します。Y を指定すると，HiRDB ファイルシステム領域長の上限が 1048575 メガバイトになります。システムログファイルの自動拡張機能を使用する場合は，Y を指定してください。

N：

2048 メガバイト以上の HiRDB ファイルシステム領域を使用しない場合に指定します。

《注意事項》

- このオペランドに N を指定してラージファイルに HiRDB ファイルシステム領域を作成すると，HiRDB ファイルシステム領域に格納した HiRDB ファイルがオープンできません。
- このオペランドの指定値を Y から N に変更する場合は注意が必要です。Y から N に変更すると，2048 メガバイト以上の HiRDB ファイルシステム領域内の RD エリア，及びシステムファイルは障害閉塞状態になります。RD エリアが障害閉塞状態になった場合は，次に示す手順で RD エリアの障害閉塞を解除してください。

手順

1. pdstop コマンドで HiRDB を正常終了します。

2. pd_large_file_use オペランドの指定値を Y に戻します。
3. pdstart コマンドで HiRDB を正常開始します。
4. pdrorg コマンドで，2048 メガバイト以上の HiRDB ファイルシステム領域内の RD エリアのデータをアンロードします。
5. pdstop コマンドで HiRDB を正常終了します。
6. pd_large_file_use オペランドの指定値を N に変更します。
7. pdstart コマンドで HiRDB を正常開始します。
8. pdmod コマンドで，2048 メガバイト以上の HiRDB ファイルシステム領域内の RD エリアを再初期化します。このとき，2048 メガバイト未満の HiRDB ファイルシステム領域を RD エリアに割り当てます。
9. pdrorg コマンドで，4 でアンロードしたデータをリロードします。

   また，システムファイルが障害閉塞状態になった場合は，出力メッセージに従って，対処してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，このオペランドの省略値は N になります。

146）　pd_ntfs_cache_disable = Y | N

HiRDB ファイルシステム領域へのアクセス時に**キャッシュ未使用アクセス**を使用するかどうかを指定します。このオペランドの指定は FAT などの NTFS 以外のファイルシステムでも有効になります。

Y：

キャッシュ未使用アクセスを使用します。この場合，Windows のキャッシュを使用しないため，Windows のキャッシュ速度の影響を受けません。このため，HiRDB の性能が安定します。また，Windows のキャッシュ使用量が減るため，Windows のリソース消費量が減少します。

なお，Y を指定する場合は，グローバルバッファのチューニングをしてください。チューニングをしないと性能が低下します。

N：

キャッシュ未使用アクセスを使用しません。この場合，Windows のキャッシュを使用します。

《指定値の目安》

次に示す場合はこのオペランドに Y を指定してください。

- KFPO00107-E 又は KFPS00700-E メッセージが出力される場合
- DLL の初期化エラーが発生する場合
- 1 サーバマシンで同時にオープンするファイルの総容量※が 100GB を超える場合

  注※　次に示すファイルの総容量を目安にしてください。

  ・RD エリア用の HiRDB ファイルシステム領域
  ・システムファイル用の HiRDB ファイルシステム領域
  ・オープンしているそのほかの HiRDB ファイルシステム領域
  ・オープンしているユーザ指定ファイル
  ・別のプログロムによってオープンしているファイル

《注意事項》

HiRDB ファイルシステム領域の使用目的によってはキャッシュ未使用アクセスを使用できません。キャッシュ未使用アクセスの適用範囲を次に示します。

HiRDB ファイルシステム領域の使用目的	適用可否
RD エリア用の HiRDB ファイルシステム領域（DB）	○
共用 RD エリア用の HiRDB ファイルシステム領域（SDB）	×※1
システムファイル用の HiRDB ファイルシステム領域（SYS）	○
ユティリティ用の，Windows のキャッシュを使用しない HiRDB ファイルシステム領域（NUTL）	○※2
作業表用ファイル用又はリスト用 RD エリア用の HiRDB ファイルシステム領域（WORK）	×
ユティリティ用の HiRDB ファイルシステム領域（UTL）	×
使用目的なしの HiRDB ファイルシステム領域（SVR）	×

（凡例）

○：キャッシュ未使用アクセスを使用できます。

×：キャッシュ未使用アクセスを使用できません。

注※1

Windows のファイルシステム上に作成できません。ダイレクトディスクアクセス専用です。

注※2

pd_ntfs_cache_disable オペランドの指定値に関係なく，キャッシュ未使用アクセスを使用します。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は N になります。

2.2.29　再編成時期予測機能に関するオペランド

再編成時期予測機能の運用方法については，マニュアル「HiRDB Version 8 システム運用ガイド」及び「HiRDB Version 8 コマンドリファレンス」を参照してください。

147）　pd_rorg_predict = Y | N

再編成時期予測機能を使用するかどうかを指定します。

Y：再編成時期予測機能を使用します。

N：再編成時期予測機能を使用しません。

《注意事項》

- 再編成時期予測機能を実行するには，解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアを作成する必要があります。
- このオペランドに Y を指定した場合，共用メモリを使用します。使用するメモリ容量については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

2.2.30　セキュリティに関するオペランド

(1)　セキュリティ監査機能に関するオペランド

セキュリティ監査機能の運用方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

148）　pd_audit = Y｜N

HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時から監査証跡を取得するかどうかを指定します。

Y：HiRDB の開始時から監査証跡を取得します。

N：HiRDB の開始時から監査証跡を取得しません。

このオペランドに N を指定しても，pdaudbegin コマンドを実行すると監査証跡を取得できます。

《前提条件》

次に示すすべての条件を満たす必要があります。満たしていない場合に Y を指定すると，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始できません。

- 監査証跡ファイル用の HiRDB ファイルシステム領域を作成している
- pd_aud_file_name オペランドに監査証跡ファイル用の HiRDB ファイルシステム領域名を指定している

149）　pd_aud_file_name = 監査証跡ファイル用の HiRDB ファイルシステム領域名

～<パス名>((150 文字以内))

セキュリティ監査機能を使用する場合はこのオペランドを必ず指定してください。**指定しないとセキュリティ監査機能を使用できません。**

監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。

複数ユニットのシステム構成でセキュリティ監査機能を使用する場合，システム全体で監査証跡を取得することを推奨します。システム全体で監査証跡を取得するには，次のどちらかの指定をしてください。

- システム共通定義で pd_aud_file_name オペランドを指定する
- すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する

ただし，同一サーバマシンで複数のユニットを稼働させるシステム構成の場合は，すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する必要があります。

《注意事項》

- このオペランドを指定した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に監査証跡ファイル用の HiRDB ファイルシステム領域に対するアクセスエラーが発生すると，HiRDB を開始できません。
- システム共通定義の pd_aud_file_name オペランドの指定によって同一サーバマシン上の複数のユニットが同じ監査証跡ファイルを指定した場合，監査証跡を正しく取得できません。

150）　pd_aud_max_generation_size = 1 監査証跡ファイルの最大容量

～<符号なし整数>((1～5240))《100》（単位：メガバイト）

1 監査証跡ファイルの最大容量をメガバイト単位で指定します。

《指定値の目安》

- HiRDB が管理用に使用する領域が 20 メガバイト必要なため，次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_aud_max_generation_size の値×pd_aud_max_generation_num の値<監査証跡ファイル用の HiRDB ファイルシステム領域長（pdfmkfs コマンドの-n オプションの値）－20（メガバイト）

- 監査証跡の 1 レコードの容量よりも小さい値を指定した場合，又は，このオペランドの指定を省略した場合で，かつ監査証跡の 1 レコードのサイズがこのオペランドの省略時仮定値よりも大きいときには，HiRDB ユニットを起動できません。

HiRDB ユニットを起動するには，次に示す条件式を満たすようにこのオペランドの値を決定してください。

pd_aud_max_generation_size の指定値≧↑監査証跡のレコードサイズの最大長÷1024↑×1024 ＋ 2048（バイト）

監査証跡のレコードサイズの最大長は，次の計算式で算出します。

監査証跡のレコードサイズの最大長＝ 1067 ＋↑pd_aud_sql_source_size の指定値÷4↑×4 ＋↑pd_aud_sql_data_size の指定値÷4↑×4（バイト）

151）　pd_aud_max_generation_num ＝ **監査証跡ファイルの最大数**

～＜符号なし整数＞((2～200))《50》

監査証跡ファイル用の HiRDB ファイルシステム領域内に作成する監査証跡ファイルの最大数（世代数）を指定します。

《指定値の目安》

- 全監査証跡ファイルに障害が発生したときのことを想定して，最大値である 200 を指定しないことをお勧めします。監査証跡ファイルに障害が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- HiRDB が管理用に使用する領域が 20 メガバイト必要なため，次に示す条件式を満たすようにこのオペランドの値を決定してください。

pd_aud_max_generation_size の値×pd_aud_max_generation_num の値＜監査証跡ファイル用の HiRDB ファイルシステム領域長（pdfmkfs コマンドの-n オプションの値）－20（メガバイト）

《注意事項》

HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，このオペランドの値よりも大きい世代番号のファイルがある場合，このオペランドの値は無効になります。この場合，HiRDB ファイルシステム領域内に作成される監査証跡ファイルの最大数には，最大世代番号が仮定されます。

152）　pd_aud_no_standby_file_opr = down｜<u>forcewrite</u>

監査証跡の出力先が満杯になった場合の HiRDB の処理方式を指定します。

pd_aud_async_buff_size オペランドの値によって，以下に示す場合に，監査証跡の出力先が満杯になります。

pd_aud_async_buff_size オペランドの値	監査証跡の出力先が満杯になるケース
0（監査証跡の出力方式に同期出力を適用する場合）	• スワップ先にできる監査証跡ファイルがない場合
4096～6553600（監査証跡の出力方式に非同期出力を適用する場合）	• スワップ先にできる監査証跡ファイルがない場合 • 監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちとなる場合

監査証跡の出力先が満杯になった場合，pd_aud_no_standby_file_opr オペランドの値によって，以下に示す HiRDB の処理を行います。

pd_aud_no_standby_file_opr オペランドの値	スワップ先にできる監査証跡ファイルがない場合	監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちとなる場合
down	スワップ先にできる監査証跡ファイルが残り一つになった場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を強制終了します。down	監査証跡の出力処理が 180 秒以上滞った場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を強制終了します。その後の対処につ

pd_aud_no_standby_file_opr オペランドの値	スワップ先にできる監査証跡ファイルがない場合	監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちとなる場合
	の指定が有効になるスワップの契機は，監査証跡ファイルの容量が一杯になった場合，及び現用ファイルに障害が発生した場合です。 このオペランドの指定によって HiRDB が強制終了した場合の対処については，マニュアル「HiRDB Version 8 システム運用ガイド」の「スワップ先にできる監査証跡ファイルがないため HiRDB が強制終了した場合」を参照してください。	いては，マニュアル「HiRDB Version 8 システム運用ガイド」の「監査証跡の非同期出力用バッファのすべての面が，バッファから監査証跡ファイルへの出力待ちとなった場合」に示す対処を行ってください。
forcewrite	スワップ先にできる監査証跡ファイルがない場合，データロード待ち（閉塞状態のファイルを除く）の監査証跡ファイルを強制的にスワップ先にして監査証跡の出力を続行します。このとき，最終更新日時が一番古いデータロード待ちの監査証跡ファイルをスワップ先にします。 ただし，pdaudswap コマンドを実行してスワップした場合，又は全ファイルが閉塞状態の場合は監査証跡の出力を中止します。	監査証跡の出力処理が 180 秒以上滞った場合，一番古いバッファに出力済みの監査証跡を破棄して，監査証跡の出力を続行します。

153）　pd_aud_async_buff_size = 監査証跡の非同期出力時に使用するバッファ長～<符号なし整数>((0，4096～6553600))《401408》（単位：バイト）

監査証跡を非同期に出力する場合に使用するバッファ長をバイト単位で指定します。0 を指定した場合は監査証跡を同期出力します。監査証跡のレコードサイズの最大長よりも小さい値を指定した場合，又は，このオペランドの指定を省略した場合で，かつ監査証跡のレコードサイズの最大長がこのオペランドの省略時仮定値よりも大きいときには，HiRDB ユニットを起動できません。

監査証跡のレコードサイズの最大長については，pd_aud_max_generation_size オペランドの説明を参照してください。

各出力方式の長所及び短所を次に示します。

pd_aud_async_buff_size の値	監査証跡の出力方式	長所	短所
0	同期出力	監査証跡を確実に監査証跡ファイルに出力できます。	SQL 処理の延長上でファイル入出力が発生するため，性能に与える影響が大きくなります。
4096～6553600	非同期出力	SQL 処理の性能に与える影響を小さくできます。	バッファへの出力後から監査証跡ファイルに出力するまでの間に，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了すると，監査証跡が失われることがあります。

《オペランドの規則》

このオペランドには 4096 の整数倍を指定してください。4096 の整数倍以外を指定した場合は，指定した値を 4096 の整数倍に切り上げてその値をこのオペランドに設定します。例えば，5000 を指定すると 8192 が設定されます。

《注意事項》

- HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，pd_aud_async_buff_size×pd_aud_async_buff_count バイト分のユニットコントローラ用共用メモリが必要です。この計算式の値がユニットコントローラ全体の共用メモリサイズの上限を超えないように設定してください。ユニットコントローラが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を小さく設定すると，監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちに陥り，監査証跡の出力処理が滞ることがあります。

  単位時間あたりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を設定してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は 4096 になります。

154）　pd_aud_async_buff_count = 監査証跡の非同期出力時に使用するバッファ面数

～<符号なし整数>((1～6500))《Max(1，ユニット内 HiRDB サーバ数×10)》（単位：面）

監査証跡を非同期に出力する場合に使用するバッファの面数を指定します。

《注意事項》

- HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，pd_aud_async_buff_size×pd_aud_async_buff_count バイト分のユニットコントローラ用共用メモリが必要です。この計算式の値がユニットコントローラ全体の共用メモリサイズの上限を超えないように設定してください。ユニットコントローラが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を小さく設定すると，監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちに陥り，監査証跡の出力処理が滞ることがあります。

  単位時間あたりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を設定してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は 3 になります。

155）　pd_aud_async_buff_retry_intvl = 監査証跡の非同期出力時に使用するバッファの確保リトライ間隔

～<符号なし整数>((1～1000))《50》（単位：ミリ秒）

監査証跡を非同期に出力する場合に使用するバッファがすべて使用中のとき，未使用のバッファが確保できるまでバッファを監視する処理のリトライ間隔を指定します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

セキュリティ監査機能使用時，UAP の実行に時間が掛かる場合にこのオペランドの値を小さくすると，UAP の実行時間が短くなることがあります。

156）　pd_aud_sql_source_size = 監査証跡に出力する SQL 文のサイズ

～<符号なし整数>((0～2000000))《0》（単位：バイト）

セキュリティ監査機能を使用する場合に監査証跡に出力する SQL 文のサイズをバイト単位で指定します。0 を指定した場合は，監査証跡に SQL 文を出力しません。指定した値よりもサイズが大きい SQL 文については，指定した値を超えた部分は監査証跡に出力されません。

《ほかのオペランドとの関連》

このオペランドを指定した場合，オペランド pd_aud_max_generation_size 及び pd_aud_async_buff_size の指定値を見積もり直してください。

157）　pd_aud_sql_data_size = 監査証跡に出力する SQL データのサイズ

～<符号なし整数>((0～1000000))《0》（単位：バイト）

セキュリティ監査機能を使用する場合に監査証跡に出力する SQL データのサイズをバイト単位で指定します。0 を指定した場合は，監査証跡に SQL データを出力しません。指定した値よりもサイズが大きい SQL データについては，指定した値を超えた部分は監査証跡に出力されません。

《ほかのオペランドとの関連》

このオペランドを指定した場合，オペランド pd_aud_max_generation_size 及び pd_aud_async_buff_size の指定値を見積もり直してください。

158）　pd_aud_file_wrn_pnt = 警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕

警告メッセージの出力契機：～<符号なし整数>((0～100))《0 又は 80》（単位：%）

スワップ先にできない監査証跡ファイル数が警告値以上になったとき，警告メッセージを出力します。このオペランドには警告値を，pd_aud_max_generation_num オペランドに指定した監査証跡ファイルの最大数に対する比率で指定します。例えば，pd_aud_max_generation_num オペランドに 100 を指定し，このオペランドに 90 を指定すると，スワップ先にできない監査証跡ファイル数が 90 以上になると警告メッセージ KFPS05123-W が出力されます。

なお，HiRDB/パラレルサーバの場合はユニット単位でチェックします。

また，このオペランドに 0 を指定すると警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定するか又は省略して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

警告メッセージの出力済み状態のリセット契機：～<符号なし整数>((0～99))（単位：%）

警告メッセージ出力済み状態のリセット契機を指定します。警告メッセージ（KFPS05123-W）が出力されると，警告メッセージ出力済み状態になります。警告メッセージ出力済み状態になると，スワップ先にできない監査証跡ファイル数がもう一度警告値以上になっても警告メッセージが出力されません。しかし，ここで指定する警告メッセージの出力済み状態のリセット契機をスワップ先にできない監査証跡ファイル数が下回ると，警告メッセージ出力済み状態が解除されます。

例えば，pd_aud_file_wrn_pnt=90,70 と指定すると，スワップ先にできない監査証跡ファイル数が監査証跡ファイルの最大数の 90%以上になると警告メッセージが出力されます。この後，スワップ先にできない監査証跡ファイル数が監査証跡ファイルの最大数の 70%を下回るまで警告メッセージは出力されません。一度 70%を下回り，再度 90%以上になったときに警告メッセージが出力されます。

《注意事項》

- この指定を省略すると，「警告メッセージの出力契機－30」の値が仮定されます。その値がマイナスになる場合は，0 が仮定されます。

- 警告メッセージの出力契機より大きい値を指定すると，その指定は無効になり，警告メッセージの出力契機の指定値と同じ値が仮定されます。

159）　pd_aud_auto_loading = Y | N

監査証跡表の自動データロード機能を使用するかどうかを指定します。監査証跡表の自動データロード機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

Y：

監査証跡表の自動データロード機能を使用します。この場合，監査証跡ファイルの世代スワップ（障害によるスワップは除く）を契機として，監査証跡表へのデータロード処理が自動実行されます。

N：

監査証跡表の自動データロード機能を使用しません。この場合，監査証跡表へのデータロード処理は自動実行されません。監査人が手動で実行してください。

《指定値の目安》

監査証跡表の自動データロード機能を使用すると監査人の負荷を低減できます。しかし，オンライン業務中にデータロードが始まるため，CPU やディスクへの入出力回数が増加し，システムの負荷が増大します。これらのメリットとデメリットを考慮した上で，監査証跡表の自動データロード機能を使用するかどうか判断してください。

《留意点》

HiRDB/パラレルサーバにおいて，本オペランドに Y を指定して監査証跡ファイルから監査証跡表へデータロードする処理を自動で実行する場合，システムマネジャを配置したユニットに pd_aud_file_name オペランドで監査証跡ファイルのパス名を設定してください。

設定方法については，システム共通定義の pd_aud_file_name オペランドを参照してください。

160）　pdaudload

〔-i インデクス作成方法〕
〔-l ログ取得方式〕
〔-n 〔一括出力用ローカルバッファ面数〕,,
〔ランダムアクセス用ローカルバッファ面数〕〕
〔-y〕
〔-X サーバ間通信の応答監視時間〕
〔-S ソート用バッファサイズ〕

監査証跡表の自動データロード機能を使用するとき，監査証跡表の自動データロード機能で動作させるデータベース作成ユティリティ（pdload）の環境情報を定義します。

《前提条件》

pd_aud_auto_loading オペランドに Y を指定している必要があります。

《指定値の目安》

マニュアル「HiRDB Version 8 システム運用ガイド」の「監査証跡表の自動データロード機能」にある「適用条件」を参照してオペランドの値を決定してください。

《注意事項》

このオペランドを 2 回以上指定した場合，1 回目に指定した値が適用されて，2 回目以降の指定値は無効となります。

-i インデクス作成方法

～《c》

インデクスの作成方法を指定します。インデクスの作成方法には次の二つがあります。

c：

インデクス一括作成モードを指定します。行データの格納中は，インデクスを作成しないでインデクス作成情報をインデクス情報ファイルに出力します。行データの格納処理が完了した後，引き続きインデクスを作成します。

s：

インデクス更新モードを指定します。行データを格納するたびにインデクスを更新します。

-l ログ取得方式

～《p》

pdload を実行するとき，データベースの更新ログを取得する方法を指定します。

a：

ログ取得モードを指定します。ロールバック及びロールフォワードに必要なデータベースの更新ログを取得します。

p：

更新前ログ取得モードを指定します。ロールバックに必要なデータベースの更新ログは取得しますが，ロールフォワードに必要なデータベースの更新ログは取得しません。

-n 〔一括出力用ローカルバッファ面数〕,,〔ランダムアクセス用ローカルバッファ面数〕

ローカルバッファを使用して表にデータロードする場合に指定します。このオプションを指定すると，ローカルバッファを使用して DB アクセスできるようになり，一括出力することで入出力回数を削減できます。

このオプションを省略した場合は，グローバルバッファを使用して 1 ページ単位で出力します。

一括出力用ローカルバッファ面数を省略してランダムアクセス用ローカルバッファ面数だけを指定する場合は，コンマ（,）を含めた内容を引用符（"）で囲んで指定してください。例えば，一括出力用ローカルバッファ面数を省略してランダムアクセス用ローカルバッファ面数だけに 1000 を指定する場合は，次のように指定してください。

pdaudload -n ",,1000"

ただし，一括出力用ローカルバッファ面数とランダムアクセス用ローカルバッファ面数の両方を同時に省略することはできません。両方を省略して-n オプションを指定した場合，定義エラーとなり KFPS01895-E メッセージが出力されます。

一括出力用ローカルバッファ面数：～<符号なし整数>((2～4096))

一括出力用ローカルバッファの面数を指定します。一括出力用ローカルバッファは，データページに対して使用します。

ランダムアクセス用ローカルバッファ面数：～<符号なし整数>((4～125000))

ランダムアクセス用ローカルバッファの面数を指定します。ランダムアクセス用ローカルバッファは，インデクスページに対して使用します。

-y

データロード時に未使用ページを使い切った場合に，データを使用中ページの未使用領域に格納したいときに指定します。このオプションを指定した場合，未使用領域にデータを格納する前に KFPH26010-I メッセージを出力します。

このオプションを指定する場合は-l オプションに a を指定してください。-l オプションに p を指定した場合，又は-l オプションの指定を省略した場合，定義エラーとなり KFPS01895-E メッセージが出力されます。

-X サーバ間通信の応答監視時間

～<符号なし整数>((1～65535))《300》

コマンドを実行したサーバで通信などに障害が発生した場合，コマンドが無応答となり，業務が停止するおそれがあります。pdload は，障害を検知するためにコマンドで実施されるディクショナリ操作についての通信の応答時間を監視できます。

-X オプションには，ディクショナリ操作での応答監視時間を秒数で設定します。ディクショナリ操作時の実行時間が-X オプションに設定した時間を超えた場合，pdload はディクショナリアクセスで障害が発生したと判断し，リターンコード 8 で処理を打ち切ります。処理を打ち切った場合，監査証跡表の自動データロード処理も停止します。

-S ソート用バッファサイズ

～<符号なし整数>((128～2097152))《1024》

インデクスの定義された監査証跡表に対してインデクス一括作成モード（-i c 指定）でデータロードをするとき，ソート用ワークファイルのバッファサイズをキロバイト単位で指定します。バッファは，HiRDB/シングルサーバの場合はシングルサーバに，HiRDB/パラレルサーバの場合は監査証跡表のあるバックエンドサーバに確保されます。

ソート用バッファサイズについては，マニュアル「HiRDB Version 8 コマンドリファレンス」の「データベース作成ユティリティ（pdload）」にある「sort 文（ソート用ワークディレクトリの情報の記述）」を参照してください。

(2)　セキュリティの強化に関するオペランド

161）　pd_security_host_group = "ホスト名"〔，"ホスト名"〕…

このオペランドは，セキュリティレベルの高いシステムを構成する場合，HiRDB サーバ構成で使用するホストを明確に定義し，セキュリティに影響する HiRDB の操作（ユティリティや関連プログラムプロダクトが定義したホスト以外からの操作）を制限するものです。

このオペランドには，HiRDB サーバを構成するネットワークで使用している全ホスト名を指定してください。このオペランドを指定することで，セキュリティ上のリスクを軽減できます。ホスト名は，IP アドレス，FQDN 形式でも指定でき，ループバックアドレスも指定できます。

また，オペランドの指定値は，強制終了，異常終了，及び計画停止後でも変更できます。

なお，ホスト名は 256 文字以内で指定してください。

《注意事項》

- DNS サーバを使用する場合は，ここで指定するホスト名を DNS サーバに登録してください。また，DNS サーバを使用しない場合は，ここで指定するホスト名を hosts ファイルに登録してください。
- 名称解決ができない場合，KFPS04693-E メッセージが出力され，HiRDB の開始処理が終了します。
- ホスト名，IP アドレス，又は FQDN の文字列を重複して指定した場合，KFPS04693-E メッセージが出力され，HiRDB の開始処理が終了します。また，IP アドレスが同じでホスト名が異なる場合，エラーにはなりません。
- pdunit オペランドの-x オプションにループバックアドレスを指定した場合は，このオペランドにもループバックアドレスを指定してください。

《備考》

pd_security_host_group オペランドの指定例を次に示します。

pd_security_host_group オペランドの指定例（HiRDB/シングルサーバの場合）

（凡例）
hostA～hostC：ホスト名
xx.x.x.x　　：IPアドレス
：HiRDBサーバを構成するネットワーク

●システム共通定義の内容

```
                    :
set pd_security_host_group=hostA,hostB
pdstart -t SDS -s sds -x hostA
                    :
```

pd_security_host_group オペランドの指定例（HiRDB/パラレルサーバの場合）

（凡例）
hostA～hostG：ホスト名
xxx.xxx.x.x：IPアドレス
- - - - - - - - ：HiRDBサーバを構成するネットワーク

●システム共通定義の内容

```
                    :
set pd_security_host_group=hostA,hostB,hostC,hostD,hostE,hostF
pdstart -t FES -s fes1 -x hostA
                    :
```

2.2.31　系切り替え機能に関するオペランド

162）　pd_ha = use | nouse

系切り替え機能を使用するかどうかを指定します。

use：系切り替え機能を使用します。

nouse：系切り替え機能を使用しません。

163）　pd_ha_ipaddr_inherit = Y | N

系切り替え機能を使用する場合に IP アドレスを引き継ぐかどうかを指定します。IP アドレスを引き継ぐかどうかによって，系切り替え機能の運用方法が異なります。詳細については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

Y：IP アドレスを引き継ぎます。

N：IP アドレスを引き継ぎません。

サーバモードで運用する場合は，IP アドレスを引き継ぐことはできません。

1：1 スタンバイレス型系切り替え機能又は影響分散スタンバイレス型系切り替え機能を使用する場合はこのオペランドを省略してください。指定しても無効になります。

《指定値の目安》

- HiRDB/シングルサーバの場合

IP アドレスを引き継ぐ場合は Y を，引き継がない場合は N を指定してください。ただし，サーバモードで運用する場合は IP アドレスを引き継げません。したがって，ここで Y を指定した場合，ユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに N を指定する必要があります。

なお，このオペランドに N を指定した場合は，ユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに Y を指定できません。

- HiRDB/パラレルサーバの場合

  IP アドレスを引き継ぐ場合は Y を，引き継がない場合は N を指定してください。ただし，サーバモードの対象にするユニットは IP アドレスを引き継げません。したがって，ここで Y を指定した場合，サーバモードの対象にするユニットのユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに N を指定する必要があります。

  なお，このオペランドに N を指定した場合は，ユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに Y を指定できません。

164）pd_ha_switch_timeout = Y | N

このオペランドはサーバモードの場合に指定できます。モニタモードの場合にこのオペランドを指定しても無効になります。

系切り替え時の HiRDB（HiRDB/パラレルサーバの場合はユニット）の内部停止処理が**サーバ障害監視時間**を超えた場合に，HiRDB の内部停止処理を待たないで系を切り替えるかどうかを指定します。ここでいうサーバ障害監視時間とは，Hitachi HA Toolkit Extension の patrol オペランドに指定した時間のことです。

Hitachi HA Toolkit Extension の patrol オペランドについては，マニュアル「Hitachi HA Toolkit」を参照してください。

Y：

系切り替え時の HiRDB の内部停止処理がサーバ障害監視時間を超えた場合，HiRDB の内部停止処理を待たないで系を切り替えます。このとき，HiRDB のスローダウンとして系を切り替えます。

1：1 スタンバイレス型系切り替え機能又は影響分散スタンバイレス型系切り替え機能を使用している場合，計画系切り替え時はこのオペランドの指定が無効になります。

N：

系切り替え時の HiRDB の内部停止処理が終了するまで系を切り替えません。

《利点》

系の切り替え時にディスク障害などで HiRDB の内部停止処理に時間が掛かると，それが系の切り替え時間の遅延要因になります。このオペランドに Y（省略値）を指定すると，HiRDB の内部停止処理に時間が掛かる場合でも，HiRDB の内部停止処理を待たないで系を切り替えられます。

《注意事項》

- patrol オペランドの値が小さいときにこのオペランドに Y を指定すると，計画系切り替えがスローダウンによる系切り替えになることがあります。これは，計画系切り替え時の HiRDB の内部停止処理が patrol オペランドで指定した時間を超えてしまうためです。
- Hitachi HA Toolkit Extension の switchtype オペランドに restart を指定した場合は注意が必要です。pd_ha_switch_timeout = Y（省略値）を指定した場合，HiRDB の内部停止処理がサーバ障害監視時間を超えると，障害が発生した系で HiRDB を再開始しません。この場合，すぐに系を切り替えます。

  Hitachi HA Toolkit Extension の switchtype オペランドについては，マニュアル「Hitachi HA Toolkit」を参照してください。

165）　pd_ha_mgr_rerun = wait | notwait

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

システムマネジャユニットの系が切り替わるときに，ほかのユニットの開始処理完了を待ち合わせるかどうかを指定します。

wait：

システムマネジャユニットの系を切り替えるときに，ほかのユニットの開始処理完了を待ち合わせます。この場合，システムマネジャユニットを開始するため，以下の確認が行われます。

- 各ユニットのバージョンチェック
- ディクショナリサーバが稼働中であるか
- フロントエンドサーバが一つ以上稼働中であるか
- バックエンドサーバが一つ以上稼働中であるか

システムマネジャユニットの系を切り替える場合に，ほかのユニットが停止しているとき，切り替えに時間が掛かる，又は失敗することがあります。一部のユニットが停止中に，システムマネジャユニットが系切り替えした場合の動作を次に示します。

縮退起動の指定（pd_start_level の値）	開始しないユニット名称の指定	動作
なし（0）	－	開始できない
あり（1）	pd_start_skip_unit オペランドに指定なし	pd_reduced_check_time オペランドで指定した時間だけ開始を待ち合わせた後に系が切り替わる※
	pd_start_skip_unit オペランドに停止中のユニットを指定	すぐに系の切り替えが完了する※

（凡例）

－：該当しません。

注※

次の条件をすべて満たしていないと，系切り替えが完了しません。

- ディクショナリサーバが稼働している
- フロントエンドサーバが一つ以上稼働している
- バックエンドサーバが一つ以上稼働している

notwait：

システムマネジャユニットの系切り替え時に，ほかのユニットの開始処理完了を待ち合わせません。このオペランドを指定すると，システムマネジャユニットの系切り替え時にほかのユニットが停止中でも，そのユニットの開始を待ち合わせることなく，系の切り替えが高速で行われます。

システムマネジャユニットの切り替えが完了すると，HiRDB の動作環境が整っていない場合でも，KFPS05210-I メッセージ（システム初期化完了メッセージ）が出力されます。システムマネジャ切り替え完了後も，UAP がエラーになるなど，業務が続行できない場合は，pdls コマンドでシステムの稼働状況を確認してください。

《前提条件》

このオペランドに notwait を指定する場合は，システム構成がある条件を満たしている必要があります。前提となるシステム構成については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値の目安》

システムマネジャユニットの系切り替えの失敗を防ぐ場合は，notwait を指定してください。システムマネジャユニットの系切り替えに失敗するときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

166）　pd_ha_transaction = error | queuing

トランザクションキューイング機能を使用するかどうかを指定します。また，系の切り替え中に HiRDB サーバへの接続数が最大同時接続数（pd_max_users オペランドの値）を超えた場合の処理を指定します。トランザクションキューイング機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

error：

- トランザクションキューイング機能を使用しません。系切り替え中のバックエンドサーバ又はディクショナリサーバで処理するトランザクションをエラーにします。
- 系の切り替え中に HiRDB サーバへの接続数が最大同時接続数を超えた場合，HiRDB サーバへの接続をエラーにします。

queuing：

- トランザクションキューイング機能を使用します。系切り替え中のバックエンドサーバ又はディクショナリサーバで処理するトランザクションをエラーにしないで，系切り替えが完了するまでフロントエンドサーバでトランザクションをキューイングします。
- 系の切り替え中に HiRDB サーバへの接続数が最大同時接続数を超えた場合，HiRDB サーバへの接続処理を pd_ha_trn_queuing_wait_time ＋ pd_ha_trn_restart_retry_time の時間だけ HiRDB クライアントでリトライします。ただし，HiRDB クライアントのバージョンが 07-00 以上の必要があります。

《前提条件》

トランザクションキューイング機能を使用する場合は，次に示す条件をすべて満たす必要があります。

- HiRDB/パラレルサーバである
- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用している

なお，HiRDB サーバへの接続処理を HiRDB クライアントでリトライする処理については，前記の前提条件を満たす必要はありません。

《クライアント環境定義との関連》

このオペランドに queuing を指定しても，クライアントごとにトランザクションキューイング機能を使用しないように変更できます。クライアントごとにトランザクションキューイング機能をキャンセルする場合は，クライアント環境定義の PDHATRNQUEUING オペランドを指定してください。PDHATRNQUEUING オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_ha_trn_queuing_wait_time
- pd_ha_trn_restart_retry_time

167）　pd_ha_trn_queuing_wait_time = トランザクションのキューイング待ち時間

～<符号なし整数>((1～3600))《180》（単位：秒）

トランザクションキューイング機能を使用する場合のトランザクションのキューイング待ち時間を指定します。このオペランドに指定した待ち時間を超えても待機系ユニット又はサーバが開始しない場合，キューイング中のトランザクションをエラーにします。これ以降発生するトランザクションはキューイングされないでエラーになります。なお，この待ち時間より先に開始処理中のユニット又はサーバが開始した場合は，この指定時間を待たないでトランザクションが再開されます。

また，1：1 スタンバイレス型系切り替え機能の場合は，正常時は代替部が待機系になり，代替中は正規 BES ユニットが待機系になります。

《前提条件》

pd_ha_transaction オペランドに queuing を指定する必要があります。

《指定値の目安》

通常，このオペランドを指定する必要はありませんが，系切り替え時のロールフォワード時間が 180 秒以上掛かる場合にこのオペランドの指定値を大きくしてください。

168）　pd_ha_trn_restart_retry_time = トランザクション開始要求エラー時のリトライ時間の上限～<符号なし整数>((1～3600))《60》（単位：秒）

トランザクションキューイング機能使用時に系切り替えが発生すると，トランザクションをフロントエンドサーバでキューイングします。しかし，系切り替えが発生してから待機系ユニット又はサーバが再開始するまでの間，フロントエンドサーバは系切り替えを検知できません。この間（系切り替えが発生してから待機系ユニット又はサーバが再開始するまでの間），フロントエンドサーバは実行系ユニット又はサーバに対してトランザクションの開始要求をしますが，実行系ユニットは既に異常終了しているため，このトランザクション開始要求はエラーになります。エラーになったトランザクションはフロントエンドサーバから開始要求が再度行われます（トランザクションの開始要求をリトライします）。

このオペランドには，このリトライ時間の上限を指定します。このオペランドの値を超えても待機系ユニット又はサーバが再開始されない場合は，リトライ中のトランザクションをエラーにします。また，これ以降発生するトランザクションはリトライされないでエラーになります。なお，このリトライ時間より前に待機系のユニット又はサーバが再開始した場合は，この指定時間リトライは行わないでキューイングされます。

《前提条件》

pd_ha_transaction オペランドに queuing を指定する必要があります。

《指定値の目安》

通常，このオペランドを指定する必要はありませんが，クラスタソフトウェアによるディスクの切り替え時間が 60 秒以上掛かる場合にこのオペランドの指定値を大きくしてください。

《備考》

pd_ha_trn_queuing_wait_time オペランドと pd_ha_trn_restart_retry_time オペランドの関連を次に示します。

〔説明〕

区間 A，D：

トランザクションを開始できる状態です（正常な状態です）。

区間 B：

バックエンドサーバがあるユニットが系切り替え中で，フロントエンドサーバがそれを検知できていない状態です。pd_ha_trn_restart_retry_time オペランドに指定した時間までトランザクション開始要求をリトライします。フロントエンドサーバが系の切り替えを検知すると，トランザクションがキューイングされます。指定した時間を過ぎても検知できない場合は，そのトランザクションをエラーにします。

区間 C：

バックエンドサーバがあるユニットが系切り替え中で，フロントエンドサーバがそれを検知している状態です。pd_ha_trn_queuing_wait_time オペランドに指定した時間までトランザクションをキューイングします。指定した時間を過ぎてもトランザクションを開始できない場合は，そのトランザクションをエラーにします。

169)　pd_ha_resource_act_wait_time = リソース活性化最大待ち時間

～<符号なし整数>((2～3600))《10》(単位：秒)

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。このオペランドには，ユニットを開始するときに，実行系サーバのリソースが活性化されるまでの最大待ち時間を秒単位で指定します。このオペランドに指定した待ち時間の間は，ユニットの開始処理を待ち合わせます。ただし，ここで指定した時間内にリソースが活性化された場合は，すぐにユニットの開始処理を再開します。

《利点》

ユニットの開始処理が完了したときに，ユニット内の実行系サーバの起動処理が完了していないと，業務を開始できません。このオペランドに適切な値を指定すると，ユニットの開始処理を待ち合わせるため，ユニットの開始処理が完了した直後から業務を開始できます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。次に示す条件をすべて満たす場合にこのオペランドを指定してください。

- 影響分散スタンバイレス型系切り替え機能を使用している場合
- KFPS05623-I メッセージが出力された場合
- メッセージ出力対象ユニットに実行系サーバがある場合

指定値の目安を次に示します。

10 ＋リソース活性化処理に掛かる時間（秒）

《備考》

ユニット内に実行系サーバが存在しない場合，このオペランドに指定した時間だけ実行系サーバの開始を待ち合わせます。ただし，ユニット内の全サーバが待機系サーバとして開始した場合は，このオペランドで指定した時間を待たないでユニットの開始処理を再開します。

170）pd_deter_restart_on_stop_fail = Y｜N

系切り替えをサーバモードで運用している場合，実行系での pdstop -f コマンド又は pdstop -z コマンドによる強制停止処理時にユニットが異常終了したときに系切り替えを抑止するかどうかを指定します。

Y：強制停止処理中にユニットが異常終了したとき，系切り替えを抑止します。

N：強制停止処理中にユニットが異常終了したとき，系切り替えを抑止しません。

《注意事項》

この機能を使用するためには，HiRDB と組み合わせる Hitachi HA Toolkit Extension のバージョンが 01-20 以降である必要があります。01-20 以前のバージョンの製品と組み合わせた場合，このオペランドに Y を指定しても無効となり，指定値に N が仮定されます。

2.2.32　HiRDB Datareplicator に関するオペランド

171）pd_rpl_init_start = Y｜N

HiRDB Datareplicator 連携機能を HiRDB 開始時から使用するかどうかを指定します。

Y：HiRDB Datareplicator 連携機能を HiRDB 開始時から使用します。

N：HiRDB Datareplicator 連携機能を HiRDB 開始時から使用しません。

《運用方法》

- このオペランドに N を指定した場合でも，pdrplstart コマンドを入力すれば，その時点から HiRDB Datareplicator 連携を開始できます。
- HiRDB 稼働中に HiRDB Datareplicator 連携を中止する場合，pdrplstop コマンドを入力します。

  なお，pdrplstop コマンドの入力によってデータ連動を中止した場合でも，pdrplstart コマンドを入力すると HiRDB Datareplicator 連携を再開始できます。ただし，pdrplstart コマンドを入力する前に，必ず抽出側データベースを基に反映側データベースを再作成してください。

172）pd_rpl_reflect_mode = server｜uap

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB Datareplicator の反映トランザクション同期機能を使用する場合，このオペランドに uap を指定してください。反映トランザクション同期機能を使用しない場合は，このオペランドを指定する必要はありません。反映トランザクション同期機能については，マニュアル「HiRDB データ連動機能 HiRDB Datareplicator Version 8」を参照してください。

server：HiRDB サーバ単位でトランザクションを反映します。

uap：UAP 単位でトランザクションを反映します。

《前提条件》

このオペランドを指定する場合は，必ず pd_rpl_hdepath オペランドを指定してください。pd_rpl_hdepath オペランドを指定していない場合，このオペランドの指定は無視されます。

《注意事項》

uap を指定した場合は更新ログ量が増加します。マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照して，システムログファイルの総容量を見積もり直してください。

《ほかのオペランドとの関連》

このオペランドに uap を指定する場合は，pd_log_max_data_size オペランドの指定値を見積もり直してください。

173）　pd_log_rpl_no_standby_file_opr = stop | continue

HiRDB Datareplicator 連携機能を使用しているときに，HiRDB Datareplicator でのシステムログの抽出が完了していないため，すべてのシステムログファイルがスワップ先にできない状態でスワップ要求が発生した場合の運用方法を指定します。

stop：HiRDB のユニットを強制終了します。

continue：HiRDB Datareplicator 連携を中止し，HiRDB だけで運用を続行します。

《前提条件》

この指定値は，pd_rpl_init_start オペランドに Y を指定するか，又は pdrplstart コマンドを入力して，かつ HiRDB Datareplicator 連携機能を使用している場合に有効になります。ただし，HiRDB/パラレルサーバの場合，フロントエンドサーバ，及びディクショナリサーバに対しては無効です。

《注意事項》

- このオペランドに stop（省略時の仮定値）を指定し，ユニットを強制終了した場合，HiRDB Datareplicator 側の抽出中のシステムログファイルが別のファイルに移った（抽出完了のファイルが一つ以上できた）ことを確認してからユニットを再開始してください。強制終了した直後に HiRDB のユニットを再開始しても，スワップ先にできる状態のファイルがないと，HiRDB は異常状態と認識して再度強制終了します。
- このオペランドに continue を指定し，HiRDB Datareplicator 連携を中止した場合，HiRDB Datareplicator 側ではデータ連動の対象となっている抽出側データベースと反映側データベースとの間で不整合が生じるため，反映側データベースの再作成が必要になります。

174）　pd_rpl_func_control = BACKWARD_CUTOFF_UPDATE | NONE

～<識別子>

HiRDB Datareplicator 連携機能を使用する場合の，抽出側 HiRDB で実行する機能を指定します。

BACKWARD_CUTOFF_UPDATE：

BLOB データ，BINARY データの後方削除更新を行います。BLOB データ，BINARY データの後方削除更新については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

NONE：

BLOB データ，BINARY データの後方削除更新を行いません。

《指定値の目安》

- BACKWARD_CUTOFF_UPDATE の指定可否を次に示します。

ユニット制御情報定義の pd_rpl_hdepath オペランドの指定	抽出側 HiRDB Datareplicator のバージョン	BACKWARD_CUTOFF_UPDATE の指定
あり	08-01 以降	※ 1
	08-00 以前	不可※ 2
なし	－	可能

（凡例）

－：該当しません。

注※ 1　マニュアル「HiRDB Datareplicator Version 8」の「BLOB 型と BINARY 型を使った表のデータ連動」を参照してください。

注※ 2　このオペランドに NONE を指定してください。

上記の条件を満たしていない場合は，BACKWARD_CUTOFF_UPDATE を指定しないでください。指定しても，抽出側 DB での抽出対象表に BLOB データ，BINARY データの後方削除更新の結果は反映側 DB に反映されません。

- pd_rpl_hdepath オペランドを指定していて，かつ，抽出側 Datareplicator のバージョンが 08-00 以前の場合は，このオペランドに NONE を指定してください。

《注意事項》

- 識別子は大文字と小文字のどちらでも指定できます。
- 同じ識別子を 2 回以上指定しても，1 回だけ指定した場合と同じ結果になります。
- BACKWARD_CUTOFF_UPDATE を指定した後に NONE を指定する場合は，変更する前にすべての更新ログが反映し終わっているかどうかを確認してください。反映されていない更新ログがある状態でオペランドの指定値を変更すると，抽出側 DB と反映側 DB で SQL の実行数が一致しないなどの問題が発生します。
- BACKWARD_CUTOFF_UPDATE を指定した状態でストアドプロシジャ，ストアドファンクション，又はトリガを定義した後に NONE を指定する場合は，それぞれ ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER を再作成してください。再作成しないと，ストアドプロシジャ，ストアドファンクション，又はトリガの実行時にエラーとなります。ただし，BLOB データ，BINARY データの後方削除更新を行っていない場合は再作成する必要はありません。ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。
- NONE を指定した状態でストアドプロシジャ，ストアドファンクション，又はトリガを定義した後に BACKWARD_CUTOFF_UPDATE を指定する場合は，それぞれ ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER を再作成してください。再作成しなくてもエラーにはなりませんが，BLOB データ，BINARY データの後方削除更新は行いません。

2.2.33　JP1 との連携に関するオペランド

175）　pd_jp1_use = Y | N

HiRDB の開始，終了などのイベントを JP1 に通知するかどうかを指定します。通知できる HiRDB のイベントについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

Y：HiRDB のイベントを JP1 に通知します。

N：HiRDB のイベントを JP1 に通知しません。

《注意事項》

Windows (x64)では，JP1 へのイベント通知をサポートしていません。Windows (x64)の場合にこのオペランドに Y を指定すると，HiRDB 開始時に KFPS04605-W メッセージが出力されます。

176）　pd_jp1_event_level = 1 | 2

JP1 に通知する HiRDB のイベントの種類（基本属性又は拡張属性）を指定します。基本属性及び拡張属性については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

1：基本属性だけを通知します。

2：基本属性及び拡張属性を通知します。

《前提条件》

このオペランドは，pd_jp1_use オペランドに Y が指定されている場合に有効です。

《指定値の目安》

- 拡張属性を通知できるのは HiRDB のバージョンが 07-02 以降の場合だけです。
- Windows (x64)の場合，このオペランドを指定できません。

177）　pd_jp1_event_msg_out = Y | N

JP1 にメッセージ出力イベントを通知するかどうかを指定します。

Y：メッセージ出力イベントを通知します。

N：メッセージ出力イベントを通知しません。

《前提条件》

このオペランドは，pd_jp1_use オペランドに Y が指定されている場合に有効です。

《指定値の目安》

HiRDB が出力するメッセージの量が多い場合，JP1 へのイベント通知回数を減らしたいときに指定してください。

2.2.34　ディレクトリサーバ連携機能に関するオペランド

178）　pd_directory_server = sods

ディレクトリサーバ連携機能を使用する場合にこのオペランドを指定します。この機能を使用すると，HiRDB を含めたシステムのユーザ情報をディレクトリサーバ（Sun Java System Directory Server）が一元管理し，HiRDB への接続ユーザの認証をディレクトリサーバが行います。ディレクトリサーバ連携機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

sods：Sun Java System Directory Server と連携する場合に指定します。

《前提条件》

- ディレクトリサーバ連携機能を使用する場合は HiRDB LDAP Option が必要です。
- HiRDB LDAP Option の稼働 OS は Windows 2000 又は Windows 2003（Windows Server 2003 (IPF)を除く）になります。
- 64 ビットモードの HiRDB では，ディレクトリサーバ連携機能を使用できません。

2.2.35　HiRDB External Data Access 機能に関するオペランド

HiRDB External Data Access 機能については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

なお，HiRDB の稼働プラットフォームによっては HiRDB External Data Access 機能を使用できません。詳細については，マニュアル「HiRDB Version 8 解説」の「HiRDB External Data Access」を参照してください。

179）　pd_max_foreign_server = 外部サーバ数の最大値

～<符号なし整数>((1～512))《1》

接続する外部サーバの最大数を指定します。

《前提条件》

HiRDB External Data Access が必要です。

《指定値の目安》

このオペランドに必要以上の値を指定すると，共用メモリを余分に使用します。使用する共用メモリが多いと HiRDB を開始できないことがあります。現在接続している外部サーバ数に，今後追加が予想される数を加えた値を指定してください。

2.2.36　OLTP に関するオペランド

180）　pd_oltp_holdcr = use | nouse

OLTP 環境下の UAP で，ホールダブルカーソル機能を使用するかどうかを指定します。

use：OLTP 環境下の UAP でホールダブルカーソル機能を使用します。

nouse：OLTP 環境下の UAP でホールダブルカーソル機能を使用しません。

《指定値の目安》

OLTP 環境下の UAP でホールダブルカーソル機能を使用する場合だけ，このオペランドに use を指定してください。

《注意事項》

OLTP 環境下の UAP でホールダブルカーソル機能を使用する場合，特定の条件を満たす UAP を使用しなければなりません。ホールダブルカーソルについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

2.2.37　バージョンアップに関するオペランド

181）　pd_auto_vrup = Y | N

HiRDB をバージョンアップするときに実行する pdvrup コマンドを自動起動するかどうかを指定します。HiRDB のバージョンアップ方法については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

Y：自動起動します。

N：自動起動しません。HiRDB 管理者が pdvrup コマンドを入力します。

182）　pd_sysdef_default_option = recommendable | v6compatible | v7compatible

HiRDB ではバージョンごとにオペランドの省略値を変更しています。以前のバージョンの省略値を適用する場合にこのオペランドを指定します。

recommendable：最新バージョンの省略値を適用します。

v6compatible：HiRDB Version 6 以前の省略値を適用します。

v7compatible：HiRDB Version 7 の省略値を適用します。

《指定値の目安》

通常はこのオペランドを省略してください。バージョンアップに伴う省略値の変更によって不都合が発生する場合にこのオペランドを指定します。対象になるオペランドについては，「1.5(1)　バージョンによって省略値が異なるオペランド」を参照してください。

2.2.38　通信処理に関するオペランド

183)　pd_service_port = スケジューラプロセスのポート番号

～<符号なし整数>((5001～65535))

次の場合に，スケジューラプロセスのポート番号を指定します。

- 高速接続機能を使用する場合

  高速接続機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

- HiRDB サーバ側にファイアウォールや NAT が設置されている場合

  この場合の HiRDB の環境設定については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合，ユニット制御情報定義では必ずこのオペランドを指定してください。また，各ユニットで異なるポート番号を指定してください。同じポート番号を指定すると，系切り替え時発生時にどちらかのユニットで系切り替えが失敗します。
- ユニット制御情報定義とシステム共通定義の両方でこのオペランドを指定した場合，ユニット制御情報定義の指定値が有効になります。
- HiRDB のポート番号（pd_name_port オペランド又は pdunit オペランドの-p オプションに指定したポート番号）と同じポート番号を指定した場合，このオペランドの指定値が無効となり，高速接続機能を使用する場合及び HiRDB サーバ側にファイアウォールや NAT が設定されている場合の運用ができなくなります。このとき，KFPS00860-W メッセージが出力されます。
- 高速接続機能を使用して pd_max_users オペランドの値以上の接続要求を同時に発行した場合，接続要求をメッセージキューから取り出すフロントエンドサーバやシングルサーバの稼働数が不足します。そのため，接続要求をメッセージキューから取り出せなくなり，メッセージキュー監視機能によってユニットが停止することがあります。pd_max_users オペランドの値以上の同時接続要求が発生しないようにしてください。メッセージキュー監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- スケジューラプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_scd_port
- pdunit -s

184)　pd_name_fixed_port_lookup = Y | N

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

ユニット間のシステムサーバと通信する場合のアドレス解決方法を指定します。ここで，アドレス解決とは，システムサーバの IP アドレスとポート番号を取得する処理をいいます。

Y：自ユニットの共用メモリでアドレス解決をします。

N：全ユニットに通信をして（ブロードキャスト）アドレス解決をします。

このオペランドに Y を指定する場合，次に示すオペランドを指定して，ユニット間通信をするシステムサーバのポート番号を指定する必要があります。ポート番号を指定しない場合，HiRDB を起動できません。ポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

ポート種別	オペランド
HiRDB のポート	pd_name_port オペランド又は pdunit オペランドの-p オプション
トランザクションサーバプロセスのポート	pd_trn_port オペランド又は pdunit オペランドの-t オプション
スケジューラプロセスのポート	pd_scd_port オペランド又は pdunit オペランドの-s オプション
ユニット監視プロセスのポート※1	pd_alv_port オペランド又は pdunit オペランドの-a オプション
メッセージログサーバプロセスのポート※2	pd_mlg_port オペランド又は pdunit オペランドの-m オプション

注※1　シングルサーバ及びシステムマネジャがないユニットでは，ユニット監視プロセスが起動しないため，ポート番号を指定する必要はありません。

注※2　pd_mlg_msg_log_unit オペランドに local を指定した場合，ブロードキャストが発生しないため，ポート番号を指定する必要はありません。また，pd_mlg_msg_log_unit オペランドに manager を指定した場合，システムマネジャがないユニットでは，メッセージログサーバプロセスが起動しないため，ポート番号を指定する必要はありません。

《利点》

このオペランドに Y を指定し，ユニット間通信をするシステムサーバのポート番号を指定すると，アドレス解決時にブロードキャストが発生しません。そのため，ユニット数が多い場合に発生する次のような問題を避けられます。

- 通信資源の圧迫（ネットワーク負荷，ポートの消費）
- 性能の劣化

《注意事項》

- このオペランドに N を指定すると，アドレス解決時に全ユニットに対するブロードキャストが発生することがあります。ブロードキャストが発生する場合を次に示します。
  - ・最初のアドレス解決時
  - ・アドレス解決したシステムサーバがダウンした後のアドレス解決時
  - ・アドレス解決したシステムサーバのユニットがダウンした後のアドレス解決時
- HiRDB/シングルサーバ又は一つのユニットだけで構成される HiRDB/パラレルサーバでは，ユニット間通信が発生しないため，このオペランドの指定有無に関係なく N を指定したものとして動作します。

クライアントの接続形態は高速接続を使用してください。通常接続を使用した場合，クライアント接続時に一時的にブロードキャストが発生します。クライアントの接続形態については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

185）　pd_scd_port = スケジューラプロセスのポート番号

～<符号なし整数>((5001～65535))

次の場合に，スケジューラプロセスのポート番号を指定します。

- pd_name_fixed_port_lookup オペランドに Y を指定した場合
- 高速接続機能を使用する場合

  高速接続機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。
- HiRDB サーバ側にファイアウォールや NAT が設置されている場合

  この場合の HiRDB の環境設定については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《注意事項》

- ユニットごとに異なるポート番号を指定する場合は，pdunit オペランドの-s オプションを指定してください。
- HiRDB のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは次に示すオペランドと関連があります。

  ・pd_service_port

  ・pdunit -s

186）　pd_trn_port = トランザクションサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，トランザクションサーバプロセスのポート番号を指定します。

《注意事項》

トランザクションサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pdunit オペランドの-t オプションと関連があります。

187）　pd_mlg_port = メッセージログサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，メッセージログサーバプロセスのポート番号を指定します。

《注意事項》

- メッセージログサーバプロセスが起動しないユニットではポート番号は使用されませんが，定義したポート番号がほかのポート番号と異なるかどうかはチェックされます。メッセージログサーバプロセスが起動しないユニットは，二つ以上のユニットで構成される HiRDB/パラレル

サーバで pd_mlg_msg_log_unit の指定値が manager の場合，システムマネジャがないユニットです。

- メッセージログサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pdunit オペランドの-m オプションと関連があります。

188）pd_alv_port = ユニット監視プロセスのポート番号

～<符号なし整数>((5001～65535))

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，ユニット監視プロセスのポート番号を指定します。

《注意事項》

- ユニット監視プロセスが起動しないユニットではポート番号は使用されませんが，定義したポート番号がほかのポート番号と異なるかどうかはチェックされます。ユニット監視プロセスは，二つ以上のユニットで構成される HiRDB/パラレルサーバでシステムマネジャがあるユニットを除いて，すべてのユニットで起動しません。
- ユニット監視プロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pdunit オペランドの-a オプションと関連があります。

189）pd_change_clt_ipaddr = 0 | 1

HiRDB サーバから HiRDB クライアントへの通信時に使用するネットワークを指定します。通常，このオペランドを指定する必要はありません。

0：

HiRDB サーバから HiRDB クライアントへの通信時，クライアント環境定義の PDCLTRCVADDR オペランドに指定した IP アドレスがあるネットワークを使用します。PDCLTRCVADDR オペランドを省略した場合は標準ホストの IP アドレスが仮定されます。

1：

HiRDB サーバから HiRDB クライアントへの通信時，HiRDB クライアントから HiRDB サーバへの通信に使用したネットワークを使用します。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は 0 になります。

190）pd_registered_port = "ポート番号の予約範囲"〔，"ポート番号の予約範囲"〕…

～〈文字列〉

HiRDB 予約ポート機能を使用する場合に，HiRDB が通信処理で使用するポート番号の範囲を指定します。

HiRDB 予約ポート機能は，サーバ間通信のときだけ有効になります。なお，ポート番号の使用量が少ない場合には，HiRDB 予約ポート機能を使用する必要はありません。

《オペランドの指定方法》

6000～8999，12500～29999，30500～44999 の計 35,000 個の範囲でポート番号を使用する場合の例を次に示します。

set pd_registered_port = "6000：8999"，"12500：29999"，"30500：44999"

《利点》

HiRDB サーバ間の通信処理で使用するポート番号は OS が自動的に割り当てています。通信処理が大量に発生すると，ポート番号の不足によって処理が中断したり，又はほかのプログラムの通信処理に影響を与えたりします。

このオペランドで HiRDB 専用のポート番号の範囲を指定すると，これらの現象を防げます。

《指定値の目安》

- 指定できるポート番号の範囲は 5001～49151 です。
- HiRDB が使用するポート数については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- %windir%¥system32¥drivers¥etc¥services（DNS 環境の場合は定義した場所）には，このオペランドで指定したポート番号は登録しないでください。pd_registered_port_check オペランドに Y，C，又は W を指定すると，%windir%¥system32¥drivers¥etc¥services との重複チェックをします。

《指定値のチューニング方法》

次に示す計算式で求めた値が，必要となるポート番号の総数の目安です。

a + b + 100

a：HiRDB 予約ポートの使用数（#OF REGISTERED PORTS）

b：HiRDB 予約ポートオーバ時の OS 自動割り当てポートの使用数（#OF ASSIGNED PORTS）

これらの情報は，統計解析ユティリティのシステムの稼働に関する統計情報で取得できます。

《オペランドの規則》

- 予約範囲は最大 10 個指定できます。
- 予約範囲を二つ以上指定した場合，範囲に含まれるポート番号が重複しないようにしてください。
- 終了ポート番号は開始ポート番号以上の値でなければいけません。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニット制御情報定義では必ずこのオペランドを指定してください。また，各ユニットで異なるポート番号を指定してください。同じポート番号を指定すると，系切り替え時発生時にどちらかのユニットで系切り替えが失敗します。
- HiRDB 予約ポート機能を使用できないポート番号では，このオペランドで指定したポート番号は使用されません。HiRDB 予約ポート機能の対象範囲については，pd_registered_port_level オペランドを参照してください。
- HiRDB 予約ポート機能を使用する場合のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- ユニット制御情報定義とシステム共通定義の両方でこのオペランドを指定した場合，ユニット制御情報定義の指定値が有効になります。

191）　pd_registered_port_check = Y | N | C | W

pd_registered_port オペランドで範囲指定したポート番号と%windir%¥system32¥drivers¥etc¥services（DNS 環境の場合は定義した場所）に登録されたポート番号との重複チェックをするかどうかを指定します。

Y：

重複チェックをします。重複している場合, KFPS00348-E メッセージを出力して HiRDB の開始処理を中断します。

N：

重複チェックをしません。

C：

重複チェックをします。重複している場合, 重複しているポート番号は HiRDB 予約ポート機能の対象外になります。

W：

重複チェックをします。重複している場合，KFPS00354-W メッセージを出力します。重複しているポート番号は HiRDB 予約ポート機能の対象外になります。

《前提条件》

pd_registered_port オペランドを指定している必要があります。

《指定値の目安》

- ポート番号の重複があると，不正電文の受信や送信電文の未達など HiRDB の通信処理に影響を及ぼすことがあります。
- Y，C，又は W を指定すると，DNS 環境のときにプロセスサーバプロセスの起動が遅くなることがあります。

192）　pd_registered_port_level = 0 | 1

このオペランドでは，HiRDB 予約ポート機能の対象範囲を指定します。

《前提条件》

pd_registered_port オペランドを指定している必要があります。

《指定値の目安》

0 を指定した場合と 1 を指定した場合の対象範囲を次に示します。

HiRDB が使用するポート番号		pd_registered_port_level の値	
		0	1
HiRDB サーバが使用するポート番号	サーバ間通信用送信ポート番号	○	○
	サーバ間通信用受信ポート番号	○	○
	pd_name_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-p オプションに指定したポート番号	×	×
	pd_service_port オペランドに指定したポート番号	×	×
	pd_scd_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-s オプションに指定したポート番号	×	×

HiRDB が使用するポート番号		pd_registered_port_level の値	
		0	1
	pd_trn_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-t オプションに指定したポート番号	×	×
	pd_mlg_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-m オプションに指定したポート番号	×	×
	pd_alv_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-a オプションに指定したポート番号	×	×
	コマンド用送信ポート番号	×	×
	コマンド用受信ポート番号	×	×
	対クライアント用送信ポート番号	×	○
	対クライアント用受信ポート番号	○	○
	連絡用ポートのサービスポート番号	×	×
	排他制御で使用するポート番号	×	×
HiRDB クライアントが使用するポート番号	送信ポート番号	×	×
	受信ポート番号	×	×

（凡例）

○:HiRDB 予約ポート機能の対象範囲となります。pd_registered_port オペランドで指定したポート番号が使用されます。

×：HiRDB 予約ポート機能の対象範囲となりません。

193）　pd_ipc_send_retrycount = プロセス間通信の送信リトライ処理の連続繰り返し回数～<符号なし整数>((1～32767))《200》（単位：回数）

プロセス間通信の送信リトライ処理の連続繰り返し回数を指定します。

このオペランドは pd_ipc_send_retrysleeptime オペランドと関連しています。

（例）

- pd_ipc_send_retrycount = 500
- pd_ipc_send_retrysleeptime = 2

上記のようにオペランドを指定すると，送信リトライ処理 1 回の繰り返し単位が 500 回連続実行で 2 秒スリープとなります。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- このオペランドの指定値を大きくすると CPU 占有率が高くなります。

194）　pd_ipc_send_retrysleeptime = プロセス間通信の送信リトライ処理のスリープ時間～<符号なし整数>((0～60))《0》（単位：秒）

プロセス間通信の送信リトライ処理のスリープ時間を指定します。

このオペランドは pd_ipc_send_retrycount オペランドと関連しています。

(例)

- pd_ipc_send_retrycount = 500
- pd_ipc_send_retrysleeptime = 2

上記のようにオペランドを指定すると，送信リトライ処理 1 回の繰り返し単位が 500 回連続実行で 2 秒スリープとなります。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- このオペランドの指定値を大きくすると送信完了時間が長くなります。

195）pd_ipc_send_count = サーバ間の送信処理のリトライ回数

～<符号なし整数>((1～32767))《60》

サーバ間の送信処理が終了するまでの送信処理のリトライ回数を指定します。なお，1 回で最大 5 秒間のリトライを続けます。省略値では 5 秒×60 回となり 300 秒間リトライを続けます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。送信タイムアウトが頻発する場合に，このオペランドの指定値を大きくしてください。

196）pd_ipc_recv_count = サーバ間の受信処理のリトライ回数

～<符号なし整数>((1～32767))《120》

サーバ間の受信処理が終了するまでの受信処理のリトライ回数を指定します。なお，1 回で最大 5 秒間のリトライを続けます。省略値では 5 秒×120 回となり 10 分間リトライを続けます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

197）pd_ipc_conn_nblock = Y | N

HiRDB のサーバ間通信（ユニット間の通信）で，コネクション確立をノンブロックモードで行うかどうか指定します。

Y：コネクション確立をノンブロックモードで行います。

N：コネクション確立をブロックモードで行います。

《利点》

このオペランドに Y を指定した場合，HiRDB のサーバ間通信（ユニット間の通信）で，connect() システムコールをノンブロックモードで発行できます。それによって，LAN 障害時に connect() システムコールが数十秒間（OS 依存）プロセスをブロックしていたのが，pd_ipc_conn_nblock_time オペランドで指定した時間まで短くなります。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_ipc_conn_nblock_time
- pd_ipc_conn_interval
- pd_ipc_conn_count
- pd_host_watch_interval

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《備考》

クライアントとサーバ間の通信でのコネクション確立については，クライアント環境定義 PDNBLOCKWAITTIME で指定してください。

198）　pd_ipc_conn_nblock_time = ノンブロックでのコネクション確立監視時間～<符号なし整数>((1～120))《8》（単位：秒）

HiRDB のサーバ間通信（ユニット間の通信）におけるノンブロックモードでのコネクション確立処理のリトライ時間と，コネクションの確立監視時間を指定します。

《前提条件》

このオペランドは，pd_ipc_conn_nblock オペランドに Y を指定した場合に有効になります。

《利点》

不当な通信エラーが多発している場合に，このオペランドの指定値を大きくすると，通信エラーを減らせることがあります。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

不当な通信エラーが多発している場合は，OS の ping コマンドなどを使用して，HiRDB のサーバ間の到達時間を計ってください。このオペランドの値より時間が掛かる場合は，到達時間より大きい値をこのオペランドに指定してください。

《注意事項》

このオペランドの指定値を大きくすると，通信できなくなった場合のタイムアウト時間が延びます。このため，ほかの処理に影響を与えることがあります。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_ipc_conn_nblock
- pd_ipc_conn_interval
- pd_ipc_conn_count

《備考》

HiRDB は一定間隔（pd_host_watch_interval オペランドの値：省略値は 10 秒）でほかのユニットに対して通信を行い，ほかのユニットの稼働状況を確認しています。

監視対象ユニットが応答を返した直後に LAN 障害が発生すると，一定時間（pd_host_watch_interval オペランドの値）待った後に次の通信処理をします。その後，通信のリトライ処理が pd_ipc_conn_nblock_time オペランドの値（デフォルト 8 秒）の間実行されます。そのため，監視対象ユニットの異常終了が検知されるまでの時間は，両オペランドの値を加算した時間になります。

HiRDB が監視対象ユニットの異常終了を検知するまでの時間の関係を次に示します。

199）　pd_ipc_conn_interval = コネクション確立処理のリトライ間隔

～<符号なし整数>((0～50))《1》

HiRDB のサーバ間通信で，データ送信時に行うコネクション確立処理のリトライ間隔を指定します。次に示す場合にコネクション確立処理をリトライします。

- 接続タイムアウトになった場合
- Listen キューが不足した場合

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- このオペランドを指定する場合は，次の条件を満たすようにしてください。

  このオペランドの指定値< pd_ipc_conn_nblock_time 及びクライアント環境定義の PDNBLOCKWAITTIME の値

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_ipc_conn_count
- pd_ipc_conn_nblock
- pd_ipc_conn_nblock_time

《備考》

ノンブロックモード及びブロックモード通信時のリトライ処理の例を次に示します。

・ノンブロックモード通信時のリトライ処理（pd_ipc_conn_nblock = Yの場合）

〔説明〕

次に示すように指定すると，ノンブロックモードのコネクション確立処理を，1 秒間隔で 8 秒間リトライします。

・pd_ipc_conn_nblock_time = 8

・pd_ipc_conn_interval = 1

・ブロックモード通信時のリトライ処理（pd_ipc_conn_nblock = Nの場合）

〔説明〕

次に示すように指定すると，ブロックモードでのコネクション確立処理を 10 秒間隔で 5 回リトライします。

・pd_ipc_conn_interval = 10

・pd_ipc_conn_count = 5

200）　pd_ipc_conn_count =コネクション確立処理のリトライ回数

～<符号なし整数>((0～600))《50》

HiRDB のサーバ間通信で，データ送信時に行うコネクション確立処理のリトライ回数を指定します。次に示す場合に，コネクション確立処理をリトライします。

- 接続タイムアウトになった場合
- Listen キューが不足した場合

このオペランドは pd_ipc_conn_interval オペランドと関連しています。

（例）

- pd_ipc_conn_interval = 10
- pd_ipc_conn_count = 5

このようにオペランドに指定すると，コネクション確立処理を 10 秒ごとに最大 5 回行います。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《注意事項》

ノンブロックモードのコネクション確立処理が実行される場合，このオペランドの指定は無効になります。

pd_ipc_conn_nblock オペランドに Y を指定すると，ノンブロックモードのコネクション確立処理が実行されます。

201）　pd_ipc_inet_bufsize = サーバ間通信で使用する送受信バッファ長

～<符号なし整数>((0～262144))《16384》（単位：バイト）

サーバ間通信（TCP INET ドメイン）で使用する送受信バッファの最大値を 4096 の倍数で指定します。

《指定値の目安》

通常は 0 を指定して OS の設定値を適用してください。

《注意事項》

- 0 を指定した場合は OS の設定値を使用します。
- TCP のバッファサイズの上限値は各 OS で異なるので注意してください。
- システム定義 pd_utl_buff_size を使用するユティリティの通信処理では，指定したバッファ長を無効にし，pd_utl_buff_size 指定値を元に，最適な値を設定します。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_utl_buff_size

202）　pd_tcp_inet_bufsize = HiRDB クライアントとの通信で使用する送受信バッファ長

～<符号なし整数>((0～262144))《0》（単位：バイト）

HiRDB クライアントとの通信（TCP INET ドメイン）で使用する送受信バッファの最大値を，4096 の倍数で指定します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。次に示す現象が発生した場合に，このオペランドを指定します。

- HiRDB サーバと HiRDB クライアント間で送受信するデータの再送がある場合，このオペランドを指定すると再送を抑止できます。
- 受信側の処理で，データ到着速度が読み込み処理速度よりも遅い場合，このオペランドを指定すると，送信データ量が増えて受信時の到着待ちによる遅延を防げます。

このオペランドには，通信に使用するバッファサイズと同じ値を指定してください。

《注意事項》

- 0 を指定した場合は OS の設定値を使用します。
- TCP のバッファサイズの上限値は各 OS で異なるので注意してください。

203)　pd_utl_buff_size = ユティリティの通信用バッファ長
～<符号なし整数>((8～64))《32》(単位：キロバイト)

次に示すユティリティのプロセスが使用する通信用バッファ 1 面のサイズをキロバイト単位で指定します。

- データベース作成ユティリティ
- データベース再編成ユティリティ（-g オプション指定時）
- リバランスユティリティ
- データベース回復ユティリティ
- データベース複写ユティリティ

《利点》

ユティリティ処理を行うサーバプロセスと HiRDB サーバ間の通信処理で，データの再送が見られる場合に指定値を大きくして HiRDB サーバ間の通信回数を減らすことができます。

《指定値の目安》

通信用バッファ長＝↑(ページ長※×2)÷1024↑（単位：キロバイト）

注※　ユティリティの処理対象となる RD エリアの最大ページ長

《オペランドの規則》

- 指定値は，4 の倍数単位に指定してください。4 の倍数でない場合は，自動的に 4 の倍数に切り上げられます。
- 指定値が 32 より小さい場合でも，データベース回復ユティリティ及びデータベース複写ユティリティ実行時には，自動的に 32 に切り上げられます。

204)　pd_utl_file_buff_size = ユティリティのファイル用バッファ長
～<符号なし整数>((32～512000))《32》(単位：キロバイト)

ユティリティがファイルにアクセスするとき，入出力に使用するバッファ長を指定します。このオペランドに大きい値を指定すると，ファイルへの入出力回数を削減できます。

数百ギガバイトのファイルにデータを入出力する場合，使用する OS やハードウェアの特性によって，入出力回数が増え，入出力に掛かる時間が長くなることがあります。この場合，バッファ長に 1 メガバイト程度を指定すると回避できることがあります。

このオペランドでバッファ長の指定が有効となるユティリティ名とそのファイル名を次に示します。

ユティリティ名	ファイル名
データベース複写ユティリティ	バックアップファイル
データベース回復ユティリティ	バックアップファイル

ユティリティ名	ファイル名
データベース再編成ユティリティ	アンロードデータファイル※1 LOB データアンロードファイル 表移行用アンロードファイル
データベース作成ユティリティ	入力データファイル※2 列単位 LOB 入力ファイル 表移行用アンロードファイル

注※1

-W オプションを指定して作成されたアンロードデータファイルの場合，バッファサイズは有効となりません。

注※2

DAT 形式の入力データファイルで，maxreclen オペランドを指定した場合，maxreclen オペランドの指定値が優先されます。この場合は，maxreclen オペランドの指定値を大きくしてください。

205）　pd_sql_send_buff_size = SQL 実行時のサーバ間通信用バッファ長

～<符号なし整数>((4～32))《4》（単位：キロバイト）

このオペランドは，HiRDB/パラレルサーバ限定のオペランドです。

HiRDB サーバ間で検索結果のデータ転送をする場合の通信用バッファ 1 面の長さをキロバイト単位で指定します。

《指定値の目安》

通信用バッファ長＝↑ページ長※÷1024↑（単位：キロバイト）

注※　ページ長＝行長× 1 回当たりの通信件数

- 可変長データ型のデータ長は実長で計算してください。
- LOB 列及び抽象データ型列は対象外のため，ページ長には含めないでください。

《注意事項》

- このオペランドの指定値を大きくすると，メモリ使用量が増えることがあります。
- インデクスなどを利用して，フロータブルサーバを用いたソート及びそのための作業表作成処理を削減するために，指定値以上の通信用バッファ長で通信を行うことがあります。

2.2.39　Java™ に関するオペランド

Java ストアドプロシジャ又は Java ストアドファンクションを使用する場合に，Java に関するオペランドを指定します。Java ストアドプロシジャ及び Java ストアドファンクションについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

206）　pd_java_option = "Java オプション"〔，"Java オプション"〕…

～<文字列>

Java 仮想マシンの起動オプションを指定します。起動オプションについては，Java 仮想マシンのドキュメントを参照してください。

《オペランドの規則》

- 一つの Java オプションの文字列長は 255 文字以内とします。
- 全 Java オプションの文字列長の合計は 1024 文字以内とします。

- Java オプションに " を指定できません。
- Java オプションは最大 20 個指定できます。

《オペランドの指定方法》

Java 仮想マシンが使用するヒープサイズの初期値を 32 メガバイト，ヒープサイズの最大値を 64 メガバイトに設定する場合の例を次に示します。

pd_java_option = "-Xms32m","-Xmx64m"

207）　pd_java_routine_stack_size = 外部 Java ルーチンが使用するスタック領域長

～＜符号なし整数＞((1024～2146435072))《131072》（単位：バイト）

外部 Java ルーチンが使用するスタック領域長をバイト単位で指定します。

《指定値の目安》

- Java オプションで指定したスタック領域長以上の値を指定してください。
- Java 仮想マシンの起動オプションに，Java 仮想マシンのスタック領域長とネイティブメソッドスタック領域長の両方を指定するオプションがある場合，このオペランドにはネイティブメソッドスタック領域長以上の値を指定してください。

《注意事項》

- Java オプションで指定したスタック領域長より小さい値を指定すると，HiRDB が正しく動作しないことがあります。
- 使用スタック領域長がこのオペランドの値を超えると，HiRDB が正しく動作しないことがあります。ただし，Java 仮想マシンがスタックオーバフローを検知した場合は正しく動作します。

208）　pd_java_archive_directory = "JAR ファイル格納ディレクトリ"

～＜パス名＞《%PDDIR%¥java》

Java ストアドプロシジャ又は Java ストアドファンクションで使用する JAR ファイルを格納するディレクトリ名を絶対パス名で指定します。

《注意事項》

- このオペランドを指定する場合，ここで指定するディレクトリを JAR ファイルのインストール前に作成しておく必要があります。
- JAR ファイル格納ディレクトリは，JAR ファイルを格納するための専用のディレクトリです。
- JAR ファイル格納ディレクトリには，インストールした JAR ファイル以外を格納しないでください。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

209）　pd_java_classpath = "Java クラスパス"

～＜パス名＞

Java 仮想マシンで使用するクラスパスを絶対パス名で指定します。

このオペランドに指定したパスに含まれるクラスは，Java ストアドプロシジャ又は Java ストアドファンクションの処理手続きとして実行される Java メソッドから参照できます。

Java ストアドプロシジャ又は Java ストアドファンクションの外部ルーチン名として指定した JAR ファイルとこのオペランドに指定したパスに同じ名称のクラスがある場合，このオペランドに指定したパスが優先されます。

《オペランドの規則》

- パス名は 1024 文字以内とします。
- パス名の大文字と小文字の区別をしません。

210）　pd_java_runtimepath = "Java Runtime Environment のルートディレクトリ"
～<パス名>《%PDDIR%¥jre》

Java Runtime Environment のルートディレクトリを絶対パス名で指定します。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

《注意事項》

JRE（Java Runtime Environment）の同梱を廃止したため，HiRDB を 07-03 より前のバージョンから 07-03 以降にバージョンアップする場合，このオペランドを追加するか，又はこのオペランドの指定値を変更してください。バージョンアップの注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「Java ストアドプロシジャ及び Java ストアドファンクションを使用する場合」を参照してください。

211）　pd_java_libpath = "Java 仮想マシンのライブラリディレクトリ"
～<パス名>

Java 仮想マシンのライブラリが格納されているディレクトリを，Java Runtime Environment のルートディレクトリ（pd_java_runtimepath オペランドの値）への相対パス名で指定します。

《オペランドの省略値》

このオペランドの省略値を次に示します。

OS の種類	省略値
Windows（下記以外）	bin¥hotspot，又は bin¥client※
Windows (IPF)	bin¥server
Windows (x64)	

注※

bin¥hotspot に jvm.dll がない場合は，bin¥client を仮定します。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

212）　pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"
～<パス名>

Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合, そのファイル名を絶対パス名で指定します。このオペランドを省略すると, Java 仮想マシンでの標準出力又は標準エラー出力が無視されます。

《指定値の目安》

このオペランドで指定するファイルの容量が非常に大きくなるため，通常はこのオペランドを指定しないでください。Java ストアドプロシジャ又は Java ストアドファンクションのデバッグ時に指定することをお勧めします。なお，このオペランドで指定するファイルの容量制限はありません。

《注意事項》

複数プロセスからの同時書き込みがあった場合は，その出力内容を保証しません。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

2.2.40　外部 C ストアドルーチンに関するオペランド

213）　pd_c_library_directory = "C ライブラリファイル格納ディレクトリ"
～<パス名>《%PDDIR%¥clib》

C ライブラリファイルを格納するディレクトリを絶対パス名で指定します。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

《注意事項》

このオペランドを指定する場合，ここで指定するディレクトリを C ライブラリファイルの登録前に作成しておく必要があります。

2.2.41　文字コードに関するオペランド

214）　pd_substr_length = 3 | 4 | 5 | 6

1 文字を表現する最大バイト数を指定します。このオペランドは，文字コードに Unicode（UTF-8）を指定している場合に有効で，スカラ関数 SUBSTR の結果の長さに影響します。

スカラ関数 SUBSTR については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。

《指定値の目安》

UCS-2 の範囲の文字（1～3 バイト）だけを使用する場合は，このオペランドを省略できます。
UCS-4 の範囲の文字（1～6 バイト）を使用する場合は，1 文字を表現する最大バイト数を指定します。

ただし，実際に使用している 1 文字を表現するバイト数以上の値を指定すると，スカラ関数 SUBSTR の結果が長くなることがあるため，使用している文字コードに基づいて指定値を検討してください。

《注意事項》

- ISO/IEC 10646 の規格では，1 文字当たり 1～4 バイトの範囲に文字が割り当てられ，5，6 バイトの範囲は将来の規格のために予約されています。HiRDB では 1 文字当たり 1～6 バイトの範囲まで使用できますが，文字が割り当てられていない 5，6 バイトの範囲を使用する場合，将来発生するおそれのある問題については保証できません。
- バージョン 06-02 でサポートした Unicode（UTF-8）は，1 文字を表現する最大バイト数は 3 バイトの範囲まで使用できます。バージョン 08-00 からは，最大 6 バイトの範囲まで使用できます。そのため，データによっては 1 文字を表現するバイト数が大きくなることがあります。
- 次の条件をすべて満たしている場合は，ルーチンの SQL オブジェクトの再作成やビュー表の再定義をする必要があります。
  ・文字コードに Unicode（UTF-8）を指定している

・新規で pd_substr_length オペランドの指定値に 3 以外を指定する，又は既に pd_substr_length オペランドを指定している環境で，ルーチンやビュー表を定義し，かつ pd_substr_length オペランドの指定値を変更する

・混在文字列型（MCHAR，MVARCHAR）を引数に指定したスカラ関数 SUBSTR をルーチンやビュー定義に指定する

ただし，クライアント環境定義の PDSUBSTRLEN オペランドを指定してコネクション単位に制御したり，SQL コンパイルオプションの SUBSTR LENGTH の指定によってルーチン単位で制御したりもできます。

- 文字コードに Unicode（UTF-8）を指定している場合の pd_substr_length オペランドと，そのほかのオペランドの指定値との優先順位を次に示します。

ルーチン単位で制御する場合

制御のタイミング	pd_substr_length オペランド	クライアント環境定義 PDSUBSTRLEN	SQL コンパイルオプション SUBSTR LENGTH	備考
定義時	有効（2）	無効	有効（1）	省略値 pd_substr_length
コール時又は関数呼び出し時	無効（定義時の指定に従う）	無効（定義時の指定に従う）	－	特になし

（凡例）

（ ）：優先順位

－：指定できません。

ビュー表単位で制御する場合

制御のタイミング	pd_substr_length オペランド	クライアント環境定義 PDSUBSTRLEN	SQL コンパイルオプション SUBSTR LENGTH	備考
定義時	有効（2）	有効（1）	－	省略値 pd_substr_length
操作時	無効（定義時の指定に従う）	無効（定義時の指定に従う）	－	特になし

（凡例）

（ ）：優先順位

－：指定できません。

2.2.42　日付・時刻に関するオペランド

215）　pd_leap_second = Y | N

時刻データ，時刻印データ，及び日時書式でうるう秒を使用するかどうかを指定します。

Y：

- 時刻データ又は時刻印データの場合

埋込み変数及び?パラメタの値，並びにそれらを文字列表現した値でうるう秒を使用できます。

- 日付書式の場合
  秒に対する値でうるう秒を使用できます。

N：

うるう秒を使用しません。

《注意事項》

うるう秒が含まれる時刻データ又は時刻印データがデータベースに格納されている場合，このオペランドでNを指定すると，うるう秒を含む時刻データ又は時刻印データを使用した時刻演算を行ったときにエラーとなることがあります。このとき，スカラ関数SECONDではうるう秒を抽出できます。

2.2.43　ユニット構成に関するオペランド

216）pdunit -x ホスト名
　　-u ユニット識別子
　　〔-d "HiRDB運用ディレクトリ名"〕
　　〔-c ホスト名〕
　　〔-p HiRDBのポート番号〕
　　〔-s スケジューラプロセスのポート番号〕
　　〔-t トランザクションサーバプロセスのポート番号〕
　　〔-m メッセージログサーバプロセスのポート番号〕
　　〔-a ユニット監視プロセスのポート番号〕

●HiRDB/シングルサーバの場合

HiRDB/シングルサーバのユニット構成を定義します。

《指定値の目安》

- HiRDB/シングルサーバの場合はこのオペランドを省略できます。
  ただし，IPアドレスを引き継がない系切り替え機能を使用する場合はこのオペランドを省略できません。
- 系切り替え機能を使用する場合は，現用系と予備系のシステム定義の内容を同じにします。
- 相互系切り替え構成の場合，マルチHiRDBの構成（1サーバマシンに二つのHiRDB/シングルサーバが共存する構成）となりますが，二つのHiRDB/シングルサーバでシステム定義の内容を同じにはできません。
- 指定するすべてのpdunitオペランドの-x及び-cオプションに指定するホスト名には，それぞれ異なる名称を指定してください。

-x ホスト名

～<ホスト名>((1～32文字))

HiRDB/シングルサーバを定義するサーバマシンのホスト名を指定します。

IPアドレス又はFQDN形式で指定することもできます。また，ループバックアドレスを指定することもできます。ただし，同じサーバマシンに対する指定形式は統一してください。例えば，同じサーバマシンのホスト名を，ホスト名形式で指定したり，IPアドレス形式で指定したりすることはできません。

指定例については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「DNSサーバでIPアドレスを管理する場合の設定」を参照してください。

ホスト名形式で指定する場合の注意事項

- hostname コマンドを実行して表示されたホスト名を指定してください。
- ホスト名は大文字と小文字を区別します。
- ホスト名には別名を指定できません。

《ほかのオペランドとの関連》

pdunit オペランドの-x オプション，-c オプション，及び pd_hostname オペランドに指定するホスト名の関係を次に示します。

システム構成の条件		pdunit オペランド		pd_hostname オペランドの指定
		-x オプションの指定	-c オプションの指定	
系切り替え機能を使用しない場合		• ホスト名※1を指定 • pdunit オペランドを省略	指定不要	• pd_hostname オペランドを省略 • 標準ホスト名※1を指定 • -x オプションと同じホスト名※1を指定
		ループバックアドレスを指定※2※4	指定不要	• pd_hostname オペランドを省略 • ループバックアドレスを指定
系切り替え機能を使用する場合	モニタモードで IP アドレスを引き継ぐ場合※6	再配置可能な IP アドレス，再配置可能な IP アドレスのホスト名，又は FQDN を指定	指定不要	• pd_hostname オペランドを省略 • 現用系の標準ホスト名※1を指定 • -x オプションと同じホスト名※1を指定
		ループバックアドレスを指定※2※3※4※5	指定不要	• pd_hostname オペランドを省略 • ループバックアドレスを指定
	IP アドレスを引き継がない場合	現用系のホスト名※1を指定	予備系のホスト名※1を指定	現用系の標準ホスト名※1を指定

注

- ここで指定したホスト名は hosts ファイル又は DNS などに登録し，名前解決しておく必要があります。
- ホスト名を FQDN 形式で指定する場合は，ホスト名を FQDN 形式で定義しておく必要があります。
- HiRDB では，localhost を通常のホスト名として扱うため，ループバックアドレスとして使用する場合は，名前解決しておく必要があります。
- pdunit オペランドを省略した場合，-x オプションに標準ホスト名が指定されたと仮定します。この場合，「はじめに」の「■このマニュアルで使用する構文要素記号」で説明している規則に従って標準ホスト名が付けられている必要があります。

  また，pdunit オペランドを省略した場合は標準ホスト名が仮定されるため，標準ホスト名を hosts ファイル又は DNS に登録して名前解決してください。

注※1

指定形式は，ホスト名，IP アドレス，又は FQDN 形式のどれかになります。

注※2

ループバックアドレスを指定する場合，hosts ファイル又は DNS などへのホスト名の登録が不要になります。

注※3

この場合，クラスタソフトウェアの IP アドレス引き継ぎの設定が不要になります。

注※4

この場合，ほかのマシンの HiRDB クライアントから HiRDB サーバに接続できません。HiRDB サーバに接続するには，マルチコネクションアドレス機能又は高速接続機能を使用してください。

注※5

マルチコネクションアドレス機能を使用する場合は，pdstart オペランドの-m オプション（HiRDB クライアントが接続するホスト名を指定するオプション）に，系の切り替え時に引き継ぐ再配置可能な IP アドレスのホスト名を指定してください。

注※6

MSCS 又は MSFC のリソース（ネットワーク名）に登録する仮想ネットワーク名を指定してください。仮想ネットワーク名の登録については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

-u ユニット識別子

～＜識別子＞((4 文字))

ユニット識別子を指定します。ユニット制御情報定義の pd_unit_id オペランドにも，ここで指定したユニット識別子を指定します。

-d "HiRDB 運用ディレクトリ名"

～＜ 200 文字以内のパス名＞《環境変数 PDDIR の値》

HiRDB 運用ディレクトリ名を指定します。

HiRDB 運用ディレクトリ名は大文字と小文字の区別をしません。C:¥hirdb_x と C:¥HIRDB_X は同じ名称として認識されます。

pdunit オペランドを省略すると，-d オプションにインストールディレクトリが指定されたと仮定します。

-c ホスト名

～＜ホスト名＞((1～32 文字))

IP アドレスを引き継がない系切り替えをする場合にこのオプションを指定します。

このオプションには予備系のホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。ただし，同じサーバマシンに対する指定形式は統一してください。例えば，同じサーバマシンのホスト名を，ホスト名形式で指定したり，IP アドレス形式で指定したりすることはできません。

ホスト名形式で指定する場合の注意事項

- ホスト名は大文字と小文字を区別します。
- ホスト名には別名を指定できません。

-p HiRDB のポート番号

～＜符号なし整数＞((5001～65535))

HiRDB/シングルサーバの場合はこのオプションを指定しないでください。

-s スケジューラプロセスのポート番号

～＜符号なし整数＞((5001～65535))

スケジューラプロセスのポート番号を指定します。

《注意事項》

- 通常，このオプションを指定する必要はありません。

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- スケジューラプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_service_port
- pd_scd_port

-t トランザクションサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

トランザクションサーバプロセスのポート番号を指定します。

《注意事項》

- 通常，このオプションを指定する必要はありません。
- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- トランザクションサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pd_trn_port オペランドと関連があります。

-m メッセージログサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

メッセージログサーバプロセスのポート番号を指定します。

《注意事項》

- 通常，このオプションを指定する必要はありません。
- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- メッセージログサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pd_mlg_port オペランドと関連があります。

●HiRDB/パラレルサーバの場合

HiRDB/パラレルサーバのユニット構成を定義します。

《指定値の目安》

- このオペランドには，HiRDB/パラレルサーバを構成するすべてのユニットの情報を定義します。例えば，ユニットが三つある場合は，pdunit オペランドを三つ指定します。
- 系切り替え機能を使用する場合はこのオペランドを省略できません。
- 系切り替え機能を使用する場合は，現用系と予備系のシステム定義の内容を同じにします。

- 相互系切り替え構成の場合，マルチ HiRDB の構成（1 ホストに異なる二つのユニットが共存する構成）となりますが，異なる二つのユニットでシステム定義の内容を同じにはできません。
- 指定するすべての pdunit オペランドの-x 及び-c オプションに指定するホスト名には，それぞれ異なる名称を指定してください。
- HiRDB/パラレルサーバを 1 ユニットだけで構成する場合は，このオペランドを省略できます。省略すると，HiRDB 運用ディレクトリにはインストールディレクトリが仮定されます。

-x ホスト名

～<ホスト名>((1～32 文字))

HiRDB/パラレルサーバの各サーバを定義するサーバマシンのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。ただし，同じサーバマシンに対する指定形式は統一してください。例えば，同じサーバマシンのホスト名を，ホスト名形式で指定したり，IP アドレス形式で指定したりすることはできません。

指定例については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「DNS サーバで IP アドレスを管理する場合の設定」を参照してください。

ホスト名形式で指定する場合の注意事項

- hostname コマンドを実行して表示されたホスト名を指定してください。
- ホスト名は大文字と小文字を区別します。
- ホスト名には別名を指定できません。

《ほかのオペランドとの関連》

pdunit オペランドの-x オプション，-c オプション，及び pd_hostname オペランドに指定するホスト名の関係を次に示します。

システム構成の条件		pdunit オペランド		pd_hostname オペランドの指定
		-x オプションの指定	-c オプションの指定	
系切り替え機能を使用しない場合		ホスト名※1を指定	指定不要	• pd_hostname オペランドを省略 • 標準ホスト名※1を指定
系切り替え機能を使用する場合	IP アドレスを引き継ぐ場合※3	再配置可能な IP アドレス，再配置可能な IP アドレスのホスト名，又は FQDN を指定	指定不要	現用系の標準ホスト名※1を指定
	IP アドレスを引き継がない場合	現用系のホスト名※1を指定	予備系のホスト名※1を指定※2	現用系の標準ホスト名※1を指定

注

- ここで指定したホスト名は hosts ファイル又は DNS などに登録し，名前解決しておく必要があります。
- ホスト名を FQDN 形式で指定する場合は，ホスト名を FQDN 形式で定義しておく必要があります。

注※1

指定形式は，ホスト名，IP アドレス，又は FQDN 形式のどれかになります。

注※2

pdunit オペランドの-x オプション及び-c オプションに指定するホスト名は重複できません。正しい例と間違った例を次に示します。

(正しい例)

```
pdunit -x hostA ... -c hostAA
pdunit -x hostB ... -c hostBB
```

(間違った例)

```
pdunit -x hostA ... -c hostB
pdunit -x hostB ... -c hostA
```

IP アドレスを引き継がない相互系切り替え構成の HiRDB システム定義の例については，「付録 B.4 HiRDB/パラレルサーバの場合：スタンバイ型系切り替え機能使用時」を参照してください。

注※3

MSCS 又は MSFC のリソース（ネットワーク名）に登録する仮想ネットワーク名を指定してください。仮想ネットワーク名の登録については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

-u ユニット識別子

～<識別子>((4 文字))

ユニット識別子を指定します。各ユニット制御情報定義の pd_unit_id オペランドにも，ここで指定したユニット識別子を指定します。

-d "HiRDB 運用ディレクトリ名"

～< 200 文字以内のパス名>《環境変数 PDDIR の値》

このユニットの HiRDB 運用ディレクトリ名を指定します。

HiRDB 運用ディレクトリ名は大文字と小文字の区別をしません。C:¥hirdb_x と C:¥HIRDB_X は同じ名称として認識されます。

《注意事項》

- 1：1 スタンバイレス型系切り替え機能を使用する場合

  正規バックエンドサーバユニットと代替バックエンドサーバユニットで同じ HiRDB 運用ディレクトリ名を指定してください。

- 影響分散スタンバイレス型系切り替え機能を使用する場合

  同じ HA グループに属する全ユニットで，同じ HiRDB 運用ディレクトリ名を指定してください。

- 環境変数 PDDIR は，pdstart -q コマンドを実行したユニットに設定してある環境変数が設定されます。複数ユニットで構成されるシステムの場合に-d オプションを省略するときは，全ユニットで同じ HiRDB

  運用ディレクトリ名を指定してください。

-c ホスト名

～<ホスト名>((1～32 文字))

IP アドレスを引き継がない系切り替えをする場合にこのオプションを指定します。

このオプションには予備系のホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。ただし，同じサーバマシンに対する指定形式は統一してください。例えば，同じサーバマシンのホスト名を，ホスト名形式で指定したり，IP アドレス形式で指定したりすることはできません。

なお，次に示すユニットにはこのオプションを指定できません。指定した場合は HiRDB を開始できません（KFPS01896-E メッセージが出力されます）。

- 1：1 スタンバイレス型系切り替え機能を適用するユニット（IP アドレスの引き継ぎが発生しない）
- 影響分散スタンバイレス型系切り替え機能を適用するユニット（IP アドレスの引き継ぎが発生しない）

ホスト名形式で指定する場合の注意事項

- ホスト名は大文字と小文字を区別します。
- ホスト名には別名を指定できません。

-p HiRDB のポート番号

～<符号なし整数>((5001～65535))

各ユニットでの HiRDB のポート番号を指定します。このオプションを省略した場合は，pd_name_port オペランドで指定した値が仮定されます。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットに同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- HiRDB のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pd_name_port オペランドと関連があります。

-s スケジューラプロセスのポート番号

～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，各ユニットでのスケジューラプロセスのポート番号を指定します。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットに同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- スケジューラプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，次に示すオペランドと関連があります。
  ・pd_service_port

・pd_scd_port

- t トランザクションサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，各ユニットでのトランザクションサーバプロセスのポート番号を指定します。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットに同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- トランザクションサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pd_trn_port オペランドと関連があります。

- m メッセージログサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，各ユニットでのメッセージログサーバプロセスのポート番号を指定します。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットに同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- メッセージログサーバプロセスが起動しないユニットではポート番号を指定しても使用されませんが，定義したポート番号がほかのポート番号と異なるかどうかはチェックされます。メッセージログサーバプロセスが起動しないユニットは，二つ以上のユニットで構成される HiRDB/パラレルサーバで pd_mlg_msg_log_unit の指定値が manager の場合，システムマネジャがないユニットです。
- メッセージログサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pd_mlg_port オペランドと関連があります。

- a ユニット監視プロセスのポート番号

～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，各ユニットでのユニット監視プロセスのポート番号を指定します。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットとで同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- ユニット監視プロセスが起動しないユニットではポート番号を指定しても使用されませんが，定義したポート番号がほかのポート番号と異なるかどうかはチェックされます。ユニット監視プロセスは，二つ以上のユニットで構成される HiRDB/パラレルサーバでシステムマネジャがあるユニットを除いて，すべてのユニットで起動しません。
- ユニット監視プロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pd_alv_port オペランドと関連があります。

2.2.44　サーバ構成に関するオペランド

217）pdstart -t サーバ種別〔-s サーバ名〕
　　-x ホスト名｜-u ユニット識別子
　　〔-m ホスト名〔，ホスト名〕…
　　〔-n ホスト名〔，ホスト名〕…〕〕
　　〔-c サーバ名｜-g HA グループ識別子〕
　　〔-k stls〕

このオペランドは省略できません。必ず指定してください。

《注意事項》

一度設定したサーバ名（-s オプションの指定）を変更する場合は，データベース初期設定ユティリティで再度システムを構築し直す必要があります。したがって，後で変更が発生するようなサーバ名は避けてください。

また，-t，-x，-u，-s，又は-c オプションの指定を変更する場合は，次に示す操作をする必要があります。したがって，後で変更が発生するような指定値は避けてください。

1. システムログファイルの初期化（オプションを変更したサーバのシステムログファイルが初期化の対象になります）
2. シンクポイントダンプファイルの初期化（オプションを変更したサーバのシンクポイントダンプファイルが初期化の対象になります）
3. ユニット用ステータスファイル及びサーバ用ステータスファイルの初期化（オプションを変更したユニット及びサーバのステータスファイルが初期化の対象になります）

●HiRDB/シングルサーバの場合

HiRDB/シングルサーバのサーバ構成を定義します。指定する項目は次のとおりです。

- サーバ種別
- サーバ名
- ホスト名又はユニット識別子

《指定値の目安》

- HiRDB/シングルサーバを複数個接続して使用する場合，サーバ名は各 HiRDB/シングルサーバ間で一意になるように指定してください。

-t サーバ種別

サーバの種別を指定します。

SDS：シングルサーバ

-s サーバ名

～<識別子>((1～8 文字))

シングルサーバのサーバ名を指定します。

サーバ名は大文字と小文字の区別をしません。sds01 と SDS01 は同じ名称として認識されます。

-x ホスト名

～<ホスト名>((1～32 文字))

pdunit オペランドの-x オプションに指定したホスト名を指定します。

pdunit オペランドを省略した場合，pdunit オペランドの-x オプションに標準ホスト名が指定されたと仮定されるため，このオプションに標準ホスト名を指定してください。

-u ユニット識別子

～<識別子>((4 文字))

サーバを実行するユニットの識別子を指定します。pdunit オペランドの-u オプションで指定したユニット識別子を指定します。

系切り替え機能を使用する場合は，「-x ホスト名」ではなく，「-u ユニット識別子」を指定することをお勧めします。

-m ホスト名〔，ホスト名〕…

～<ホスト名>((1～32 文字))

マルチコネクションアドレス機能を使用する場合に，HiRDB クライアントが接続する HiRDB/シングルサーバのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。

マルチコネクションアドレス機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

HiRDB クライアントと HiRDB サーバ間で使用しているネットワークと，HiRDB サーバのサーバマシン間で使用しているネットワークが異なる場合にこのオプションを指定します。また，pd_hostname オペランドにループバックアドレスを指定している場合に，ほかのホストの HiRDB クライアントから HiRDB サーバにアクセスするときは，このオプションを指定する必要があります。

ホスト名形式で指定する場合の注意事項

- ホスト名は大文字と小文字を区別します。
- ホスト名には別名を指定できません。

《注意事項》

- このオプションは四つまで指定できます。
- 同じネットワーク上のホスト名を複数指定した場合，最初に指定したホスト名が有効になります。

- HiRDB/シングルサーバがないサーバマシンのホスト名を指定した場合，その指定は無視されます。
- サブネットによって，HiRDB クライアントの接続用ネットワークと HiRDB サーバ間の通信用ネットワークを切り分けている場合は，マルチコネクションアドレス機能が適用できません。-m オプションを指定した場合，HiRDB クライアントから HiRDB サーバへ接続ができなくなるときがあります。
- -m オプションと-n オプションの指定方法を以下に示します。

HiRDB/シングルサーバの構成		pdstart オペランドの指定方法	
系切り替え構成	系切り替え時に HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレス（クライアントの接続先となる IP アドレス）を引き継ぐ構成	-m オプションの指定	-n オプションの指定
適用	適用	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスに対応するホスト名を指定してください。	オプションの指定を省略してください。
	非適用	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの現用系のホスト名を指定してください。	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの予備系のホスト名を指定してください。
非適用	-	HiRDB クライアントと HiRDB サーバ間で使用するネットワークのホスト名を指定してください。	オプションの指定を省略してください。

（凡例）

-：IP アドレスを引き継ぐ構成の適用対象外

- 系切り替え構成を適用する際は，HiRDB サーバ間で使用するネットワークの IP アドレスを引き継がない場合でも，HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスは引き継ぐ構成としてください。HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスを引き継がない構成とした場合，予備系のホストに系が切替わると UAP の接続処理時間が遅くなります。
- 複数の IP アドレスが起動しているマシン上では，このオプションのホスト名に標準ホスト名を指定しないでください。標準ホスト名を指定すると，HiRDB クライアントの接続に失敗することがあります。詳細はマニュアル「HiRDB Version 8 システム導入・設計ガイド」の「ホスト名の登録」を参照してください。
- IP アドレスを引き継ぐ系切り替え機能を使用する場合に，pdunit オペランドの-x オプションにループバックアドレスを指定しているときは，再配置可能な IP アドレスのホスト名を-mオプションに指定してください。

-n ホスト名〔，ホスト名〕…

～<ホスト名>((1～32 文字))

このオプションは-mオプションと一緒に指定してください。

マルチコネクションアドレス機能を使用する場合に，HiRDB クライアントが接続する HiRDB/シングルサーバのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。

HiRDB クライアントと HiRDB サーバ間で使用しているネットワークと，HiRDB サーバのサーバマシン間で使用しているネットワークが異なる場合にこのオプションを指定します。IP アドレスを引き継がない系切り替え機能を使用する場合に，予備系のホスト名を指定してください。

ホスト名形式で指定する場合の注意事項

- ホスト名は大文字と小文字を区別します。
- ホスト名には別名を指定できません。

《注意事項》

- このオプションは四つまで指定できます。
- 同じネットワーク上のホスト名を複数指定した場合，最初に指定したホスト名が有効になります。
- HiRDB/シングルサーバがないサーバマシンのホスト名を指定した場合，その指定は無視されます。
- サブネットによって，HiRDB クライアントの接続用ネットワークと HiRDB サーバ間の通信用ネットワークを切り分けている場合は，マルチコネクションアドレス機能が適用できません。-n オプションを指定した場合，HiRDB クライアントから HiRDB サーバへ接続ができなくなるときがあります。
- -m オプションと-n オプションの指定方法は，-m オプションの項を参照してください。
- 複数の IP アドレスが起動しているマシン上では，このオプションのホスト名に標準ホスト名を指定しないでください。標準ホスト名を指定すると，HiRDB クライアントの接続に失敗することがあります。詳細はマニュアル「HiRDB Version 8 システム導入・設計ガイド」の「ホスト名の登録」を参照してください。
- このオプションを指定する場合は，pd_hostname オペランドに標準ホスト名を指定してください。

-c サーバ名

～<識別子>((1～8 文字))

このオプションは HiRDB/パラレルサーバ限定のオプションのため，省略してください。

-g HA グループ識別子

～<識別子>((1～8 文字))

このオプションは HiRDB/パラレルサーバ限定のオプションのため，省略してください。

-k stls

～<識別子>((4 文字))

このオプションは HiRDB/パラレルサーバ限定のオプションのため，省略してください。

●HiRDB/パラレルサーバの場合

HiRDB/パラレルサーバのサーバ構成を定義します。指定する項目は次のとおりです。

- サーバ種別
- サーバ名
- ホスト名又はユニット識別子

《指定値の目安》

サーバ構成の目安を次に示します。

- 各サーバを別サーバマシンに定義すると，SQL 処理の並列度を向上できます。
- TCP/IP プロトコルで接続されているサーバマシンにシステムマネジャ及びフロントエンドサーバを定義すると，SQL 処理の並列度を向上できます。
- バックエンドサーバの CPU 負荷によっては，1 サーバマシンに複数のバックエンドサーバを割り当てた方がよい場合もあります。

《注意事項》

システムマネジャ及びディクショナリサーバは複数個指定できません。

-t サーバ種別

サーバの種別を指定します。

MGR：システムマネジャ

FES：フロントエンドサーバ

BES：バックエンドサーバ

DIC：ディクショナリサーバ

-s サーバ名

～<識別子>((1～8 文字))

サーバ名を指定します。サーバ種別が MGR（システムマネジャ）の場合は指定する必要はありません。

サーバ名は大文字と小文字の区別をしません。bes01 と BES01 は同じ名称として認識されます。

-x ホスト名

～<ホスト名>((1～32 文字))

pdunit オペランドの-x オプションに指定したホスト名を指定します。

-u ユニット識別子

～<識別子>((4 文字))

サーバを実行するユニットの識別子を指定します。pdunit オペランドの-u オプションで指定したユニット識別子を指定します。

系切り替え機能を使用する場合は，「-x ホスト名」ではなく，「-u ユニット識別子」を指定することをお勧めします。

-m ホスト名〔，ホスト名〕…

～<ホスト名>((1～32 文字))

マルチコネクションアドレス機能を使用する場合に，HiRDB クライアントが接続するフロントエンドサーバのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。

マルチコネクションアドレス機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

HiRDB クライアントと HiRDB サーバ間で使用しているネットワークと，HiRDB サーバのサーバマシン間で使用しているネットワークが異なる場合にこのオプションを指定します。

ホスト名形式で指定する場合の注意事項

- ホスト名は大文字と小文字を区別します。
- ホスト名には別名を指定できません。

《注意事項》

- このオプションは四つまで指定できます。
- 同じネットワーク上のホスト名を複数指定した場合，最初に指定したホスト名が有効になります。

- フロントエンドサーバがないサーバマシンのホスト名を指定した場合，その指定は無視されます。
- サブネットによって，HiRDB クライアントの接続用ネットワークと HiRDB サーバ間の通信用ネットワークを切り分けている場合は，マルチコネクションアドレス機能が適用できません。-m オプションを指定した場合，HiRDB クライアントから HiRDB サーバへ接続ができなくなるときがあります。
- -m オプションと-n オプションの指定方法を以下に示します。

HiRDB/シングルサーバの構成		pdstart オペランドの指定方法	
系切り替え構成	系切り替え時に HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレス（クライアントの接続先となる IP アドレス）を引き継ぐ構成	-m オプションの指定	-n オプションの指定
適用	適用	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスに対応するホスト名を指定してください。	オプションの指定を省略してください。
	非適用	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの現用系のホスト名を指定してください。	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの予備系のホスト名を指定してください。
非適用	-	HiRDB クライアントと HiRDB サーバ間で使用するネットワークのホスト名を指定してください。	オプションの指定を省略してください。

（凡例）

-：IP アドレスを引き継ぐ構成の適用対象外

- 系切り替え構成を適用する際は，HiRDB サーバ間で使用するネットワークの IP アドレスを引き継がない場合でも，HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスは引き継ぐ構成としてください。HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスを引き継がない構成とした場合，予備系のホストに系が切替わると UAP の接続処理時間が遅くなります。
- 複数の IP アドレスが起動しているマシン上では，このオプションのホスト名に標準ホスト名を指定しないでください。標準ホスト名を指定すると，HiRDB クライアントの接続に失敗することがあります。詳細はマニュアル「HiRDB Version 8 システム導入・設計ガイド」の「ホスト名の登録」を参照してください。
- IP アドレスを引き継ぐ系切り替え機能を使用する場合に，pdunit オペランドの-x オプションにループバックアドレスを指定しているときは，再配置可能な IP アドレスのホスト名を-mオプションに指定してください。

-n ホスト名〔，ホスト名〕…

～<ホスト名>((1～32 文字))

このオプションは-mオプションと一緒に指定してください。

マルチコネクションアドレス機能を使用する場合に，HiRDB クライアントが接続するフロントエンドサーバのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。

HiRDB クライアントと HiRDB サーバ間で使用しているネットワークと，HiRDB サーバのサーバマシン間で使用しているネットワークが異なる場合にこのオプションを指定します。IP アドレスを引き継がない系切り替え機能を使用する場合に，予備系のホスト名を指定してください。

ホスト名形式で指定する場合の注意事項

- ホスト名は大文字と小文字を区別します。
- ホスト名には別名を指定できません。

《注意事項》

- このオプションは四つまで指定できます。
- 同じネットワーク上のホスト名を複数指定した場合，最初に指定したホスト名が有効になります。
- フロントエンドサーバがないサーバマシンのホスト名を指定した場合，その指定は無視されます。
- サブネットによって，HiRDB クライアントの接続用ネットワークと HiRDB サーバ間の通信用ネットワークを切り分けている場合は，マルチコネクションアドレス機能が適用できません。-n オプションを指定した場合，HiRDB クライアントから HiRDB サーバへ接続ができなくなるときがあります。
- -m オプションと-n オプションの指定方法は，-m オプションの項を参照してください。
- 複数の IP アドレスが起動しているマシン上では，このオプションのホスト名に標準ホスト名を指定しないでください。標準ホスト名を指定すると，HiRDB クライアントの接続に失敗することがあります。詳細はマニュアル「HiRDB Version 8 システム導入・設計ガイド」の「ホスト名の登録」を参照してください。
- このオプションを指定する場合は，pd_hostname オペランドに標準ホスト名を指定してください。

-c サーバ名

～<識別子>((1～8 文字))

このオプションは 1:1 スタンバイレス型系切り替え機能に関するオプションです。-s オプションで指定したバックエンドサーバの代替 BES 名を指定します。

-c オプションを指定するには，次に示す条件をすべて満たす必要があります。満たさない場合は HiRDB を開始できません。なお，ここでは pdstart オペランドで-u オプションを指定したとして説明しています。

- -t オプションに BES が指定されている必要があります。
- 正規 BES ユニット及び代替 BES ユニット内にバックエンドサーバ以外のサーバがあってはなりません。
- 正規 BES ユニット（-u オプションに指定したユニット）内の全バックエンドサーバに対して代替 BES を指定してください。また，その代替 BES は同じユニット（代替 BES ユニット）に定義する必要があります。
- 正規 BES ユニット内で-c オプションに指定するサーバ名を重複できません。
- 正規 BES ユニット内のバックエンドサーバ数と代替 BES ユニット内のバックエンドサーバ数は同じ必要があります。
- 代替 BES の pdstart オペランドに-c オプションを指定する場合，-c オプションには正規 BES ユニット内のバックエンドサーバを指定する必要があります。

-g HA グループ識別子

～<識別子>((1～8 文字))

影響分散スタンバイレス型系切り替え機能を使用する場合に HA グループ識別子を指定します。HA グループ識別子は-s オプションで指定したサーバの移動先となるユニットの集合で，pdhagroup オペランドに指定しておく必要があります。HA グループについては，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

なお，このオプションを指定する場合は次に示す条件をすべて満たす必要があります。すべて満たさない場合は HiRDB を開始できません。

●-s オプションに指定するサーバの条件

- -t オプションに BES を指定している（バックエンドサーバを指定する必要があります）
- -c オプションは指定していない（このサーバに対する代替 BES は指定できません）

●-u オプションに指定するユニットの条件

- 正規ユニットは-g オプションで指定した HA グループに属している
- 正規ユニットを構成するサーバの種類はバックエンドサーバだけである
- 正規ユニットを構成するサーバの移動先に HA グループを指定している

●-g オプションに指定する HA グループの条件

- HA グループ内の全ユニットが同じネットワークセグメントに属している
- 影響分散スタンバイレス型系切り替え機能を適用している（pd_ha_agent = activeunits を指定している）
- ユニット内に一つ以上のホスト BES がある
- ユニット内のサーバの種類はバックエンドサーバだけである
- ユニット内のサーバはすべて HA グループに属している

-k stls

～<識別子>((4 文字))

回復不要 FES を使用する場合に指定します。回復不要 FES を使用する場合は HiRDB Non Recover FES が必要になります。

なお，次に示すどれかの条件を満たす場合は HiRDB を開始できません。

- このオプションの指定に誤りがある
- -t オプションに指定したサーバがフロントエンドサーバでない
- フロントエンドサーバ以外のサーバがあるユニットに対してこのオプションを指定している
- HiRDB Non Recover FES をセットアップしていない

《ほかのオペランドとの関連》

- このオプションを指定する場合は，pd_log_max_data_size オペランドの指定値を見積もり直してください。
- このオプションを指定したフロントエンドサーバのユニットを pd_start_skip_unit オペランドに指定しても，指定は無視されます。
- 回復不要 FES ユニットでは，系切り替え機能を適用できません。系切り替え機能を適用するシステムの場合，回復不要 FES ユニットの pd_ha_unit オペランドに必ず nouse を指定してください。

2.2.45　グローバルバッファに関するオペランド

218）pdbuffer -a グローバルバッファ名

{-r RD エリア名〔，RD エリア名〕… |

-b RD エリア名〔，RD エリア名〕… |

-o |

-i 認可識別子．インデクス識別子 }

-n バッファ面数〔- l バッファサイズ〕

〔-m 同時実行最大プリフェッチ数〕

〔-p 一括入力最大ページ数〕

〔-w デファードライトトリガ時の更新ページ出力比率〕

〔-c〕

〔-y デファードライトトリガ契機の更新バッファ面数〕

グローバルバッファをどの RD エリアに割り当てるかを定義します。グローバルバッファとは，RD エリアへ表のデータを入出力するときにデータを確保する領域のことであり，共用メモリ上に確保されます。

すべての RD エリアにグローバルバッファを割り当ててください。グローバルバッファを割り当てない RD エリアに対しては，SQL 文及び pdopen コマンドを実行できません。グローバルバッファの設計方法については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《オペランドの規則》

pdbuffer オペランドは 2,000,000 個まで指定できます。HiRDB/パラレルサーバの場合は 1 サーバ当たり 2,000,000 個までになります。ただし，全体で 2,147,483,647 個が上限になります。

《指定値のチューニング方法》

グローバルバッファのチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

pdbuffer オペランドを省略すると，次に示す条件でグローバルバッファを割り当てます。

- グローバルバッファ名：gbuf00
- グローバルバッファの種別：-o オプション指定相当
- グローバルバッファサイズ：6 メガバイト
- バッファ面数：6 メガバイト÷RD エリアのページ長の最大値

-a グローバルバッファ名

～<識別子>((1～16 文字))

グローバルバッファの名称を指定します。同じ名称は指定できません。

-r RD エリア名〔，RD エリア名〕…

～<識別子>((1～30 文字))

グローバルバッファを割り当てる RD エリアの名称を指定します。次に示す RD エリアの名称を指定してください。

- マスタディレクトリ用 RD エリア
- データディクショナリ用 RD エリア
- データディレクトリ用 RD エリア
- ユーザ用 RD エリア

- データディクショナリ LOB 用 RD エリア※
- ユーザ LOB 用 RD エリア※
- レジストリ用 RD エリア
- レジストリ LOB 用 RD エリア※
- リスト用 RD エリア

注※

これらの LOB 用 RD エリアは，-b オプションの LOB 用グローバルバッファも割り当てることをお勧めします（-r オプションと-b オプションの両方を指定します）。LOB 用グローバルバッファを割り当てると性能が向上します。

《指定値の目安》

ページ長が同じか，又はページ長が近い RD エリアを同じグローバルバッファに割り当てると，入出力回数を削減できます。ただし，同じページ長の RD エリアでも，次に示すような RD エリアは別々のグローバルバッファに割り当てた方が入出力回数を削減できます。

- 用途の異なる表を格納する RD エリア
- ランダムアクセスが多い RD エリアと順次アクセスが多い RD エリア

《オペランドの規則》

- RD エリア名に半角英数字以外の文字を含む場合は引用符（"）で囲んでください。
- RD エリア名を引用符（"）で囲んだ場合は英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。
- 一つのグローバルバッファには最大 3200 個の RD エリアを定義できます。

-b RD エリア名〔，RD エリア名〕…

～<識別子>((1～30 文字))

LOB 用グローバルバッファを割り当てる RD エリアの名称を指定します。次に示す RD エリアの名称を指定してください。

- データディクショナリ LOB 用 RD エリア
- ユーザ LOB 用 RD エリア
- レジストリ LOB 用 RD エリア

ここで指定した LOB 用 RD エリアは，-r オプションでも指定しておく必要があります。-b オプションの指定だけでは LOB 用 RD エリアをアクセスできません。指定例を次に示します。

(例)

LOB 用 RD エリア（RDLOB01）に-r 及び-b オプションのグローバルバッファを割り当てます。

```
pdbuffer -a gbuf01 -r RDLOB01 -n 1000
pdbuffer -a gbuf02 -b RDLOB01 -n 1000
```

備考

- LOB 用 RD エリアにはディレクトリ部とデータ部があります。ディレクトリ部とデータ部はそれぞれ異なるグローバルバッファで管理しています。-r 指定のグローバルバッファにはディレクトリ部がキャッシュされ，-b 指定のグローバルバッファにはデータ部がキャッシュされます。このため，-r（又は-o）及び-b オプションを同時に指定する必要があります。
- -r 指定のグローバルバッファと-b 指定のグローバルバッファは異なる用途のグローバルバッファのため，別々に見積もる必要があります。-r 指定のグローバルバッファはディレクトリ部分だけをキャッシュするため，比較的小さくてもよいです。-b 指定のグローバルバッファは LOB のデータ部をキャッシュするため，使用可能なメモリ容量及びヒット率を考慮して見積もってく

ださい。グローバルバッファの見積もり方法については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値の目安》

次に示すような LOB 用 RD エリアを指定します。

- プラグインインデクスを格納する LOB 用 RD エリア
- データ量が少なく，アクセス頻度が高いデータを格納する LOB 用 RD エリア

なお，一つの LOB 用グローバルバッファには一つの LOB 用 RD エリアを割り当てることをお勧めします。

《オペランドの規則》

- RD エリア名称が半角英数字以外の文字を含む場合は，引用符（"）で囲んでください。
- RD エリア名を引用符（"）で囲んだ場合は英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。
- 一つのグローバルバッファには，最大 3200 個の LOB 用 RD エリアを定義できます。

《注意事項》

LOB 用グローバルバッファは，プリフェッチ機能及びデファードライトトリガ機能を使用しません。したがって，-m，-p，及び-w オプションを指定する必要はありません。

-o

-r オプションで指定していないすべての RD エリアに，グローバルバッファを割り当てる場合に指定します。-o オプションは 1 回だけ指定できます。2 回以上指定した場合は最初の指定を有効とします。

-i 認可識別子．インデクス識別子

認可識別子：～＜識別子＞((1～8 文字))

インデクス識別子：～＜識別子＞((1～30 文字))

インデクス用グローバルバッファを割り当てるインデクスの名称（認可識別子．インデクス識別子）を指定します。

《指定値の目安》

使用頻度の高いインデクスを指定してください。使用頻度の高いインデクスにグローバルバッファを割り当てると，インデクスページのメモリ常駐度が上がり，入出力回数の削減効果が出ます。

特に，クラスタキー又はユニークキーに定義したインデクスをグローバルバッファに割り当てると効果が大きくなります。なお，クラスタキーのインデクス識別子は HiRDB が決めるため，表を定義した後にディクショナリ表（SQL_INDEXES 表の INDEX_NAME 列）を検索してインデクス識別子を確認してください。

また，すべてのインデクスをインデクス用のグローバルバッファに割り当てると，グローバルバッファ全体の使用効率が悪くなります。したがって，メモリ量に応じてグローバルバッファに割り当てるインデクスを厳選してください。

《オペランドの規則》

- 「認可識別子．インデクス識別子」は，同じ名称で複数指定できません。
- 認可識別子又はインデクス識別子に半角英数字以外の文字を含む場合は，認可識別子及びインデクス識別子をそれぞれ引用符（"）で囲んでください。
- 認可識別子及びインデクス識別子をそれぞれ引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合は，すべて大文字として扱われます。

《注意事項》

- -i オプションに指定したインデクスを格納している RD エリアにもグローバルバッファを割り当てる必要があります。-r オプションで RD エリアを指定するか，又は-o オプションでグローバルバッファを割り当ててください。

  RD エリアには，ディレクトリ部（ディレクトリページ），データ部（表データのページ），及びインデクス部（インデクスページ）が格納されています。インデクス用グローバルバッファにはインデクス部だけがキャッシュされるため，ディレクトリ部とデータ部をキャッシュするための-r 又は-o 指定のグローバルバッファが必要になります。

- DROP TABLE 又は DROP INDEX を実行した場合，削除したインデクスに割り当てていたインデクス用グローバルバッファは使用されなくなります。また，削除したインデクスと同じ名称のインデクスを定義しても，そのインデクス用グローバルバッファは使用できません。そのインデクス用グローバルバッファを使用するには，HiRDB を一度正常終了して再度開始する必要があります。

- HiRDB の開始時に存在しないインデクスを-i オプションに指定した場合，そのインデクス用グローバルバッファは割り当てられません。このとき，KFPH23014-W メッセージが出力されます。

  また，HiRDB の開始後，-i オプションに指定したインデクスを定義しても，インデクス用グローバルバッファは使用できません。インデクス用グローバルバッファを使用するには，HiRDB を一度正常終了して再度開始する必要があります。

- インデクス用グローバルバッファのバッファサイズより大きなページ長のインデクス格納 RD エリアを ALTER TABLE で追加した場合，追加した RD エリアのインデクスページはインデクス用グローバルバッファにキャッシュされません。インデクス用グローバルバッファにキャッシュするには，HiRDB を一度正常終了して再度開始する必要があります。

- データベース構成変更ユティリティの RD エリアの再初期化で，RD エリアのページ長をインデクス用グローバルバッファのバッファサイズより大きくした場合，再初期化した RD エリアのインデクスページはインデクス用グローバルバッファにキャッシュされません。インデクス用グローバルバッファにキャッシュするには，HiRDB を一度正常終了して再度開始する必要があります。

-n バッファ面数

～<符号なし整数>

- 32 ビットモードの場合：((4～460000))
- 64 ビットモードの場合：((4～1073741824))

グローバルバッファの面数を指定します。このオプションは必ず指定してください。

《指定値の目安》

- 必要以上に大きな値を指定すると，入出力回数は減るがバッファを検索するためのオーバヘッドが増えます。
- グローバルバッファは，共用メモリに確保するので必要以上に大きくするとほかのメモリの使用時にページングが多発し，性能が低下します。
- シンクポイント取得のタイミング，及びグローバルバッファ内の更新ページの比率によっては，データベースへの出力が集中します。そのため，入出力回数のバランスも考慮する必要があります。
- バッファ面数が大き過ぎると共用メモリを確保できないことがあるので注意してください。
- バッファ面数を拡張する場合，共用メモリの割り当て先は，ページングファイルにすることを推奨します。運用ディレクトリ下のファイルを割り当てると，NTFS のキャッシュフラッシュの影響を受けやすくなり，OS のスローダウンを引き起こす場合があります。

共用メモリの割り当て先は，pdntenv コマンドの-shmfile オプションで設定できます。
pdntenv コマンドについては，マニュアル「HiRDB Version 9 コマンドリファレンス」を参照してください。

- グローバルバッファの面数は，次の表から求めてください。

条件		グローバルバッファ面数の求め方
-r 又は-o オプション指定のグローバルバッファの場合※1	HiRDB/シングルサーバの場合	同時に発生する SQL 処理要求の数×1SQL で使用するページ数（3～6 程度）
	HiRDB/パラレルサーバの場合	同時実行ユーザ数×1 トランザクション内の平均同時アクセス表数×3×n※2
-b オプション指定のグローバルバッファの場合		LOB 用 RD エリアの総ページ数×常駐度※3 (%)
-i オプション指定のグローバルバッファの場合※1		インデクスの格納ページ数×常駐度※4 (%)

注※1

最低限必要なバッファ面数は次のとおりです。バッファ面数にはこの値以上を指定するようにしてください。この値より小さい値を指定した場合はバッファ不足による SQL エラーが発生することがあります。

- HiRDB/シングルサーバの場合
  同時に発生する SQL 処理要求の数×4
- HiRDB/パラレルサーバの場合
  同時実行ユーザ数×1 トランザクション内の平均同時アクセス表数×4

注※2

n=1 として見積もったバッファ面数では，バッファのヒット率を高くできません。このため，係数としてどれくらいの余裕をバッファに持たせるか（バッファのヒット率をどのくらいにするか）を検討して指定してください。

注※3

常駐度は 1（100%）が望ましいが，メモリ容量，データのアクセス頻度などを考慮して設定します。ユーザ LOB 用 RD エリアの総ページ数，及びレジストリ用 RD エリアの総ページ数については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

注※4

常駐度は 1（100%）が望ましいが，メモリ容量，インデクスの重要度などを考慮して設定します。インデクスの格納ページ数については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値のチューニング方法》

バッファのヒット率が HiRDB 稼働業務で 80%以上となるように設定してください。バッファのヒット率は次に示す方法で確認できます。

- 統計解析ユティリティ（グローバルバッファに関する統計情報）の「更新バッファヒット率」及び「参照バッファヒット率」で確認できます。
- pdbufls コマンドのヘッダ名 HIT で確認できます。

-l バッファサイズ

～<符号なし整数>((4～30 の偶数))

グローバルバッファのバッファ 1 面のサイズをキロバイト単位で指定します。このオプションは-n オプションと一緒に指定します。

《指定値の目安》

通常，このオプションは省略してください。このオプションを省略すると，このグローバルバッファに割り当てた RD エリアの最大ページ長がバッファサイズになります。なお，HiRDB/パラレルサーバの場合はサーバ内の RD エリアの最大ページ長がバッファサイズになるため，サーバごとにバッファサイズが異なることがあります。

また，次に示す場合は指定値の変更を考えてください。

- 将来，このオプションで指定した値を超えるページ長の RD エリアを追加したり，再初期化して RD エリアのページ長を大きくしたりすることがある場合，このオプションの指定値を大きめに設定しておくことをお勧めします。ただし，HiRDB を一度終了できる場合，HiRDB の次回開始時にはバッファサイズに最大ページ長が仮定されるため，このオプションの指定値を大きめに設定しておく必要はありません。

《オペランドの規則》

- このオプションの指定値が RD エリアの最大ページ長より小さい場合は，RD エリアの最大ページ長がバッファサイズになります。
- このオプションに奇数を指定した場合，バッファサイズは指定した値に 1 を加算した値になります。

《注意事項》

このオプションの値を変更しても，HiRDB を正常開始しないと，その値が反映されません。HiRDB を再開始した場合は，前回稼働時（変更前）のバッファサイズとなります。

-m 同時実行最大プリフェッチ数

～<符号なし整数>((0～95000))《0》

同時にプリフェッチ機能を使用する最大数を指定します。プリフェッチ機能は，rawI/O 機能を使用して大量検索をする場合に入出力時間を削減する効果があります。

このオプションに 0 を指定するか，又は省略した場合は，プリフェッチ機能は動作しません。プリフェッチ機能を動作させるには，1 以上を指定してください。

《指定値の目安》

指定値の設定は，このグローバルバッファに割り当てられた RD エリアの表の中で，プリフェッチ機能が適用される SQL を同時に実行する回数を指定します。プリフェッチ機能が適用される SQL を次に示します。ただし，3.に関しては実行数を 2 として算出します。

1. インデクスを使用しない SELECT，UPDATE，DELETE 文（=条件，IN 条件を除く）
2. インデクスを使用した昇順検索※をする SELECT，UPDATE，DELETE 文（=条件，IN 条件を除く）
3. クラスタキーを使用した昇順検索※をする SELECT，UPDATE，DELETE 文（=条件，IN 条件を除く）

注※　複数列インデクスの場合はインデクス定義で指定した順序

《注意事項》

プリフェッチ機能を使用する場合，グローバルバッファ用の共用メモリからグローバルバッファとは別に一括入力専用のバッファが確保されるため，再度共用メモリを見積もる必要があります。

プリフェッチ機能，及びグローバルバッファが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

-p 一括入力最大ページ数

～<符号なし整数>((2～256))《32》

プリフェッチ機能の一括入力最大ページ数を指定します。この指定は，-m オプションで 1 以上を指定したときだけ有効になります。

《指定値の目安》

共用メモリ量と削減効果のコストパフォーマンスから，指定の目安として次に示す計算式を満足するように指定します。

a×b = 64～128　　（単位：キロバイト）

a：プリフェッチ対象表のデータ，又はインデクスが格納されている RD エリアのページ長

b：一括入力最大ページ数

-w デファードライトトリガ時の更新ページ出力比率

～<符号なし整数>((0～100))《20》

デファードライトトリガでの更新ページ出力比率をパーセントで指定します。デファードライトトリガについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

このオプションに 0 を指定した場合は，デファードライトトリガでの更新ページ出力をしません。

《指定値の目安》

- グローバルバッファごとの入出力回数，及び更新ページのヒット率などを統計解析ユティリティで確認し，指定値を決めてください。更新ページのヒット率が高いグローバルバッファでは出力比率を低く，ヒット率の低いグローバルバッファでは出力比率を高く指定してください。
- 必要以上に大きい値を指定すると，更新の頻度が多い場合，入出力が多発します。また，遅延書き込み時にデータベースに書き込むページ数が増え，スループットが低下します。反対に，必要以上に少ない値を指定すると，シンクポイントダンプ出力時にデータベースに書き出すページ数が多くなることがあります。このため，できるだけ，シンクポイントダンプ出力間隔の間に同じページがデータベースに書き出される回数が少なくなるようにしてください。
- すべてのページがグローバルバッファ上にあり，複数トランザクションで同一ページの更新が多く発生する場合は 0 を指定してください。0 を指定すると，シンクポイント間隔内でデータベースに書き出される総ページ数を減らせます。

-c

このオプションはスタンバイレス型系切り替え機能に関するオプションです。この機能を使用しない場合にはこのオプションを指定しないでください。

- 1：1 スタンバイレス型系切り替え機能の場合

  正規 BES の RD エリア又はインデクスが使用するグローバルバッファを，切り替え先の代替 BES 側にも作成する場合にこのオプションを指定します。このオプションを省略した場合，代替 BES 側にはグローバルバッファは作成されないため，代替中は代替 BES に定義している-o オプション指定のグローバルバッファを使用します。

  なお，-c 及び-o オプションの両方を省略した場合は代替 BES ユニットを開始できません。

  このオプションの指定によって代替中のグローバルバッファの割り当て方式が決まります。代替中のグローバルバッファの割り当て方式については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

- 影響分散スタンバイレス型系切り替え機能の場合

  影響分散スタンバイレス型系切り替えの対象となるバックエンドサーバに配置されている RD エリア又はインデクスのグローバルバッファを定義する場合はこのオプションを指定します。ここで定義したグローバルバッファは同じ HA グループ内の全ユニットに作成されます。

なお，-c オプションを省略した場合，このグローバルバッファが割り当てられないため，このグローバルバッファを使用する RD エリアにアクセスできません。

このオプションの指定によって影響分散スタンバイレス型系切り替え機能の対象となるバックエンドサーバの RD エリア，又はインデクスに対するグローバルバッファの割り当て方式が決まります。影響分散スタンバイレス型系切り替え機能使用時のグローバルバッファの割り当て方式については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

-y デファードライトトリガ契機の更新バッファ面数

～<符号なし整数>((2～2147483647))

デファードライトトリガのトリガ契機を更新バッファ面数で指定します。更新バッファ面数が指定値に達すると，更新ページをディスクに書き込みます。グローバルバッファごとにデファードライトトリガのトリガ契機を設定したい場合に，このオプションを指定してください。

《指定値の目安》

通常はこのオプションを省略してください。デファードライトの書き込み処理がシンクポイントダンプ取得間隔時間内で終了しない場合など，更新バッファ数を減らし，更新バッファヒット率を多少抑えてでも書き込み時間を短縮したいときにこのオプションを指定します。指定する場合は，50%（HiRDB が設定する初期値）を目安にし，マニュアル「HiRDB Version 8 システム運用ガイド」の「デファードライト処理適用時のシンクポイント処理時間のチューニング」を参照してオペランドの値を決定してください。

《オペランドの規則》

このオプションにグローバルバッファ面数より大きい値を指定した場合は，グローバルバッファ面数と同じ値が適用されます。

《注意事項》

このオプションの指定値が小さ過ぎると，デファードライト処理の回数が増え，デファードライト処理による負荷を高めるため，指定値の目安に従った設定をお勧めします。

《ほかのオペランドとの関連》

このオペランドは，pd_dbbuff_rate_updpage オペランドと次に示す関係があります。

- pd_dbbuff_rate_updpage オペランドの指定値は全グローバルバッファに適用されます。
- pdbuffer オペランドの-y オプションの指定値はグローバルバッファごとに適用されます。
- pdbuffer オペランドの-y オプションの指定値が優先されます。
- pdbuffer オペランドの-y オプションを指定しない場合，デファードライトトリガ契機の更新バッファ面数は，pd_dbbuff_rate_updpage オペランドの指定によって次のように異なります。

pd_dbbuff_rate_updpage オペランドの指定	デファードライトトリガ契機の更新バッファ面数
あり	グローバルバッファ面数×pd_dbbuff_rate_updpage オペランドの指定値
なし	HiRDB が自動計算します。

2.2.46　HA グループに関するオペランド

219）　pdhagroup -g HA グループ識別子 -u ユニット識別子〔，ユニット識別子〕…

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドで HA グループを定義します。

一つのシステムに定義できる HA グループ数の上限はありません。

HA グループについては，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

-g HA グループ識別子

～<識別子>((1～8 文字))

HA グループ識別子を指定します。システム内で HA グループを一意に識別できるように指定してください。

-u ユニット識別子〔，ユニット識別子〕…

～<識別子>((4 文字))

HA グループを構成するユニットのユニット識別子を指定します。指定できるユニット数は 2～32 です。ユニットを重複して指定すると，エラーになります。なお，このオプションに指定するユニットは次に示す条件をすべて満たす必要があります。

- HA グループ内の全ユニットが同じネットワークセグメントに属している
- ほかの HA グループに属していない
- 影響分散スタンバイレス型系切り替え機能を適用している（pd_ha_agent = activeunits を指定している）
- ユニット内に一つ以上のホスト BES がある（受け入れ専用ユニットは定義できない）
- ユニット内のホスト BES 数と受け入れ可能なゲスト BES の最大数（pd_max_act_guest_servers オペランドの値）の合計は 34 以内である
- ユニット内のサーバはすべてバックエンドサーバである
- ユニット内のサーバはすべて HA グループに属している

2.2.47　統計情報に関するオペランド

220）　pdstbegin

●HiRDB/シングルサーバの場合のオペランド指定形式

pdstbegin 〔-k 統計情報種別〔，統計情報種別〕…〕
〔-m 時間間隔〕

●HiRDB/パラレルサーバの場合のオペランド指定形式

pdstbegin 〔-k 統計情報種別〔，統計情報種別〕…〕
〔-m 時間間隔〕
〔{-x ホスト名｜-u ユニット識別子}〕
〔{-a｜-s サーバ名〔，サーバ名〕…}〕

HiRDB の開始時から統計情報を取得する場合に指定します。統計情報の出力を中止するときは pdstend コマンドを実行してください。出力先ファイルは統計ログファイルになります。

-k 統計情報種別〔，統計情報種別〕…

～≪sys≫

出力する統計情報の種別を指定します。なお，統計情報の出力種別の指定状況は，pdls -d stj コマンドで確認できます。

統計情報の種類（-k オプションの指定）	出力する統計情報の種類	統計情報の出力契機
sys	システムの稼働に関する統計情報	-m オプションに指定した時間間隔ごとに統計情報を出力します。
uap	UAP に関する統計情報	HiRDB への接続時及び HiRDB からの切り離し時に統計情報を出力します。

統計情報の種類（-k オプションの指定）	出力する統計情報の種類	統計情報の出力契機
sql	SQL に関する統計情報	SQL の実行開始時及び SQL の実行終了時に統計情報を出力します。
buf	グローバルバッファに関する統計情報※1	シンクポイント時に統計情報を出力します。
fil	データベース操作に関する HiRDB ファイルの統計情報	
dfw	デファードライト処理に関する統計情報	
idx	インデクスに関する統計情報	
sop	SQL 静的最適化に関する統計情報	動的 SQL の前処理時，及び静的 SQL の前処理時に，SQL オブジェクトバッファにヒットしなかった場合に統計情報を出力します。
dop	SQL 動的最適化に関する統計情報	FETCH 文及び CLOSE 文以外の SQL 文を実行したときに統計情報を出力します。
pcd	SQL オブジェクト実行に関する統計情報	SQL オブジェクトの実行時に統計情報を出力します。
sqh	SQL 文の履歴に関する統計情報 ※2	PREPARE 時，及び埋込み型 SQL 文の実行時に統計情報を出力します。
obj	SQL オブジェクト転送に関する統計情報※3	SQL オブジェクトの実行時に統計情報を出力します。
fsv	外部サーバの稼働に関する統計情報※3	トランザクション終了時に統計情報を出力します。
hba	外部サーバの利用状況に関する統計情報※3	外部サーバに対して SQL が実行されたときに統計情報を出力します。

注※1

これらの統計情報はシンクポイント時に取得され，シンクポイント間の情報が編集されます。このため，シンクポイントが 2 回以上発生しないと統計情報が取得されません。これらの統計情報を確実に取得するためには，pdstend コマンド実行直前に，pdlogswap 又は pdlogsync コマンドを実行してシンクポイントを発生させてください。

注※2

SQL 文の履歴に関する統計情報は，SQL に関する統計情報（sql 指定）を出力している場合に出力されます。

注※3

HiRDB/パラレルサーバ限定の統計情報です。HiRDB/シングルサーバの場合は出力しません。

-m 時間間隔

～<符号なし整数>（(1～1440)) ≪10≫

このオプションは「システムの稼働に関する統計情報」を取得する場合（-k オプションで sys を指定した場合）に指定します。

「システムの稼働に関する統計情報」を統計ログファイルに出力する間隔を分単位で指定します。

-x ホスト名

～<ホスト名>（(1～32 文字))

HiRDB/パラレルサーバの場合に，統計情報の出力対象となるユニットを限定するとき，pdunit オペランドの-x オプションに指定したホスト名を指定します。統計情報の出力対象となるユニットのホスト名を指定してください。

このユニットに影響分散スタンバイレス型系切り替えを適用している場合は受け入れ中のゲスト BES の統計情報も出力します。

-u ユニット識別子

～<識別子>（(4 文字))

HiRDB/パラレルサーバのときに，統計情報の出力対象となるユニットを限定する場合に指定します。統計情報の出力対象となるユニットのユニット名を指定します。

このホスト上のユニットに影響分散スタンバイレス型系切り替えを適用している場合は受け入れ中のゲスト BES の統計情報も出力します。

-a

HiRDB/パラレルサーバのときに，統計情報の出力対象となるサーバを限定しない場合に指定します。このオプションを指定すると，全サーバが統計情報の出力対象となります。

-s サーバ名〔，サーバ名〕…

～<識別子>（(1～8 文字))

HiRDB/パラレルサーバのときに，統計情報の出力対象となるサーバを限定する場合に指定します。統計情報の出力対象となるサーバのサーバ名を指定します。

サーバの種類によっては統計情報の出力対象にならない場合があります。統計情報を出力するサーバの種類を次に示します。

統計情報の種類 (-k オプションの指定)	サーバの種類		
	FES	DS	BES
sys	○	○	○
uap	○	×	×
sql	○	×	×
buf	×	○	○
fil	×	○	○
dfw	×	○	○
idx	×	○	○
sop	○	×	×
dop	○	×	×
pcd	○	○	○
sqh	○	×	×
obj	×	○	○
fsv	×	×	○
hba	×	×	○

（凡例）

○：統計情報を出力します。

×：統計情報を出力しません。

《ほかのオペランドとの関連》

このオペランドを指定すると，pd_statistics オペランドの指定は無視されます。

《注意事項》

- pdstbegin オペランドは一つだけ指定できます。二つ以上指定した場合は一つ目の指定が有効になり，二つ目以降は無視されます。
- pdstbegin オペランドを指定した場合は，pdstend コマンドを入力しないと，HiRDB を終了するまで統計情報が出力されます。
- 1：1 スタンバイレス型系切り替え機能を使用している場合で，代替 BES ユニットに系を切り替えるときはこのオペランドの指定が無効になります。
- 影響分散スタンバイレス型系切り替え機能を使用している場合の注意事項です。HiRDB の正常開始時，各サーバはこのオペランドの指定値に従います。HiRDB の再開始時（系切り替えを含む）はこのオペランドの指定値が無効になります。再開始時は前回の統計情報の取得状態を引き継ぎます。系の切り替え時は切り替え前の統計情報の取得状態を引き継ぎます。
- ユニット内に開始しているサーバが存在しない場合は，ユニットの統計情報は取得されません。
- pdstbegin 及び pdstend コマンドの入力タイミングによっては，UAP に関する統計情報と SQL に関する統計情報の取得情報が一致しないことがあります。pdstbegin 及び pdstend コマンドの入力タイミングと出力される統計情報の関係を次に示します。

（凡例）○：統計情報を出力します。
×：統計情報を出力しません。
注※　ＯＬＴＰ環境下では出力されません。

- HiRDB，ユニット，又はサーバを終了（異常終了を含む）して，その後，開始した場合，開始後に統計情報の取得状態が引き継がれないことがあります。HiRDB，ユニット，又はサーバ開始時の，統計情報の取得状態の引き継ぎ有無を次に示します。

開始モード	統計情報の取得環境	開始条件			
		HiRDB の開始	ユニット単位の開始	サーバ単位の開始	
				影響分散スタンバイレス型系切り替え機能を使用していない場合	影響分散スタンバイレス型系切り替え機能を使用している場合
正常開始	pdstbegin オペランドを指定して統計情報を取得している場合	○	○	×※	○
	pdstbegin コマンドを実行して統計情報を取得している場合	×※	×※	×※	×※

開始モード	統計情報の取得環境	開始条件			
		HiRDBの開始	ユニット単位の開始	サーバ単位の開始	
				影響分散スタンバイレス型系切り替え機能を使用していない場合	影響分散スタンバイレス型系切り替え機能を使用している場合
再開始	pdstbegin オペランドを指定して統計情報を取得している場合	○	○	－	△
	pdstbegin コマンドを実行して統計情報を取得している場合	×※	×※	－	○

（凡例）

○：統計情報の取得状態を引き継ぎます。

△：統計情報の取得状態を引き継ぎます。pdstbegin コマンド実行時はコマンド指定の統計情報の取得状態を引き継ぎます。

×：統計情報の取得状態を引き継ぎません。

－：該当しません。

注※

統計情報を取得する場合，HiRDB の開始後に pdstbegin コマンドを実行する必要があります。

221）　pdhibegin -k 履歴情報種別〔，履歴情報種別〕…

HiRDB の開始時から履歴情報を取得する場合に指定します。このオペランドを指定すると，HiRDB が終了するまで履歴情報を出力し続けます。出力先ファイルはシステムログファイルになります。

-k 履歴情報種別：

出力する履歴情報種別を指定します。

cnc：CONNECT/DISCONNECT に関する統計情報

HiRDB/パラレルサーバの場合，出力先ファイルはフロントエンドサーバのシステムログファイルになります。

2.2.48　クライアントグループに関するオペランド

222）　pdcltgrp -g クライアントグループ名称 -u グループごとの接続保証ユーザ数

クライアントグループの接続枠保証機能を使用する場合に指定します。クライアントグループの接続枠保証機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

-g クライアントグループ名称

～<英字>((1～2 文字))

クライアントグループの名称を指定します。

クライアントの種別	クライアントグループ名称	備考
X/Open XA インタフェース	XA※1	X/Open XA インタフェースで HiRDB にアクセスするクライアントのことです。クライアントが PC 又は WS であっても，X/Open XA インタフェースで HiRDB にアクセスすると，クライアントグループは XA になります。

クライアントの種別	クライアントグループ名称	備考
PC クライアント	PC※2	Windows 系及び Linux 系クライアントのことです。
WS クライアント	WS※2	UNIX 系クライアントのことです。
メインフレーム系クライアント	MF	VOS3 クライアントのことです。
ユーザが任意に決めたクライアントグループ※3	英大文字1文字※1※4	－

（凡例）－：該当しません。

注※1

このクライアントグループ名称を指定するには，HiRDB クライアントのバージョンが次に示すどちらかの条件を満たす必要があります。

- 04-05
  05-00 及び 05-01 では指定できません。
- 05-02 以降

注※2

このクライアントグループ名称を指定するには，HiRDB クライアントのバージョンが次に示す条件を満たす必要があります。

- 04-00 以降

注※3

ユーザが任意に決めたクライアントグループは，10 グループまで指定できます。

注※4

ここで指定したクライアントグループ名称を，クライアント環境定義の PDCLTGRP オペランドに指定してください。pdcltgrp オペランドと PDCLTGRP オペランドの指定値が合っていない場合，ここでの指定を無視します。

-u　グループごとの接続保証ユーザ数

～<符号なし整数>

- HiRDB/シングルサーバの場合：((1～2999))
- HiRDB/パラレルサーバの場合：((1～1999))

各クライアントグループの接続保証数を指定します。

ほかのクライアントグループ又はユティリティからの HiRDB へのアクセスが集中した場合でも，ここで指定したユーザ数までは，該当するクライアントグループから HiRDB への接続が保証されます。

《注意事項》

- -u オプションで指定したユーザ数の合計が，pd_max_users オペランドの指定値以上にならないようにしてください。指定値以上になると，HiRDB を開始しません。
- データベース定義ユティリティ以外のユティリティは，この機能の対象外です。

2.2.49　プラグインに関するオペランド

223）　pdplugin -n プラグイン名称

～<識別子>((1～30 文字))

HiRDB で使用するプラグインの名称を指定します。使用しない場合は，省略してください。

ここで指定するプラグイン名称については，プラグインに付いているマニュアルを参照してください。

《前提条件》

ここで指定するプラグインは，あらかじめ pdplgrgst コマンドで HiRDB に登録しておいてください。

2.2.50　共用メモリに関するオペランド

224）　SHMMAX 共用メモリセグメントサイズの上限値

～<符号なし整数>（単位：メガバイト）

- 32 ビットモードの場合：((6～2047))《200》
- 64 ビットモードの場合：((6～4194304))《1024》

グローバルバッファ用の共用メモリセグメントサイズの上限値をメガバイト単位で指定します。

HiRDB はこのオペランドの指定値を上限としてグローバルバッファ用の共用メモリセグメントを確保します。サーバマシン内の RD エリアに割り当てるグローバルバッファサイズの合計がこのオペランドの値を超える場合は，複数の共用メモリセグメントを割り当てます。

HiRDB は開始時に 1 サーバ当たり最大 16 セグメントのグローバルバッファプール用共用メモリを確保します。グローバルバッファプールが使用している共用メモリセグメントの情報は pdls コマンド（-d mem 指定）によって確認できます。

《指定値の目安》

<32 ビットモード>

- SHMMAX を特に指定する必要はありません。ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合には，200 を指定してください。
- KFPH23005-E メッセージ（エラーコード 20）が出力される場合，マニュアル「HiRDB Version 8 メッセージ」を参照して KFPH23005-E メッセージの対策に示される SHMMAX の値に変更してください。

<64 ビットモード>

- グローバルバッファプール用共用メモリサイズが 16GB 以下の場合，SHMMAX を指定する必要はありません。
- グローバルバッファプール用共用メモリサイズが 16GB より大きい場合，SHMMAX の指定値に以下の式を満たす値を指定してください。

  （SHMMAX の指定値）×16 ＝（グローバルバッファプール用共用メモリサイズ）

《注意事項》

- グローバルバッファ動的変更機能を使用している場合，SHMMAX の指定値の変更は正常停止後のみできます。

  そのため，強制停止または計画停止後の再開始時に KFPH23005-E メッセージ（エラーコード 20）が発生し，SHMMAX の値の変更が必要な場合，マニュアル「HiRDB Version 8 メッセージ」の KFPH23005-E メッセージ（エラーコード 20）の表 項番 3 を参照して，SHMMAX の指定値を変更してください。

- 共用メモリは，共用メモリファイルとして HiRDB 運用ディレクトリがあるディスクに確保されます。このため，ディスクの空き容量が十分でないと，ディスク容量不足になります。
- 共用メモリをページファイルに割り当てる場合，ページファイル容量が十分でないと，ページファイル容量不足になります。

  共用メモリを割り当てる方法については，マニュアル「HiRDB Version 8 コマンドリファレンス」の pdntenv コマンドを参照してください。
- 共用メモリのページ固定機能を使用する場合は，本オペランドにはラージページページ長の整数倍の値を指定してください。共用メモリのページ固定は以下の場合に使用されます。

  ・pd_dbbuff_attribute オペランドに fixed を指定してください。

  ・pd_max_resident_rdarea_no オペランドに 1 以上を指定ください。

  指定値がページ長の整数倍でない場合は HiRDB がページ長の整数倍の値に補正します。
- 本オペランドを省略した場合，デフォルト値が仮定されます。また，本オペランドに指定可能範囲外の値を指定した場合，デフォルト値が仮定されたものとみなし，警告メッセージ KFPH23011-W を出力した後に HiRDB 開始処理を続行します。
- グローバルバッファプール用共用メモリが使用する管理領域は SHMMAX に指定したサイズで分割することができません。そのため，SHMMAX 指定値には上記管理領域のサイズより大きい値を指定してください。グローバルバッファプール用共用メモリが使用する管理領域サイズの詳細はマニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_dbbuff_modify
- pdbuffer
- pd_max_add_dbbuff_no
- pd_max_add_dbbuff_shm_no
- pd_sysdef_default_option（32 ビットモードで v6compatible を指定している場合，このオペランドの省略値は 6 になります）

2.2.51　日付・時刻に関するオペランド

225）TZ タイムゾーン

～<文字列>

出力するログの日付・時刻情報のタイムゾーンを指定します。タイムゾーンとは，時刻などを表示する場合の環境を設定する環境変数のことをいいます。

このオペランドの指定値（省略時は JST-9）と，サーバマシンのシステム環境変数 TZ が同じであることを確認してください。コマンドプロンプトで set コマンドを実行すると，サーバマシンの環境変数が表示されます。システム環境変数 TZ が未設定，又は値がこのオペランドと異なる場合は HiRDB の動作を保証できません。

2.2.52　メッセージの出力抑止機能に関するオペランド

226）pdmlgput -s 出力有無

{-c ALL｜〔-l メッセージの重要度〕 -m メッセージ ID〔，メッセージ ID〕…}

HiRDB がイベントログに出力するメッセージに対して，次に示す指定ができます。

- 全メッセージを出力抑止する

- 出力対象とするメッセージを指定する
- 指定したメッセージの重要度表示を変更して出力する

-s 出力有無

メッセージの出力，又は出力抑止を指定します。

Y：メッセージを出力します。

N：メッセージの出力を抑止します。

イベントログへのメッセージ出力抑止については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

-c ALL

このオプションは，HiRDB がイベントログに出力するすべてのメッセージを出力抑止する場合に指定します。

このオプションを指定する場合は，-s オプションに N を指定します。

-l メッセージの重要度

出力されるメッセージの変更後の重要度を指定します。

このオプションを指定する場合は，-s オプションに Y を指定します。

E：エラーメッセージ

W：警告メッセージ

Q：応答待ちメッセージ

I：インフォメーションメッセージ

イベントログに出力するメッセージの重要度の変更については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

-m メッセージ ID〔，メッセージ ID〕…

出力，又は出力抑止をするメッセージ ID を指定します。ただし，メッセージ ID のハイフン（-）以降の重要度は記述しません（KFPS01820-E の場合には KFPS01820 を指定します）。

このオプションに，元からイベントログに出力されないメッセージ ID を指定しても，指定しない場合と同じです。

《オペランドの規則》

- このオペランドは，複数行指定できます。
- ALL 指定，及びメッセージ ID 指定によって，同一メッセージに対する制御行が複数存在する場合は，後から指定した行が有効となります。

《ほかのオペランドとの関連》

- pd_mlg_file_size：メッセージの出力抑止をする場合，メッセージログファイルの最大容量を指定します。
- pd_mlg_msg_log_unit：メッセージを出力する場合，メッセージログの出力先（システムマネジャユニット，又はメッセージ出力元のユニット）を指定します。

3 ユニット制御情報定義

この章では，ユニット制御情報定義の各オペランドの内容について説明します。

3.1　オペランドの形式

ユニット制御情報定義ではユニットの情報を定義します。ここでは，ユニット制御情報定義の各オペランドの形式について説明します。なお，表中の「番号」は，「3.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

●影響分散スタンバイレス型系切り替え機能を使用される方へ

影響分散スタンバイレス型系切り替え機能を使用する場合，指定できるオペランドが決まっています。詳細については，「付録 F　影響分散スタンバイレス型系切り替え機能使用時に指定できるオペランド一覧（ユニット制御情報定義）」を参照してください。

(1)　システム構成

番号	形式
1	set pd_unit_id = ユニット識別子
2	〔set pd_hostname = ホスト名〕

(2)　同時実行最大数

番号	形式
3	〔set pd_max_server_process = 最大同時起動サーバプロセス数〕※

(3)　HiRDB の処理方式

番号	形式
4	〔set pd_server_entry_queue = spnfifo｜fifo｜loop〕※
5	〔set pd_thdlock_wakeup_lock = Y｜N〕※
6	〔set pd_thdlock_pipe_retry_interval = スレッド間ロックの解放調査間隔〕※
7	〔set pd_thdlock_retry_time = スレッド間ロックスリープ時間〕※
8	〔set pd_thdspnlk_spn_count = スレッド間スピンロックのスピン回数〕※
9	〔set pd_db_io_error_action = dbhold｜unitdown〕※

(4)　全面回復処理

番号	形式
10	〔set pd_max_recover_process = 全面回復処理の並列実行プロセス数〕※

(5)　システム監視

番号	形式
11	〔set pd_watch_time = 最大応答待ち時間〕※
12	〔set pd_down_watch_proc = サーバプロセスの異常終了回数の上限値〔，監視間隔〕〕※

(6)　SQL 実行時間警告出力機能

番号	形式
13	〔set pd_cwaittime_report_dir = SQL 実行時間警告情報ファイルの出力先ディレクトリ〕※
14	〔set pd_cwaittime_report_size = SQL 実行時間警告情報ファイルの最大容量〕※

(7)　拡張 SQL エラー情報出力機能

番号	形式
15	〔set pd_uap_exerror_log_dir = SQL エラーレポートファイルの格納ディレクトリ〕※
16	〔set pd_uap_exerror_log_size = SQL エラーレポートファイルの最大サイズ〕※
17	〔set pd_uap_exerror_log_param_size = エラーログファイル及び SQL エラーレポートファイルへ出力するパラメタ情報の最大データ長〕※

(8)　排他制御

番号	形式
18	〔set pd_lck_wait_timeout = 排他待ち限界経過時間〕※
19	〔set pd_lck_release_detect = interval｜pipe〕※
20	〔set pd_lck_release_detect_interval = 排他解除検知インターバル時間〕※
21	〔set pd_lck_deadlock_info = Y｜N〕※
22	〔set pd_lck_deadlock_check = Y｜N〕※
23	〔set pd_lck_deadlock_check_interval = デッドロック監視時間間隔〕※

(9)　共用メモリ

番号	形式
24	〔set pd_shmpool_attribute = free｜fixed〕※
25	〔set pd_dbbuff_attribute = free｜fixed〕※

(10)　統計情報

番号	形式
26	〔set pd_stj_file_size = 統計ログファイルの最大容量〕※
27	〔set pd_stj_buff_size = 統計ログバッファ長〕※

(11)　RPC トレース情報

番号	形式
28	〔set pd_rpc_trace = Y｜N〕※
29	〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕※
30	〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕※

(12)　トラブルシュート情報

番号	形式
31	〔set pd_cancel_dump = put ｜ noput〕※
32	〔set pd_dump_suppress_watch_time =トラブルシュート情報の出力抑止時間〕※
33	〔set pd_pth_trace_max = 通信トレース格納最大数〕
34	〔set pd_spool_cleanup_interval = トラブルシュート情報の削除処理間隔〕※
35	〔set pd_spool_cleanup_interval_level = 日数〔，削除種別〕〕※
36	〔set pd_spool_cleanup = normal｜force｜no〕※
37	〔set pd_spool_cleanup_level = 日数〔，削除種別〕〕※
38	〔set pd_module_trace_max = モジュールトレース格納最大数〕※
39	〔set pd_module_trace_timer_level = モジュールトレース出力時刻取得方法〕※

(13)　グローバルバッファ

番号	形式
40	〔set pd_dbbuff_wait_interval = グローバルバッファの占有状態の調査間隔〕※
41	〔set pd_dbbuff_wait_spn_count = グローバルバッファの占有状態調査のスピンループ回数上限値〕※

(14)　ユニット用ステータスファイル

番号	形式
42	set pd_syssts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"
	〔set pd_syssts_file_name_2 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

番号	形式
	[set pd_syssts_file_name_3 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"]
	[set pd_syssts_file_name_4 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"]
	[set pd_syssts_file_name_5 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"]
	[set pd_syssts_file_name_6 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"]
	[set pd_syssts_file_name_7 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"]

(15)　ユニット用ステータスファイル（障害発生時）

番号	形式
43	[set pd_syssts_initial_error = stop | continue | excontinue]
44	[set pd_syssts_singleoperation = stop | continue]
45	[set pd_syssts_last_active_file = "論理ファイル名"]
46	[set pd_syssts_last_active_side = A | B]

(16)　セキュリティ

番号	形式
47	[set pd_audit = Y | N] ※
48	[set pd_aud_file_name = 監査証跡ファイル用の HiRDB ファイルシステム領域名] ※
49	[set pd_aud_max_generation_size = 1 監査証跡ファイルの最大容量] ※
50	[set pd_aud_max_generation_num = 監査証跡ファイルの最大数] ※
51	[set pd_aud_async_buff_size = 監査証跡の非同期出力時に使用するバッファ長] ※
52	[set pd_aud_async_buff_count = 監査証跡の非同期出力時に使用するバッファ面数] ※
53	[set pd_aud_async_buff_retry_intvl = 監査証跡の非同期出力時に使用するバッファの確保リトライ間隔] ※
54	[set pd_aud_sql_source_size = 監査証跡に出力する SQL 文のサイズ]
55	[set pd_aud_sql_data_size = 監査証跡に出力する SQL データのサイズ]

(17)　系切り替え機能

番号	形式
56	[set pd_ha_acttype = monitor | server]
57	[set pd_ha_unit = nouse]
58	[set pd_ha_switch_timeout = Y | N] ※
59	[set pd_ha_server_process_standby = Y | N]
60	[set pd_ha_agent = standbyunit | server | activeunits]

番号	形式
61	〔set pd_ha_max_act_guest_servers = 受け入れ可能なゲスト BES の最大数〕
62	〔set pd_ha_max_server_process = 受け入れユニット内の最大ユーザサーバプロセス数〕
63	〔set pd_ha_process_count = ゲスト BES 受け入れ後のユニット内常駐プロセス数〕
64	〔set pd_ha_resource_act_wait_time = リソース活性化の最大待ち時間〕※
65	〔set pd_ha_ipaddr_inherit = Y｜N〕※

(18)　HiRDB Datareplicator

番号	形式
66	〔set pd_rpl_hdepath = 抽出側 HiRDB Datareplicator 運用ディレクトリ名〕

(19)　通信処理

番号	形式
67	〔set pd_service_port = スケジューラプロセスのポート番号〕※
68	〔set pd_change_clt_ipaddr = 0｜1 〕※
69	〔set pd_registered_port = "ポート番号の予約範囲"〔, "ポート番号の予約範囲"〕…〕※
70	〔set pd_registered_port_check = Y｜N｜C｜W〕※
71	〔set pd_registered_port_level = 0｜1〕※
72	〔set pd_ipc_send_retrycount = プロセス間通信の送信リトライ処理の連続繰り返し回数〕※
73	〔set pd_ipc_send_retrysleeptime = プロセス間通信の送信リトライ処理のスリープ時間〕※
74	〔set pd_ipc_send_count = サーバ間の送信処理のリトライ回数〕※
75	〔set pd_ipc_recv_count = サーバ間の受信処理のリトライ回数〕※
76	〔set pd_ipc_inet_bufsize = サーバ間通信で使用する送受信バッファ長〕※
77	〔set pd_tcp_inet_bufsize = HiRDB クライアントとの通信で使用する送受信バッファ長〕※

(20)　Java

番号	形式
78	〔set pd_java_archive_directory = "JAR ファイル格納ディレクトリ"〕※
79	〔set pd_java_classpath = "Java クラスパス"〕※
80	〔set pd_java_runtimepath = "Java Runtime Environment のルートディレクトリ"〕※
81	〔set pd_java_libpath = "Java 仮想マシンのライブラリディレクトリ"〕※
82	〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※

(21)　外部 C ストアドルーチン

番号	形式
83	[set pd_c_library_directory = "C ライブラリファイル格納ディレクトリ"]

(22)　ワークファイル出力先ディレクトリの変更

番号	形式
84	[set pd_tmp_directory = ワークファイル出力先ディレクトリ名]

(23)　共用メモリ

番号	形式
85	[putenv SHMMAX 共用メモリセグメントサイズの上限値] ※

注※

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が仮定されます。

3.2　オペランドの説明

3.2.1　システム構成に関するオペランド

1）　pd_unit_id = ユニット識別子　～<識別子>((4 文字))

このオペランドは省略できません。必ず指定してください。

ユニット識別子を指定します。システム共通定義の pdunit -u オペランドに指定したユニット識別子を指定してください。

複数の HiRDB/シングルサーバを使用している場合は，システム間でユニット識別子が一意になるようにしてください。

HiRDB/パラレルサーバの場合は，システム内でユニット識別子が一意になるように指定してください。

《注意事項》

ユニット識別子を変更する場合は，次に示すファイルを再初期化する必要があります。

- ユニット用ステータスファイル
- システムログファイル

2）　pd_hostname = ホスト名　～<ホスト名>((1～32 文字))

ユニットを定義した現用系のサーバマシンの標準ホスト名を指定します。ホスト名は，ホスト名，IP アドレス，又は FQDN のどれかの指定形式で指定してください。

また，システム共通定義の pdunit オペランドの-x オプションに指定したホスト名を指定できます。ホスト名は，ホスト名，IP アドレス，又は FQDN のどれかの指定形式で指定してください。ただし，pdunit オペランドの-x オプションの指定形式と合わせてください。例えば，pdunit オペランドの-x オプションで IP アドレスを指定した場合は，このオペランドにも IP アドレスを指定してください。

《指定値の目安》

- このオペランドは，システム共通定義の pdunit 及び pdstart オペランドと関係が深いため，pdunit 及び pdstart オペランドの説明を読んでから，このオペランドを指定してください。
- コマンドプロンプトで hostname コマンドを実行して表示されたホスト名を指定してください。
- ホスト名は大文字と小文字を区別します。

《注意事項》

マルチコネクションアドレス機能を使用するために，pdstart オペランドの-n オプションを指定した場合は，このオペランドを省略できません。この場合，このオペランドには標準ホスト名を指定してください。

3.2.2　同時実行最大数に関するオペランド

3）　pd_max_server_process = 最大同時起動サーバプロセス数
～<符号なし整数>((50～10000))

ユニット内で，同時に起動するサーバのプロセス数の最大値を指定します。

サーバのプロセス数には，システムサーバ，各サーバ，ユティリティなどのプロセス数を含めます。システムサーバとは，HiRDB が内部的に使用するサーバのことです。

《指定値の目安》

- 通常はこのオペランドを省略してください。このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次に示す計算式の値が仮定されます。なお，変数の説明中に出てくるオペランドの値を変更した

場合は HiRDB が自動的に再計算します。また，最大同時起動サーバプロセス数が pd_max_server_process の最大値(10000)を超える場合は，各種定義値を小さくし，定義省略時の計算式が最大値内に収まるようにしてください。

省略値＝a＋b×（c＋30）＋70＋i＋j＋k＋n

ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合は 100 が仮定されます。

- このオペランドを指定する場合は次に示す計算式を参考にして求めてください。HiRDB/パラレルサーバの場合はユニットごとに求めてください。その結果のうち一番大きな値を目安としてください。

  推奨値の目安＝a＋b×（c＋d×e＋f＋3）＋d×g＋h＋i＋j＋k＋n

変数	変数の説明
a	HiRDB/シングルサーバの場合 pd_max_users の値 HiRDB/パラレルサーバの場合 ユニット内のサーバごとに次の計算式を実行して求めた値を合計してください。 バックエンドサーバの場合：pd_max_bes_process の値 ディクショナリサーバの場合：pd_max_dic_process の値 フロントエンドサーバの場合：pd_max_users の値 • pd_max_bes_process 又は pd_max_dic_process オペランドを省略している場合は pd_max_users の値で計算してください。 • ユニット内に複数のバックエンドサーバがある場合は，バックエンドサーバごとに計算してください。 • マルチフロントエンドサーバの場合，フロントエンドサーバごとに 1 を加算してください。 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_server_process オペランドの値を代入します。
b	HiRDB/シングルサーバの場合 1 HiRDB/パラレルサーバの場合 ユニット内サーバ数（システム共通定義の pdstart オペランドで該当ユニットに割り当てたサーバ数） • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_servers オペランドの値（省略している場合はデフォルト値）を加算します。
c	全面回復処理の並列実行プロセス数（pd_max_recover_process オペランドの値）
d	同時実行するユティリティの最大数（このオペランドを省略した場合 2 が仮定されます）
e	ユティリティがサーバごとに起動するプロセス数（10 としてください）
f	HiRDB がサーバ制御のために起動するプロセス数（7 としてください）
g	ユティリティがユニット単位に起動するプロセス数（10 としてください）
h	HiRDB がユニット制御のために起動するプロセス数（50 としてください）
i	HiRDB/シングルサーバの場合 1 HiRDB/パラレルサーバの場合 ユニット内バックエンドサーバ数 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_servers オペランドの値を加算します。

変数	変数の説明
j	**HiRDB/シングルサーバの場合** pd_max_ard_process オペランドの値 **HiRDB/パラレルサーバの場合** ユニット内バックエンドサーバ数×pd_max_ard_process オペランドの値＋ユニット内ディクショナリサーバ数×pd_max_ard_process オペランドの値 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，次に示す値を加算してください。 ゲスト BES の pd_max_ard_process オペランドの最大値×pd_ha_max_act_guest_servers オペランドの値
k	pd_process_terminator オペランドの指定値によって値が変わります。 • resident の場合：1 • fixed の場合（省略値）：pd_process_terminator_max オペランドの値 • nonresident の場合：0
n	**HiRDB/シングルサーバの場合** pd_dfw_awt_process オペランドの値 **HiRDB/パラレルサーバの場合** バックエンドサーバ数×pd_dfw_awt_process オペランドの値＋ディクショナリサーバ数×pd_dfw_awt_process オペランドの値 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，次に示す値を加算してください。 ゲスト BES の pd_dfw_awt_process オペランドの最大値×pd_ha_max_act_guest_servers オペランドの値

《ほかのオペランドとの関連》

このオペランドの指定値で，pd_process_count オペランドの上限値が制限されます。

《注意事項》

- この指定値にはユニット内のサーバ，ユティリティなどのプロセス数が含まれます。したがって，この値が小さ過ぎると，次に示す現象が発生することがあります。
  ・ユニット又はサーバの開始処理がエラーになります。
  ・トランザクションの回復処理ができなくなります。
  ・HiRDB の計画停止ができなくなります。
- 実際に起動できるプロセス数は，システムのリソースなどに影響されるため，リソースの調整やサーバの配置変更などが必要となる場合があります。

3.2.3　HiRDB の処理方式に関するオペランド

4)　pd_server_entry_queue = spnfifo | fifo | loop

UAP の多重実行時に HiRDB サーバプロセスで処理が競合すると，処理要求を一時的にキューイングすることがあります。このときの HiRDB の処理方式を指定します。なお，ここでいう処理の競合とは，HiRDB サーバプロセス上でトランザクションが稼働するときに，複数のプロセスが表や RD エリアなどの内部資源に対して行う排他制御の競合を意味しています。内部資源に対して排他制御を行えるプロセスは常に一つだけです。また，以下の説明でいうスピンとは，排他制御の実行権を得るための処理を意味しています。ほかのプロセスが排他制御の実行権を解放した場合，スピン中のプロセスが排他制御の実行権を得る可能性が高くなります。

spnfifo：

先に発生した処理要求を優先的に処理します。ただし，キューに登録される前に一定回数のスピンをするため，完全に順序が守られるわけではありません。これはバージョン 06-00 以前の処理方式です。

fifo：

先に発生した処理要求を spnfifo 指定時より優先的に処理します。キューに登録される前に一定回数のスピンをしないため，spnfifo 指定より処理順序は守られます。また，CPU の負荷を軽減する効果があります。

loop：

すべての処理要求を均等な優先度で処理します。キューに登録されている間は高速なスピンをします。loop を指定すると，UAP の多重実行時のレスポンスが向上することがあります。しかし，そのほかの指定より CPU に負荷が掛かります。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

UAP の多重実行時の処理性能が向上しない場合に指定値を変更してください。性能が向上することがあります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，spnfifo が仮定されます。

5)　pd_thdlock_wakeup_lock = Y | N

スレッド間ロックの解放通知方式を指定します。このオペランドに Y を指定することで，解放通知が確実に送信されることを保証します。

Y：

スレッド間ロックの解放通知をするときに，これとは別の一時的なロックを新たに取得します。

N：

スレッド間ロックの解放通知をするときに，これとは別の一時的なロックを新たに取得しません。

これは，バージョン 06-02 以前の HiRDB の処理方式です。

《指定値の目安》

このオペランドには Y を指定してください。

《注意事項》

N を指定した場合，ほかのトランザクションに比べて，実行時間が長いトランザクションが発生し，UAP の多重実行時のレスポンスが低下することがあります。

6)　pd_thdlock_pipe_retry_interval = スレッド間ロックの解放調査間隔

～<符号なし整数>((0～2147483647))（単位：マイクロ秒）

スレッド間ロックの解放調査を行う間隔をマイクロ秒単位で指定します。

《指定値の目安》

次の条件をすべて満たす場合にデフォルト値以上の値にすると，CPU 使用率が下がることがあります。

- pd_thdlock_wakeup_lock = Y を指定している
- UAP の多重実行時の処理性能が向上しない
- CPU 使用率が高い

ただし，ほかのトランザクションに比べて，実行時間が長いトランザクションが発生するようになります。

- 上記に該当しない場合は，このオペランドを指定しないでください。

《注意事項》

このオペランドにデフォルト値未満の値を指定すると，解放調査処理を短時間で繰り返すため，CPU 使用率が高騰します。

7)　pd_thdlock_retry_time = スレッド間ロックスリープ時間
～<符号なし整数>((1～1000000))（単位：マイクロ秒）

スレッド間ロックのスリープ時間をマイクロ秒単位で指定します。次に示す条件をすべて満たす場合にこのオペランドを指定すると，CPU 使用率が下がることがあります。

- CPU 使用率が非常に高い
- 性能が低下してでも CPU 使用率を下げたい
- pd_thdlock_sleep_func オペランドに 0 を指定している

上記に該当しない場合，このオペランドは指定しないでください。pd_thdlock_sleep_func と pd_thdlock_retry_time オペランドの指定値の組み合わせによる HiRDB の処理を次に示します。

pd_thdlock_sleep_func オペランドの値	pd_thdlock_retry_time オペランドの値	
	1～10000	10001～1000000
0	select()又は Sleep()を使用して指定したスレッド間ロックのスリープ時間だけ，各プロセスが待機します。	
1	sched_yield()又は SwitchToThread()を使用して，OS によってプロセスの割り当てを決定します（pd_thdlock_retry_time オペランドの値は無効となります）。※	select()又は Sleep()を使用して指定したスレッド間ロックのスリープ時間だけ，各プロセスが待機します。

注※　プロセスが待機しないため，CPU の使用率が高騰します。

《指定値の目安》

- このオペランドを指定する場合，最初は 10000 を指定してください。
- 10000 を指定して CPU 使用率が高くなり過ぎる場合は，指定値を大きくしてください。

《注意事項》

- 指定値を小さくしても，性能が変わらない場合があります。
- 1000 以上の値を指定すると，性能が低下することがあります。

《ほかのオペランドとの関連》

- pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，このオペランドの省略値は 10000 になります。

8)　pd_thdspnlk_spn_count = スレッド間スピンロックのスピン回数
～<符号なし整数>((0～2147483647))

スレッド間スピンロックのスピン回数を指定します。次に示す条件をすべて満たす場合にこのオペランドを指定すると，性能が向上することがあります。それ以外の場合は指定する必要はありません。

- CPU 使用率に余裕がある
- CPU 使用率を上げてでも性能を良くしたい

《指定値の目安》

- このオペランドを指定する場合は 512 より大きい値を指定してください。
- 指定値は OS の種類，プロセッサの種類，マシンの性能，ディスクの性能，UAP の内容，UAP の多重実行数などに依存するため，明確な目安はありません。指定値の変更と性能測定を行って適切な値を求めてください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，512 が仮定されます。

《注意事項》

- このオペランドの値が大き過ぎると，CPU 使用率が高くなって OS がスローダウンするなどの弊害が発生することがあります。その場合はこのオペランドの値を小さくしてください。
- このオペランドの値を大きくしても，性能が向上しないこともあります。

9）pd_db_io_error_action = dbhold | unitdown

RD エリア（マスタディレクトリ用 RD エリアを除く）の入出力エラーが発生したときの HiRDB の処理を指定します。なお，マスタディレクトリ用 RD エリアに入出力エラーが発生した場合，このオペランドの指定に関係なく常に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。RD エリアの入出力エラーが発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

ここでいう入出力エラーとは, HiRDB がファイルを操作したときに HiRDB が判別できない理由でファイル操作に失敗したエラーを意味し，HiRDB ファイルシステムに対するアクセス要求から返されるエラーコードに-1544 が出力されるエラーのことです。

dbhold：

RD エリアの入出力エラーが発生した場合，その RD エリアを障害閉塞します。

unitdown：

RD エリアの入出力エラーが発生した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。ただし，異常終了後に再度入出力エラーが発生した場合，その RD エリアを障害閉塞します。再び，unitdown の指定を有効にするには，次に示すどちらかの方法を実行してください。

- HiRDB を正常開始する
- システム構成変更コマンド（pdchgconf コマンド）を実行する

《指定値の目安》

マニュアル「HiRDB Version 8 システム運用ガイド」の「RD エリアの入出力エラーが発生したときの対処方法」を参照して，このオペランドの指定値を決めてください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，dbhold が仮定されます。

《注意事項》

- unitdown を指定したときに入出力エラーが発生すると HiRDB が異常終了するため，次の場合に，処理対象の RD エリアが障害閉塞することがあります。

  ・更新前ログ取得モード又はログレスモードで UAP やユティリティを実行している場合

  ・CREATE TABLE の RECOVERY オペランドで NO を指定してログレスモードにしたユーザ LOB 用 RD エリアに対して，UAP 又はユティリティを実行している場合

- unitdown を指定する場合は，データベースの更新ログを取得する運用にしてください。データベースの更新ログを取得しない運用にする場合は，UAP 又はユティリティの実行前にバックアップを取得して，RD エリアの障害閉塞を回復できるようにしてください。バックアップの取得については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- 開始処理中又は終了処理中の入出力エラーについては，unitdown を指定しても HiRDB を異常終了しません。
- データベース回復ユティリティ（pdrstr）での回復処理中は，unitdown を指定していても HiRDB を異常終了しません。この場合は，データベース回復ユティリティ（pdrstr）を再度実行して回復してください。

3.2.4　全面回復処理に関するオペランド

10）　pd_max_recover_process = 全面回復処理の並列実行プロセス数

～<符号なし整数>((1～10))

全面回復処理をするときの回復処理プロセス（REDO プロセス）数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（ディクショナリサーバ又はバックエンドサーバ）当たりの回復処理プロセス（REDO プロセス）数になります。

《前提条件》

rawI/O 機能を使用する必要があります。

《指定値の目安》

- RD エリアを定義した論理ボリュームの数が 3 以上のとき（1 サーバ当たり）：3
- RD エリアを定義した論理ボリュームの数が 3 未満のとき（1 サーバ当たり）：論理ボリューム数
- このオペランドの値を大きくすると，全面回復処理時の入出力並列度が増すため，回復処理時間を短縮できます。ただし，「このオペランドの値×サーバ数」分のプロセスが起動されるため，前記の指定値の目安と HiRDB のリソースを考慮して指定値を決めてください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，3 が仮定されます。

3.2.5　システム監視に関するオペランド

11）　pd_watch_time = 最大応答待ち時間

～<符号なし整数>((0～65535))（単位：秒）

このオペランドは旧バージョンとの互換性保証のための指定です。したがって指定する必要はありません。

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB サーバプロセスが他の HiRDB サーバプロセス（ディクショナリサーバ/バックエンドサーバ）から応答が返ってくるまでの最大待ち時間を指定します。

HiRDB サーバプロセスの中断を許したくない時間を指定しますが，指定した時間を過ぎても該当 HiRDB サーバプロセスの処理を確実に中断するわけではありません。

《利点》

HiRDB クライアントが SQL 実行を中断（クライアントプロセスの強制終了など），又はコマンド・ユティリティの実行を中断（強制停止等）しても，HiRDB サーバが当該 SQL やコマンド・ユティリティの実行を中止しない場合，HiRDB サーバはこの SQL やコマンド・ユティリティの実行を続

けて長時間にわたって排他資源などを占有することがあります。このオペランドを指定すると，この占有時間を短くできます。

《指定値の目安》

次のうちの最も長い時間を指定してください。

- クライアント環境定義の PDCWAITTIME オペランドで指定する時間
- pd_lck_wait_timeout オペランドで指定する時間
- 実行時間が最も掛かる SQL の処理時間
- 実行時間が最もかかるコマンド・ユーティリティの時間値

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

- このオペランドに 0 を指定した場合は，SQL 実行時間の監視をしません。
- SQL の最大実行時間を設定する場合は，このオペランドに 0 を設定し，クライアント環境変数 PDCWAITTIME で SQL 最大実行時間を設定することをお勧めします。クライアント環境定義 PDCWAITTIME については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
- このオペランドに SQL 実行時間より短い値を指定した場合，SQL 実行中に処理が中断され，HiRDB クライアントに SQL エラーか，又は HiRDB サーバの異常終了が報告されることがあります。
- HiRDB/シングルサーバの場合にこのオペランドを指定しても，SQL 最大実行時間を監視しません。ただし，このオペランドを指定した場合，その指定値が pd_lck_wait_timeout オペランドの省略値として使用されます。このため，HiRDB/シングルサーバの場合はこのオペランドを省略することをお勧めします。

12）　pd_down_watch_proc = サーバプロセスの異常終了回数の上限値〔，監視間隔〕

このオペランドは，HiRDB のサーバプロセスの異常終了回数を監視するオペランドです。PDCWAITTIME オーバ，又はアボートによって異常終了するプロセスを監視対象とします。

サーバプロセスの異常終了が多発すると，新たなサービスを受け付けられないことがあります。しかし，サーバプロセスの異常終了では HiRDB は異常終了しないため，実質オンライン停止状態になります。このオペランドを指定すると，HiRDB の再開始によってこの状態を抜け出せます。

サーバプロセスの異常終了回数の上限値：～<符号なし整数>((0～65535))

サーバプロセスの異常終了回数がこのオペランドに指定した値を超えた場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を異常終了させます。これを**プロセスの異常終了回数監視機能**といいます。プロセスの異常終了回数監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

HiRDB/シングルサーバの場合は，シングルサーバプロセスの異常終了回数がカウントされます。HiRDB/パラレルサーバの場合は，ユニット内のフロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバプロセスの異常終了回数の合計値がカウントされます。

なお，0 を指定すると，サーバプロセスの異常終了回数を監視しません。

監視間隔：～<符号なし整数>((10～3600))（単位：秒）

サーバプロセスの異常終了回数を監視する時間を秒単位で指定します。

例えば，100 を指定すると，100 秒間隔でサーバプロセスの異常終了回数を監視します。

《利点》

- HiRDB の再開始によってメモリ及びリソース状態がリフレッシュされて処理効率が向上します。
- サーバプロセスの異常終了が多発した場合，HiRDB が異常終了するため，すぐに系を切り替えられます。

《注意事項》

- サーバプロセスが異常終了すると，KFPS01820-E メッセージが出力されます。pdcancel コマンドでサーバプロセスが異常終了したときにも KFPS01820-E メッセージが出力されますが，これは異常終了回数のカウントに含まれません。
- 相互系切り替え構成の場合，系が切り替わると同一サーバマシンで複数の HiRDB を稼働するため，逆にトラフィックが上昇して効果が得られないことがあります。このオペランドを指定する場合は，異常終了した系で HiRDB を再開始する運用をお勧めします。

《オペランドの規則》

サーバプロセスの異常終了回数の上限値を指定しないで監視間隔だけの指定はできません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- サーバプロセスの異常終了回数の上限値：0
- 監視間隔：600

《備考》

- プロセスの異常終了回数監視機能によって HiRDB が異常終了する場合，KFPS01821-E 及び KFPS00729-E メッセージを出力します。
- サーバプロセスが異常終了する要因と異常終了回数のカウント対象を次に示します。

サーバプロセスが異常終了する要因	異常終了回数のカウント対象			
	シングルサーバプロセス	フロントエンドサーバプロセス	ディクショナリサーバプロセス	バックエンドサーバプロセス
クライアント環境定義の PDCWAITTIME オペランドの値を超えた	○	○	×※1	×※1
pdcancel コマンド	×	×※2	×	×
内部強制終了（HiRDB が内部的に SIGKILL を発行してプロセスを終了させる場合）	○※3	○※3	×※1	×※1
アボート	○	○	○	○
XA 接続の UAP でロールバックが発生した	○	○	×	×
上記以外のプロセス異常終了	○	○	○	○

（凡例）

○：プロセスの異常終了回数にカウントされます。

×：プロセスの異常終了回数にカウントされません。

注※1

トランザクションブランチが発生した状態で障害を検知した場合，同一トランザクションブランチから発生したフロントエンドサーバプロセスの異常終了がカウントの対象になります。

注※2

pdcancel コマンドでバックエンドサーバプロセス又はディクショナリサーバプロセスを強制終了すると，フロントエンドサーバプロセスを内部的に強制終了します。この場合，フロントエンドサーバプロセスの異常終了がカウントされることがあります。

注※3

OLTP システムによるグローバルトランザクションが発生した状態で障害を検知した場合，同一グローバルトランザクションから発生したシングルサーバプロセス又はフロントエンドサーバプロセスの異常終了がカウントの対象になります。

3.2.6　SQL 実行時間警告出力機能に関するオペランド

13）　pd_cwaittime_report_dir = SQL 実行時間警告情報ファイルの出力先ディレクトリ～<パス名>

SQL 実行時間警告情報ファイルを出力する場合に，出力先のディレクトリを絶対パス名で指定します。ここで指定したディレクトリ下に SQL 実行時間警告情報ファイルが二つ（pdcwwrn1 及び pdcwwrn2）作成されます。

HiRDB/パラレルサーバの場合，警告対象となった SQL を発行する UAP が接続するフロントエンドサーバがあるサーバマシンに SQL 実行時間警告情報ファイルを出力します。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，SQL 実行時間警告情報ファイルを出力しません。ただし，警告メッセージ（KFPA20009-W）は出力されます。

14）　pd_cwaittime_report_size = SQL 実行時間警告情報ファイルの最大容量～<符号なし整数>((2048～2147473627))（単位：バイト）

SQL 実行時間警告情報ファイルの最大容量を指定します。このオペランドに指定した値は SQL 実行時間警告情報ファイル一つ分の容量です。SQL 実行時間警告情報ファイルは二つ作成されるため，指定値に注意してください。例えば，10,000 を指定すると，ディレクトリ下に最大 10,000 バイトのファイルが二つ作成されます。

《指定値の目安》

次に示す計算式を目安にこのオペランドの指定値を決めてください。

{1280 ＋ SQL 文のサイズ（バイト）}　×ファイルに格納する警告情報数

SQL 文に注釈（コメント）や SQL の最適化を指定している場合は，注釈及び SQL の最適化の指定サイズ（バイト）も，SQL 文のサイズに含めてください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，100000 が仮定されます。

《備考》

- SQL 実行時間警告情報ファイルへの出力容量がこのオペランドの値を超えると，出力先をもう片方のファイルに切り替えます。これを繰り返して二つのファイルを交互に使用します。このとき，切り替え先の古い情報は削除されます。
- 1 回で出力する SQL 実行時間警告情報の大きさがファイル容量を超えた場合，SQL 実行時間警告情報をすべて出力しません。ファイル容量までの SQL 実行時間警告情報を出力します。この場合，SQL 実行時間警告情報の最後に#を付けます。

3.2.7　拡張 SQL エラー情報出力機能に関するオペランド

15）　pd_uap_exerror_log_dir = SQL エラーレポートファイルの格納ディレクトリ
～< 255 文字以内のパス名>

SQL エラーレポートファイルの格納ディレクトリを絶対パス名で指定します。

指定したディレクトリに，二つの SQL エラーレポートファイルが作成されます。ファイル名は pduaperrlog1 と pduaperrlog2 です。

拡張 SQL エラー情報出力機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，SQL エラー情報が SQL エラーレポートファイルに出力されません。

《注意事項》

指定したパス名のアルファベットの大文字と小文字を同一文字として扱います。

16）　pd_uap_exerror_log_size = SQL エラーレポートファイルの最大サイズ
～<符号なし整数>((2048～2147483647))（単位：バイト）

SQL エラーレポートファイルの最大サイズを指定します。このオペランドで指定した値は，二つ作成される SQL エラーレポートファイルのうち一つ分の容量になります。SQL エラーレポートファイルへの出力容量がこのオペランドの値を超えると，出力先をもう片方のファイルに切り替えます。これを繰り返して二つのファイルを交互に使用します。1 回の処理で出力する SQL エラー情報のサイズがこのオペランドの値を超える場合，先頭からこの指定値－1 バイトまで（このオペランドに 1000000 を指定した場合 999999）の SQL エラー情報に#（シャープ 1 文字）を付けて出力します。

《指定値の目安》

このオペランドの指定値は，残しておきたい SQL エラー情報件数を考慮して決定してください。計算式を次に示します。

(A+B) ×保持したい件数

- A = 1100+SQL 文のサイズ（単位：バイト）

  パラメタ情報の出力サイズを除いた SQL エラー情報の 1 件当たりのサイズです。SQL 文に注釈（コメント）や SQL 最適化指定を記述している場合は，注釈（コメント）及び SQL 最適化指定分のサイズも，SQL 文のサイズに含めます。注釈及び SQL 最適化指定については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。

- B = (↑pd_uap_exerror_log_param_size オペランドの値÷16↑+1) ×89×パラメタの個数

  パラメタ情報の出力サイズです。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，1000000 が仮定されます。

17）　pd_uap_exerror_log_param_size = エラーログファイル及び SQL エラーファイルへ出力するパラメタ情報の最大データ長

～<符号なし整数>((0～32008))（単位：バイト）

エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。

- 1 以上を指定した場合

  エラーログファイル及び SQL エラーレポートファイルにパラメタ情報を出力します。

- 0 を指定した場合

  エラーログファイル及び SQL エラーレポートファイルにパラメタ情報を出力しません。

可変長文字列型，BLOB 型，及び BINARY 型の場合，データ長の領域もこの指定値に含まれます。

エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の大きさがこのオペランドの指定値を超えた場合，指定値の長さまでパラメタ情報を出力します。この場合，超過分は切り捨てられます。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDUAPEXERLOGPRMSZ オペランドを指定してください。このオペランドと，クライアント環境定義 PDUAPEXERLOGPRMSZ の両方を指定した場合は，クライアント環境定義 PDUAPEXERLOGPRMSZ が優先されます。

PDUAPEXERLOGPRMSZ オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

3.2.8　排他制御に関するオペランド

18）　pd_lck_wait_timeout = 排他待ち限界経過時間

～<符号なし整数>((0～65535))（単位：秒）

排他待ち時間を監視する最大時間を秒単位で指定します。排他待ち時間とは，排他要求が待ち状態になってから解除されるまでの時間のことです。

指定した時間内に待ち状態が解除されない場合，SQL はエラーリターンします。0 を指定した場合，排他待ち時間を監視しないで，待ち状態が解除されるまで待ち続けます。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次のうち，大きい方の値が仮定されます。

- 180
- pd_watch_time の値

ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合は 180 が仮定されます。

《注意事項》

このオペランドを省略すると，pd_watch_time オペランドの値が省略値として仮定されることがあります。pd_watch_time オペランドは HiRDB/シングルサーバに指定しても無効になりますが，pd_lck_wait_timeout オペランドの省略値に使用されることがあります。このため，HiRDB/シングルサーバの場合は pd_watch_time オペランドを省略することをお勧めします。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDLCKWAITTIME を指定してください。PDLCKWAITTIME については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_lck_deadlock_check
- pd_lck_deadlock_check_interval

19）pd_lck_release_detect = interval | pipe

このオペランドでは，HiRDB が排他の解除を検知する方法（資源を占有している処理が，資源の占有を解除したかどうかを知る方法）について指定します。

interval：一定間隔で排他管理用領域を参照して排他解除状態を調べます。

pipe：pipe ファイル（プロセス間通信のパイプ）を使用して排他解除通知を受け取ります。

《指定値の目安》

このオペランドの指定値を変更するときの目安を次に示します。

指定値	HiRDB の処理方式	利点・適用基準
interval	共用メモリにある排他管理用領域を参照して，排他が解除されたかどうかを検知します。HiRDB は排他管理用領域を一定間隔（一定のインターバル）で参照しています。インターバル時間は，pd_lck_release_detect_interval オペランドで指定します。	資源を占有している処理が排他を解除しても，次の排他管理用領域の参照時までそれを検知できません。このため，1 トランザクションの処理時間が短い UAP では，排他待ち時間が長くなることがあります。ただし，この待ちでは CPU に負荷を掛けたり，ファイルをオープンしたりしません。 pd_lck_release_detect_interval オペランドに小さい値を指定すると，CPU 利用率が上がり過ぎてスループットが伸びないことがあります。低速な CPU を使用している場合に，CPU 負荷を軽減する効果があります。
pipe	セマフォを使用して排他が解除されたかどうかを検知します。資源を占有している処理から排他待ちをしている処理に排他解除が通知されます。排他占有プロセスが排他を解除した時点で，排他待ちプロセスは排他の解除を検知できます。	• 1 トランザクションの処理時間が pd_lck_release_detect_interval オペランドの値以下の場合は，スループットが向上します。 • 排他待ちが多発した場合，CPU に負荷が掛かります。 • 高速な CPU を使用している場合は，スループットを向上させる効果があります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，HiRDB/シングルサーバの場合は pipe が，HiRDB/パラレルサーバの場合は interval が仮定されます。

20）　pd_lck_release_detect_interval = 排他解除検知インターバル時間
～<符号なし整数>((1～1000))（単位：ミリ秒）

排他管理用領域の参照は一定のインターバルをおいて行われます。このオペランドにはそのインターバル時間を指定します。

・このオペランドに 49 以下を指定した場合

インターバル時間はこのオペランドで指定した値から始まり，最大 50 ミリ秒になります。

・このオペランドに 50 以上を指定した場合

インターバル時間は 50 ミリ秒から始まり，最大このオペランドで指定した値になります。

《前提条件》

pd_lck_release_detect オペランドに interval を指定している必要があります。

なお，HiRDB/パラレルサーバの場合は interval が省略値です。

《指定値の目安》

- 基本的にはこのオペランドを省略してください。
  短時間の間に排他が多発し，排他獲得待ちが発生しているため，トランザクション処理性能が大幅に低下している場合に指定値の変更を検討してください。
- 指定値を小さくすると，排他待ちが多発したときに CPU に負荷が掛かります。
- 指定値を大きくすると，排他待ち時間が長くなることがあります。
- 統計解析ユティリティのシステムの稼働に関する統計情報の「排他待ち時間（WAIT TIME)」を参考にして指定値を決定してください。統計情報に出力された排他待ち時間がこのオペランドの値に比べて短い場合は，このオペランドの値を小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，10 が仮定されます。

21）　pd_lck_deadlock_info = Y | N

デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報を出力するかどうかを指定します。デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報は，%PDDIR%¥spool¥pdlckinf 下に出力されます。デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

Y：デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報を出力します。

N：デッドロック情報，タイムアウト情報，及び排他資源管理テーブル情報を出力しません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，Y が仮定されます。

《ほかのオペランドとの関連》

- pd_lck_deadlock_check オペランドに N を指定した場合は，このオペランドには常に N が仮定されます。
- このオペランドは pd_lck_deadlock_check オペランドと関連があります。

22）　pd_lck_deadlock_check = Y | N

デッドロックの発生を監視するかどうかを指定します。

Y：デッドロックの発生を監視します。

N：デッドロックの発生を監視しません。

《指定時の目安》

デッドロックが発生しない業務システムでは，このオペランドにNを指定するとSQLの実行性能が改善することがあります。特に，デッドロックの検出方法がインターバル監視方式の場合，排他制御用プールパーティション数が増加するとデッドロックを検出するたびに排他制御の性能が劣化することがあるため，デッドロックが発生しない業務システムを構築した上でこのオペランドにNを指定することをお勧めします。

デッドロックが発生する業務システムでは，このオペランドにYを指定してください。Nを指定すると，デッドロックが発生した場合，pd_lck_timeoutオペランドに指定した時間が経過するまでSQLが終了しません。また，HiRDBがデッドロック情報を出力しないため，デッドロックが発生した要因が分からなくなるおそれがあります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，Yが仮定されます。

《注意事項》

このオペランドにNを指定すると，デッドロックが発生したトランザクションはエラーとなりません。この場合，トランザクションは次の要因でキャンセルされます。

- 排他待ち時間を監視する最大時間が経過し，SQLがエラーリターンする
- HiRDBクライアントの最大待ち時間が経過し，要求がUAPにエラーリターンする

《クライアント環境定義との関連》

このオペランドは次に示すクライアント環境定義と関連があります。

- PDCWAITTIME
- PDLCKWAITTIME

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_deadlock_info
- pd_lck_pool_partition
- pd_fes_lck_pool_partition
- pd_lck_deadlock_check_interval
- pd_lck_wait_timeout

23)　pd_lck_deadlock_check_interval = デッドロック監視時間間隔

～<符号なし整数>((1～2000000000))（単位：ミリ秒）

インターバル監視方式でデッドロックの発生を監視する場合の，監視を実施する間隔を指定します。

《前提条件》

このオペランドの前提条件は，サーバ種別によって異なります。

シングルサーバ，バックエンドサーバ，又はディクショナリサーバの場合は，次に示す条件をすべて満たしている必要があります。

- pd_lck_deadlock_checkオペランドにYを指定している
- pd_lck_pool_partitionオペランドに2以上の値を指定している

フロントエンドサーバの場合は，次に示す条件をすべて満たしている必要があります。

- pd_lck_deadlock_checkオペランドにYを指定している
- pd_fes_lck_pool_partitionオペランドに2以上の値を指定している

《指定時の目安》

デッドロックが発生してから検出するまでの時間を短くしたい場合は，このオペランドの指定値を小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，1000 が仮定されます。

《注意事項》

- このオペランドの指定値を小さくし過ぎると，システムのパフォーマンスが低下する場合があります。
- このオペランド指定値を大きくし過ぎると，次の要因でトランザクションがキャンセルされることがあります。
  - ・排他待ち時間を監視する最大時間が経過し，SQL がエラーリターンする
  - ・HiRDB クライアントの最大待ち時間が経過し，要求が UAP にエラーリターンする

《クライアント環境定義との関連》

このオペランドは次に示すクライアント環境定義と関連があります。

- PDCWAITTIME

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_deadlock_check
- pd_lck_pool_partition
- pd_fes_lck_pool_partition
- pd_lck_wait_timeout

3.2.9　共用メモリに関するオペランド

24)　pd_shmpool_attribute = free | fixed

HiRDB のユニットコントローラが使用する共用メモリをメモリ上に固定するかどうかを指定します。なお，Windows のバージョンによって，サポートの有無が異なります。Windows の前提条件については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「システム設計」の「共用メモリのページ固定」を参照してください。サポートされていない場合は，このオペランドに fixed を指定しても無効になります。

free：

メモリ上に固定しません。実メモリ量によっては，別の共用メモリのページングが発生し，性能に影響する場合があります。

fixed：

メモリ上に固定します。

《利点》

HiRDB が使用する共用メモリをメモリ上に固定（fixed 指定）すると，共用メモリのページングを防止できます。このため，共用メモリに対するアクセス性能が向上します。

《指定値の目安》

算出した共用メモリサイズとサーバマシンの実メモリから，メモリを固定するかどうかを決定してください。HiRDB がメモリを固定できるサイズに上限があるため，メモリを固定するとほかのメモ

リのページングが多発することがあります。したがって，実メモリや全使用量に対する共用メモリプールの占める割合をよく検討する必要があります。

ユニットコントローラが使用する共用メモリの計算式，及び各サーバが使用する共用メモリについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，free が仮定されます。

《注意事項》

ページ固定に対応している Windows で fixed を指定する場合の注意事項を次に示します。

- pdntenv コマンドを使用して，共用メモリをページングファイルに割り当てる必要があります。pdntenv コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。
- 共用メモリは Windows の Large Page でのページサイズに切り上げて取得するため，メモリ所要量を見積もるときは注意してください。Windows の Large Page のページサイズについては，pdntenv -os コマンドで確認してください。

25)　pd_dbbuff_attribute = free | fixed

グローバルバッファが使用する共用メモリを実メモリ上に固定するかどうかを指定します。Windows のバージョンによって，サポートの有無が異なります。Windows の前提条件については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「システム設計」の「共用メモリのページ固定」を参照してください。サポートされていない場合は，このオペランドに fixed を指定しても無効になります。

free：

グローバルバッファが使用する共用メモリを実メモリにページ固定しません。

fixed：

グローバルバッファが使用する共用メモリを実メモリにページ固定します。

《利点》

グローバルバッファが使用する共用メモリを実メモリにページ固定（fixed 指定）すると，共用メモリのページングを防止できます。このため，共用メモリに対するアクセス性能が向上します。

《指定値の目安》

- 算出した共用メモリサイズとサーバマシンの実メモリから，メモリを固定するかどうかを決定してください。実メモリに比べて多大なメモリをページ固定すると，ページングの多発や仮想メモリ不足の原因になります。したがって，実メモリや全仮想メモリに対する共用メモリプールの占める割合を検討する必要があります。グローバルバッファが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- ページ固定は，実メモリサイズからページ固定する共用メモリサイズを減算したサイズが，スワップ領域サイズからページ固定する共用メモリサイズを減算したサイズの 1/2 以上にならないことを目安にしてください。

《ほかの機能との関連》

- fixed を指定した場合，動的変更したグローバルバッファが使用する共用メモリも実メモリ上に固定されます。そのため，実メモリサイズを十分考慮してグローバルバッファを追加又は変更してください。

《注意事項》

ページ固定に対応している Windows で fixed を指定する場合の注意事項を次に示します。

- pdntenv コマンドを使用して，共用メモリをページングファイルに割り当てる必要があります。pdntenv コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。
- 共用メモリは Windows の Large Page でのページサイズに切り上げて取得するため，メモリ所要量を見積もるときは注意してください。Windows の Large Page のページサイズについては，pdntenv -os コマンドで確認してください。
- SHMMAX オペランドに Windows の Large Page のページサイズに切り上げた値を指定してください。
- HiRDB 起動時にグローバルバッファプールが使用する共用メモリがメモリ上に固定できない場合，KFPH23045-W メッセージを出力します。その場合，fixed を指定しても共用メモリの一部又はすべてがメモリ上にページ固定されないで，処理を続行します。

3.2.10　統計情報に関するオペランド

26)　pd_stj_file_size = 統計ログファイルの最大容量

～<符号なし整数>((64～1000000))（単位：キロバイト）

統計ログファイルの最大容量をキロバイト単位で指定します。統計ログファイルは二つあり，統計ログの出力量がここで指定した最大容量に達したときに統計ログファイルをスワップします。

《指定値の目安》

- このオペランドの指定値の見積もり方法については，「付録 C.1　統計ログファイル（pd_stj_file_size）の見積もり式」を参照してください。
- pd_stj_buff_size オペランドとの間に次に示す関係が保たれるように指定してください。

  pd_stj_file_size≧pd_stj_buff_size×2

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，1024 が仮定されます。

27)　pd_stj_buff_size = 統計ログバッファ長

～<符号なし整数>((32～512))（単位：キロバイト）

統計ログバッファの大きさを指定します。

《指定値の目安》

次に示す統計情報を出力しない場合は，仮定値である 32 で問題ありません。

- SQL オブジェクト実行に関する統計情報
- SQL オブジェクト転送に関する統計情報

これらの統計情報を出力する場合は，次に示す計算式で求めた値に 32 を加算して指定してください。ただし，512 を超えた場合は，512 を指定してください。

(a÷1024) × (0.03÷b)

a：統計ログ出力量（バイト）

「付録 C.1　統計ログファイル（pd_stj_file_size）の見積もり式」を参照してください。

b：統計情報の出力時間（秒）

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，32 が仮定されます。

3.2.11　RPC トレース情報に関するオペランド

28）　pd_rpc_trace = Y | N

RPC トレース情報を取得するかどうかを指定します。RPC トレース情報には, HiRDB の保守情報が出力されます。

通常，このオペランドは省略してください。

Y：RPC トレースを取得します。

N：RPC トレースを取得しません。

《注意事項》

このオペランドに Y を指定すると，通信性能が低下します。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，N が仮定されます。

29）　pd_rpc_trace_name = "RPC トレースを取得するファイル名称"

～< 254 文字以内のパス名>

RPC トレースファイルの名称を絶対パス名で指定します。RPC トレースファイルとして，指定したファイル名の後に 1，2，l（エル）を付けた三つのファイルが作成されます。

《注意事項》

このオペランドで指定したディレクトリ下に，最大「pd_rpc_trace_size の値× 2」の大きさのファイルが作成されます。ファイル容量に注意してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

%PDDIR%¥spool¥rpctr

30）　pd_rpc_trace_size = RPC トレースを取得するファイルの容量

～<符号なし整数>((1024～2147483648))（単位：バイト）

RPC トレースファイルの容量をバイト単位で指定します。

このオペランド指定値は，"RPC トレースファイル"1 及び"RPC トレースファイル"2 のそれぞれの容量となります。

《指定値の目安》

RPC トレースは，すべてのプロセス間の通信電文が一つのファイルに時系列に蓄積されます。その量がこの指定値を超えると，もう一方のファイルにトレースデータの出力を切り替えます。古いトレースは上書きされて消去されます。このため，指定値が小さいと十分なトレースを取得できなくなり, トラブルシュートが困難になることがあります。したがって, このオペランドには 1,000,000 以上の値を指定してください。

《注意事項》

このオペランドで指定した値は RPC トレースファイル l（エル）の容量に影響を与えません。RPC トレースファイル l（エル）の容量は 0 バイト固定です。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，4096 が仮定されます。

3.2.12　トラブルシュート情報に関するオペランド

31）pd_cancel_dump = put | noput

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

次に示す場合にトラブルシュート情報を取得するかどうかを指定します。

- クライアント環境定義の PDCWAITTIME オペランドで指定した監視時間内に SQL が終了しなかった場合
- pdcancel コマンドで実行中の UAP を中断させた場合

取得されるトラブルシュート情報については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

put：

トラブルシュート情報を取得します。トラブルシュート情報が%PDDIR%¥spool 下のファイルに出力されるため，ファイルシステムが圧迫されるおそれがあります。

なお，出力されたトラブルシュート情報は次に示すタイミングで HiRDB が自動的に削除します。

- HiRDB が稼働中の場合は 24 時間ごと（削除間隔を pd_spool_cleanup_interval オペランドで変更できます）
- HiRDB を開始したとき（削除するかどうかを pd_spool_cleanup オペランドで変更できます）

HiRDB 管理者がトラブルシュート情報を削除する場合は, pdcspool コマンドを実行してください。

noput：

トラブルシュート情報を取得しません。トラブルシュート情報が出力されないため，ファイルシステムへの負荷を軽減できます。通常の運用で UAP の中断が頻繁に発生し，かつ，その原因を調査する必要がない場合に指定してください。

異常終了時に出力される障害情報については，pd_dump_suppress_watch_time オペランドの「表 2-1　異常終了時に出力される障害情報」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，put が仮定されます。

32）pd_dump_suppress_watch_time =トラブルシュート情報の出力抑止時間～<符号なし整数>((0～3600))（単位：秒）

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

次に示す現象が発生した場合に出力されるトラブルシュート情報（%PDDIR%¥spool の下のファイル）の再出力を抑止する時間を秒単位で指定します。

- PDCWAITTIME タイムオーバが発生した場合
- pdcancel コマンドで実行中の UAP を中断させた場合
- プロセスが異常終了した場合

トラブルシュート情報を出力してから，このオペランドで指定した時間を経過するまでトラブルシュート情報を出力しません。例えば，このオペランドに 60 を指定した場合，トラブルシュート情報を出力した後，60 秒間はトラブルシュート情報を出力しません。

なお，このオペランドに 0 を指定した場合は，トラブルシュート情報の出力を抑止しません。

《利点》

HiRDB サーバのプロセス数が複数あると，タイムアウトなどでサーバプロセスの異常終了が連続して発生することがあります。サーバプロセスの異常終了が連続して発生すると，core 及び簡易ダン

プなどのトラブルシュート情報を繰り返し出力するため，HiRDB 運用ディレクトリがあるディスクの容量を圧迫します。HiRDB 運用ディレクトリがあるディスクの容量が不足すると HiRDB が異常終了することがあるため，このオペランドを指定してディスク容量の圧迫を防いでください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

- 内部矛盾による異常終了，外部からのシグナル受信処理などは，このオペランドの指定値に関係なくトラブルシュート情報を出力します。
- 異常終了時に出力される障害情報を次の表に示します。

表 3-1　異常終了時に出力される障害情報

異常終了の要因		障害情報	pd_dump_suppress_watch_time の値			
			0		0 以外	
			pd_cancel_dump の値			
			put	noput	put	noput
PDSWAITTIME オーバ		退避コアファイル	×	×	×	×
		デバック情報ファイル	×	×	×	×
		簡易ダンプファイル	×	×	×	×
		KFPA20009-W メッセージ	×	×	×	×
		SQL 実行時間警告情報ファイル	×	×	×	×
PDSWATCHTIME オーバ		退避コアファイル	×	×	×	×
		デバック情報ファイル	×	×	×	×
		簡易ダンプファイル	×	×	×	×
		KFPA20009-W メッセージ	×	×	×	×
		SQL 実行時間警告情報ファイル	×	×	×	×
PDCWAITTIME オーバ	pd_client_waittime_over_abort=Y	退避コアファイル	○	○	●	●
		デバック情報ファイル	○	○	●	●
		デバック情報ファイル 2	○	○	●	●
		簡易ダンプファイル	○	○	●	●
		KFPA20009-W メッセージ	○	○	●	●
		SQL 実行時間警告情報ファイル	○	○	●	●
		共用メモリダンプファイル	▲	▲	▲	▲
	pd_client_waittime	退避コアファイル	×	×	×	×

異常終了の要因		障害情報	pd_dump_suppress_watch_time の値			
			0		0 以外	
			pd_cancel_dump の値			
			put	noput	put	noput
	_over_abort=N	デバック情報ファイル	○	×	●	×
		デバック情報ファイル 2	×	×	×	×
		簡易ダンプファイル	○	×	●	×
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
		共用メモリダンプファイル	×	×	×	×
pdcancel コマンド		退避コアファイル	×	×	×	×
		デバック情報ファイル	○	×	●	×
		簡易ダンプファイル	○	×	●	×
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
内部 kill9※1		退避コアファイル	×	×	×	×
		デバック情報ファイル	○	×	●	×
		簡易ダンプファイル	○	×	●	×
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
内部 kill3※2		退避コアファイル	○	○	●	●
		デバック情報ファイル	○	○	●	●
		簡易ダンプファイル	○	○	●	●
		KFPA20009-W メッセージ	○	○	●	●
		SQL 実行時間警告情報ファイル	○	○	●	●
Abort※3		退避コアファイル	○	○	●	●
		デバック情報ファイル	○	○	●	●
		簡易ダンプファイル	○	○	●	●
		KFPA20009-W メッセージ	○	○	●	●

異常終了の要因	障害情報	pd_dump_suppress_watch_time の値			
		0		0 以外	
		pd_cancel_dump の値			
		put	noput	put	noput
	SQL 実行時間警告情報ファイル	○	○	●	●
	アボート情報ファイル	○	○	●	●
上記以外※4	退避コアファイル	△	△	△	△
	デバック情報ファイル	○	○	○	○
	簡易ダンプファイル	△	△	△	△
	KFPA20009-W メッセージ	△	△	△	△
	SQL 実行時間警告情報ファイル	△	△	△	△

（凡例）

○：障害情報を出力します。pd_dump_suppress_watch_time オペランドの指定は無効になります。

×：障害情報を出力しません。

●：障害情報を出力します。pd_dump_suppress_watch_time オペランドの指定が有効になります。

△：プロセスの終了形態によっては障害情報が出力されないことがあります。

▲：ユニットの開始後，1 回目のダンプ出力時に出力されます。pd_clt_waittime_over_dump_level オペランドに shm_fesonly を指定すると，共用メモリダンプを出力するユニットを制限できます。

注※ 1

OpenTP1 の UAP 異常終了処理など，内部的に SIGKILL を発行したときのことです。
PDCWAITTIME オーバ，pdcancel コマンドによる異常終了を含みません。

注※ 2

異常検知時など, 内部的に SIGQUIT を発行したときのことです。PDCWAITTIME オーバ, pdcancel コマンドによる異常終了を含みません。

注※ 3

HiRDB が矛盾を検知して abort() したときのことです。

注※ 4

SIGSEGV，SIGBUS，外部からのシグナル受信，exit，そのほかの予期しない障害などのことです。

33)　pd_pth_trace_max = 通信トレース格納最大数

～<符号なし整数>((1024～8388608))

トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。

《指定値の目安》

通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，1024 が仮定されます。

《注意事項》

通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

34）pd_spool_cleanup_interval = トラブルシュート情報の削除処理間隔

～<符号なし整数>((0～744))（単位：時間）

このオペランドは，出力したトラブルシュート情報及び作業用一時ファイルを削除するためのオペランドです。トラブルシュート情報及び作業用一時ファイルを残しておくと HiRDB 運用ディレクトリがあるディスクの容量を圧迫する原因になります。HiRDB 運用ディレクトリがあるディスクの容量が不足すると HiRDB が異常終了することがあるため，HiRDB は次に示すファイルを定期的に削除します。

- トラブルシュート情報ファイル（%PDDIR%¥spool 下のファイル）
- 作業用一時ファイル（%PDDIR%¥tmp 下のファイル）

このオペランドでは，その削除間隔を時間単位で指定します。例えば，このオペランドに 48 を指定すると，48 時間ごとにこれらのファイルを削除します。通常は（このオペランドを省略すると），24 時間ごとにファイルを削除します。

なお，HiRDB の正常開始時点が時間のカウント開始時点になります。HiRDB を正常終了すると，時間のカウントも止まります。そして，次の正常開始時に時間のカウントが 0 に戻ります。

削除対象のファイルは，次で説明している pd_spool_cleanup_interval_level オペランドで指定します。

《オペランドの規則》

0 を指定すると，ファイルの削除処理を実行しません。

《指定値の目安》

このオペランドに 24，48，72，…を指定すると，決まった時間にファイルの削除処理が実行されます。削除処理はシステムに負荷が掛かる時間帯を外して実施するようにしてください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，24 が仮定されます。

《注意事項》

HiRDB が計画停止，強制終了，及び異常終了して HiRDB が停止している間は，時間がカウントされます。ただし，HiRDB の停止中に削除時間になった場合は，ファイルを削除しません。次の削除時間になるまでファイルを削除しません。ファイルを削除してから HiRDB を再開始する場合は，pdcspool コマンドを実行してください。

《備考》

pd_spool_cleanup_interval オペランドと pd_spool_cleanup オペランドの違いについて説明します。

- pd_spool_cleanup_interval オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup オペランドの指定を検討してください。

35)　pd_spool_cleanup_interval_level = 日数〔，削除種別〕

このオペランドは，出力したトラブルシュート情報及び作業用一時ファイルを削除するためのオペランドです。トラブルシュート情報ファイル及び作業用一時ファイルの定期削除の削除条件を指定します。

日数：～<符号なし整数>((1～24855))（単位：日）

ここで指定した日よりも前に作成したトラブルシュート情報ファイルを削除します。例えば，3 と指定した場合，3 日以内（3 日×24 時間＝ 72 時間以内）に作成したトラブルシュート情報ファイルだけを残し，残りをすべて削除します。

削除種別：<文字列>

削除するトラブルシュート情報ファイルの種類を指定します。

all：すべてのファイルを削除対象にします。

dump：HiRDB が内部取得するファイルだけを削除対象にします。

削除対象のファイルを次に示します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
デッドロック・タイムアウト情報	pdlckinf	○	×	排他制御でエラーが発生したときに出力します。
アクセスパス情報	pdsqldump	○	×	アクセスパス表示ユティリティ使用時に出力します。
退避コアファイルなど	save	○	○	プロセスが異常終了したときに出力します。
共用メモリダンプファイル	pdshmdump	○	○	プロセス又はユニットが異常終了したときに出力します。
簡易ダンプファイル	pdsysdump	○	○	なし。
	pdsdsdump	○	○	HiRDB/パラレルサーバには存在しません。
	pdfesdump pddicdump pdbesdump	○	○	HiRDB/シングルサーバには存在しません。
システムログファイルの状態情報ファイル	pdjnlinf	○	×	¥pdjnlinf¥errinf 下のファイルは削除しません。

（凡例）

○：削除対象のファイルです。

×：削除対象外のファイルです。

注　ディレクトリ名は%PDDIR%¥spool 下のディレクトリ名です。

作業用一時ファイルについては，削除種別の指定に関係なく次に示すファイル以外を削除します。（　）内は%PDDIR%¥tmp 下のディレクトリ名です。

- HiRDB が起動するプロセスのカレントワーキングディレクトリ（home）
- 共用メモリ情報ファイル（pdommenv）
- pdbufls コマンドの差分情報ファイル（ファイル名称が"CMb"で始まるファイル）

《前提条件》

pd_spool_cleanup_interval オペランドに 0 以外を指定している必要があります。

《指定値の目安》

コマンド（ユティリティを含む）の実行時間より多い日数を指定してください。例えば，バックアップを取得する pdcopy コマンドの実行に 24 時間（1 日）必要な場合は，日数に 2 以上の値を指定してください。コマンドの実行時間より多い日数を指定しないと，コマンドが使用中の作業用一時ファイルも削除されるため，コマンドが正常に動作しないことがあります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次の値が仮定されます。

- 日数：7
- 削除種別：all（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は dump）

《オペランドの規則》

日数を指定しないで削除種別だけの指定はできません。

《注意事項》

pd_tmp_directory オペランドを指定しないで環境変数 TMP だけを指定している場合，コマンド又はユティリティが使用する作業用一時ファイルの出力先は環境変数 TMP に指定したディレクトリになります。環境変数 TMP に指定したディレクトリに出力された作業用一時ファイルは定期削除の対象外になるため，エクスプローラなどで削除してください。

《備考》

pd_spool_cleanup_interval_level オペランドと pd_spool_cleanup_level オペランドの違いについて説明します。

- pd_spool_cleanup_interval_level オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup_level オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval_level オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup_level オペランドの指定を検討してください。

36）pd_spool_cleanup = normal | force | no

このオペランドは，出力したトラブルシュート情報を削除するためのオペランドです。

HiRDB が出力したトラブルシュート情報ファイル（%PDDIR%¥spool 下のファイル）を HiRDB 開始時に削除するかどうかを指定します。削除対象のファイルは，次で説明している pd_spool_cleanup_level オペランドで指定します。

normal：HiRDB の正常開始及び計画停止後の再開始時に削除します。

force：HiRDB の開始モードに関係なく HiRDB の開始時に削除します。

no：削除しません。

《指定値の目安》

トラブルシュート情報ファイルがディスク容量を圧迫する場合に normal 又は force を指定してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，force が仮定されます。

《備考》

pd_spool_cleanup_interval オペランドと pd_spool_cleanup オペランドの違いについて説明します。

- pd_spool_cleanup_interval オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup オペランドの指定を検討してください。

37）　pd_spool_cleanup_level = 日数〔，削除種別〕

このオペランドは，出力したトラブルシュート情報を削除するためのオペランドです。HiRDB 開始時のトラブルシュート情報ファイルの削除条件を指定します。

日数：～<符号なし整数>((0～24855))（単位：日）

ここで指定した日よりも前に作成したトラブルシュート情報ファイルを削除します。例えば，3 と指定した場合，3 日以内（3 日×24 時間＝ 72 時間以内）に作成したトラブルシュート情報ファイルだけを残し，残りをすべて削除します。

0 を指定した場合はすべてのファイルを削除します。

削除種別：<文字列>

削除するトラブルシュート情報ファイルの種類を指定します。

all：すべてのファイルを削除対象にします。

dump：HiRDB が内部取得するファイルだけを削除対象にします。

削除対象のファイルを次に示します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
デッドロック・タイムアウト情報	pdlckinf	○	×	排他制御でエラーが発生したときに出力します。
アクセスパス情報	pdsqldump	○	×	アクセスパス表示ユティリティ使用時に出力します。
退避コアファイルなど	save	○	○	プロセスが異常終了したときに出力します。
共用メモリダンプファイル	pdshmdump	○	○	プロセス又はユニットが異常終了したときに出力します。
簡易ダンプファイル	pdsysdump	○	○	なし。
	pdsdsdump	○	○	HiRDB/パラレルサーバには存在しません。
	pdfesdump pddicdump pdbesdump	○	○	HiRDB/シングルサーバには存在しません。
システムログファイルの状態情報ファイル	pdjnlinf	○	×	¥pdjnlinf¥errinf 下のファイルは削除しません。

（凡例）

○：削除対象のファイルです。

×：削除対象外のファイルです。

注　ディレクトリ名は%PDDIR%¥spool 下のディレクトリ名です。

《前提条件》

pd_spool_cleanup オペランドで normal 又は force（省略値）を指定している必要があります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次の値が仮定されます。

- 日数：7
- 削除種別：all（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は dump）

《オペランドの規則》

日数を指定しないで削除種別だけを指定できません。

《備考》

pd_spool_cleanup_interval_level オペランドと pd_spool_cleanup_level オペランドの違いについて説明します。

- pd_spool_cleanup_interval_level オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup_level オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval_level オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup_level オペランドの指定を検討してください。

38）pd_module_trace_max = モジュールトレース格納最大数

～<符号なし整数>((126～16383))

HiRDB のプロセスは，実行した関数やマクロの履歴をプロセス固有メモリ中に記録しています。この履歴をモジュールトレースといい，このオペランドではモジュールトレースのレコード数を指定します。この履歴の内容は core ファイルに取り込まれ，プロセス異常終了時に出力されます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，126 が仮定されます。

《注意事項》

各プロセスでは，以下のサイズのプロセス固有メモリが確保されます。

32 ビットモードの場合：64 ＋ 48×pd_module_trace_max オペランドの値（バイト）

64 ビットモードの場合：64 ＋ 64×pd_module_trace_max オペランドの値（バイト）

39）pd_module_trace_timer_level = 0 | 10 | 20

モジュールトレースに出力する時刻の取得方法を指定します。このオペランドの指定値の意味を次に示します。

指定値	時刻取得方法
0	すべてのモジュールトレース出力箇所で，秒単位で取得します。
10	すべてのモジュールトレース出力箇所のうち，入出力処理前後など性能調査で問題となりやすい箇所だけマイクロ秒単位で取得し，そのほかの箇所では秒単位で取得します。
20	すべてのモジュールトレース出力箇所で，マイクロ秒単位で取得します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

このオペランドに 0 以外の値を指定すると，時刻をマイクロ秒で取得する関数を発行するため，性能が低下することがあります。

3.2.13　グローバルバッファに関するオペランド

40）　pd_dbbuff_wait_interval = グローバルバッファの占有状態の調査間隔

～<符号なし整数>((0～2147483647))（単位：ミリ秒）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

グローバルバッファの占有状態を調査する間隔を指定します。また，このオペランドを指定すると，グローバルバッファの占有状態を調査する方式が変わります。

このオペランドを指定していない場合

次に示す処理方式になります。

このオペランドを指定している場合

次に示す処理方式になります。

《指定値の目安》

次に示す条件をすべて満たす場合にこのオペランドを指定してください。性能が向上することがあります。なお，このオペランドを指定する場合は通常 1 を指定します。

- グローバルバッファの排他待ちが発生している (pdbufls コマンドの実行結果の WAITL で確認できます)
- CPU 使用率を高くしても性能を良くしたい

このオペランドに 1 を指定したため，CPU 使用率が高くなり過ぎた場合は，値を大きくしてください。また，このオペランドに 1 を指定しても CPU 使用率に余裕がある場合は，pd_dbbuff_wait_spn_count オペランドの値を大きくしてください。性能が向上することがあります。

41） pd_dbbuff_wait_spn_count = グローバルバッファの占有状態調査のスピンループ回数上限値 ～<符号なし整数>((0～2147483646))

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

グローバルバッファの占有状態の調査処理中に発生する，インターバルループ内のスピンループ回数の上限値を指定します。詳細については，pd_dbbuff_wait_interval オペランドの説明を参照してください。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。pd_dbbuff_wait_interval オペランドに 1 を指定している場合にこのオペランドを指定してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

3.2.14　ユニット用ステータスファイルに関するオペランド

42）　pd_syssts_file_name_1 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"

：

pd_syssts_file_name_7 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"

ユニット用ステータスファイルを定義します。pd_syssts_file_name_2～7 オペランドは省略できますが，pd_syssts_file_name_1 オペランドは省略できません。

"論理ファイル名"：～<識別子>((1～8 文字))

ユニット用ステータスファイルの論理ファイル名を指定します。ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系ステータスファイル名"：～<パス名>((167 文字以内))

A 系のステータスファイル名を絶対パス名で指定します。

"B 系ステータスファイル名"：～<パス名>((167 文字以内))

B 系のステータスファイル名を絶対パス名で指定します。

《指定値の目安》

- A 系及び B 系のステータスファイル名には，pdstsinit コマンドで作成したステータスファイルを指定してください。pdstsinit コマンドで作成していないステータスファイルを指定すると，そのステータスファイルは**実体のないステータスファイル**になります。
- ステータスファイルに障害が発生すると，HiRDB はステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要になります。
- A 系と B 系には同じレコード長，及び同じ容量のステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- ステータスファイルは A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系ステータスファイル名，及び B 系ステータスファイル名には同じ名称を指定できません。
- HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥usts01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。usts01 については大文字と小文字の区別をします。

《留意事項》

- HiRDB を正常開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，全ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときは pd_syssts_file_name_1～7 の中で最初に指定したステータスファイルが現用ファイルになります。残りのファイルのうちオープンできたファイルが予備ファイルになり，オープンできないファイルは予約ファイルになります。
- HiRDB を再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。

●実体のないステータスファイルの使い方

実体のないステータスファイルを指定すると，HiRDB の稼働中にステータスファイルを新規追加できます。例えば，ステータスファイルの障害などで予備ファイルが少なくなったときに，実体のないステータスファイルを予備ファイルにします。実体のないステータスファイルを予備ファイルにする手順を次に示します。

〈手順〉

1. pdstsinit コマンドで，システムファイル用の HiRDB ファイルシステム領域にステータスファイルを作成します。
2. pdstsopen コマンドでステータスファイルをオープンします。

この操作は HiRDB 稼働中に実行できます。HiRDB を一度停止する必要はありません。

- メリット及びデメリット

  実体のないステータスファイルを定義すると，HiRDB ファイルシステム領域の占有量は小さくなります。しかし，実体のないステータスファイルを予備ファイルとして追加するときに，システムファイル用の HiRDB ファイルシステム領域に十分な空き（ファイルを追加するだけの空き領域）がないと追加できないため，システムの信頼性は低くなります。

  実体のないステータスファイルを定義しないと，HiRDB ファイルシステム領域の占有量は大きくなるが，ファイル障害によるスワップ先が保証されるため，信頼性は高くなります。

- 注意事項

  実体のないステータスファイルを定義した場合，HiRDB は HiRDB の開始時にステータスファイルに異常があると認識します。このため，pd_syssts_initial_error オペランドに stop（省略値）を指定していると，HiRDB を開始できないので注意してください。実体のないステータスファイルを定義する場合は，pd_syssts_initial_error オペランドに continue 又は excontinue を指定してください。また，HiRDB を開始する前に現用ファイルを pd_syssts_last_active_file オペランドに指定する必要があります。

3.2.15　ユニット用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

43）　pd_syssts_initial_error = stop | continue | excontinue

HiRDB（又はユニット）の開始時，HiRDB はユニット用ステータスファイルの現用ファイルを特定する処理を実行します。この特定処理で，次に示す異常を検知したときの HiRDB の処置を指定します。

- ユニット用ステータスファイルの実体がない
- ユニット用ステータスファイルの障害を検知した

現用ファイルの特定処理は pd_syssts_file_name_1～7 オペランドに指定したユニット用ステータスファイルを対象とします。

stop：

現用ファイルの特定処理時にユニット用ステータスファイルの異常を検知した場合，HiRDB（又はユニット）の開始処理を中止します。この場合，異常を検知したステータスファイルの対策を実施した後に HiRDB を開始してください。

continue 又は excontinue：

現用ファイルの特定処理時にユニット用ステータスファイルの異常を検知しても，現用ファイルが正常な場合は HiRDB（又はユニット）の開始処理を続行します。ただし，

pd_syssts_singleoperation オペランドの値（ステータスファイルの片系運転をするかどうか）によっては開始処理を中止することがあります。pd_syssts_singleoperation オペランドとの関係を次に示します。

●pd_syssts_singleoperation オペランドとの関係

pd_syssts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
continue	ユニット用ステータスファイルの異常を検知した場合，HiRDB は現用ファイルを特定できないため，HiRDB（又はユニット）の開始処理を中止します。	HiRDB 管理者が現用ファイルを特定して pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドを指定します。その後，HiRDB（又はユニット）を開始してください。
stop（省略値）	ユニット用ステータスファイルの異常を検知した場合，HiRDB が現用ファイルを特定して HiRDB（又はユニット）の開始処理を続行します。ただし，A 系と B 系のファイルが次に示す表（**HiRDB が現用ファイルを特定できないケース**）の条件のどれかに当てはまる場合，HiRDB は現用ファイルを特定できないため，HiRDB の開始処理を中止します。	HiRDB が現用ファイルを特定できない場合は，HiRDB 管理者が現用ファイルを特定して pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドを指定します。その後，HiRDB（又はユニット）を開始してください。

●HiRDB が現用ファイルを特定できないケース

pd_syssts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
continue	障害閉塞	障害閉塞
	障害閉塞	オープン（初期状態）
	障害閉塞	ファイル実体なし
	オープン（初期状態）	障害閉塞
	オープン（初期状態）	ファイル実体なし
	ファイル実体なし	オープン（初期状態）
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし
excontinue	障害閉塞	障害閉塞
	障害閉塞	ファイル実体なし
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし

したがって，HiRDB 管理者の操作をなるべく少なくしたい場合（pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドを指定するケースを減らしたい場合）は，次に示すようにオペランドを指定してください。

- pd_syssts_initial_error オペランドに excontinue を指定
- pd_syssts_singleoperation オペランドに stop を指定

《指定値の目安》

各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_syssts_initial_error オペランドの値	
	stop	continue 又は excontinue
HiRDB 又はユニット開始時の HiRDB の処理	ユニット用ステータスファイルの異常を検知した場合，HiRDB（又はユニット）の開始処理を中止します。	一部のユニット用ステータスファイルに異常があっても，現用ファイルが正常な場合は HiRDB（又はユニット）の開始処理を続行します。
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB 開始時の障害対策を簡易化する場合は continue 又は excontinue を指定します。
メリット	HiRDB の開始時に全ユニット用ステータスファイルの正常が保証されます。このため，HiRDB の開始後に現用ファイルの障害が発生した場合，予備ファイルにスワップできます。	HiRDB の開始時に一部のユニット用ステータスファイルの障害を検知しても，残りの正常なファイルだけで HiRDB を開始できます。このため，HiRDB の停止時間を短くできます。この場合，予備ファイルが少なくなっているため，障害状態のステータスファイルをすぐに修復する必要があります。
デメリット	ユニット用ステータスファイルの障害によって HiRDB の開始処理が中止される可能性が高くなります。	予備ファイルが少ない状態で HiRDB を稼働することがあるため，システムの信頼性が低下します。予備ファイルの数によってはユニット用ステータスファイルをスワップできないことがあります。

《注意事項》

- 現用ファイルが両系とも異常の場合は，このオペランドの指定値に関係なく HiRDB（又はユニット）の開始処理を中止します。
- HiRDB を開始する前に現用ファイルを pdstsinit コマンドで初期化しないでください。初期化すると HiRDB を再開始できません。
- このオペランドに excontinue を指定し，すべての世代のステータスファイルが片系運転でかつオープン（初期状態）の場合，HiRDB は起動しません。

《備考》

このオペランドの指定値と HiRDB の処理及び HiRDB 管理者の処置を次に示します。

●このオペランドの指定値と HiRDB の処理

pd_syssts_initial_error オペランドの値	ステータスファイル障害	pd_syssts_singleoperationオペランドの値	HiRDBが現用ファイルを特定できるか	pd_syssts_last_active_fileの指定	pd_syssts_last_active_fileオペランドの値と，オープンできたステータスのうち，最新のファイルが一致するか	現用ファイル障害	pd_syssts_last_active_sideオペランドの指定	pd_syssts_last_active_sideオペランドに指定されたファイルが使用できるか	表「HiRDBの処理及びHiRDB管理者の処置」の対応番号
stop（省略値）	なし	－	－	－	－	－	－	－	[1]
	あり	－	－	－	－	－	－	－	[5]
continue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
excontinue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]

（凡例）－：該当しません（条件がHiRDBの処理に影響を与えません）。

●HiRDB の処理及び HiRDB 管理者の処置

対応番号	HiRDB の処理	HiRDB 管理者の処置
[1]	HiRDB の開始処理を続行します。	なし
[2]	HiRDB が最新の現用ファイルを特定して，開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[3]	pd_syssts_last_active_file オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[4]	pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[5]	pd_syssts_initial_error オペランドに stop が指定されているため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000010 が示す対策をしてください。

対応番号	HiRDB の処理	HiRDB 管理者の処置
[6]	前回稼働時の現用ファイルが特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000016 が示す対策をしてください。
[7]	HiRDB が特定した現用ファイルの正常な系と，pd_syssts_last_active_side オペランドに指定された系が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000017 が示す対策をしてください。
[8]	HiRDB が特定した現用ファイル名と，pd_syssts_last_active_file オペランドに指定したファイル名が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000015 が示す対策をしてください。
[9]	前回稼働時の現用ファイルの正常な系が特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000018 が示す対策をしてください。

44）pd_syssts_singleoperation = stop | continue

ユニット用ステータスファイルの片系運転をするかどうかを指定します。予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，正常な系（片方の系）だけで処理を続行することを**ステータスファイルの片系運転**といいます。ステータスファイルの片系運転については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

なお，予備ファイルがある状況で現用ファイルの片系に障害が発生した場合は，このオペランドの指定値に関係なく，ステータスファイルをスワップして処理を続行します（片系運転にはなりません）。

stop：

片系運転をしません。片系運転になる場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。HiRDB が異常終了した場合，予備ファイルを準備した後に HiRDB を開始してください。

continue：

片系運転をします。片系運転になると KFPS01044-I メッセージが出力されます。片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了したりすると，HiRDB を再開始できなくなるため，片系運転になった場合はすぐに予備ファイルを準備してください。

《指定値の目安》

- stop を指定してシステムの信頼性を上げることを推奨します。また，現用ファイルの障害に対しては，予備ファイルの数を増やして備えることを推奨します。
- 各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_syssts_singleoperation オペランドの値	
	stop	continue
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB を停止させないことを重視する場合は continue を指定します。
メリット	予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，片系運転をしないで HiRDB	予備ファイルがない状況で現用ファイルの片系に障害が発生しても，処理を続行できます。この

項目	pd_syssts_singleoperation オペランドの値	
	stop	continue
	を異常終了します。このため，現用ファイルの内容が失われる可能性が低くなります。	ため，ステータスファイルの障害によってHiRDB が停止する可能性が低くなります。
デメリット	ステータスファイルの障害によって HiRDB が停止する可能性が高くなります。ただし，予備ファイルの数を増やすことで可能性を低くできます。	片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，現用ファイルの内容が失われるため，HiRDB を再開始できなくなります。

《ほかのオペランドとの関連》

pd_syssts_singleoperation 及び pd_syssts_initial_error オペランドの指定値の組み合わせによって，ステータスファイル障害時の HiRDB の処理が決定します。したがって，これら二つのオペランドの指定値は一緒に考えるようにしてください。

45）　pd_syssts_last_active_file = "論理ファイル名"

～<識別子>((1～8 文字))

HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に現用にするステータスファイルの論理ファイル名を指定します。HiRDB は，このオペランドに指定されたファイルと HiRDB が特定した現用ファイルを比較します。ファイルが一致する場合は HiRDB を開始しますが，一致しない場合は開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_syssts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **全ステータスファイルを初期化した直後に HiRDB を開始する場合**

   pd_syssts_file_name_1～7 オペランドに指定した最も小さい番号の正常な論理ファイル名を指定してください。この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，全ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_syssts_last_active_side オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   全ステータスファイルを初期化してください。その後，1 の方法を実施してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

5. **実体のないステータスファイルを指定している場合**

   現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I
- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージのうちで，最も直前に出力されたメッセージに表示されているステータスファイルが現用ファイルになります。

46）　pd_syssts_last_active_side = A｜B

現用ファイルの片系に障害が発生している状態で HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB を開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_syssts_initial_error オペランドに continue，又は excontinue を指定している
- pd_syssts_last_active_file オペランドを指定している

3.2.16　セキュリティに関するオペランド

セキュリティ監査機能の運用方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

47）　pd_audit = Y｜N

HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時から監査証跡を取得するかどうかを指定します。

Y：HiRDB の開始時から監査証跡を取得します。

N：HiRDB の開始時から監査証跡を取得しません。

このオペランドに N を指定しても，pdaudbegin コマンドを実行すると監査証跡を取得できます。

《前提条件》

次に示すすべての条件を満たす必要があります。満たしていない場合に Y を指定すると，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始できません。

- 監査証跡ファイル用の HiRDB ファイルシステム領域を作成している
- pd_aud_file_name オペランドに監査証跡ファイル用の HiRDB ファイルシステム領域名を指定している

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，N が仮定されます。

48）　pd_aud_file_name = 監査証跡ファイル用の HiRDB ファイルシステム領域名

～<パス名>((150 文字以内))

セキュリティ監査機能を使用する場合はこのオペランドを必ず指定してください。**指定しないとセキュリティ監査機能を使用できません。**

監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。

複数ユニットのシステム構成でセキュリティ監査機能を使用する場合，システム全体で監査証跡を取得することを推奨します。システム全体で監査証跡を取得するには，次のどちらかの指定をしてください。

- システム共通定義で pd_aud_file_name オペランドを指定する
- すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する

ただし，同一サーバマシンで複数のユニットを稼働させるシステム構成の場合は，すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する必要があります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。

《注意事項》

- このオペランドを指定した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に監査証跡ファイル用の HiRDB ファイルシステム領域に対するアクセスエラーが発生すると，HiRDB を開始できません。
- システム共通定義の pd_aud_file_name オペランドの指定によって同一サーバマシン上の複数のユニットが同じ監査証跡ファイルを指定した場合，監査証跡を正しく取得できません。

49）pd_aud_max_generation_size = 1 監査証跡ファイルの最大容量

～<符号なし整数>((1～5240))（単位：メガバイト）

1 監査証跡ファイルの最大容量をメガバイト単位で指定します。

《指定値の目安》

- HiRDB が管理用に使用する領域が 20 メガバイト必要なため，次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_aud_max_generation_size の値×pd_aud_max_generation_num の値<監査証跡ファイル用の HiRDB ファイルシステム領域長（pdfmkfs コマンドの-n オプションの値）－20（メガバイト）

- 監査証跡の 1 レコードの容量よりも小さい値を指定した場合，又は，このオペランドの指定を省略した場合で，かつ監査証跡の 1 レコードのサイズがこのオペランドの省略時仮定値よりも大きいときには，HiRDB ユニットを起動できません。

  HiRDB ユニットを起動するには，次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_aud_max_generation_size の指定値≧↑監査証跡のレコードサイズの最大長÷1024↑×1024 + 2048（バイト）

  監査証跡のレコードサイズの最大長は，次の計算式で算出します。

  監査証跡のレコードサイズの最大長＝ 1067 ＋↑pd_aud_sql_source_size の指定値÷4↑×4＋↑pd_aud_sql_data_size の指定値÷4↑×4（バイト）

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，100 が仮定されます。

50）pd_aud_max_generation_num = 監査証跡ファイルの最大数

～<符号なし整数>((2～200))

監査証跡ファイル用の HiRDB ファイルシステム領域内に作成する監査証跡ファイルの最大数（世代数）を指定します。

《指定値の目安》

- 全監査証跡ファイルに障害が発生したときのことを想定して，最大値である 200 を指定しないことをお勧めします。監査証跡ファイルに障害が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- HiRDB が管理用に使用する領域が 20 メガバイト必要なため，次に示す条件式を満たすようにこのオペランドの値を決定してください。

pd_aud_max_generation_size の値×pd_aud_max_generation_num の値＜監査証跡ファイル用の HiRDB ファイルシステム領域長（pdfmkfs コマンドの-n オプションの値）－20（メガバイト）

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，50 が仮定されます。

《注意事項》

HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，このオペランドの値よりも大きい世代番号のファイルがある場合，このオペランドの値は無効になります。この場合，HiRDB ファイルシステム領域内に作成される監査証跡ファイルの最大数には，最大世代番号が仮定されます。

51）　pd_aud_async_buff_size = 監査証跡の非同期出力時に使用するバッファ長

～＜符号なし整数＞((0，4096～6553600))（単位：バイト）

監査証跡を非同期に出力する場合に使用するバッファ長をバイト単位で指定します。0 を指定した場合は監査証跡を同期出力します。監査証跡のレコードサイズの最大長よりも小さい値を指定した場合，又は，このオペランドの指定を省略した場合で，かつ監査証跡のレコードサイズの最大長がこのオペランドの省略時仮定値よりも大きいときには，HiRDB ユニットを起動できません。

監査証跡のレコードサイズの最大長については，pd_aud_max_generation_size オペランドの説明を参照してください。

各出力方式の長所及び短所を次に示します。

pd_aud_async_buff_size の値	監査証跡の出力方式	長所	短所
0	同期出力	監査証跡を確実に監査証跡ファイルに出力できます。	SQL 処理の延長上でファイル入出力が発生するため，性能に与える影響が大きくなります。
4096～6553600	非同期出力	SQL 処理の性能に与える影響を小さくできます。	バッファへの出力後から監査証跡ファイルに出力するまでの間に，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了すると，監査証跡が失われることがあります。

《オペランドの規則》

このオペランドには 4096 の整数倍を指定してください。4096 の整数倍以外を指定した場合は，指定した値を 4096 の整数倍に切り上げてその値をこのオペランドに設定します。例えば，5000 を指定すると 8192 が設定されます。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，401408 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 4096 が仮定されます）。

《注意事項》

- HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，pd_aud_async_buff_size×pd_aud_async_buff_count バイト分のユニットコントローラ用共用メモリが必要です。この計算式の値がユニットコントローラ全体の共用メモリサイズの上限を超えないように設定してください。ユニットコントローラが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を小さく設定すると，監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちに陥り，監査証跡の出力処理が滞ることがあります。

  単位時間あたりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を設定してください。

52）　pd_aud_async_buff_count = 監査証跡の非同期出力時に使用するバッファ面数

～<符号なし整数>((1～6500))

監査証跡を非同期に出力する場合に使用するバッファの面数を指定します。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次のうち，大きい方の値が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 3 が仮定されます）。

- 1（ユニット内 HiRDB サーバが存在しない場合）
- ユニット内 HiRDB サーバ数×10

《注意事項》

- HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，pd_aud_async_buff_size×pd_aud_async_buff_count バイト分のユニットコントローラ用共用メモリが必要です。この計算式の値がユニットコントローラ全体の共用メモリサイズの上限を超えないように設定してください。ユニットコントローラが使用する共用メモリの計算式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を小さく設定すると，監査証跡の非同期出力時に使用するバッファのすべての面が，バッファから監査証跡ファイルへの出力待ちに陥り，監査証跡の出力処理が滞ることがあります。

  単位時間あたりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランドと pd_aud_async_buff_count オペランドの値を設定してください。

53）　pd_aud_async_buff_retry_intvl = 監査証跡の非同期出力時に使用するバッファの確保リトライ間隔

～<符号なし整数>((1～1000))（単位：ミリ秒）

監査証跡を非同期に出力する場合に使用するバッファがすべて使用中のとき，未使用のバッファが確保できるまでバッファを監視する処理のリトライ間隔を指定します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

セキュリティ監査機能使用時，UAP の実行に時間が掛かる場合にこのオペランドの値を小さくすると，UAP の実行時間が短くなることがあります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，50 が仮定されます。

54)　pd_aud_sql_source_size = 監査証跡に出力する SQL 文のサイズ

～<符号なし整数>((0～2000000))（単位：バイト）

セキュリティ監査機能を使用する場合に監査証跡に出力する SQL 文のサイズをバイト単位で指定します。0 を指定した場合は，監査証跡に SQL 文を出力しません。指定した値よりもサイズが大きい SQL 文については，指定した値を超えた部分は監査証跡に出力されません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

《ほかのオペランドとの関連》

このオペランドを指定した場合，オペランド pd_aud_max_generation_size 及び pd_aud_async_buff_size の指定値を見積もり直してください。

55)　pd_aud_sql_data_size = 監査証跡に出力する SQL データのサイズ

～<符号なし整数>((0～1000000))（単位：バイト）

セキュリティ監査機能を使用する場合に監査証跡に出力する SQL データのサイズをバイト単位で指定します。0 を指定した場合は，監査証跡に SQL データを出力しません。指定した値よりもサイズが大きい SQL データについては，指定した値を超えた部分は監査証跡に出力されません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

《ほかのオペランドとの関連》

このオペランドを指定した場合，オペランド pd_aud_max_generation_size 及び pd_aud_async_buff_size の指定値を見積もり直してください。

3.2.17　系切り替え機能に関するオペランド

56)　pd_ha_acttype = monitor | server

系切り替え機能の運用をモニタモードで行うか，サーバモードで行うかを指定します。モニタモード及びサーバモードについては，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

monitor：系切り替え機能の運用をモニタモードで行います。

server：系切り替え機能の運用をサーバモードで行います。

次に示す機能を使用する場合はサーバモードでの運用が前提になります。

- ユーザサーバホットスタンバイ
- 高速系切り替え機能
- 1：1 スタンバイレス型系切り替え機能
- 影響分散スタンバイレス型系切り替え機能

《前提条件》

このオペランドを指定するには pd_ha オペランドに use を，pd_ha_ipaddr_inherit オペランドに N を指定する必要があります。また，server を指定する場合は次に示す条件をすべて満たす必要があります。

- Hitachi HA Toolkit Extension を導入している
- IP アドレスを引き継がない指定をしている（pd_ha_ipaddr_inherit = N を指定している）

これらの製品を導入しないで server を指定すると，HiRDB を開始できません。

57）pd_ha_unit = nouse

このオペランドはユニットに系切り替え機能を適用しない場合に指定します。系切り替え機能を適用するユニットと，適用しないユニットを分ける場合に指定してください。このオペランドを指定すると，この項で説明しているオペランドの指定がすべて無効になります。

《ほかのオペランドとの関連》

フロントエンドサーバの pdstart オペランドに -k stls オプションを指定すると回復不要 FES ユニットになります。回復不要 FES ユニットでは，系切り替え機能を適用できません。系切り替え機能を適用するシステムの場合，回復不要 FES ユニットのユニット制御情報定義には，必ずこのオペランドを指定してください。

58）pd_ha_switch_timeout = Y | N

このオペランドはサーバモードの場合に指定できます。モニタモードの場合にこのオペランドを指定しても無効になります。

系切り替え時の HiRDB（HiRDB/パラレルサーバの場合はユニット）の内部停止処理が**サーバ障害監視時間**を超えた場合に，HiRDB の内部停止処理を待たないで系を切り替えるかどうかを指定します。ここでいうサーバ障害監視時間とは，Hitachi HA Toolkit Extension の patrol オペランドに指定した時間のことです。

Hitachi HA Toolkit Extension の patrol オペランドについては，マニュアル「Hitachi HA Toolkit」を参照してください。

Y：

系切り替え時の HiRDB の内部停止処理がサーバ障害監視時間を超えた場合，HiRDB の内部停止処理を待たないで系を切り替えます。このとき，HiRDB のスローダウンとして系を切り替えます。

1：1 スタンバイレス型系切り替え機能又は影響分散スタンバイレス型系切り替え機能を使用している場合，計画系切り替え時はこのオペランドの指定が無効になります。

N：

系切り替え時の HiRDB の内部停止処理が終了するまで系を切り替えません。

《利点》

系の切り替え時にディスク障害などで HiRDB の内部停止処理に時間が掛かると，それが系の切り替え時間の遅延要因になります。このオペランドに Y（省略値）を指定すると，HiRDB の内部停止処理に時間が掛かる場合でも，HiRDB の内部停止処理を待たないで系を切り替えられます。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，Y が仮定されます。

《注意事項》

- patrol オペランドの値が小さいときにこのオペランドに Y を指定すると，計画系切り替えがスローダウンによる系切り替えになることがあります。これは，計画系切り替え時の HiRDB の内部停止処理が patrol オペランドで指定した時間を超えてしまうためです。
- Hitachi HA Toolkit Extension の switchtype オペランドに restart を指定した場合は注意が必要です。pd_ha_switch_timeout = Y（省略値）を指定した場合，HiRDB の内部停止処理がサーバ障害監視時間を超えると，障害が発生した系で HiRDB を再開始しません。この場合，すぐに系を切り替えます。

Hitachi HA Toolkit Extension の switchtype オペランドについては，マニュアル「Hitachi HA Toolkit」を参照してください。

59）　pd_ha_server_process_standby = Y | N

ユーザサーバホットスタンバイを使用するかどうかを指定します。ユーザサーバホットスタンバイについては，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

Y：ユーザサーバホットスタンバイを使用します。

N：ユーザサーバホットスタンバイを使用しません。

なお，1：1 スタンバイレス型系切り替え機能又は影響分散スタンバイレス型系切り替え機能を使用する場合はこのオペランドを省略してください。

《前提条件》

Y（省略値）を指定する場合はサーバモードである必要があります。モニタモードの場合は常に N が仮定されます。

《ほかのオペランドとの関連》

高速系切り替え機能はユーザサーバホットスタンバイを包括しています。このため，pd_ha_agent オペランドに standbyunit を指定した場合（高速系切り替え機能を使用する指定をした場合），このオペランドに N を指定しても Y が仮定されます。

60）　pd_ha_agent = standbyunit | server | activeunits

次に示す機能を使用する場合にこのオペランドを指定します。

- 高速系切り替え機能
- 1：1 スタンバイレス型系切り替え機能
- 影響分散スタンバイレス型系切り替え機能

これらの機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

standbyunit：高速系切り替え機能を使用する場合に指定します。

server：1：1 スタンバイレス型系切り替え機能を使用する場合に指定します。

activeunits：影響分散スタンバイレス型系切り替え機能を使用する場合に指定します。

なお，同一システム内で，1：1 スタンバイレス型系切り替え機能と影響分散スタンバイレス型系切り替え機能を混在して指定できません。

《前提条件》

このオペランドを指定する場合はサーバモードである必要があります。モニタモードの場合にこのオペランドを指定すると KFPS01896-E エラーになります。

《注意事項》

- 高速系切り替え機能の対象ユニットは IP アドレスを引き継げません。したがって，HiRDB/シングルサーバの場合はシステム共通定義又はユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに N を指定してください。HiRDB/パラレルサーバで IP アドレスを引き継ぐ構成の場合は，高速系切り替え機能対象ユニットのユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに N を指定してください。
- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用する場合は，グローバルバッファを動的変更できません。グローバルバッファの動的変更については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《ほかのオペランドとの関連》

- 高速系切り替え機能はユーザサーバホットスタンバイを包括しています。このため，このオペランドに standbyunit を指定すると pd_ha_server_process_standby オペランドに Y が仮定されます。
- 1：1 スタンバイレス型系切り替え機能及び影響分散スタンバイレス型系切り替え機能は，系を切り替えた場合，切り替え先のユニットのサーバプロセスを利用します。そのため，このオペランドに server 又は activeunits を指定すると pd_ha_server_process_standby オペランドの指定が無効になります。
- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用する場合は，pd_rdarea_open_attribute_use オペランド（RD エリアのオープン契機の指定）に Y が仮定されます。
- 影響分散スタンバイレス型系切り替え機能を使用している場合，HA グループ内の全ユニットが起動していなくてもよいため，pd_start_skip_unit オペランドの指定が無効になります。ただし，HA グループ内の全バックエンドサーバがどこかのユニットで起動していないと，HiRDB を開始できません。
- このオペランドを指定した場合，次に示すオペランドの指定値を確認してください。指定値に誤りがある場合，KFPS01896-E エラーになります。

オペランド名	pd_ha_agent オペランドの値		
	standbyunit	server	activeunits
pd_ha	use	use	use
pd_ha_acttype	server	server	server
pd_ha_ipaddr_inherit（ユニット制御情報定義）	N	－	－
pd_ha_agent	－	ほかのユニットに activeunits を指定してはいけない	ほかのユニットに server を指定してはいけない
pdstart オペランドの-c オプション	－	○	－
pdstart オペランドの-g オプション	－	－	○
pdhagroup	－	－	○

（凡例）

○：オペランドの指定が必要です。

－：該当しません。

61）　pd_ha_max_act_guest_servers = 受け入れ可能なゲスト BES の最大数

～<符号なし整数>((0～33))

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。ユニット内で実行系として稼働するゲスト BES の最大数を指定します。ユニット内で実行系として稼働するゲスト BES 数がこのオペランドの指定値に達すると，ゲスト BES の受け入れ可能状態を解除します。

このオペランドの値によって，受け入れユニットで確保するゲスト用領域の数が決定します。このオペランドの値を変更してゲスト BES 用のリソース所要量を調整してください。

なお，このオペランドの値が小さ過ぎると，系の切り替えが発生したとき，一部のバックエンドサーバの処理が実行されないことがあります。HA グループ内の障害バックエンドサーバ数が HA グループ内の受け入れ可能ゲスト BES 数の合計を超えると，超えた分のバックエンドサーバの処理がどのユニットにも受け入れられないため，そのバックエンドサーバの処理が実行されません。

《前提条件》

このオペランドを指定する場合は次に示すオペランドを指定する必要があります。指定しないと，システム定義の不正によるエラーになります（KFPS01896-E メッセージが出力されます）。

- pdhagroup
- pd_ha_agent = activeunits

《指定値の目安》

このオペランドを省略すると，このオペランドの値を HiRDB が計算します。また，指定値の上限値は次に示す値の小さい方になります。

- 34－ユニット内のホスト BES 数
- HA グループ内のサーバ数－ユニット内のホスト BES 数

上限値以上の値を指定しても，この上限値に補正されます（KFPS05613-W メッセージが出力されます）。

なお，このオペランドを省略したときに HiRDB が算出する受け入れ可能なゲスト BES の最大数は，次に示す条件をすべて満たしていることを想定しています。

- HA グループ内の各ユニットのバックエンドサーバ数が同じである
- 2 点障害が発生した

したがって，HA グループ内の各ユニットのバックエンドサーバ数が異なる場合，又は 3 点障害以上が発生した場合，このオペランドの省略値では一部のバックエンドサーバの処理が実行されないことがあります。

《備考》

このオペランドの省略値の計算式を次に示します。

- HA グループ内のユニット数が 2 の場合：a
- HA グループ内のユニット数が 3 以上の場合：↑（a×2）÷（b－2）↑

a：ユニット内のホスト BES 数

b：HA グループ内のユニット数

このオペランドを指定する場合，次に示す計算式（推奨値）からこのオペランドの指定値を求めてください。この計算式は，多点障害時に自ユニットを除くユニットで，ホスト BES 数が多いユニットから順に障害が発生した場合，残りのユニットが障害ユニットのホスト BES を均等に受け入れることを想定しています。

$$\text{推奨値} = \uparrow \sum_{i=1}^{c} a_i \div (b-c) \uparrow$$

a：自ユニット以外で，i 番目にホスト BES 数が多いユニットのホスト BES 数

b：HA グループ内のユニット数

c：想定している多点障害数（＜b）

62）　pd_ha_max_server_process = 受け入れユニット内の最大ユーザサーバプロセス数

～＜符号なし整数＞((1～10000))《ユニット内ホスト BES の pd_max_bes_process の合計値》

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。ユニット内で起動するユーザサーバプロセスの最大数を指定します。受け入れユニットでは，ホスト BES 及びゲスト BES の各 pd_max_bes_process オペランドの値（最大起動プロセス数）によってサーバプロセス数が制限されます。このオペランドを指定すると，それに加えてユニット内のサーバプロセス数の合計値が制限できます。

《前提条件》

このオペランドを指定する場合は次に示すオペランドを指定する必要があります。指定しないと，このオペランドの指定は無効になります。

- pdhagroup
- pd_ha_agent = activeunits

《指定値の目安》

系の切り替え後，ユニット内のプロセス数を増加させたくない場合はこのオペランドを省略してください。ただし，同時に処理できるサービス要求数の上限が制限されることがあります。リソースが十分にある場合は，ユニット内のホスト BES と，受け入れ可能なゲスト BES の最大起動プロセス数（pd_max_bes_process オペランドの値）の合計値を指定してください。そうすると，系の切り替え後も同時に処理できるサービス要求数の上限が系の切り替え前と同じになります。

このオペランドの指定によって，受け入れユニットに過剰な負荷が掛かることを抑止できます。系切り替え発生後のユニットの負荷上昇，及び同時に処理できるサービス要求数の両者を考慮してこのオペランドの指定値を決定してください。

《注意事項》

ユニット内ホスト BES の pd_max_bes_process オペランドの合計値より小さい値をこのオペランドに指定した場合，その指定値は無効になります。この場合，ユニット内ホスト BES の pd_max_bes_process オペランドの合計値が仮定されます（KFPS05614-W メッセージが出力されます）。

63)　pd_ha_process_count = ゲスト BES 受け入れ後のユニット内常駐プロセス数

～<符号なし整数>((0～10000))

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。ゲスト BES 受け入れ後のユニット内のホスト BES とゲスト BES の常駐プロセス数の合計値を指定します。

ゲスト BES 受け入れ後の各サーバの常駐プロセス数をこのオペランドの指定で調整できます。

《前提条件》

このオペランドを指定する場合は次に示すオペランドを指定する必要があります。指定しないと，このオペランドの指定は無効になります。

- pdhagroup
- pd_ha_agent = activeunits
- サーバ共通定義又はバックエンドサーバ定義の pd_process_count

《指定値の目安》

- このオペランドを省略すると，このオペランドの値を HiRDB が計算します。
- 非サービス中の常駐プロセスがプロセスを占有していて，一部のバックエンドサーバがプロセスを起動できないため，サービスの要求を処理できない場合，このオペランドの値を小さくしてください。ただし，系の切り替え後，トランザクションのスループットが悪くなることがあります。系の切り替え後，トランザクションのスループットを良くしたい場合は，このオペランドの値を大きくしてください。ただし，このオペランドの値が大き過ぎると，一部のバックエンドサーバでサービスの要求を処理できないことがあります。

《注意事項》

ユニット内ホスト BES の pd_process_count オペランドの合計値より小さい値をこのオペランドに指定した場合，その指定値は無効になります。この場合，ユニット内ホスト BES の pd_process_count オペランドの合計値が仮定されます（KFPS05615-W メッセージが出力されます）。

《備考》

ゲスト BES を受け入れた後，各バックエンドサーバに常駐するプロセス数は，次に示す値のうち小さい方になります。

- このオペランドの値を各バックエンドサーバの pd_process_count オペランドの値で比例配分した値
- 各バックエンドサーバの pd_process_count オペランドの値

$$\mathrm{MIN}\left\{\begin{array}{l}\mathrm{pd_process_count}\ ,\\ \downarrow \mathrm{pd_ha_process_count} \times \dfrac{\mathrm{pd_process_count}}{\sum_{\text{実行中BES}} \mathrm{pd_process_count}} \downarrow\end{array}\right\}$$

このオペランドの省略値は，ゲスト BES を受け入れた後の最大起動ユーザサーバプロセス数に，ゲスト BES を受け入れる前の最大起動プロセス数に対する常駐プロセス数の比率を掛けた値です。計算式を次に示します。

$$\uparrow \mathrm{pd_ha_max_server_process} \times \frac{\sum_{\text{ホストBES}} \mathrm{pd_process_count}}{\sum_{\text{ホストBES}} \mathrm{pd_max_bes_process}} \uparrow$$

64)　pd_ha_resource_act_wait_time = リソース活性化最大待ち時間

～<符号なし整数>((2～3600))（単位：秒）

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。このオペランドには，ユニットを開始するときに，実行系サーバのリソースが活性化されるまでの最大待ち時間を秒単位で指定します。このオペランドに指定した待ち時間の間は，ユニットの開始処理を待ち合わせます。ただし，ここで指定した時間内にリソースが活性化された場合は，すぐにユニットの開始処理を再開します。

《利点》

ユニットの開始処理が完了したときに，ユニット内の実行系サーバの起動処理が完了していないと，業務を開始できません。このオペランドに適切な値を指定すると，ユニットの開始処理を待ち合わせるため，ユニットの開始処理が完了した直後から業務を開始できます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。次に示す条件をすべて満たす場合にこのオペランドを指定してください。

- 影響分散スタンバイレス型系切り替え機能を使用している場合
- KFPS05623-I メッセージが出力された場合
- メッセージ出力対象ユニットに実行系サーバがある場合

指定値の目安を次に示します。

10 ＋リソース活性化処理に掛かる時間（秒）

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，10 が仮定されます。

《備考》

ユニット内に実行系サーバが存在しない場合，このオペランドに指定した時間だけ実行系サーバの開始を待ち合わせます。ただし，ユニット内の全サーバが待機系サーバとして開始した場合は，このオペランドで指定した時間を待たないでユニットの開始処理を再開します。

65）　pd_ha_ipaddr_inherit = Y｜N

系切り替え機能を使用する場合に IP アドレスを引き継ぐかどうかを指定します。HiRDB/パラレルサーバの場合にこのオペランドを指定するか検討してください。IP アドレスを引き継ぐかどうかによって，系切り替え機能の運用方法が異なります。詳細については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

Y：IP アドレスを引き継ぎます。

N：IP アドレスを引き継ぎません。

サーバモードで運用する場合は，IP アドレスを引き継ぐことはできません。

1：1 スタンバイレス型系切り替え機能又は影響分散スタンバイレス型系切り替え機能を使用する場合はこのオペランドを省略してください。指定しても無効になります。

《指定値の目安》

- HiRDB/パラレルサーバで IP アドレスを引き継ぐ構成の場合は，サーバモードの対象になるユニットについてだけ，N（IP アドレスを引き継がない構成）を指定してください。

  なお，システム共通定義の pd_ha_ipaddr_inherit オペランドに N を指定した場合は，このオペランドに Y を指定できません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，Y が仮定されます。

3.2.18　HiRDB Datareplicator に関するオペランド

66）　pd_rpl_hdepath = 抽出側 HiRDB Datareplicator 運用ディレクトリ名

HiRDB Datareplicator 連携機能を使用する場合，抽出側 HiRDB Datareplicator 運用ディレクトリ名を指定します。抽出側 HiRDB Datareplicator の環境変数 HDEPATH で指定した名称をこのオペランドに指定します。

《前提条件》

pd_rpl_init_start オペランドに Y を指定，又は pdrplstart コマンドを入力して HiRDB Datareplicator 連携機能を使用している場合に，このオペランドは有効になります。

《注意事項》

- 抽出側 HiRDB Datareplicator 運用ディレクトリ名を変更する場合は，HiRDB Datareplicator の環境変数 HDEPATH などを変更する必要があります。

《ほかのオペランドとの関連》

pd_rpl_reflect_mode オペランドを指定する場合は，このオペランドを必ず指定してください。

3.2.19　通信処理に関するオペランド

67)　pd_service_port = スケジューラプロセスのポート番号

～<符号なし整数>((5001～65535))

次の場合に，スケジューラプロセスのポート番号を指定します。

- 高速接続機能を使用する場合

  高速接続機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

- HiRDB サーバ側にファイアウォールや NAT が設置されている場合

  この場合の HiRDB の環境設定については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合，ユニット制御情報定義では必ずこのオペランドを指定してください。また，各ユニットで異なるポート番号を指定してください。同じポート番号を指定すると，系切り替え時発生時にどちらかのユニットで系切り替えが失敗します。
- ユニット制御情報定義とシステム共通定義の両方でこのオペランドを指定した場合，ユニット制御情報定義の指定値が有効になります。
- HiRDB のポート番号（pd_name_port オペランド又は pdunit オペランドの-p オプションに指定したポート番号）と同じポート番号を指定した場合，このオペランドの指定値が無効となり，高速接続機能を使用する場合及び HiRDB サーバ側にファイアウォールや NAT が設定されている場合の運用ができなくなります。このとき，KFPS00860-W メッセージが出力されます。
- 高速接続機能を使用して pd_max_users オペランドの値以上の接続要求を同時に発行した場合，接続要求をメッセージキューから取り出すフロントエンドサーバやシングルサーバの稼働数が不足します。そのため，接続要求をメッセージキューから取り出せなくなり，メッセージキュー監視機能によってユニットが停止することがあります。pd_max_users オペランドの値以上の同時接続要求が発生しないようにしてください。メッセージキュー監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。
- スケジューラプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_scd_port
- pdunit -s

68)　pd_change_clt_ipaddr = 0 | 1

HiRDB サーバから HiRDB クライアントへの通信時に使用するネットワークを指定します。通常，このオペランドを指定する必要はありません。

0：

HiRDB サーバから HiRDB クライアントへの通信時，クライアント環境定義の PDCLTRCVADDR オペランドに指定した IP アドレスがあるネットワークを使用します。PDCLTRCVADDR オペランドを省略した場合は標準ホストの IP アドレスが仮定されます。

1：

HiRDB サーバから HiRDB クライアントへの通信時，HiRDB クライアントから HiRDB サーバへの通信に使用したネットワークを使用します。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，1 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 0 が仮定されます）。

69）　pd_registered_port = "ポート番号の予約範囲"〔，"ポート番号の予約範囲"〕…

～〈文字列〉

HiRDB 予約ポート機能を使用する場合に，HiRDB が通信処理で使用するポート番号の範囲を指定します。

HiRDB 予約ポート機能は，サーバ間通信のときだけ有効になります。なお，ポート番号の使用量が少ない場合には，HiRDB 予約ポート機能を使用する必要はありません。

《オペランドの指定方法》

6000～8999，12500～29999，30500～44999 の計 35,000 個の範囲でポート番号を使用する場合の例を次に示します。

set pd_registered_port = "6000：8999"，"12500：29999"，"30500：44999"

《利点》

HiRDB サーバ間の通信処理で使用するポート番号は OS が自動的に割り当てています。通信処理が大量に発生すると，ポート番号の不足によって処理が中断したり，又はほかのプログラムの通信処理に影響を与えたりします。

このオペランドで HiRDB 専用のポート番号の範囲を指定すると，これらの現象を防げます。

《指定値の目安》

- 指定できるポート番号の範囲は 5001～49151 です。
- HiRDB が使用するポート数については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- %windir%¥system32¥drivers¥etc¥services（DNS 環境の場合は定義した場所）には，このオペランドで指定したポート番号は登録しないでください。pd_registered_port_check オペランドに Y，C，又は W を指定すると，%windir%¥system32¥drivers¥etc¥services との重複チェックをします。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。

《指定値のチューニング方法》

次に示す計算式で求めた値が，必要となるポート番号の総数の目安です。

a + b + 100

a：HiRDB 予約ポートの使用数（#OF REGISTERED PORTS）

b：HiRDB 予約ポートオーバ時の OS 自動割り当てポートの使用数（#OF ASSIGNED PORTS）

これらの情報は，統計解析ユティリティのシステムの稼働に関する統計情報で取得できます。

《オペランドの規則》

- 予約範囲は最大 10 個指定できます。
- 予約範囲を二つ以上指定した場合，範囲に含まれるポート番号が重複しないようにしてください。

- 終了ポート番号は開始ポート番号以上の値でなければいけません。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニット制御情報定義では必ずこのオペランドを指定してください※。また，各ユニットで異なるポート番号を指定してください。同じポート番号を指定すると，系切り替え時発生時にどちらかのユニットで系切り替えが失敗します。
- HiRDB 予約ポート機能を使用できないポート番号では，このオペランドで指定したポート番号は使用されません。HiRDB 予約ポート機能の対象範囲については，pd_registered_port_level オペランドを参照してください。
- HiRDB 予約ポート機能を使用する場合のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- ユニット制御情報定義とシステム共通定義の両方でこのオペランドを指定した場合，ユニット制御情報定義の指定値が有効になります。

70）　pd_registered_port_check = Y | N | C | W

pd_registered_port オペランドで範囲指定したポート番号と%windir%¥system32¥drivers¥etc¥services（DNS 環境の場合は定義した場所）に登録されたポート番号との重複チェックをするかどうかを指定します。

Y：

重複チェックをします。重複している場合, KFPS00348-E メッセージを出力して HiRDB の開始処理を中断します。

N：

重複チェックをしません。

C：

重複チェックをします。重複している場合, 重複しているポート番号は HiRDB 予約ポート機能の対象外になります。

W：

重複チェックをします。重複している場合，KFPS00354-W メッセージを出力します。重複しているポート番号は HiRDB 予約ポート機能の対象外になります。

《前提条件》

pd_registered_port オペランドを指定している必要があります。

《指定値の目安》

- ポート番号の重複があると，不正電文の受信や送信電文の未達など HiRDB の通信処理に影響を及ぼすことがあります。
- Y，C，又は W を指定すると，DNS 環境のときにプロセスサーバプロセスの起動が遅くなることがあります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，Y が仮定されます。

71）　pd_registered_port_level = 0 | 1

このオペランドでは，HiRDB 予約ポート機能の対象範囲を指定します。

《前提条件》

pd_registered_port オペランドを指定している必要があります。

《指定値の目安》

0 を指定した場合と 1 を指定した場合の対象範囲を次に示します。

HiRDB が使用するポート番号		pd_registered_port_level の値	
		0	1
HiRDB サーバが使用するポート番号	サーバ間通信用送信ポート番号	○	○
	サーバ間通信用受信ポート番号	○	○
	pd_name_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-p オプションに指定したポート番号	×	×
	pd_service_port オペランドに指定したポート番号	×	×
	pd_scd_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-s オプションに指定したポート番号	×	×
	pd_trn_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-t オプションに指定したポート番号	×	×
	pd_mlg_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-m オプションに指定したポート番号	×	×
	pd_alv_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-a オプションに指定したポート番号	×	×
	コマンド用送信ポート番号	×	×
	コマンド用受信ポート番号	×	×
	対クライアント用送信ポート番号	×	○
	対クライアント用受信ポート番号	○	○
	連絡用ポートのサービスポート番号	×	×
	排他制御で使用するポート番号	×	×
HiRDB クライアントが使用するポート番号	送信ポート番号	×	×
	受信ポート番号	×	×

（凡例）

○:HiRDB 予約ポート機能の対象範囲となります。pd_registered_port オペランドで指定したポート番号が使用されます。

×：HiRDB 予約ポート機能の対象範囲となりません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

72）　pd_ipc_send_retrycount = プロセス間通信の送信リトライ処理の連続繰り返し回数

～<符号なし整数>((1～32767))（単位：回数）

プロセス間通信の送信リトライ処理の連続繰り返し回数を指定します。

このオペランドは pd_ipc_send_retrysleeptime オペランドと関連しています。

（例）

- pd_ipc_send_retrycount = 500
- pd_ipc_send_retrysleeptime = 2

上記のようにオペランドを指定すると，送信リトライ処理 1 回の繰り返し単位が 500 回連続実行で 2 秒スリープとなります。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- このオペランドの指定値を大きくすると CPU 占有率が高くなります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，200 が仮定されます。

73）　pd_ipc_send_retrysleeptime = プロセス間通信の送信リトライ処理のスリープ時間

～<符号なし整数>((0～60))（単位：秒）

プロセス間通信の送信リトライ処理のスリープ時間を指定します。

このオペランドは pd_ipc_send_retrycount オペランドと関連しています。

（例）

- pd_ipc_send_retrycount = 500
- pd_ipc_send_retrysleeptime = 2

上記のようにオペランドを指定すると，送信リトライ処理 1 回の繰り返し単位が 500 回連続実行で 2 秒スリープとなります。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- このオペランドの指定値を大きくすると送信完了時間が長くなります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

74）　pd_ipc_send_count = サーバ間の送信処理のリトライ回数

～<符号なし整数>((1～32767))

サーバ間の送信処理が終了するまでの送信処理のリトライ回数を指定します。なお，1 回で最大 5 秒間のリトライを続けます。省略値では 5 秒×60 回となり 300 秒間リトライを続けます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。送信タイムアウトが頻発する場合に，このオペランドの指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，60 が仮定されます。

75）　pd_ipc_recv_count = サーバ間の受信処理のリトライ回数

～<符号なし整数>((1～32767))

サーバ間の受信処理が終了するまでの受信処理のリトライ回数を指定します。なお，1 回で最大 5 秒間のリトライを続けます。省略値では 5 秒×120 回となり 10 分間リトライを続けます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，120 が仮定されます。

76）　pd_ipc_inet_bufsize = サーバ間通信で使用する送受信バッファ長

～<符号なし整数>((0～262144))（単位：バイト）

サーバ間通信（TCP INET ドメイン）で使用する送受信バッファの最大値を 4096 の倍数で指定します。

《指定値の目安》

通常は 0 を指定して OS の設定値を適用してください。

《注意事項》

- 0 を指定した場合は OS の設定値を使用します。
- TCP のバッファサイズの上限値は各 OS で異なるので注意してください。
- システム定義 pd_utl_buff_size を使用するユティリティの通信処理では，指定したバッファ長を無効にし，pd_utl_buff_size 指定値を元に，最適な値を設定します。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。

システム共通定義の同じオペランドも省略すると，16384 が仮定されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_utl_buff_size

77）　pd_tcp_inet_bufsize = HiRDB クライアントとの通信で使用する送受信バッファ長

～<符号なし整数>((0～262144))（単位：バイト）

HiRDB クライアントとの通信（TCP INET ドメイン）で使用する送受信バッファの最大値を，4096 の倍数で指定します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。次に示す現象が発生した場合に，このオペランドを指定します。

- HiRDB サーバと HiRDB クライアント間で送受信するデータの再送がある場合，このオペランドを指定すると再送を抑止できます。
- 受信側の処理で，データ到着速度が読み込み処理速度よりも遅い場合，このオペランドを指定すると，送信データ量が増えて受信時の到着待ちによる遅延を防げます。

このオペランドには，通信に使用するバッファサイズと同じ値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

- 0 を指定した場合は OS の設定値を使用します。
- TCP のバッファサイズの上限値は各 OS で異なるので注意してください。

3.2.20　Java に関するオペランド

Java ストアドプロシジャ又は Java ストアドファンクションを使用する場合に，Java に関するオペランドを指定します。Java ストアドプロシジャ及び Java ストアドファンクションについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

78)　pd_java_archive_directory = "JAR ファイル格納ディレクトリ"

～<パス名>

Java ストアドプロシジャ又は Java ストアドファンクションで使用する JAR ファイルを格納するディレクトリ名を絶対パス名で指定します。

《注意事項》

- このオペランドを指定する場合，ここで指定するディレクトリを JAR ファイルのインストール前に作成しておく必要があります。
- JAR ファイル格納ディレクトリは，JAR ファイルを格納するための専用のディレクトリです。
- JAR ファイル格納ディレクトリには，インストールした JAR ファイル以外を格納しないでください。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，%PDDIR%¥java が仮定されます。

79)　pd_java_classpath = "Java クラスパス"

～<パス名>

Java 仮想マシンで使用するクラスパスを絶対パス名で指定します。

このオペランドに指定したパスに含まれるクラスは，Java ストアドプロシジャ又は Java ストアドファンクションの処理手続きとして実行される Java メソッドから参照できます。

Java ストアドプロシジャ又は Java ストアドファンクションの外部ルーチン名として指定した JAR ファイルとこのオペランドに指定したパスに同じ名称のクラスがある場合，このオペランドに指定したパスが優先されます。

《オペランドの規則》

- パス名は 1024 文字以内とします。
- パス名の大文字と小文字の区別をしません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。

80）　pd_java_runtimepath = "Java Runtime Environment のルートディレクトリ"

～<パス名>

Java Runtime Environment のルートディレクトリを絶対パス名で指定します。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

《注意事項》

JRE（Java Runtime Environment）の同梱を廃止したため，HiRDB を 07-03 より前のバージョンから 07-03 以降にバージョンアップする場合，このオペランドを追加するか，又はこのオペランドの指定値を変更してください。バージョンアップの注意事項については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「Java ストアドプロシジャ及び Java ストアドファンクションを使用する場合」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，%PDDIR%¥jre が仮定されます。

81）　pd_java_libpath = "Java 仮想マシンのライブラリディレクトリ"

～<パス名>

Java 仮想マシンのライブラリが格納されているディレクトリを，Java Runtime Environment のルートディレクトリ（pd_java_runtimepath オペランドの値）への相対パス名で指定します。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

OS の種類	省略値
Windows（下記以外）	bin¥hotspot，又は bin¥client※
Windows (IPF)	bin¥server
Windows (x64)	

注※

bin¥hotspot に jvm.dll がない場合は，bin¥client を仮定します。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

82）　pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"

～<パス名>

Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合, そのファイル名を絶対パス名で指定します。

《指定値の目安》

このオペランドで指定するファイルの容量が非常に大きくなるため，通常はこのオペランドを指定しないでください。Java ストアドプロシジャ又は Java ストアドファンクションのデバッグ時に指定することをお勧めします。なお，このオペランドで指定するファイルの容量制限はありません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，Java 仮想マシンでの標準出力又は標準エラー出力が無視されます。

《注意事項》

複数プロセスからの同時書き込みがあった場合は，その出力内容を保証しません。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

3.2.21　外部 C ストアドルーチンに関するオペランド

83)　pd_c_library_directory = "C ライブラリファイル格納ディレクトリ"

～<パス名>

C ライブラリファイルを格納するディレクトリを絶対パス名で指定します。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

《注意事項》

このオペランドを指定する場合，ここで指定するディレクトリを C ライブラリファイルの登録前に作成しておく必要があります。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，%PDDIR%¥clib が仮定されます。

3.2.22　ワークファイル出力先ディレクトリの変更

84)　pd_tmp_directory = ワークファイル出力先ディレクトリ名

～<絶対パス名>((3～200))

HiRDB が出力するワークファイルの出力先ディレクトリ名を指定します。指定するディレクトリは HiRDB を起動する前に作成し，全ユーザに対してアクセス権限を付与しておいてください。

HiRDB は，各コマンドで決められた優先順位に従ってワークファイル出力先ディレクトリを決定します。出力先を決定する上でこのオペランドの指定値が有効となる出力ファイルについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。

指定したディレクトリに格納されたワークファイルは，pdcspool コマンドで定期的に削除できます。pdcspool コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。

《前提条件》

このオペランドに指定するディレクトリは，HiRDB を起動する前に作成してください。また，ディレクトリのアクセス権限に「Everyone に対してフルコントロール」を指定してください。

HiRDB 停止中に誤ってディレクトリを削除するなどして，HiRDB の起動時にこのオペランドに指定した名称のディレクトリが存在しなくなった場合は，HiRDB が指定した名称のディレクトリを作成し，アクセス権限に「Everyone に対してフルコントロール」を付与します。

《利点》

- HiRDB の運用ディレクトリ以外に HiRDB が出力するワークファイルの出力先を 1 か所に統一できます。
- pdcspool コマンドと併用することで，コマンド実行後に残るワークファイルを定期的に削除できます。

これらの効果によって，ディスク管理が容易になります。

《注意事項》

- このオペランドには，ドライブ直下（C:¥など）は指定できません。
- HiRDB 以外のプログラムで使用しているディレクトリ名を指定しないでください。HiRDB 以外のプログラムで使用するディレクトリ名を指定すると，pdcspool コマンドを実行した際に，HiRDB 以外のプログラムが出力したファイルが削除されることがあります。

  ただし，次の名称を指定した場合には，pdcspool コマンドを実行してもファイルが削除されないため，ユーザが手動でファイルを削除する必要があります。

  ・ドライブ直下（C:¥など）
- 指定したパス名が正しくない，又は，指定したディレクトリにユーザのアクセス権限がない場合，HiRDB はこのオペランドで指定したディレクトリにアクセスできません。このとき，警告メッセージが出力され，このオペランドの指定値に%PDDIR%¥tmp が仮定されます。
- このオペランドを指定しなかった場合，環境変数 TMP に値が正しく設定されていれば，このオペランドの指定値に環境変数 TMP の設定値が仮定されます。このとき，環境変数に設定されたパス名の長さはチェックしないため，パス名の長さが 513 バイト以上でも設定値は正しいものとみなします。
- 環境変数 TMP に値が設定されていないとき，又は環境変数の設定値が不正のときは，このオペランドの指定値に%PDDIR%¥tmp が仮定されます。

3.2.23　共用メモリに関するオペランド

85）　SHMMAX 共用メモリセグメントサイズの上限値

～<符号なし整数>（単位：メガバイト）

- 32 ビットモードの場合：((6～2047))《200》
- 64 ビットモードの場合：((6～4194304))《1024》

グローバルバッファ用の共用メモリセグメントサイズの上限値をメガバイト単位で指定します。

HiRDB はこのオペランドの指定値を上限としてグローバルバッファ用の共用メモリセグメントを確保します。サーバマシン内の RD エリアに割り当てるグローバルバッファサイズの合計がこのオペランドの値を超える場合は，複数の共用メモリセグメントを割り当てます。

HiRDB は開始時に 1 サーバ当たり最大 16 セグメントのグローバルバッファプール用共用メモリを確保します。グローバルバッファプールが使用している共用メモリセグメントの情報は pdls コマンド（-d mem 指定）によって確認できます。

《指定値の目安》

<32 ビットモード>

- SHMMAX を特に指定する必要はありません。ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合には，200 を指定してください。
- KFPH23005-E メッセージ（エラーコード 20）が出力される場合，マニュアル「HiRDB Version 8 メッセージ」を参照して KFPH23005-E メッセージの対策に示される SHMMAX の値に変更してください。

<64 ビットモード>

- グローバルバッファプール用共用メモリサイズが 16GB 以下の場合，SHMMAX を指定する必要はありません。
- グローバルバッファプール用共用メモリサイズが 16GB より大きい場合，SHMMAX の指定値に以下の式を満たす値を指定してください。

  （SHMMAX の指定値）×16 ＝（グローバルバッファプール用共用メモリサイズ）

《注意事項》

- グローバルバッファ動的変更機能を使用している場合，SHMMAX の指定値の変更は正常停止後のみできます。

  そのため，強制停止または計画停止後の再開始時に KFPH23005-E メッセージ（エラーコード 20）が発生し，SHMMAX の値の変更が必要な場合，マニュアル「HiRDB Version 8 メッセージ」の KFPH23005-E メッセージ（エラーコード 20）の表 項番 3 を参照して，SHMMAX の指定値を変更してください。
- 共用メモリは，共用メモリファイルとして HiRDB 運用ディレクトリがあるディスクに確保されます。このため，ディスクの空き容量が十分でないと，ディスク容量不足になります。
- 共用メモリをページファイルに割り当てる場合，ページファイル容量が十分でないと，ページファイル容量不足になります。

  共用メモリを割り当てる方法については，マニュアル「HiRDB Version 8 コマンドリファレンス」の pdntenv コマンドを参照してください。
- 共用メモリのページ固定機能を使用する場合は，本オペランドにはラージページページ長の整数倍の値を指定してください。共用メモリのページ固定は以下の場合に使用されます。

  ・pd_dbbuff_attribute オペランドに fixed を指定してください。

  ・pd_max_resident_rdarea_no オペランドに 1 以上を指定ください。

  指定値がページ長の整数倍でない場合は HiRDB がページ長の整数倍の値に補正します。
- 本オペランドに指定可能範囲外の値を指定した場合，本オペランドの省略値が仮定されたものとみなし，警告メッセージ KFPH23011-W を出力した後に HiRDB 開始処理を続行します。
- グローバルバッファプール用共用メモリが使用する管理領域は SHMMAX に指定したサイズで分割することができません。そのため，SHMMAX 指定値には上記管理領域のサイズより大きい値を指定してください。グローバルバッファプール用共用メモリが使用する管理領域サイズの詳細はマニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_dbbuff_modify
- pdbuffer
- pd_max_add_dbbuff_no
- pd_max_add_dbbuff_shm_no
- pd_sysdef_default_option（32 ビットモードで v6compatible を指定している場合，このオペランドの省略値は 6 になります）

4 サーバ共通定義

この章では，サーバ共通定義の各オペランドの内容について説明します。

4.1　オペランドの形式

サーバ共通定義では，各サーバ定義（フロントエンドサーバ定義，ディクショナリサーバ定義，及びバックエンドサーバ定義）で共通な情報を定義します。ここで定義した内容は，各サーバ定義で指定しなかったオペランドの仮定値となります。ここでは，サーバ共通定義の各オペランドの形式について説明します。

なお，表中の「番号」は，「4.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

番号	形式
1	〔set pd_max_bes_process = 1 バックエンドサーバ当たりの最大起動プロセス数〕
2	〔set pd_max_dic_process = 1 ディクショナリサーバ当たりの最大起動プロセス数〕
3	〔set pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕〕
4	〔set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔〕
5	〔set pd_max_ard_process = 非同期 READ プロセス数〕
6	〔set pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数〕

(2)　作業表

番号	形式
7	〔set pd_work_buff_mode = each｜pool〕
8	〔set pd_work_buff_size = 作業表用バッファ長〕
9	〔set pd_work_buff_expand_limit = 作業表用バッファの増分上限値〕

(3)　システム監視

番号	形式
10	〔set pd_watch_pc_client_time = クライアントからの要求間隔監視時間〕
11	〔set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〕
12	〔set pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値〕

(4)　排他制御

番号	形式
13	〔set pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ〕

番号	形式
14	[set pd_fes_lck_pool_size = フロントエンドサーバの排他制御用プールサイズ]
15	[set pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数]
16	[set pd_fes_lck_pool_partition = フロントエンドサーバの排他制御用プール分割数]
17	[set pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和]
18	[set pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数]
19	[set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数]
20	[set pd_dbsync_lck_release_count = シンクポイント時のグローバルバッファ排他解除間隔]

(5)　バッファ

番号	形式
21	[set pd_sql_object_cache_size = SQL オブジェクト用バッファ長] ※
22	[set pd_table_def_cache_size = 表定義情報用バッファ長]
23	[set pd_auth_cache_size = ユーザ権限情報用バッファ長]
24	[set pd_view_def_cache_size = ビュー解析情報用バッファ長]
25	[set pd_type_def_cache_size = ユーザ定義型情報用バッファ長]
26	[set pd_routine_def_cache_size = ルーチン定義情報用バッファ長]
27	[set pd_registry_cache_size = レジストリ情報用バッファ長]

(6)　共用メモリ

番号	形式
28	[set pd_sds_shmpool_size = シングルサーバ用共用メモリサイズ]
29	[set pd_dic_shmpool_size = ディクショナリサーバ用共用メモリサイズ]
30	[set pd_bes_shmpool_size = バックエンドサーバ用共用メモリサイズ]

(7)　RPC トレース情報

番号	形式
31	[set pd_rpc_trace = Y | N] ※
32	[set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"] ※
33	[set pd_rpc_trace_size = RPC トレースを取得するファイルの容量] ※

(8)　トラブルシュート情報

番号	形式
34	〔set pd_module_trace_max = モジュールトレース格納最大数〕※
35	〔set pd_module_trace_timer_level = モジュールトレース出力時刻取得方法〕※
36	〔set pd_pth_trace_max = 通信トレース格納最大数〕

(9)　グローバルバッファ

番号	形式
37	〔set pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値〕
38	〔set pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値〕

(10)　セキュリティ

番号	形式
39	〔set pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長〕

(11)　Java

番号	形式
40	〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※
41	〔set pd_java_castoff = Y ｜ N〕

(12)　システムログファイル

番号	形式
42	〔set pd_log_dual = Y ｜ N〕
43	〔set pd_log_remain_space_check = warn ｜ safe〕
44	〔set pd_log_singleoperation = Y ｜ N〕
45	〔set pd_log_rerun_reserved_file_open = Y ｜ N〕
46	〔set pd_log_rerun_swap = Y ｜ N〕
47	〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕
48	〔set pd_log_unload_check = Y ｜ N〕
49	〔set pd_log_max_data_size = ログ入出力バッファ長〕
50	〔set pd_log_write_buff_count = ログ出力バッファ面数〕
51	〔set pd_log_rec_leng = システムログファイルのレコード長〕
52	〔set pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数〕

番号	形式
53	〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕

(13)　シンクポイントダンプファイル

番号	形式
54	〔set pd_spd_dual = Y｜N〕
55	〔set pd_spd_assurance_msg = Y｜N〕
56	〔set pd_spd_assurance_count = 有効保証世代数〕
57	〔set pd_spd_reduced_mode = 縮退運転オプション〕
58	〔set pd_spd_reserved_file_auto_open = Y｜N〕
59	〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕
60	〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕

(14)　サーバ用ステータスファイル（障害発生時）

番号	形式
61	〔set pd_sts_initial_error = stop｜continue｜excontinue〕
62	〔set pd_sts_singleoperation = stop｜continue〕

(15)　バックエンドサーバ接続保持機能

番号	形式
63	〔set pd_bes_connection_hold = Y｜N〕
64	〔set pd_bes_conn_hold_trn_interval = バックエンドサーバ接続保持時間〕

注※

このオペランドを省略すると，システム共通定義又はユニット制御情報定義の同じオペランドの指定値が仮定されます。

4.2　オペランドの説明

4.2.1　プロセスに関するオペランド

1）　pd_max_bes_process = 1 バックエンドサーバ当たりの最大起動プロセス数

～<符号なし整数>((1～2048))《pd_max_users の値》

1 バックエンドサーバ当たりの**最大起動プロセス数**を指定します。マルチフロントエンドサーバの場合，1 バックエンドサーバに pd_max_users の値を超える処理が集中することがあります。このオペランドには，1 バックエンドサーバのプロセスを pd_max_users の値を超えて起動する場合の最大起動プロセス数を指定します。

《前提条件》

このオペランドはマルチフロントエンドサーバのときに指定します。

《指定値の目安》

- 1 バックエンドサーバに集中する可能性がある処理の最大値を次に示します。

  pd_max_users オペランドの値×フロントエンドサーバ数

  ここで求めた値を最大値として，バックエンドサーバへの処理の集中度を考慮してこのオペランドの値を指定してください。必要以上に大きな値を指定すると，メモリを圧迫する原因となります。

- pd_max_users の値よりも小さな値を指定した場合は pd_max_users の値を仮定します。このとき，警告メッセージ（KFPS01888-W）を出力します。

《指定値のチューニング方法》

最大起動プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

2）　pd_max_dic_process = 1 ディクショナリサーバ当たりの最大起動プロセス数

～<符号なし整数>((1～2048))《pd_max_users の値》

1 ディクショナリサーバ当たりの**最大起動プロセス数**を指定します。マルチフロントエンドサーバの場合，ディクショナリサーバに pd_max_users の値を超える処理が集中することがあります。また，マルチフロントエンドサーバでない場合も，RD エリアやグローバルバッファに関連する運用コマンド（pdbufls，pddbls，pdopen，pdclose，pdhold，pdrels）を pd_max_users の値以上同時に実行すると，ディクショナリサーバに pd_max_users の値を超える処理が集中することがあります。

このオペランドには，1 ディクショナリサーバのプロセスを pd_max_users の値を超えて起動する場合の最大起動プロセス数を指定します。

《指定値の目安》

- 1 ディクショナリサーバに集中する可能性がある処理の最大値を次に示します。

  pd_max_users オペランドの値×フロントエンドサーバ数

  ここで求めた値を最大値として，ディクショナリサーバへの処理の集中度を考慮してこのオペランドの値を指定してください。必要以上に大きな値を指定すると，メモリを圧迫する原因となります。

- RD エリアやグローバルバッファに関する運用コマンドの実行で，ディクショナリサーバに処理が集中する場合は，同時に実行する運用コマンドの数が最大値になります。

- pd_max_users の値よりも小さな値を指定した場合は pd_max_users の値を仮定します。このとき，警告メッセージ（KFPS01888-W）を出力します。

《指定値のチューニング方法》

最大起動プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

3)　pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕

～<符号なし整数>

- シングルサーバの場合：((0～3000))《最大起動プロセス数》
- バックエンドサーバの場合：((0～2048))《最大起動プロセス数》
- フロントエンドサーバ又はディクショナリサーバの場合：((0～2048))《↑最大起動プロセス数÷2↑》

常駐プロセス数：

各サーバの常駐プロセス数を指定します。常駐プロセスとは，サーバ開始時からあらかじめ起動しておくプロセスのことです。

《利点》

各サーバで並行処理できるトランザクションが使用するプロセスを，あらかじめシステム開始時に起動して常駐しておくことで，新たにトランザクションが入力されてもプロセス起動時間を削減できます。ただし，HiRDB の開始に時間が掛かります。

《指定値の目安》

- 各サーバプロセスのプロセス固有領域とプロセサの実メモリ量から値を求めてください。サーバプロセスのプロセス固有領域については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- マルチフロントエンドサーバ構成で，更に pd_max_bes_process 又は pd_max_dic_process オペランドを指定している場合は，次に示す条件を満たすようにこのオペランドを指定してください。

  pd_process_count の値≦pd_max_bes_process 又は pd_max_dic_process の値
- このオペランドには，次の表に示す各サーバの**最大起動プロセス数**の範囲内で値を指定します。

サーバの種類		最大起動プロセス数
シングルサーバ		pd_max_users の値
フロントエンドサーバ	マルチフロントエンドサーバの場合	pd_max_users の値＋1
	マルチフロントエンドサーバでない場合	pd_max_users の値
ディクショナリサーバ		pd_max_dic_process※の値
バックエンドサーバ		pd_max_bes_process※の値

注※

pd_max_dic_process 又は pd_max_bes_process オペランドを省略している場合は，pd_max_users の値が仮定されます。

《指定値のチューニング方法》

常駐プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- 常駐プロセス数はメモリと密接に関連しているため，必要以上に大きい値を指定すると，HiRDB を開始できなくなることがあります。
- 常駐プロセス数を超えたプロセスが必要になると，最大起動プロセス数までプロセスを動的に起動します。ただし，pd_max_server_process オペランドの指定値によっては，最大起動プロセス数分のプロセスを起動できないことがあります。

《オペランドの省略値》

このオペランドを省略（又は 0 を指定）し，更に各サーバ定義でもオペランドを省略（又は 0 を指定）すると，次に示す値が仮定されます。

- シングルサーバ又はバックエンドサーバの場合：最大起動プロセス数
- フロントエンドサーバ又はディクショナリサーバの場合：↑最大起動プロセス数÷2↑

サーバ開始時の常駐プロセス数：

HiRDB の開始処理時に常駐化する常駐プロセスの数を指定します。

通常，常駐プロセスの起動処理は HiRDB の開始処理中に行われます。このため，常駐プロセス数が多くなると，それに比例して HiRDB の開始処理時間が長くなります。目安として，100MIPS 程度のサーバマシンでプロセスを一つ起動するのに約 1 秒掛かります。

サーバ開始時の常駐プロセス数の指定有無による処理の違いを次に示します。

●サーバ開始時の常駐プロセス数を指定しない場合（pd_process_count = 500 と指定した場合）

すべての常駐プロセスの起動処理を HiRDB の開始処理中に実行します。常駐プロセス（この場合は 500 個）がすべて起動されないと，HiRDB を開始できません。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 500 秒掛かります。

●サーバ開始時の常駐プロセス数を指定する場合（pd_process_count = 500,50 と指定した場合）

一部の常駐プロセスの起動処理を HiRDB の開始処理中に実行し，残りの常駐プロセスの起動処理を HiRDB の開始処理終了後に実行します。一部の常駐プロセス（この場合は 50 個）が起動されれば，HiRDB を開始できます。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 50 秒掛かります。残りの常駐プロセス（この場合は 450 個）は，HiRDB の開始処理終了後に起動されます。

《利点》

HiRDB の開始処理時間を短縮できます。系切り替え機能を使用している場合など，HiRDB の開始処理時間をなるべく短くしたいときに適用します。

《指定値の目安》

HiRDB の開始処理終了直後に必要なプロセス数を指定します。

《注意事項》

サーバ開始時の常駐プロセス数を指定する場合は，クライアント環境定義の PDCWAITTIME オペランドの値を見直してください。

HiRDB の開始直後に，サーバ開始時の常駐プロセス数の値を超える UAP が実行されると，残りの常駐プロセスを起動した後にトランザクション処理が実行されます。したがって，クライアント環境定義の PDCWAITTIME オペランドの値が小さいと，一部の UAP がタイムアウトで処理できない場合があります。

PDCWAITTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

4）　pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔

～<符号なし整数>((0～1440))《0》（単位：分）

HiRDB の非常駐サーバプロセスの停止処理をする間隔を分単位で指定します。この機能は，稼働中のサーバプロセス数が常駐プロセス数（pd_process_count オペランドの値）より多くなった場合に適用されます。この機能によって停止するサーバプロセス数は HiRDB が自動的に算出して決定します。

《利点》

業務（稼働中のサーバプロセス数）がピークのときに再利用できる非常駐サーバプロセスが増えるため，メモリなどのプロセス資源の利用効率が向上します。

《指定値の目安》

- 例えば，一日のうち 1 時間だけ業務のピーク時間があり，その時間内でピークとなる間隔が 2 分ぐらいの場合，このオペランドに 2 を指定します。
- ピーク時に同時実行されるサーバプロセス数が常駐プロセス数以下の場合，この機能を使用しても効果は期待できないため，このオペランドを省略します。

《指定値のチューニング方法》

システムの稼働に関する統計情報をサーバ単位に 1 週間取得してください。「サービス実行中のサーバプロセス数（# OF PROCESS ON SERVICE）」の値から業務のピークを求めてください。そのピークが現在設定されている常駐プロセス数（pd_process_count オペランドの値）を超えている場合は，ピークごとの間隔を求めてその時間を設定します。

ただし，サーバマシンのメモリ，CPU などの資源に余裕がある場合は，不足している分のプロセス数を常駐プロセス数に加算する方（pd_process_count オペランドの値を大きくする方）がこのオペランドを指定するより性能が向上します。

《注意事項》

このオペランドを省略するか，又は 0 を指定した場合は 10 秒間隔でサービス待ち状態の非常駐サーバプロセスを調査し，サービス待ち状態の非常駐サーバプロセスがあれば，その非常駐サーバプロセスを停止します。

5）pd_max_ard_process = 非同期 READ プロセス数

～<符号なし整数>((0～256))《0》

非同期 READ 機能を使用する場合にこのオペランドを指定します。このオペランドには非同期 READ 処理に必要なプロセス数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（バックエンドサーバ又はディクショナリサーバ）当たりのプロセス数を指定します。非同期 READ 機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《前提条件》

pdbuffer オペランドの-m オプションで 1 以上を指定している必要があります。

《指定値の目安》

- 0 又は 1 を指定してください。ただし，pdbuffer オペランドの-m オプションの値が 2～256 の場合は，-m オプションと同じ値を指定してください。-m オプションの値が 257 以上の場合は，RD エリアやシステムファイルを格納するディスク装置の数（HiRDB/パラレルサーバの場合は 1 サーバ当たりの数）と同じ値を指定するか，又は 256 を指定してください。
- このオペランドの値を大きくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に処理時間を短縮できます。このオペランドの値を小さくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に，非同期 READ プロセスの処理完了待ちが発生する分，処理時間が余計に掛かることがあります。
- このオペランドの値×サーバ数分のプロセスが起動されるため，リソース（共用メモリ及びメッセージキュー）を考慮してこのオペランドの値を決定してください。共用メモリ及びメッセージキューの見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値のチューニング方法》

指定値（非同期 READ プロセス数）のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの規則》

このオペランドに 0 を指定すると，非同期 READ 機能を使用しません。

1：1 スタンバイレス型系切り替え機能を使用している場合，正規 BES とそれに対応する代替 BES に同じ値を指定してください。

《ほかのオペランドとの関連》

このオペランドの値を変更した場合は pd_max_server_process オペランドの値を見直してください。

6）pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数

～<符号なし整数>((2～255))

すべてのバッファプールに対して，**デファードライト処理の並列 WRITE 機能**を使用する場合にこのオペランドを指定します。このオペランドには並列処理するプロセス数を指定します。プロセス数を増加させることで書き込み処理時間を短縮できます。デファードライト処理の並列 WRITE 機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値の目安》

デファードライト処理の並列 WRITE 機能が有効になる最小値の 2 を指定してください。また，マニュアル「HiRDB Version 8 システム運用ガイド」の「デファードライト処理のチューニング」を参照してオペランドの値を決定してください。

《注意事項》

デファードライト処理の並列 WRITE 機能を指定すると，プロセス数が増加するため，CPU 利用率が上がります。

4.2.2　作業表に関するオペランド

7）pd_work_buff_mode = each | pool

HiRDB が作業表を作成するときのバッファの確保方式を指定します。

each：個々の作業表ごとにバッファを確保します。

pool：サーバプロセス単位にバッファプールとして一括して確保します。

《指定値の目安》

- 通常は pool（省略値）を指定してください。特に大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合には pool を指定してください。
- あらかじめ，作業表用バッファに使用できるプロセス固有領域サイズが決まっている場合，pool を指定してください。pool を指定すると，HiRDB が効率良く作業表用バッファを各作業表に配分します。

  この場合，pd_work_buff_size に指定した値でプロセス固有領域を占有し，そのプール内で作業表の入出力をバッファリングします。したがって，pd_work_buff_size に指定した値以上にプロセス固有メモリを占有することはありません。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible を指定している場合，このオペランドの省略値は each になります。

8)　pd_work_buff_size = 作業表用バッファ長
～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((128～1000000))
- 64 ビットモードの場合：((128～4000000000))

HiRDB が作成する作業表のバッファの大きさをキロバイト単位で指定します。

項目	pd_work_buff_mode = each の場合	pd_work_buff_mode = pool の場合
利点	作業表バッファ長が大きいと，データ操作時に発生する作業表用ファイルへの入出力回数を削減できるため，作業表を使用する SQL の実行時間を削減できます。ただし，各サーバのプロセス固有メモリを使用するので，システム全体のメモリ量（実メモリ，仮想メモリ）を考慮して指定してください。 特に pd_work_buff_mode=each の場合，メモリは「pd_work_buff_size の値×必要作業表数」分確保されます。したがって，必要以上に大きな値を指定すると，ほかのプロセスで仮想メモリ不足が生じることがあります。	
適用基準	大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合に指定します。このとき，pd_work_buff_mode = pool を指定してください。	
指定値の目安	• 個々の作業表ごとに確保するバッファの大きさを指定します。 • 作業表用バッファ長に，作業表メモリ容量より大きな値を指定すると，作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。作業表メモリ容量の見積もり式は次のとおりです。 **作業表メモリ容量＝対象となる作業表容量※÷2**	• サーバプロセス単位に一括して確保する作業表用バッファプールの大きさを指定します。 • 大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合は，4352～5120 を指定してください。指定すると，ソートでの入出力単位が大きくなるため，ソート処理時間が短くなります。 • 作業表用バッファ長に，SQL 文単位の作業表メモリ総容量を上限として，大きな値を指定するほど作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。SQL 文単位の作業表メモリ総容量の見積もり式は次のとおりです。 **SQL 文単位の作業表メモリ総容量＝ a×b ＋ c×d**
注意事項	複数のユーザが同時に実行する場合，及び複数個の作業表を使用する SQL 文などを実行する場合には，指定したサイズのバッファが，個々の作業表ごとに確保されます。そのため，大きな値を指定するとメモリを圧迫する原因となります。	SQL 文単位に使用する作業表の数に比べて作業表用バッファ長に指定した値が小さいと，each を指定した場合に比べて処理時間が増加することがあります。具体的には，「SQL 文単位の作業表最大数×128」以上の値を指定してください。SQL 文単位の作業表最大数の見積もり式は次のとおりです。 **SQL 文単位の作業表最大数＝ b ＋ d**
オペランドの規則	指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。	• 指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。 • 384 以上の値を指定してください。384 未満の値が指定されると，自動的に 384 に切り上げられます。
省略値	• このオペランドを省略し，更にシングルサーバ定義の pd_work_buff_size オペランドを省略した場合，128 が仮定されます。	• このオペランドを省略し，更にシングルサーバ定義の pd_work_buff_size オペランドを省略した場合，384（32 ビットモードの場合）又

項目	pd_work_buff_mode = each の場合	pd_work_buff_mode = pool の場合
	• このオペランドを省略して，更にバックエンドサーバ定義及びディクショナリサーバ定義の pd_work_buff_size オペランドを省略した場合，512 が仮定されます。	は 5120（64 ビットモードの場合）が仮定されます。 • このオペランドを省略して，更にバックエンドサーバ定義及びディクショナリサーバ定義の pd_work_buff_size オペランドを省略した場合，1024（32 ビットモードの場合）又は 5120（64 ビットモードの場合）が仮定されます。

a：↑｛作業表（列情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

b：作業表（列情報格納用）の最大数※

c：↑｛作業表（位置情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

d：作業表（位置情報格納用）の最大数※

注※

これらの項目の見積もり方法については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

9)　pd_work_buff_expand_limit = 作業表用バッファの増分上限値

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((128～1000000))
- 64 ビットモードの場合：((128～4000000000))

HiRDB が作成する作業表用バッファ長は pd_work_buff_size オペランドで指定します。この作業表用バッファの不足時，作業表用バッファを自動増分する場合にこのオペランドを指定します。このオペランドで指定した値になるまで作業表用バッファを増分します。

例えば，オペランドに次の指定をした場合，通常 1,024 キロバイトの作業表用バッファを確保します。作業表用バッファが 1,024 キロバイトで不足すると，2,048 キロバイトまで作業表用バッファを増分します。

- pd_work_buff_size = 1024
- pd_work_buff_expand_limit = 2048

次に示すときに HiRDB は作業表用バッファを増分します。

- 結合方式がハッシュジョイン，又は副問合せの実行方式にハッシュ実行が適用されている場合に，必要な作業表用バッファが確保できないとき
- 複数の作業表を同時に使用する場合，各作業表に作業表用バッファを 128 キロバイト割り当てて作業表用バッファが不足するとき

《前提条件》

pd_work_buff_mode オペランドを省略するか，又は pool を指定している必要があります。

《利点》

作業表用バッファ不足（pd_work_buff_size オペランドの指定値不足）で UAP がエラーになるのを防げます。

《注意事項》

次に示すどちらかの条件を満たす場合は作業表用バッファを増分しません。

- pd_work_buff_expand_limit オペランドを指定しない
- pd_work_buff_expand_limit オペランドの値≦pd_work_buff_size オペランドの値

《オペランドの規則》

指定値は 128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。

《ほかのオペランドとの関連》

各サーバプロセスでの作業表用バッファの初回増分時に KFPH29008-I メッセージを出力します。なお，pd_work_table_option オペランドで KFPH29008-I メッセージの出力を抑止できます。

《留意事項》

作業表用バッファを増分した場合，該当するサーバプロセスで使用中の作業表数が 0 になると，増分した作業表用バッファを解放します。作業表の使用数が 0 になるのは次に示すときです。

- 使用中のカーソルをすべてクローズしたとき（この場合，作業表の使用数が 0 にならないことがあります）
- ホールダブルカーソルを使用していない場合は，トランザクションを正常終了するか，又は取り消したとき
- ホールダブルカーソルを使用している場合は，UAP を HiRDB から切り離したとき（DISCONNECT したとき）

《備考》

「ハッシュジョイン，副問合せのハッシュ実行」は，次に示す場合に適用されます。

- pd_additional_optimize_level オペランド，クライアント環境定義の PDADDITIONALOPTLVL オペランド，又は SQL コンパイルオプションの ADD OPTIMIZE LEVEL オペランドで，「コストベース最適化モード 2 の適用」と一緒に「ハッシュジョイン，副問合せのハッシュ実行」を指定する
- SQL 文中の結合方式の SQL 最適化指定に HASH を指定する
- SQL 文中の副問合せ実行方式の SQL 最適化指定に HASH を指定する

4.2.3　システム監視に関するオペランド

10）pd_watch_pc_client_time = クライアントからの要求間隔監視時間

～<符号なし整数>((0～65535))《3600》（単位：秒）

Windows 版の HiRDB クライアントからの要求に対する応答を HiRDB サーバが返してから，次に HiRDB クライアントからの要求があるまでの最大待ち時間（サーバ側の最大待ち時間）を秒単位で指定します。

指定した時間内に HiRDB クライアントから全く要求がない場合には，クライアントに異常が発生したものとして，強制的にサーバとクライアントとの接続を切り離します。この時に HiRDB クライアントに対しては，接続の切り離しについての通知はしません。

時間監視の対象となるのは，CONNECT から DISCONNECT までの間で，非トランザクション状態（SQL の実行を開始してから COMMIT 又は ROLLBACK までの間以外）の間です。

《注意事項》

- 0 を指定した場合には，HiRDB クライアントからの次の要求を無限に待ち続けます。
- このオペランドの指定値を小さい値（1～600 程度）にした場合には，HiRDB クライアントが，SQL 実行時にサーバダウンを検知するなど，不当に異常終了することがあります。
- UNIX 版の HiRDB クライアント（Linux 版の HiRDB クライアントも含む）の場合は，このオペランドの指定に関係なく時間を監視しません。UNIX 版の HiRDB クライアントに対して時間監視をする場合は，HiRDB クライアントのクライアント環境定義 PDSWATCHTIME を指定してください。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSWATCHTIME オペランドを指定してください。

PDSWATCHTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

11）　pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0，2～100000))

UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると, HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。

この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。

このオペランドでは，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。シンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，対象トランザクションを強制的に中断してロールバックをします。これを**シンクポイントダンプ有効化のスキップ回数監視機能**といいます。

シンクポイントダンプ有効化のスキップ回数監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

このオペランドを指定すると，UAP の無限ループなどを防止できます。

《指定値の目安》

通常は 0 を指定してください。0 を指定すると，スキップ回数の上限値を HiRDB が計算します。0 を指定して不都合が発生した場合，又は KFPS02101-I メッセージが出力された場合に，このオペランドの値を変更してください。そのときの指定値の目安については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの規則》

このオペランドを省略すると，シンクポイントダンプ有効化のスキップ回数監視機能を使用しません。

《注意事項》

- このオペランドを指定すると，ログレスモードで実行している UAP も監視対象になります。ログレスモードで実行している UAP の処理が中断すると，データベースを自動回復できないため RD エリアを障害閉塞します。このため，上限値の設定に際しては，ログレスモードで実行する UAP のトランザクション処理中に，該当するサーバ内のほかのトランザクションから出力されるシステムログ量も考慮に入れてください。
- pdload，pdmod，pdrorg，pdexp，pddbst，pdgetcst，pdrbal，pdvrup，及び pdmemdb コマンドはこの機能の監視対象外になります。

12）　pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0～100000))《0》

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになると，シンクポイントダンプの有効化処理ができずにスキップされます。これは，デファードライト処理によってシンクポイントダンプの取得が遅れることで，シンクポイントで出力される更新バッファ数がシンクポイントダンプの取得間隔内で出力できる更新バッファ数を超えるためです。

シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

このオペランドでは，デファードライト処理によってシンクポイントダンプの取得が遅れた場合に，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。

デファードライト処理によってシンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，シンクポイントダンプの取得間隔内でシンクポイントダンプの取得が完了するように，HiRDB が更新バッファ数の上限値を決定します。そして，更新バッファ数が上限値を超えた時点で最も古い更新バッファを出力して，シンクポイント時の更新バッファ総量を制限します。これを**更新バッファ量抑制機能**といいます。

《利点》

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生する，ユニット異常終了を回避できます。

《指定値の目安》

通常はこのオペランドを省略してください。更新処理のスループットを低下させてでも，シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生するユニット異常終了を回避したい場合に，1 を指定します。

許容できるシンクポイントダンプの有効化処理のスキップ回数がログ容量などから分かる場合は，その回数を指定します。

《オペランドの規則》

このオペランドに 0 を指定すると，更新バッファ量抑制機能を使用しません。

《注意事項》

更新バッファ量抑制機能の有効期間について，注意事項を次に示します。なお，更新バッファ量抑制機能の有効期間とは，KFPH23035-I メッセージが出力されてから KFPH23036-I メッセージが出力されるまでの間です。

- 更新バッファ数が，HiRDB が決定した更新バッファの上限値を超えている場合，更新処理の実行後に更新バッファを出力するため，更新処理のスループットが低下します。HiRDB が決定する更新バッファの上限値は，次の計算式で求められます。

（シンクポイントインターバル時間÷WRITE 単価※）
×（1－（前回のシンクポイントからプレシンクまでのログ出力量÷シンクポイント間のログ出力量）
×（バッファプールのバッファ面数÷シンクポイントで更新があったバッファプールの総バッファ面数）

注※　WRITE 単価についての詳細は，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

- pd_dbbuff_rate_updpage オペランド，又は pdbuffer -y オペランドによってデファードライトトリガのトリガ契機を指定している場合，それぞれの指定値が，HiRDB が決定した更新バッファの上限値より大きくなると，HiRDB が決定した更新バッファの上限値をデファードライトトリガのトリガ契機となる更新バッファ数に変更します。

  また，pdbuffer -w オペランドの値は，各バッファの更新バッファ上限値まで出力されるように自動調整されます。

- シンクポイントダンプの有効化処理がスキップした場合，シンクポイント時の更新バッファ出力処理中に検知されます。そのため，更新バッファ量抑制機能が有効となる時期は，シンクポイントダンプの有効化処理がスキップされ，エラーメッセージが出力されるときよりも後になる場合があります。
- 通常，並列 write 機能使用時のシンクポイント処理では各デファードライト処理用並列 WRITE プロセスに対する出力要求はシンクポイントごとに 1 回ですが，更新バッファ量抑制機能使用時はシンクポイントスキップの検出を早めるため複数回に分割して行われます。

4.2.4　排他制御に関するオペランド

13)　pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～2000000))《16000》
- 64 ビットモードの場合：((1～2000000000))《32000》

HiRDB/シングルサーバの場合は，シングルサーバの排他制御で使用する共用メモリ領域（排他制御用プール）の大きさをキロバイト単位で指定します。

HiRDB/パラレルサーバの場合は，バックエンドサーバ及びディクショナリサーバの排他制御で使用する共用メモリ領域の大きさをキロバイト単位で指定します。なお，フロントエンドサーバの排他制御で使用する共用メモリ領域の大きさは，pd_fes_lck_pool_size オペランドで指定してください。

このオペランドの指定対象の領域は，排他資源管理テーブルとして使用されます。排他資源管理テーブル数，排他要求数，及び排他制御用プールサイズの間には，次の関係があります。

「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数

係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。

《指定値の目安》

- 32 ビットモードの場合，6 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 64 ビットモードの場合，4 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 次に示す計算式でこのオペランドの指定値を見積もってください。

HiRDB の種類	計算式（単位：キロバイト）
HiRDB/シングルサーバの場合（32 ビットモード）	↑↑a÷pd_lck_pool_partition の値↑÷6↑×pd_lck_pool_partition の値
HiRDB/パラレルサーバの場合（32 ビットモード）	↑↑b÷pd_lck_pool_partition の値↑÷6↑×pd_lck_pool_partition の値
HiRDB/シングルサーバの場合（64 ビットモード）	↑↑a÷pd_lck_pool_partition の値↑÷4↑×pd_lck_pool_partition の値
HiRDB/パラレルサーバの場合（64 ビットモード）	↑↑b÷pd_lck_pool_partition の値↑÷4↑×pd_lck_pool_partition の値

a：シングルサーバで同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については，「付録 D　排他資源数の見積もり」を参照してください。

b：各サーバ（ディクショナリサーバ又はバックエンドサーバ）で同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については，「付録 D　排他資源数の見積もり」を参照してください。

注

特に，定義系 SQL の DROP TABLE 又は DROP SCHEMA を実行する場合は，このオペランドの指定値を正確に見積もって指定してください。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報※）の「排他資源管理テーブル使用率（%OF USE LOCK TABLE)」を参照してください。使用率の最大値が 80%以上の場合は，これからのデータベース拡張に備えて，このオペランドの値を大きくすることをお勧めします。使用率の最大値が 10%以下の場合は，共用メモリを節約するために，このオペランドの値を小さくすることをお勧めします。

注※

HiRDB/パラレルサーバの場合は「サーバごとのシステムの稼働に関する統計情報」で，ディクショナリサーバ及びバックエンドサーバの「排他資源管理テーブル使用率（%OF USE LOCK TABLE)」を参照してください。

《注意事項》

- このオペランドの指定値が小さ過ぎると，SQL がエラーリターンすることがあります。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

- pd_sysdef_default_option オペランドに v6compatible を指定している場合，このオペランドの省略値は 1024 になります。
- このオペランドは，pd_lck_pool_partition オペランドと関連があります。

14）pd_fes_lck_pool_size = フロントエンドサーバの排他制御用プールサイズ～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～2000000))
- 64 ビットモードの場合：((1～2000000000))

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

フロントエンドサーバの排他制御で使用する共用メモリ領域（排他制御用プール）の大きさをキロバイト単位で指定します。なお，シングルサーバ，ディクショナリサーバ，又はバックエンドサーバの排他制御で使用する共用メモリ領域の大きさは，pd_lck_pool_size オペランドで指定してください。

このオペランドの指定対象の領域は，排他資源管理テーブルとして使用されます。排他資源管理テーブル数，排他要求数，及び排他制御用プールサイズの間には，次の関係があります。

「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数

係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。

《指定値の目安》

- 32 ビットモードの場合，6 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 64 ビットモードの場合，4 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 次に示す計算式を基に，このオペランドの指定値を見積もってください。

  ↑↑（a＋b）÷pd_fes_lck_pool_partition の値↑÷ c ↑×pd_fes_lck_pool_partition の値　（キロバイト）

  a：フロントエンドサーバで同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については,「付録 D　排他資源数の見積もり」を参照してください。

b：（pd_max_users の値＋ 3）×（pd_max_access_tables の値＋ 4）

c：32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入してください。

《指定値のチューニング方法》

統計解析ユティリティ（サーバごとのシステムの稼働に関する統計情報）で，フロントエンドサーバの「排他資源管理テーブル使用率（%OF USE LOCK TABLE)」を参照してください。使用率の最大値が 80%以上の場合は，これからのデータベース拡張に備えて，このオペランドの値を大きくすることをお勧めします。使用率の最大値が 10%以下の場合は，共用メモリを節約するために，このオペランドの値を小さくすることをお勧めします。

《オペランドの省略値》

このオペランドの省略値を次に示します。

・32 ビットモードの場合

{(pd_max_users の値＋ 3）×（pd_max_access_tables の値＋ 4)}　÷ 6

・64 ビットモードの場合

{(pd_max_users の値＋ 3）×（pd_max_access_tables の値＋ 4)}　÷ 4

《注意事項》

- このオペランドの指定値が小さ過ぎると，SQL がエラーリターンすることがあります。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは，pd_fes_lck_pool_partition オペランドと関連があります。

15）　pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数

～<符号なし整数>((1～5000))《1》

排他制御処理の分散をする場合にこのオペランドを指定してください。

HiRDB/シングルサーバの場合は，排他制御処理の分散をするときにシングルサーバの排他制御で使用する排他制御用プールパーティション数を指定します。

HiRDB/パラレルサーバの場合は，排他制御処理の分散をするときにバックエンドサーバ及びディクショナリサーバの排他制御で使用する排他制御用プールパーティション数を指定します。なお，フロントエンドサーバの排他制御で使用する排他制御用プールパーティション数は，pd_fes_lck_pool_partition オペランドで指定してください。

排他制御処理の分散については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値のチューニング方法》

排他制御用プールパーティション数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- このオペランドの指定値を大きくし過ぎると，各排他制御用プールパーティションに割り当てられる共用メモリのサイズが小さくなるため，排他制御用プールパーティションが容量不足となり SQL がエラーリターンすることがあります。また，排他制御の処理に時間が掛かるため，システムのパフォーマンスが下がります。この場合，このオペランドの指定値を小さくしてください。

- 排他制御用プールサイズは 1 キロバイト以上必要であるため，pd_lck_pool_size の値よりも大きな値を指定した場合，このオペランドには pd_lck_pool_size の値が仮定されます。また，このとき，KFPS00421-W メッセージが出力されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_pool_size
- pd_lck_deadlock_check_interval

16）pd_fes_lck_pool_partition = フロントエンドサーバの排他制御用プール分割数

～<符号なし整数>((1～5000))《1》

排他制御処理の分散をするときにフロントエンドサーバの排他制御で使用する排他制御用プールパーティション数を指定します。なお，シングルサーバ，ディクショナリサーバ，又はバックエンドサーバの排他制御で使用する排他制御用プールパーティション数は，pd_lck_pool_partition オペランドで指定してください。

排他制御処理の分散については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値のチューニング方法》

排他制御用プールパーティション数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- このオペランドの指定値を大きくし過ぎると，各排他制御用プールパーティションに割り当てられる共用メモリのサイズが小さくなるため，排他制御用プールパーティションが容量不足となり SQL がエラーリターンすることがあります。また，排他制御の処理に時間が掛かるため，システムのパフォーマンスが下がります。この場合，このオペランドの指定値を小さくしてください。
- 排他制御用プールサイズは 1 キロバイト以上必要であるため，pd_lck_pool_size の値よりも大きな値を指定した場合，このオペランドには pd_lck_pool_size の値が仮定されます。また，このとき，KFPS00421-W メッセージが出力されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_fes_lck_pool_size
- pd_lck_deadlock_check_interval

17）pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和

～<符号なし整数>((0～140000))《256》

トランザクションを越えて保持する表，及び RD エリアの排他資源数を指定します。このオペランドの値に従って，表，RD エリアの UNTIL DISCONNECT 指定の排他を管理するブロックを共用メモリに確保します。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。次に示す場合に指定値を変更するかどうかを検討してください。

- 同時に実行するユティリティの数を大量に増やす場合
- ホールダブルカーソルを使用する場合

- pdlbuffer オペランドで指定したローカルバッファを使用する場合
- 共用 RD エリアを使用する場合

このオペランドの指定値の見積もり方法については，「付録 C.5　1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式」を参照してください。

《指定値のチューニング方法》

このオペランドの指定値が小さいと，トランザクションがロールバックしたり，リターンコード 8 でユティリティが異常終了したりします。このとき, KFPA11914-E, 又は KFPH28001-E メッセージが出力されます。この現象が発生したら，このオペランドの値を大きくしてください。

なお，このオペランドの値を大きくすると，それに比例して必要なメモリも増加します。必要なメモリは「このオペランドの値×48（64 ビットモードの場合は 64)」バイトです。

18）　pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数

～<符号なし整数>((16～1024))《16》

UNTIL DISCONNECT 指定の LOCK 文を実行しない表に対してホールダブルカーソルを使用する場合に，そのホールダブルカーソルのトランザクション当たりの最大同時オープン数を指定します。

《注意事項》

このオペランドに省略時解釈値以外の値を指定すると，共用メモリ使用量が増加します。

《ほかのオペランドとの関連》

このオペランドと次に示すオペランドの指定値は，ロックサーバの共用メモリサイズ計算に使用されます。32 ビットモードの HiRDB の場合，各オペランドの指定値が大き過ぎると，ロックサーバの共用メモリサイズが 2 ギガバイトを超えるため，HiRDB を開始できなくなることがあります。ロックサーバの共用メモリサイズが 2 ギガバイトを超えないように，このオペランドと次に示すオペランドの指定値を調整してください。

- pd_max_access_tables
- pd_max_users
- pd_max_bes_process
- pd_max_dic_process
- pd_lck_hash_entry
- pd_lck_pool_size

共用メモリについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

19）　pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数

～<符号なし整数>((0～2147483647))《0》

排他制御用プールで使用するハッシュテーブルのエントリ数を指定します。HiRDB はここで指定された値に従ってサーバ（シングルサーバ，フロントエンドサーバ，バックエンドサーバ，ディクショナリサーバ）ごとにユニットコントローラ用の共用メモリに排他制御用プールを確保します。

《指定値の目安》

通常はこのオペランドを省略してください。

次に示す条件に該当する場合は指定値を検討してください。

- バージョン 06-02 以降にバージョンアップする場合に共用メモリサイズをできるだけ変更したくないときは 11261 を指定してください。この場合，バージョンアップ前と同数のハッシュエ

ントリ数を確保するため，排他制御用プール中のハッシュテーブルサイズがバージョンアップ前と同じになります。

- このオペランドに推奨値より大きい値を指定すると，性能が向上することがあります。ただし，《推奨値の求め方》の変数 a より大きい値を指定しても，a を指定したときよりも性能が向上することはありません。

《オペランドの規則》

- このオペランド及び各サーバ定義の pd_lck_hash_entry オペランドを省略するか，又はこのオペランドに 0 を指定すると，HiRDB はサーバごとに推奨値を計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 11261 が仮定されます）。推奨値については《推奨値の求め方》を参照してください。
- このオペランドに 0 でも素数でもない値を指定した場合，HiRDB はその指定値を超えない最大の素数が指定されたと仮定します。

《注意事項》

このオペランドの指定値が小さ過ぎると，ハッシュエントリ不足が発生して性能が低下することがあります。このオペランドを省略すれば，ハッシュエントリ不足及びハッシュエントリ不足による性能低下は発生しません。

《推奨値の求め方》

推奨値を次に示します。

推奨値＝ MAX（↑a÷10↑，11261）を超えない最大の素数

変数	変数の算出式		
a	シングルサーバの場合	（pd_max_users の値＋ 3）×（pd_max_access_tables の値＋ 14）＋ pd_lck_pool_size×c	
	フロントエンドサーバの場合	pd_fes_lck_pool_size が省略されている場合	（b ＋ 3）×（pd_max_access_tables の値＋ 4）
		pd_fes_lck_pool_size が指定されている場合	pd_fes_lck_pool_size 値×c
	バックエンドサーバ, ディクショナリサーバの場合	（b ＋ 3）×10 ＋ pd_lck_pool_size の値×c	
b	フロントエンドサーバの場合	マルチフロントエンドサーバの場合	pd_max_users の値＋ 1
		マルチフロントエンドサーバでない場合	pd_max_users の値
	バックエンドサーバの場合	pd_max_users の値＞ pd_max_bes_process の値の場合	pd_max_users の値
		pd_max_users の値≦ pd_max_bes_process の値の場合	pd_max_bes_process の値
	ディクショナリサーバの場合	pd_max_users の値＞ pd_max_dic_process の値の場合	pd_max_users の値
		pd_max_users の値≦ pd_max_dic_process の値の場合	pd_max_dic_process の値
c	32 ビットモードの場合は 6，64 ビットモードの場合は 4		

20）　pd_dbsync_lck_release_count = シンクポイント時のグローバルバッファ排他解除間隔

～<符号なし整数>((0，100～1073741824))《10000》

シンクポイント時に発生するグローバルバッファへの排他を解除する間隔を指定します。

シンクポイント時には，ディスクに反映する必要があるバッファ（更新バッファ）のサーチ処理が発生します。通常は，更新バッファのサーチ処理中に，一定間隔でグローバルバッファへの排他を解除します。

例えば 100 を指定した場合，100 面（グローバルバッファの面数）のサーチ処理が完了すると，一度排他を解除します。その後，再度排他を掛けてサーチ処理を続行します。このように 100 面ごとに排他の解除をします。

《利点》

このオペランドを指定すると，シンクポイント時のグローバルバッファの排他占有時間を調整できます。このオペランドの値を小さくすると，グローバルバッファの排他占有時間が短くなり，シンクポイント時のトラザクション性能が向上することがあります。

グローバルバッファプールの排他占有時間については，統計解析ユティリティを実行して，グローバルバッファプールに関する統計情報の「シンクポイント時のバッファプール排他占有時間（SYNCL）」で確認できます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。次に示す条件をすべて満たす場合にこのオペランドの指定を検討してください。

- シンクポイント時にトランザクション性能が低下する
- pdbuffer オペランドの-n オプションに指定したバッファ面数が非常に大きい

《オペランドの規則》

- 1 から 99 までの値を指定した場合，自動的に 100 が設定されます。
- 0 を指定した場合，更新バッファのサーチ処理が完了するまで，グローバルバッファに対して排他が掛かります。

《注意事項》

このオペランドの値を小さくすると，ほかのトランザクションの割り込みによって，更新バッファのサーチ時間が長くなります。その間に更新されたグローバルバッファもシンクポイント時の出力対象になります。このため，シンクポイント時に出力される更新バッファ数が増加します。シンクポイント時に出力対象となる更新バッファ数については，統計解析ユティリティの，グローバルバッファプールに関する統計情報の「シンクポイント出力ページ数（SYNCW）」で確認できます。

4.2.5　バッファに関するオペランド

21）　pd_sql_object_cache_size = SQL オブジェクト用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((22～256000))
- 64 ビットモードの場合：((22～2000000))

SQL オブジェクトを格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- SQL オブジェクトは，ユーザごとにトランザクションが終了するまでバッファに保存されます。したがって，同時に実行するトランザクションの SQL オブジェクトがバッファに格納できる大きさが必要です。

- 静的 SQL の SQL オブジェクトはトランザクション終了後もバッファに空きがなくなるまで保存して，同一 UAP を実行する複数のユーザで共用することで，SQL 解析処理の削減を図ります。バッファを有効に利用するためには，使用頻度の高い UAP の SQL オブジェクトがバッファに常駐するようにバッファを確保してください。
- バッファ長の見積もり時は，まず UAP で発行する SQL 文の SQL オブジェクト長から UAP 実行に必要なバッファ長を見積もります。次に，同時に実行される UAP と同時実行ユーザ数を考慮して必要なバッファ長を算出してください。
- 1SQL 文の SQL オブジェクト長の見積もり方法については，「付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

（pd_max_users の値＋ 3）×22

《指定値のチューニング方法》

SQL オブジェクト用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

22)　pd_table_def_cache_size = 表定義情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((100～65535))
- 64 ビットモードの場合：((100～2000000))
- HiRDB/シングルサーバの場合

《↑ $\sqrt{\text{pd_max_users の値}+3}$ ↑×100》

- HiRDB/パラレルサーバの場合

《↑ $\sqrt{\text{(pd_max_bes_process 又は pd_max_dic_process の指定値の最大値)}+3}$ ↑×100》

一度使用した表と順序数生成子の定義情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。表と順序数生成子の定義情報は，SQL 文の前処理時に使用します。また，このバッファ内の定義情報は，LRU で管理されます。

《利点》

- 一度使用した表と順序数生成子の定義情報を共用メモリ内に格納して，次回使用するときは入出力なしで使用できます。
- 動的 SQL を多く使用する場合，性能が向上します。
- ディクショナリサーバとの通信回数を削減できます（HiRDB/パラレルサーバの場合）。

《指定値の目安》

- 使用頻度が高い表と順序数生成子の定義情報長の合計値を指定してください。
- 1 順序数生成子当たりの表定義情報バッファサイズは，8 キロバイトです。
- 1 表当たりの表定義情報用バッファ長の求め方については，「付録 C.4　表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

表定義情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

23）　pd_auth_cache_size = ユーザ権限情報用バッファ長

～<符号なし整数>((1～100))《1》（単位：キロバイト）

ユーザ権限情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- ユーザ権限情報用バッファには CONNECT 権限，DBA 権限，及び監査権限の情報を格納します。このバッファに情報がない場合は HiRDB 接続時にディクショナリ表から情報を取得するため，応答時間が遅くなります。したがって，常時接続しているユーザ数分の情報を格納できるようにバッファ長を指定してください。
- 1 ユーザ当たりのユーザ権限情報を格納するのに 68 バイト必要になります。それを基に計算してください。

《指定値のチューニング方法》

ユーザ権限情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

24）　pd_view_def_cache_size = ビュー解析情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((0～32000))
- 64 ビットモードの場合：((0～2000000))

$$《\uparrow \sqrt{(\text{pd_max_users の値}+3)} \uparrow \times 8》$$

ビュー解析情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《利点》

一度使用したビュー解析情報を共用メモリ内に格納して，次回使用するときは入出力なしで使用できます。

《指定値の目安》

使用頻度が高いビュー表のビュー解析情報用バッファ長の合計値を指定してください。ビュー解析情報用バッファ長の見積もりについては，「付録 C.3　ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

ビュー解析情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

25）　pd_type_def_cache_size = ユーザ定義型情報用バッファ長

～<符号なし整数>《0》（単位：キロバイト）

- 32 ビットモードの場合：((100～65536))
- 64 ビットモードの場合：((100～2000000))

このオペランドは，ユーザ定義型に関するオペランドです。ユーザ定義型を使用する場合は，このオペランドを指定することをお勧めします。

ユーザ定義型の情報を格納するバッファ（共用メモリ）の大きさを，キロバイト単位で指定します。ユーザ定義型の情報を使用すると，その情報がこのバッファ内に格納されます。この情報は SQL 文の前処理時に使用します。

《利点》

- このバッファにユーザ定義型の情報を格納すると，このユーザ定義型の情報を次回使用するときに，ディクショナリ表をアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- ディクショナリサーバとの通信回数が削減できます。
- 動的 SQL を多く使用する場合，このオペランドを指定すると性能が向上します。

《指定値の目安》

使用頻度が高い表に定義してあるユーザ定義型の定義情報長を合計します。一つ当たりのユーザ定義型の定義情報長は，次に示す計算式から求めてください。

↑｛(（0.3＋0.2×a＋0.1×b）＋3）÷4｝↑×4

（単位：キロバイト）

a：ユーザ定義型の属性数

b：スーパタイプを継承したサブタイプ数

《指定値のチューニング方法》

ユーザ定義型情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

26）pd_routine_def_cache_size＝ルーチン定義情報用バッファ長

～<符号なし整数>《100》（単位：キロバイト）

- 32 ビットモードの場合：((0，20～65536))
- 64 ビットモードの場合：((0，20～2000000))

次に示す定義情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。これらの定義情報は SQL 文の前処理時に使用します。

- プラグイン関数の定義情報
- システム定義スカラ関数の定義情報
- ルーチンの定義情報

《利点》

- このバッファに前記の定義情報を格納すると，これらの情報を次回使用するときに，ディクショナリ表をアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- ディクショナリサーバとの通信回数が削減できます。

《適用基準》

次に示す SQL を多く使用する場合に指定します。

- プラグインを使用する SQL
- システム定義スカラ関数を使用する SQL
- ルーチンを使用する SQL

《指定値の目安》

このオペランドの指定値の求め方については，「付録 C.6　ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

ルーチン定義情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

このオペランドの指定値が全プラグイン提供関数の定義情報の合計値より小さい場合，プラグイン提供関数の定義情報はバッファ中に確保されません。

27)　pd_registry_cache_size = レジストリ情報用バッファ長

～<符号なし整数>((0～65536))《10》(単位：キロバイト)

このオペランドは，プラグインに関するオペランドです。レジストリ情報を使用するプラグインを使用する場合は，このオペランドを指定することをお勧めします。HiRDB Text Search Plug-in を使用する場合は，指定することをお勧めします。

レジストリ情報を格納するバッファ（共用メモリ）の大きさを，キロバイト単位で指定します。レジストリ情報を使用すると，レジストリ情報がバッファ内に格納されます。レジストリ情報は SQL 文の実行時に使用します。

《利点》

- このバッファにレジストリ情報を格納すると，レジストリ情報を次回使用するときに，レジストリをアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- ディクショナリサーバとの通信回数が削減できます。
- レジストリ情報を多く使用するプラグインを使用する場合，このオペランドを指定すると性能が向上します。

《指定値の目安》

レジストリ情報用バッファ長は，次に示す計算式から求めてください。

↑（0.3＋a）↑×b　　（単位：キロバイト）

a：レジストリキー長の平均長（単位：キロバイト）

レジストリキー値の平均長は次に示す SQL で求められます。

この SQL の結果の単位はバイトのため，キロバイトに変換してください。

```
SELECT AVG(KEY_LENGTH) FROM MASTER.SQL_REGISTRY
```

b：レジストリキーの登録数

レジストリキーの登録数は次に示す SQL で求められます。

```
SELECT COUNT(*) FROM MASTER.SQL_REGISTRY
```

《指定値のチューニング方法》

レジストリ情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

4.2.6　共用メモリに関するオペランド

28)　pd_sds_shmpool_size = シングルサーバ用共用メモリサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～200000))
- 64 ビットモードの場合：((1～4000000000))

このオペランドは，HiRDB/シングルサーバ限定のオペランドです。ユニットコントローラ用の共用メモリの中で，シングルサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。サーバ共通定義及びシングルサーバ定義の pd_sds_shmpool_size オペランドを省略すると，このオペランドの値を HiRDB が計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます）。HiRDB はマニュアル「HiRDB Version 8 システム導入・設計ガイド」に記載されている「シングルサーバが使用する共用メモリの計算式」を基に計算をします。この計算式中の変数の説明に出てくるオペランドの値を変更した場合，このオペランドの値を HiRDB が自動的に再計算します。なお，計算式中の変数「サーバ内のインデクス総数」は pd_assurance_index_no オペランドの値が仮定されます。また，pd_assurance_table_no オペランドを省略した場合は，pd_assurance_table_no オペランドの値として 100 が仮定されます。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

- 必要以上にこのオペランドの値が大きいと，共用メモリをむだに確保してしまいます。また，このオペランド値が小さ過ぎると，次に示す現象が発生します。
  ・HiRDB を開始できなくなります。
  ・UAP 又はユティリティが実行できなくなります。
- このオペランドを省略する場合は，必ず pd_assurance_table_no，及び pd_assurance_index_no オペランドに適切な値を指定してください。

29）pd_dic_shmpool_size = ディクショナリサーバ用共用メモリサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～200000))
- 64 ビットモードの場合：((1～4000000000))

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。ユニットコントローラ用の共用メモリの中で，ディクショナリサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。サーバ共通定義及びディクショナリサーバ定義の pd_dic_shmpool_size オペランドを省略すると，このオペランドの値を HiRDB が計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます）。HiRDB はマニュアル「HiRDB Version 8 システム導入・設計ガイド」に記載されている「ディクショナリサーバが使用する共用メモリの計算式」を基に計算をします。この計算式中の変数の説明に出てくるオペランドの値を変更した場合，このオペランドの値を HiRDB が自動的に再計算します。なお，計算式中の変数「サーバ内のインデクス総数」はデータディクショナリ表のインデクス数が仮定されます。また，計算式中の変数「インデクス用のグローバルバッファプール数」，及び「グローバルバッファ総数（pdbuffer オペランドの指定数）」は，32 ビットモードの場合は 500，64 ビットモードの場合は 1000 が仮定されます。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

- 必要以上にこのオペランドの値が大きいと，共用メモリをむだに確保してしまいます。また，このオペランド値が小さ過ぎると，次に示す現象が発生します。
  ・ユニットを開始できなくなります。
  ・UAP 又はユティリティが実行できなくなります。
- このオペランドを省略する場合は，必ず pd_assurance_index_no オペランドに適切な値を指定してください。

30）　pd_bes_shmpool_size = バックエンドサーバ用共用メモリサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：**((1～200000))**
- 64 ビットモードの場合：**((1～4000000000))**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。ユニットコントローラ用の共用メモリの中で，バックエンドサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。サーバ共通定義及びバックエンドサーバ定義の pd_bes_shmpool_size オペランドを省略すると，このオペランドの値を HiRDB が計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます）。HiRDB はマニュアル「HiRDB Version 8 システム導入・設計ガイド」に記載されている「バックエンドサーバが使用する共用メモリの計算式」を基に計算をします。この計算式中の変数の説明に出てくるオペランドの値を変更した場合，このオペランドの値を HiRDB が自動的に再計算します。なお，計算式中の変数「サーバ内のインデクス総数」は pd_assurance_index_no オペランドの値が仮定されます。また，計算式中の変数「インデクス用のグローバルバッファプール数」，及び「グローバルバッファ総数（pdbuffer オペランドの指定数）」は，32 ビットモードの場合は 500，64 ビットモードの場合は 1000 が仮定されます。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

- 必要以上にこのオペランドの値が大きいと，共用メモリをむだに確保してしまいます。また，このオペランド値が小さ過ぎると，次に示す現象が発生します。
  ・ユニットを開始できなくなります。
  ・UAP 又はユティリティが実行できなくなります。

- このオペランドを省略する場合は，必ず pd_assurance_table_no，及び pd_assurance_index_no オペランドに適切な値を指定してください。

4.2.7　RPC トレース情報に関するオペランド

31）　pd_rpc_trace = Y | N

RPC トレース情報を取得するかどうかを指定します。RPC トレース情報には, HiRDB の保守情報が出力されます。

通常，このオペランドは省略してください。

Y：RPC トレースを取得します。

N：RPC トレースを取得しません。

《注意事項》

このオペランドに Y を指定すると，通信性能が低下します。

《オペランドの省略値》

このオペランドを省略すると，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。ユニット制御情報定義とシステム共通定義の同じオペランドも省略すると，N が仮定されます。

32）　pd_rpc_trace_name = "RPC トレースを取得するファイル名称"

～< 254 文字以内のパス名>

RPC トレースファイルの名称を絶対パス名で指定します。RPC トレースファイルとして，指定したファイル名の後に 1，2，l（エル）を付けた三つのファイルが作成されます。

《注意事項》

このオペランドで指定したディレクトリ下に，最大「pd_rpc_trace_size の値× 2」の大きさのファイルが作成されます。ファイル容量に注意してください。

《オペランドの省略値》

このオペランドを省略すると，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。ユニット制御情報定義とシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

%PDDIR%¥spool¥rpctr

33）　pd_rpc_trace_size = RPC トレースを取得するファイルの容量

～<符号なし整数>((1024～2147483648))（単位：バイト）

RPC トレースファイルの容量をバイト単位で指定します。

このオペランド指定値は，"RPC トレースファイル"1 及び"RPC トレースファイル"2 のそれぞれの容量となります。

《指定値の目安》

RPC トレースは，すべてのプロセス間の通信電文が一つのファイルに時系列に蓄積されます。その量がこの指定値を超えると，もう一方のファイルにトレースデータの出力を切り替えます。古いトレースは上書きされて消去されます。このため，指定値が小さいと十分なトレースを取得できなくなり, トラブルシュートが困難になることがあります。したがって, このオペランドには 1,000,000 以上の値を指定してください。

《注意事項》

このオペランドで指定した値は RPC トレースファイル l（エル）の容量に影響を与えません。RPC トレースファイル l（エル）の容量は 0 バイト固定です。

《オペランドの省略値》

このオペランドを省略すると，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。ユニット制御情報定義とシステム共通定義の同じオペランドも省略すると，4096 が仮定されます。

4.2.8　トラブルシュート情報に関するオペランド

34）　pd_module_trace_max = モジュールトレース格納最大数

～<符号なし整数>((126～16383))

HiRDB のプロセスは，実行した関数やマクロの履歴をプロセス固有メモリ中に記録しています。この履歴をモジュールトレースといい，このオペランドではモジュールトレースのレコード数を指定します。この履歴の内容は core ファイルに取り込まれ，プロセス異常終了時に出力されます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。ユニット制御情報定義とシステム共通定義の同じオペランドも省略すると，126 が仮定されます。

《注意事項》

各プロセスでは，以下のサイズのプロセス固有メモリが確保されます。

32 ビットモードの場合：64 ＋ 48×pd_module_trace_max オペランドの値（バイト）

64 ビットモードの場合：64 ＋ 64×pd_module_trace_max オペランドの値（バイト）

35）　pd_module_trace_timer_level = 0 | 10 | 20

モジュールトレースに出力する時刻の取得方法を指定します。このオペランドの指定値の意味を以下に示します。

指定値	時刻取得方法
0	すべてのモジュールトレース出力箇所で，秒単位で取得します。
10	すべてのモジュールトレース出力箇所のうち，入出力処理前後など性能調査で問題となりやすい箇所だけマイクロ秒単位で取得し，そのほかの箇所では秒単位で取得します。
20	すべてのモジュールトレース出力箇所で，マイクロ秒単位で取得します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。ユニット制御情報定義とシステム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

このオペランドに 0 以外の値を指定すると，時刻をマイクロ秒で取得する関数を発行するため，性能が低下することがあります。

36)　pd_pth_trace_max = 通信トレース格納最大数

～<符号なし整数>((1024～8388608))

トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。

《指定値の目安》

通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。ユニット制御情報定義とシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。

《注意事項》

通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

4.2.9　グローバルバッファに関するオペランド

37)　pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値

～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加するグローバルバッファ数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

- pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，その数に余裕を持たせた値を指定してください。
- 次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_max_add_dbbuff_no の値≦2000000－HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《オペランドの省略値》

このオペランドの省略値を次に示します。

条件		省略値
32 ビットモードの場合	a ≧500 の場合	256
	a ＜ 500 の場合	500－ a
64 ビットモードの場合	a ≧1000 の場合	256
	a ＜ 1000 の場合	1000－ a

a：HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《注意事項》

このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_shm_no

38）pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値

～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加したときに割り当てられる，共用メモリセグメント数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，値を指定してください。

《オペランドの省略値》

このオペランドの省略値を次に示します。

条件		省略値
pd_max_add_dbbuff_no オペランドを省略した場合	32 ビットモードの場合	（16－a）＋ 500
	64 ビットモードの場合	（16－a）＋ 1000
pd_max_add_dbbuff_no オペランドを指定した場合	32 ビットモードの場合	↓pd_max_add_dbbuff_no の値×1.5 ＋ 16↓ （計算値が 32752 以上だった場合は自動的に 32752 が設定されます）
	64 ビットモードの場合	

a：HiRDB 開始時に各サーバに割り当てられている共用メモリセグメント数

《注意事項》

- 次に示す条件式を満たす場合は，このオペランドに pd_max_add_dbbuff_no オペランドの値が仮定されます。

  pd_max_add_dbbuff_shm_no の値＜ pd_max_add_dbbuff_no の値

  省略値が上記の条件を満たす場合も，pd_max_add_dbbuff_no オペランドの値が仮定されます。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- 追加する共用メモリセグメントサイズが SHMMAX オペランドの値を超える場合，SHMMAX オペランドの値を上限値とした複数の共用メモリセグメントに分割されます。追加する共用メモリセグメントサイズを想定して SHMMAX オペランドの値を増やすか，又は分割されたときに不足しないように pd_max_add_dbbuff_shm_no オペランドの値を増やしてください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_no

4.2.10　セキュリティに関するオペランド

39)　pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長

～<符号なし整数>((1～2000000))（単位：キロバイト）

セキュリティ監査機能の情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

セキュリティ監査情報用バッファ長は，次に示す計算式から求めてください。

↑0.3 + a ×0.25↑　　（単位：キロバイト）

a ：セキュリティ監査機能の監査証跡に指定した絞り込み条件のオブジェクト数

《オペランドの省略値》

このオペランドを省略した場合，次に示すサイズのバッファ長が HiRDB の開始時に取得されます。このメモリが確保できない場合，HiRDB は開始できないため注意してください。

↑0.3 + MAX｛(a + 100)，(a×1.2)｝×0.25↑

（単位：キロバイト）

a ：セキュリティ監査機能の監査証跡に指定した絞り込み条件のオブジェクト数

《注意事項》

このオペランドで指定した大きさのメモリが確保できない場合，HiRDB は開始できません。

4.2.11　Java に関するオペランド

Java ストアドプロシジャ又は Java ストアドファンクションを使用する場合に，Java に関するオペランドを指定します。Java ストアドプロシジャ及び Java ストアドファンクションについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

40)　pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"

～<パス名>

Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合, そのファイル名を絶対パス名で指定します。

《指定値の目安》

このオペランドで指定するファイルの容量が非常に大きくなるため，通常はこのオペランドを指定しないでください。Java ストアドプロシジャ又は Java ストアドファンクションのデバッグ時に指定することをお勧めします。なお，このオペランドで指定するファイルの容量制限はありません。

《オペランドの省略値》

このオペランドを省略すると，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。ユニット制御情報定義とシステム共通定義の同じオペランドも省略すると，Java 仮想マシンでの標準出力又は標準エラー出力が無視されます。

《注意事項》

複数プロセスからの同時書き込みがあった場合は，その出力内容を保証しません。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

41）　pd_java_castoff = Y | N

HiRDB のサーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，バックエンドサーバ）プロセスで，Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。

Y：Java 仮想マシンを起動した場合，以下の契機でサーバプロセスを終了させます。

#	サーバの種類	プロセス名称	プロセス終了契機
1	シングルサーバ	pdsds	UAP の切り離し時
2	フロントエンドサーバ	pdfes	UAP の切り離し時
3	ディクショナリサーバ	pddic	トランザクション決着時, 又は UAP の切り離し時
4	バックエンドサーバ	pdbes	トランザクション決着時, 又は UAP の切り離し時

N：Java 仮想マシンを起動してもサーバプロセスが終了しない。

ここでの指定値は，各サーバ定義の pd_java_castoff オペランドの省略時の解釈値となります。

《指定値のチューニング方法》

Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。

Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に, pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。

#	懸案
1	Java 仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。
2	Java 仮想マシンを使用しないコネクションで実行する SQL でも，別のコネクションで Java 仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多い SQL を実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。 なお，Java 仮想マシンの機能については Java 仮想マシンのドキュメントのオプションを参照してください。

《注意事項》

Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- pd_process_count

4.2.12　システムログファイルに関するオペランド

42)　pd_log_dual = Y | N

システムログファイルを二重化するかどうかを指定します。

Y：システムログファイルを二重化します。

N：システムログファイルを二重化しません。

《利点》

システムログファイルを二重化すると，HiRDB はA系及びB系の両方に同じシステムログを取得します。取得したシステムログを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシステムログを読み込めるため，システムの信頼性を向上できます。

《ほかのオペランドとの関連》

システムログファイルを二重化する場合は，各サーバ定義の pdlogadpf オペランドで B 系のシステムログファイル名を指定してください。

43)　pd_log_remain_space_check = warn | safe

システムログファイルの空き率が**警告値**未満になったときの HiRDB の処理方式を指定します。この機能を**システムログファイルの空き容量監視機能**といいます。システムログファイルの空き容量監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

warn：

システムログファイルの空き率が警告値未満になった場合，警告メッセージ KFPS01162-W を出力します。

safe：

システムログファイルの空き率が警告値未満になった場合，新規トランザクションのスケジューリングを抑止して，サーバ内の全トランザクションを強制終了します。このとき，KFPS01160-E メッセージを出力します。

《指定値の目安》

システムログファイルの空き容量不足によるユニットの異常終了の可能性を低くできるため，safe の指定を推奨します。ただし，safe を指定すると，システムログファイルの空き容量が不足したときに，サーバ内の全トランザクションが強制終了されます。このため，システムログファイルの設計をより正確に行う必要があります。システムログファイルの設計については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

44)　pd_log_singleoperation = Y | N

このオペランドは，システムログファイルを二重化している場合に指定してください。二重化していない場合は指定する必要はありません。

システムログファイルの片系運転をするかどうかを指定します。システムログファイルに障害が発生して，両系ともに使用できるシステムログファイルがない場合でも，HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了しないで正常な片方の系だけで処理を続行します。これを**システムログファイルの片系運転**といいます。

また，両方のシステムログファイルで処理を続行すること(通常の処理形態)を**システムログファイルの両系運転**といいます。

Y：システムログファイルの片系運転をします。

N：システムログファイルの片系運転をしません。常に両系運転とします。

《前提条件》

pd_log_dual オペランドに Y を指定している必要があります。

45）　pd_log_rerun_reserved_file_open = Y｜N

システムログファイルの自動オープンをするかどうかを指定します。

HiRDB（又はユニット）を再開始するときに，「上書きできる状態」のシステムログファイルがない場合，予約のファイルがあればHiRDBが予約のファイルをオープンして「上書きできる状態」にし，処理を続行します。これを**システムログファイルの自動オープン**といいます。

予約ファイルを使用するのは，次の場合です。

- 再開始後，最初のシンクポイントダンプを取得するまでの間
- オープンされたファイルグループすべてが「上書きできない状態」の場合

Y：システムログファイルの自動オープンをします（予約ファイルをオープンして使用します）。

N：システムログファイルの自動オープンをしません（予約ファイルを使用しません）。

《利点》

このオペランドにYを指定すると，ユニットの再開始時に「スワップ先にできる状態」のファイルがなくても，予約のファイルがあればユニットを再開始できます。

ただし，「アンロード待ち状態」のファイルがあるときは，ユニットを停止します。「アンロード待ち状態」のファイルのアンロードを実行した後，再度ユニットを開始してください。

46）　pd_log_rerun_swap = Y｜N

HiRDB（又はユニット）の再開始時に，システムログファイルをスワップするかどうかを指定します。

Y：システムログファイルをスワップします。

N：システムログファイルをスワップしません。

《利点》

このオペランドにYを指定すると，再開始前後でシステムログファイルを物理的に分けられます。このため，再開始前のシステムログファイルをサーバ稼働中に使用できます。

47）　pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間

～<符号なし整数>((1～32580))《180》（単位：秒）

システムログファイルのスワップが完了するまでの待ち時間を秒単位で指定します。指定した時間を過ぎてもシステムログファイルのスワップが完了しない場合は，ユニットが異常終了します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。マシンの性能が低いなどの理由によってシステムログファイルのスワップに時間が掛かる場合は，このオペランドの値を大きくしてください。また，ディスク障害などによってシステムログファイルのスワップに遅延が発生した場合に，より短い時間で異常を検知し，ユニットを異常終了させたいときは，このオペランドの値を小さくしてください。

48）　pd_log_unload_check = Y｜N

システムログファイルのアンロード状態を，HiRDBがチェックするかどうかを指定します。

Y：

アンロード状態をチェックします（通常の運用となります）。

N：

アンロード状態をチェックしません。アンロードの状態に関係なく，次に示す条件がすべて満たされれば，システムログファイルを「スワップ先にできる状態」にします。

- 上書きできる状態
- 抽出完了状態（HiRDB Datareplicator）

このときシステムログファイルの運用方法が，アンロード状態のチェックを解除する運用になります。アンロード状態のチェックを解除する運用については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

N を指定したときの利点を次に示します。

- システムログファイルのアンロード操作がなくなるため，運用方法が簡単になります。
- アンロードログファイルを保管するためのファイル容量が不要になります。

《指定値の目安》

データベースを回復するときにシステムログを使用しない場合（バックアップ取得時点に回復できればよい場合）に N を指定します。

《注意事項》

N を指定したときの注意事項を次に示します。

- バックアップ取得時点にしかデータベースを回復できません。
- データベースの回復にシステムログが必要なのにこの運用をした場合はデータベースの回復手段がなくなります。

49)　pd_log_max_data_size = ログ入出力バッファ長

～<符号なし整数>((32000～523000))《400000》(単位：バイト)

システムログの入出力に使用するバッファの大きさをバイト単位で指定します。

《指定値の目安》

- **HiRDB/シングルサーバの場合**

  指定値のチューニング方法に従って指定値を変更してください。

- **HiRDB/パラレルサーバの場合**

  条件式 1 を満たす値を指定してください。pd_rpl_reflect_mode オペランドに uap を指定している場合，又は回復不要 FES を使用している場合は，条件式 1 と条件式 2 の両方を満たす値を指定してください。指定値を最適化する場合は，指定値のチューニング方法に従って指定値を変更してください。

  条件式 1：ログ入出力バッファ長≧a

  a：72×（一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数）＋ 1344

  条件式 2：ログ入出力バッファ長≧b

  b：(一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数＋ 1）×128 ＋ 64

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）の次に示す情報を調べて，このオペランドの指定値を変更してください。

- 入出力待ちバッファ面数（# OF BUFFER FOR WAIT I/O）

  入出力待ちバッファ面数の平均値が 100 を大きく超える場合は，このオペランドの値を大きくして，平均値が 100 に近付くようにしてください。

- カレントバッファなしによる待ち回数（# OF WAIT THREAD）

  カレントバッファなしによる待ち回数が 0 以外の値になる場合は，このオペランドの値を大きくしてください。

《ほかのオペランドとの関連》

- pd_log_write_buff_count オペランドとともに，ログ入出力バッファ長を決定してください。
- pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は 32000 になります。

50）　pd_log_write_buff_count = ログ出力バッファ面数

～<符号なし整数>((3～65000))《10》

システムログの出力に使用するバッファの面数を指定します。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）のカレントバッファなしによる待ち回数（# OF WAIT THREAD）を調べて，待ち回数が多ければ，スループットが向上するよう指定値を大きくしてください。

《ほかのオペランドとの関連》

- pd_log_max_data_size オペランドとともに，ログ出力バッファ面数を決定してください。
- pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，このオペランドの省略値は 3 になります。

51）　pd_log_rec_leng = システムログファイルのレコード長

～<符号なし整数>((1024，2048，4096))《4096》（単位：バイト）

システムログファイルのレコード長を指定します。1024，2048，4096 のどれかを指定します。

レコード長は pdloginit コマンドの-l オプションで設定します。-l オプションで設定するレコード長をこのオペランドに指定してください。

《注意事項》

- pdloginit コマンドの-l オプションで設定するレコード長と異なるレコード長をこのオペランドに指定すると，そのシステムログファイルはオープンできません。
- システムログファイルのレコード長を変更する方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

52）　pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数

～<符号なし整数>((0～256))

ロールバック処理でシステムログの入力に使用するバッファの面数を指定します。このオペランドに 0 を指定した場合，HiRDB がロールバック用ログ入力バッファ面数を決定します。

《指定値のチューニング方法》

このオペランドに 0 を指定してください。0 を指定してメモリ不足が発生する場合は，このオペランドの指定を省略してください。

《オペランドの省略値》

このオペランドの省略値を次に示します。

- ユニットに 1：1 スタンバイレス型系切り替え機能を適用している場合
  代替 BES 数×2
- ユニットに影響分散スタンバイレス型系切り替えを適用している場合
  （ホスト BES 数＋ pd_ha_max_act_guest_servers の値）×2
- 上記以外の場合
  ユニット内サーバ数×2

53)　pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔,拡張上限サイズ〕

～<符号なし整数>((0～104857600))《0，0》(単位：レコード)

このオペランドは，システムログファイルの自動拡張機能を使用する場合に指定します。

一つのシステムログファイルが拡張契機 1 回当たりに拡張するサイズ，及び拡張できるファイルサイズの上限をレコード数で指定します。

1 回当たりに拡張するサイズを省略するか，又は 0 を指定した場合，システムログファイルの自動拡張を行いません。拡張上限サイズを省略するか，又は 0 を指定した場合，ファイルシステム領域のあるディスクが満杯になるか，システムログファイルの容量が上限に達するまでシステムログファイルの自動拡張を行います。また，1 回当たりに拡張するサイズに拡張上限サイズより大きい値を指定した場合，拡張上限サイズに指定した値まで自動拡張を行います。

システムログファイルの自動拡張機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《前提条件》

pd_large_file_use オペランドに Y を指定しているか，又は指定を省略している必要があります。

《指定値の目安》

- 1 回当たりに拡張するサイズには，pdloginit コマンドの-n オプションでシステムログファイルを作成した際に指定したレコード数を基に指定します。サーバごとにシステムログファイルのレコード数の平均値の 1/10 を計算し，最大値を持つサーバの値を指定してください。

  ・1:1 スタンバイレス型系切り替え機能を適用している BES の場合，代替 BES のシステムログファイルを含めた平均値の 1/10 を計算してください。

  ・影響分散スタンバイレス型系切り替え機能を適用している BES の場合，ゲスト BES のシステムログファイルを含めた平均値の 1/10 を計算してください。

- 通常，拡張上限サイズは省略してください。

《指定値のチューニング方法》

システムログの出力量が自動拡張で拡張するサイズを超えると，システムログファイルが満杯になってユニットダウンする場合があります。その場合は，1 回当たりに拡張するサイズを大きくしてください。また，拡張処理に時間が掛かり，トランザクション性能に影響を及ぼす場合は，1 回当たりに拡張するサイズを小さくしてください。

4.2.13　シンクポイントダンプファイルに関するオペランド

54)　pd_spd_dual = Y | N

シンクポイントダンプファイルを二重化するかどうかを指定します。

Y：シンクポイントダンプファイルを二重化します。

N：シンクポイントダンプファイルを二重化しません。

《利点》

シンクポイントダンプファイルを二重化すると，HiRDB はA系及びB系の両方に同じシンクポイントダンプを取得します。取得したシンクポイントダンプを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシンクポイントダンプを読み込めるため，システムの信頼性を向上できます。

《ほかのオペランドとの関連》

シンクポイントダンプファイルを二重化する場合は，各サーバ定義の pdlogadpf オペランドで B 系のシンクポイントダンプファイル名も指定してください。

55）　pd_spd_assurance_msg = Y｜N

シンクポイントダンプが有効化されたとき，メッセージ KFPS02183-I を出力するかどうかを指定します。

Y：出力します。

N：出力しません。

56）　pd_spd_assurance_count = 有効保証世代数

～<符号なし整数>((1～2))《1》

システム回復時のシンクポイントダンプファイルの入力障害などに備え，HiRDB 稼働中に保存しておくシステムログファイルの範囲を，シンクポイントダンプファイルの世代数によって指定します。この世代数のことを**有効保証世代数**といいます。ここで指定した世代数分のシンクポイントダンプファイルは，上書きできない状態になります。

《利点》

有効保証世代数を 2 にすると，最新世代のシンクポイントダンプファイルに障害が発生しても，1 世代前のシンクポイントからシステムを回復できるため，信頼性が向上します。

《指定値の目安》

- 信頼性を向上させる場合は有効保証世代数を 2 にしてください。ただし，有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます（二つになります）。
- シンクポイントダンプファイルを二重化して信頼性を向上している場合は，このオペランドを省略するか，又は 1 を指定することをお勧めします。

《注意事項》

- シンクポイントダンプファイルの必要最低数は**有効保証世代数＋ 1** となります。
- 有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます。上書きできない状態のシンクポイントダンプファイルに対応するシステムログファイルは上書きできない状態になります。したがって，有効保証世代数を 2 にすると，上書きできない状態のシステムログファイルが増えるため，スワップ先にできる状態のシステムログファイル数が不足することがあります。これを防ぐためにはシステムログファイルの容量を考慮してください。

57）　pd_spd_reduced_mode = 縮退運転オプション

～<符号なし整数>((0～2))《0》

シンクポイントダンプファイルの縮退運転をするかどうかを指定します。

縮退運転とは，HiRDB 稼働中又は再開始処理中のファイル障害などによって，シンクポイントダンプファイルの数が運用に必要なファイル数（**有効保証世代数**※＋ 1）以下になった場合でも，最低二つのファイルが使用できれば処理を続行する機能です。

注※　pd_spd_assurance_count オペランドの指定値です。

0：縮退運転を使用しません。

1：縮退運転を使用します。

2：縮退運転を使用し，縮退運転でシンクポイントダンプ取得の契機ごとに警告メッセージを出力します。

58）　pd_spd_reserved_file_auto_open = Y｜N

シンクポイントダンプファイルの自動オープンをするかどうかを指定します。

シンクポイントダンプファイルに障害が発生して，運用に必要なファイル数（**有効保証世代数**※＋1）を下回った場合，予約のファイルがあればHiRDBが予約のファイルをオープンして上書きできる状態にし，処理を続行します。これを**シンクポイントダンプファイルの自動オープン**といいます。

注※　pd_spd_assurance_countオペランドの指定値です。

Y：

シンクポイントダンプファイルの自動オープンをします。運用に必要なファイル数（有効保証世代数＋1）を下回った場合，予約ファイルを自動的にオープンします。

N：

シンクポイントダンプファイルの自動オープンをしません。運用に必要なファイル数（有効保証世代数＋1）を下回っても，予約ファイルを自動的にオープンしません。

《ほかのオペランドとの関連》

このオペランドの指定はpd_spd_reduced_modeオペランドより優先されます。

59）pd_spd_max_data_size＝シンクポイントダンプファイルのバッファ長

～<符号なし整数>((32000～4000000))《32768》（単位：バイト）

シンクポイントダンプをシンクポイントダンプファイルに入出力するときに使用するバッファ（共用メモリ）の大きさをバイト単位で指定します。

ここで指定した値によってシンクポイントダンプファイルの入出力回数を制御します。

《指定値の目安》

- 通常はこのオペランドを指定する必要はありません。
- 指定値を大きくするほど，シンクポイントダンプファイルに対する入出力回数が少なくなります。

60）pd_log_sdinterval＝システムログ出力量〔，経過時間〕

シンクポイントダンプの取得間隔を指定します。次に示す条件で指定できます。

- システムログの出力量
- 前回シンクポイントダンプを取得してからの経過時間

システムログ出力量：～<符号なし整数>((100～100000))《5000》（単位：ログブロック数）

シンクポイントダンプの取得間隔を指定します。ここで指定したログブロック数分のシステムログが出力されるたびに，シンクポイントダンプを取得します。

経過時間：～<符号なし整数>((0,10～1440))《60》（単位：分）

シンクポイントダンプの取得間隔を指定します。前回シンクポイントダンプを取得してから，ここで指定した経過時間が過ぎると，シンクポイントダンプを取得します。

- 経過時間に0を指定すると，HiRDBは経過時間でのシンクポイントダンプを取得しません。
- 前回シンクポイントダンプを取得してからトランザクションが一度も発生しない場合，ここで指定した経過時間が過ぎてもシンクポイントダンプを取得しません。

《指定値の目安》

- HiRDBを再開始するときの所要時間に特に規定のない場合は，このオペランドを指定する必要はありません。
- このオペランドの指定値によって，HiRDBを再開始するときの所要時間が決まります。

  このオペランドの指定値を小さくすれば，HiRDBを再開始するときのデータベース回復に掛かる所要時間が短くなります。ただし，シンクポイントダンプの取得回数が増えるため，オンライン中の性能が低下する場合があります。

逆に，このオペランドの指定値を大きくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が長くなります。ただし，シンクポイントダンプの取得回数が減るため，オンライン中の性能が向上する場合があります。

《指定値のチューニング方法》

シンクポイントダンプがどの程度の間隔で取得されているかは，統計解析ユティリティのシステムの稼働に関する統計情報の「シンクポイントダンプ取得間隔時間（SYNC POINT GET INTERVAL）」で確認できます。SYNC POINT GET INTERVAL の平均値を見て，シンクポイントダンプを取得する間隔が長いと判断したら，ここでの指定値を小さくしてください。また，反対に短いと判断したら，ここでの指定値を大きくしてください。

《注意事項》

シンクポイントダンプの取得間隔はシステムログの出力量で決まります。このため，更新系のトランザクションがほとんどない時間帯は，メモリからデータベースに反映するまでの時間が長くなります。このような状態で障害が発生すると，その間に発生したトランザクションの回復時間が長くなります。このようなケースが考えられる場合は，「経過時間」でもシンクポイントダンプの取得間隔を設定してください。

《ほかのオペランドとの関連》

pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合，システムログ出力量の省略値は 1000 になります。

4.2.14　サーバ用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

61）　pd_sts_initial_error = stop | continue | excontinue

サーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，又はバックエンドサーバ）の開始時，HiRDB はサーバ用ステータスファイルの現用ファイルを特定する処理を実行します。この特定処理で，次に示す異常を検知したときの HiRDB の処置を指定します。

- サーバ用ステータスファイルの実体がない
- サーバ用ステータスファイルの障害を検知した

現用ファイルの特定処理は pd_sts_file_name_1～7 オペランドに指定したサーバ用ステータスファイルを対象とします。

stop：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。この場合，異常を検知したステータスファイルの対策を実施した後に HiRDB を開始してください。

continue 又は excontinue：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知しても，現用ファイルが正常な場合はサーバの開始処理を続行します。ただし，pd_sts_singleoperation オペランドの値（ステータスファイルの片系運転をするかどうか）によっては開始処理を中止することがあります。pd_sts_singleoperation オペランドとの関係を次に示します。

●pd_sts_singleoperation オペランドとの関係

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
continue	サーバ用ステータスファイルの異常を検知した場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。
stop（省略値）	サーバ用ステータスファイルの異常を検知した場合，HiRDB が現用ファイルを特定してサーバの開始処理を続行します。ただし，A 系と B 系のファイルが次に示す表（**HiRDB が現用ファイルを特定できないケース**）の条件のどれかに当てはまる場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB が現用ファイルを特定できない場合は，HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。

●HiRDB が現用ファイルを特定できないケース

pd_sts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
continue	障害閉塞	障害閉塞
	障害閉塞	オープン（初期状態）
	障害閉塞	ファイル実体なし
	オープン（初期状態）	障害閉塞
	オープン（初期状態）	ファイル実体なし
	ファイル実体なし	オープン（初期状態）
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし
excontinue	障害閉塞	障害閉塞
	障害閉塞	ファイル実体なし
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし

したがって，HiRDB 管理者の操作をなるべく少なくしたい場合（pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定するケースを減らしたい場合）は，次に示すようにオペランドを指定してください。

- pd_sts_initial_error オペランドに excontinue を指定
- pd_sts_singleoperation オペランドに stop を指定

《指定値の目安》

各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
サーバ開始時の HiRDB の処理	サーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	一部のサーバ用ステータスファイルに異常があっても，現用ファイルが正常な場合はサーバの開始処理を続行します。
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB 開始時の障害対策を簡易化する場合は continue 又は excontinue を指定します。
メリット	サーバの開始時に該当するサーバの全サーバ用ステータスファイルの正常が保証されます。このため，HiRDB の開始後に現用ファイルの障害が発生した場合, 予備ファイルにスワップできます。	サーバの開始時に一部のサーバ用ステータスファイルの障害を検知しても，残りの正常なファイルだけで HiRDB を開始できます。このため，HiRDB の停止時間を短くできます。この場合，予備ファイルが少なくなっているため，障害状態のステータスファイルをすぐに修復する必要があります。
デメリット	サーバ用ステータスファイルの障害によって HiRDB の開始処理が中止される可能性が高くなります。	予備ファイルが少ない状態で HiRDB を稼働することがあるため，システムの信頼性が低下します。予備ファイルの数によってはサーバ用ステータスファイルをスワップできないことがあります。

《注意事項》

- 現用ファイルが両系とも異常の場合は，このオペランドの指定値に関係なくサーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。
- HiRDB を開始する前に現用ファイルを pdstsinit コマンドで初期化しないでください。初期化すると HiRDB を再開始できません。
- このオペランドに excontinue を指定し，すべての世代のステータスファイルが片系運転でかつオープン（初期状態）の場合，HiRDB は起動しません。

《備考》

オペランドの指定値と HiRDB の処理及び HiRDB 管理者の処置を次に示します。

●オペランドの指定値と HiRDB の処理

pd_sts_initial_errorオペランドの値	ステータスファイル障害	pd_sts_singleoperationオペランドの値	HiRDBが現用ファイルを特定できるか	pd_sts_last_active_fileの指定	pd_sts_last_active_fileオペランドの値と，オープンできたステータスのうち，最新のファイルが一致するか	現用ファイル障害	pd_sts_last_active_sideオペランドの指定	pd_sts_last_active_sideオペランドに指定されたファイルが使用できるか	表「HiRDBの処理及びHiRDB管理者の処置」の対応番号
stop（省略値）	なし	－	－	－	－	－	－	－	[1]
	あり	－	－	－	－	－	－	－	[5]
continue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
excontinue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]

（凡例）－：該当しません（条件がHiRDBの処理に影響を与えません）。

●HiRDB の処理及び HiRDB 管理者の処置

対応番号	HiRDB の処理	HiRDB 管理者の処置
[1]	HiRDB の開始処理を続行します。	なし
[2]	HiRDB が最新の現用ファイルを特定して，開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[3]	pd_sts_last_active_file オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[4]	pd_sts_last_active_file 及び pd_sts_last_active_side オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[5]	pd_sts_initial_error オペランドに stop が指定されているため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000010 が示す対策をしてください。

対応番号	HiRDB の処理	HiRDB 管理者の処置
[6]	前回稼働時の現用ファイルが特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000016 が示す対策をしてください。
[7]	HiRDB が特定した現用ファイルの正常な系と，pd_sts_last_active_side オペランドに指定された系が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000017 が示す対策をしてください。
[8]	HiRDB が特定した現用ファイル名と，pd_sts_last_active_file オペランドに指定したファイル名が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000015 が示す対策をしてください。
[9]	前回稼働時の現用ファイルの正常な系が特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000018 が示す対策をしてください。

62）pd_sts_singleoperation = stop | continue

サーバ用ステータスファイルの片系運転をするかどうかを指定します。予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，正常な系（片方の系）だけで処理を続行することを**ステータスファイルの片系運転**といいます。ステータスファイルの片系運転については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

なお，予備ファイルがある状況で現用ファイルの片系に障害が発生した場合は，このオペランドの指定値に関係なく，ステータスファイルをスワップして処理を続行します（片系運転にはなりません）。

stop：

片系運転をしません。片系運転になる場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。HiRDB が異常終了した場合，予備ファイルを準備した後に HiRDB を開始してください。

continue：

片系運転をします。片系運転になると KFPS01044-I メッセージが出力されます。片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了したりすると，HiRDB を再開始できなくなるため，片系運転になった場合はすぐに予備ファイルを準備してください。

《指定値の目安》

- stop を指定してシステムの信頼性を上げることを推奨します。また，現用ファイルの障害に対しては，予備ファイルの数を増やして備えることを推奨します。
- 各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB を停止させないことを重視する場合は continue を指定します。
メリット	予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，片系運転をしないで HiRDB	予備ファイルがない状況で現用ファイルの片系に障害が発生しても，処理を続行できます。この

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
	を異常終了します。このため，現用ファイルの内容が失われる可能性が低くなります。	ため，ステータスファイルの障害によってHiRDB が停止する可能性が低くなります。
デメリット	ステータスファイルの障害によって HiRDB が停止する可能性が高くなります。ただし，予備ファイルの数を増やすことで可能性を低くできます。	片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，現用ファイルの内容が失われるため，HiRDB を再開始できなくなります。

《ほかのオペランドとの関連》

pd_sts_singleoperation 及び pd_sts_initial_error オペランドの指定値の組み合わせによって，ステータスファイル障害時の HiRDB の処理が決定します。したがって，これら二つのオペランドの指定値は一緒に考えるようにしてください。

4.2.15　バックエンドサーバ接続保持機能に関するオペランド

バックエンドサーバ接続保持機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

63)　pd_bes_connection_hold = Y | N

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

バックエンドサーバ接続保持機能を使用するかどうかを指定します。

Y：バックエンドサーバ接続保持機能を使用します。

N：バックエンドサーバ接続保持機能を使用しません。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDBESCONHOLD オペランドを指定します。PDBESCONHOLD オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《注意事項》

バックエンドサーバ接続保持機能を使用する場合は，次に示す計算式を満たすようにしてください。

各バックエンドサーバプロセス数（pd_max_bes_process オペランドの値）

≧全フロントエンドサーバプロセス数 (pd_max_users オペランドの値×フロントエンドサーバ数)

この計算式を満たさない場合，バックエンドサーバプロセス数が不足して SQL エラーになることがあります。また，HiRDB の稼働中にユティリティなどを実行する場合は，バックエンドサーバプロセス数にユティリティの分の余裕値を確保してください。

64)　pd_bes_conn_hold_trn_interval = バックエンドサーバ接続保持時間

～<符号なし整数>((0～3600))《1》（単位：秒）

バックエンドサーバ接続保持時間を秒単位で指定します。

バックエンドサーバ接続保持機能を使用すると，HiRDB はトランザクションの終了後から次のトランザクションが実行されるまでの時間を監視します。次のトランザクションが実行されるまでの時間が，指定値の範囲内の場合はフロントエンドサーバとバックエンドサーバの接続を保持します。指定値を超えた場合は，トランザクション終了後にフロントエンドサーバとバックエンドサーバの接続を切断します。

なお，このオペランドに 0 を指定した場合は時間を監視しません。SQL DISCONNECT（XA ライブラリを使用時は xa_close），PDCWAITTIME オーバなどでフロントエンドサーバとクライアント間の接続が切断されたときだけ，フロントエンドサーバとバックエンドサーバの接続を切断します。

5 シングルサーバ定義

この章では，シングルサーバ定義の各オペランドの内容について説明します。

5.1　オペランドの形式

シングルサーバ定義ではシングルサーバの実行環境を定義します。ここでは，シングルサーバ定義の各オペランドの形式について説明します。なお，表中の「番号」は，「5.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

番号	形式
1	〔set pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕〕※
2	〔set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔〕※
3	〔set pd_max_ard_process = 非同期 READ プロセス数〕※
4	〔set pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数〕

(2)　作業表

番号	形式
5	〔set pd_work_buff_mode = each｜pool〕※
6	〔set pd_work_buff_size = 作業表用バッファ長〕※
7	〔set pd_work_buff_expand_limit = 作業表用バッファの増分上限値〕※

(3)　システム監視

番号	形式
8	〔set pd_watch_pc_client_time = クライアントからの要求間隔監視時間〕※
9	〔set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〕※
10	〔set pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値〕※

(4)　SQL 実行時間警告出力機能

番号	形式
11	〔set pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜SQL 実行時間警告情報の出力条件（時間指定）〕※

(5)　拡張 SQL エラー情報出力機能

番号	形式
12	[set pd_uap_exerror_log_use = YES | NO] ※

(6)　排他制御

番号	形式
13	[set pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ] ※
14	[set pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数] ※
15	[set pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和] ※
16	[set pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数] ※
17	[set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数] ※
18	[set pd_dbsync_lck_release_count = シンクポイント時のグローバルバッファ排他解除間隔] ※

(7)　バッファ

番号	形式
19	[set pd_sql_object_cache_size = SQL オブジェクト用バッファ長] ※
20	[set pd_table_def_cache_size = 表定義情報用バッファ長] ※
21	[set pd_auth_cache_size = ユーザ権限情報用バッファ長] ※
22	[set pd_view_def_cache_size = ビュー解析情報用バッファ長] ※
23	[set pd_type_def_cache_size = ユーザ定義型情報用バッファ長] ※
24	[set pd_routine_def_cache_size = ルーチン定義情報用バッファ長] ※
25	[set pd_registry_cache_size = レジストリ情報用バッファ長] ※

(8)　共用メモリ

番号	形式
26	[set pd_sds_shmpool_size = シングルサーバ用共用メモリサイズ] ※

(9)　RPC トレース情報

番号	形式
27	[set pd_rpc_trace = Y | N] ※
28	[set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"] ※

番号	形式
29	〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕※

(10)　トラブルシュート情報

番号	形式
30	〔set pd_module_trace_max = モジュールトレース格納最大数〕※
31	〔set pd_module_trace_timer_level = モジュールトレース出力時刻取得方法〕※
32	〔set pd_pth_trace_max = 通信トレース格納最大数〕

(11)　グローバルバッファ

番号	形式
33	〔set pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値〕※
34	〔set pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値〕※

(12)　セキュリティ

番号	形式
35	〔set pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長〕※

(13)　プラグインインデクスの遅延一括作成

番号	形式
36	〔set pd_plugin_ixmk_dir = "インデクス情報ファイルを作成するディレクトリ名"又は "インデクス情報ファイルを作成する HiRDB ファイルシステム領域名"〕

(14)　Java

番号	形式
37	〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※
38	〔set pd_java_castoff = Y ｜ N〕

(15)　システムログファイル

番号	形式
39	〔set pd_log_dual = Y ｜ N〕※
40	〔set pd_log_remain_space_check = warn ｜ safe〕※
41	〔set pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔, "アンロードログファイルの出力先ディレクトリ"〕…〕

番号	形式
42	〔set pd_log_singleoperation = Y｜N〕※
43	〔set pd_log_rerun_reserved_file_open = Y｜N〕※
44	〔set pd_log_rerun_swap = Y｜N〕※
45	〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕※
46	〔set pd_log_unload_check = Y｜N〕※
47	〔set pd_log_max_data_size = ログ入出力バッファ長〕※
48	〔set pd_log_write_buff_count = ログ出力バッファ面数〕※
49	〔set pd_log_rec_leng = システムログファイルのレコード長〕※
50	〔set pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数〕※
51	〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕※

(16)　シンクポイントダンプファイル

番号	形式
52	〔set pd_spd_dual = Y｜N〕※
53	〔set pd_spd_assurance_msg = Y｜N〕※
54	〔set pd_spd_assurance_count = 有効保証世代数〕※
55	〔set pd_spd_reduced_mode = 縮退運転オプション〕※
56	〔set pd_spd_reserved_file_auto_open = Y｜N〕※
57	〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕※
58	〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕※

(17)　サーバ用ステータスファイル

番号	形式
59	set pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"
	〔set pd_sts_file_name_2 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_3 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_4 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_5 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_6 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

(18)　サーバ用ステータスファイル（障害発生時）

番号	形式
60	〔set pd_sts_initial_error = stop｜continue｜excontinue〕※
61	〔set pd_sts_singleoperation = stop｜continue〕※
62	〔set pd_sts_last_active_file = "論理ファイル名"〕
63	〔set pd_sts_last_active_side = A｜B〕

(19)　作業表用ファイル

番号	形式
64	pdwork -v "HiRDB ファイルシステム領域名称"〔，"HiRDB ファイルシステム領域名称"〕…

(20)　システムログファイルの構成

番号	形式
65	{{pdlogadfg -d sys -g ファイルグループ名〔ONL〕}}
66	{{pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名" 〔-b "システムログファイル名"〕}}

(21)　シンクポイントダンプファイルの構成

番号	形式
67	{{pdlogadfg -d spd -g ファイルグループ名〔ONL〕}}
68	{{pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕}}

(22)　プラグイン

番号	形式
69	{{〔 pdplgprm -n プラグイン名称 〔 -s 共用メモリサイズ〕〕}}

注※

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

ただし，次に示すオペランドについては，サーバ共通定義ではなく，システム共通定義の同じオペランドの指定値が有効となります。

- pd_cwaittime_wrn_pnt
- pd_uap_exerror_log_use

5.2　オペランドの説明

5.2.1　プロセスに関するオペランド

1）　pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕

～<符号なし整数>((0～3000))

常駐プロセス数：

シングルサーバの常駐プロセス数を指定します。常駐プロセスとは，サーバ開始時からあらかじめ起動しておくプロセスのことです。

《利点》

シングルサーバで並行処理できるトランザクションが使用するプロセスを，あらかじめシステム開始時に起動して常駐しておくことで，新たにトランザクションが入力されてもプロセス起動時間を削減できます。ただし，HiRDB の開始に時間が掛かります。

《指定値の目安》

- シングルサーバのサーバプロセスのプロセス固有領域とプロセサの実メモリ量から値を求めてください。サーバプロセスのプロセス固有領域については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- マルチフロントエンドサーバ構成で，更に pd_max_bes_process 又は pd_max_dic_process オペランドを指定している場合は，次に示す条件を満たすようにこのオペランドを指定してください。

  pd_process_count の値≦pd_max_bes_process 又は pd_max_dic_process の値
- このオペランドには，シングルサーバの**最大起動プロセス数**（pd_max_users の値）の範囲内で値を指定します。

《指定値のチューニング方法》

常駐プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- 常駐プロセス数はメモリと密接に関連しているため，必要以上に大きい値を指定すると，HiRDB を開始できなくなることがあります。
- 常駐プロセス数を超えたプロセスが必要になると，最大起動プロセス数までプロセスを動的に起動します。ただし，pd_max_server_process オペランドの指定値によっては，最大起動プロセス数分のプロセスを起動できないことがあります。

《オペランドの省略値》

このオペランドを省略（又は 0 を指定）すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，最大起動プロセス数が仮定されます。

サーバ開始時の常駐プロセス数：

HiRDB の開始処理時に常駐化する常駐プロセスの数を指定します。

通常，常駐プロセスの起動処理は HiRDB の開始処理中に行われます。このため，常駐プロセス数が多くなると，それに比例して HiRDB の開始処理時間が長くなります。目安として，100MIPS 程度のサーバマシンでプロセスを一つ起動するのに約 1 秒掛かります。

サーバ開始時の常駐プロセス数の指定有無による処理の違いを次に示します。

●サーバ開始時の常駐プロセス数を指定しない場合（pd_process_count = 500 と指定した場合）

すべての常駐プロセスの起動処理を HiRDB の開始処理中に実行します。常駐プロセス(この場合は 500 個）がすべて起動されないと，HiRDB を開始できません。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 500 秒掛かります。

●サーバ開始時の常駐プロセス数を指定する場合(pd_process_count = 500,50 と指定した場合)

一部の常駐プロセスの起動処理を HiRDB の開始処理中に実行し，残りの常駐プロセスの起動処理を HiRDB の開始処理終了後に実行します。一部の常駐プロセス（この場合は 50 個）が起動されれば，HiRDB を開始できます。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 50 秒掛かります。残りの常駐プロセス（この場合は 450 個）は，HiRDB の開始処理終了後に起動されます。

《利点》

HiRDB の開始処理時間を短縮できます。系切り替え機能を使用している場合など，HiRDB の開始処理時間をなるべく短くしたいときに適用します。

《指定値の目安》

HiRDB の開始処理終了直後に必要なプロセス数を指定します。

《注意事項》

サーバ開始時の常駐プロセス数を指定する場合は，クライアント環境定義の PDCWAITTIME オペランドの値を見直してください。

HiRDB の開始直後に，サーバ開始時の常駐プロセス数の値を超える UAP が実行されると，残りの常駐プロセスを起動した後にトランザクション処理が実行されます。したがって，クライアント環境定義の PDCWAITTIME オペランドの値が小さいと，一部の UAP がタイムアウトで処理できない場合があります。

PDCWAITTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

2) pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔
～<符号なし整数>((0～1440))（単位：分）

HiRDB の非常駐サーバプロセスの停止処理をする間隔を分単位で指定します。この機能は，稼働中のサーバプロセス数が常駐プロセス数（pd_process_count オペランドの値）より多くなった場合に適用されます。この機能によって停止するサーバプロセス数は HiRDB が自動的に算出して決定します。

《利点》

業務（稼働中のサーバプロセス数）がピークのときに再利用できる非常駐サーバプロセスが増えるため，メモリなどのプロセス資源の利用効率が向上します。

《指定値の目安》

- 例えば，一日のうち 1 時間だけ業務のピーク時間があり，その時間内でピークとなる間隔が 2 分ぐらいの場合，このオペランドに 2 を指定します。
- ピーク時に同時実行されるサーバプロセス数が常駐プロセス数以下の場合，この機能を使用しても効果は期待できないため，このオペランドを省略します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《指定値のチューニング方法》

システムの稼働に関する統計情報をサーバ単位に 1 週間取得してください。「サービス実行中のサーバプロセス数（# OF PROCESS ON SERVICE)」の値から業務のピークを求めてください。そのピークが現在設定されている常駐プロセス数（pd_process_count オペランドの値）を超えている場合は，ピークごとの間隔を求めてその時間を設定します。

ただし，サーバマシンのメモリ，CPU などの資源に余裕がある場合は，不足している分のプロセス数を常駐プロセス数に加算する方（pd_process_count オペランドの値を大きくする方）がこのオペランドを指定するより性能が向上します。

《注意事項》

このオペランドを省略するか，又は 0 を指定した場合は 10 秒間隔でサービス待ち状態の非常駐サーバプロセスを調査し，サービス待ち状態の非常駐サーバプロセスがあれば，その非常駐サーバプロセスを停止します。

3）　pd_max_ard_process = 非同期 READ プロセス数

～<符号なし整数>((0～256))

非同期 READ 機能を使用する場合にこのオペランドを指定します。このオペランドには非同期 READ 処理に必要なプロセス数を指定します。非同期 READ 機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《前提条件》

pdbuffer オペランドの-m オプションで 1 以上を指定している必要があります。

《指定値の目安》

- 0 又は 1 を指定してください。ただし，pdbuffer オペランドの-m オプションの値が 2～256 の場合は，-m オプションと同じ値を指定してください。-m オプションの値が 257 以上の場合は，RD エリアやシステムファイルを格納するディスク装置の数と同じ値を指定するか，又は 256 を指定してください。
- このオペランドの値を大きくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に処理時間を短縮できます。このオペランドの値を小さくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に，非同期 READ プロセスの処理完了待ちが発生する分，処理時間が余計に掛かることがあります。
- このオペランドの値×サーバ数分のプロセスが起動されるため，リソース（共用メモリ及びメッセージキュー）を考慮してこのオペランドの値を決定してください。共用メモリ及びメッセージキューの見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値のチューニング方法》

指定値（非同期 READ プロセス数）のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《オペランドの規則》

このオペランドに 0 を指定すると，非同期 READ 機能を使用しません。

《ほかのオペランドとの関連》

このオペランドの値を変更した場合は pd_max_server_process オペランドの値を見直してください。

4）　pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数

～<符号なし整数>((2～255))

すべてのバッファプールに対して，**デファードライト処理の並列 WRITE 機能**を使用する場合にこのオペランドを指定します。このオペランドには並列処理するプロセス数を指定します。プロセス数を増

加させることで書き込み処理時間を短縮できます。デファードライト処理の並列 WRITE 機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値の目安》

デファードライト処理の並列 WRITE 機能が有効になる最小値の 2 を指定してください。また，マニュアル「HiRDB Version 8 システム運用ガイド」の「デファードライト処理のチューニング」を参照してオペランドの値を決定してください。

《注意事項》

デファードライト処理の並列 WRITE 機能を指定すると，プロセス数が増加するため，CPU 利用率が上がります。

5.2.2　作業表に関するオペランド

5）pd_work_buff_mode = each | pool

HiRDB が作業表を作成するときのバッファの確保方式を指定します。

each：個々の作業表ごとにバッファを確保します。

pool：サーバプロセス単位にバッファプールとして一括して確保します。

《指定値の目安》

- 通常は pool を指定してください。特に大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合には pool を指定してください。
- あらかじめ，作業表用バッファに使用できるプロセス固有領域サイズが決まっている場合，pool を指定してください。pool を指定すると，HiRDB が効率良く作業表用バッファを各作業表に配分します。

  この場合，pd_work_buff_size に指定した値でプロセス固有領域を占有し，そのプール内で作業表の入出力をバッファリングします。したがって，pd_work_buff_size に指定した値以上にプロセス固有メモリを占有することはありません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，pool が仮定されます（pd_sysdef_default_option オペランドに v6compatible を指定している場合は each が仮定されます）。

6）pd_work_buff_size = 作業表用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((128～1000000))
- 64 ビットモードの場合：((128～4000000000))

HiRDB が作成する作業表のバッファの大きさをキロバイト単位で指定します。

項目	pd_work_buff_mode = each の場合	pd_work_buff_mode = pool の場合
利点	作業表バッファ長が大きいと，データ操作時に発生する作業表用ファイルへの入出力回数を削減できるため，作業表を使用する SQL の実行時間を削減できます。ただし，各サーバのプロセス固有メモリを使用するので，システム全体のメモリ量（実メモリ，仮想メモリ）を考慮して指定してください。 特に pd_work_buff_mode=each の場合，メモリは「pd_work_buff_size の値×必要作業表数」分確保されます。したがって，必要以上に大きな値を指定すると，ほかのプロセスで仮想メモリ不足が生じることがあります。	
適用基準	大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合に指定します。このとき，pd_work_buff_mode = pool を指定してください。	

項目	pd_work_buff_mode = each の場合	pd_work_buff_mode = pool の場合
指定値の目安	• 個々の作業表ごとに確保するバッファの大きさを指定します。 • 作業表用バッファ長に，作業表メモリ容量より大きな値を指定すると，作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。作業表メモリ容量の見積もり式は次のとおりです。 **作業表メモリ容量＝対象となる作業表容量※÷2**	• サーバプロセス単位に一括して確保する作業表用バッファプールの大きさを指定します。 • 大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合は，4352～5120 を指定してください。指定すると，ソートでの入出力単位が大きくなるため，ソート処理時間が短くなります。 • 作業表用バッファ長に，SQL 文単位の作業表メモリ総容量を上限として，大きな値を指定するほど作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。SQL 文単位の作業表メモリ総容量の見積もり式は次のとおりです。 **SQL 文単位の作業表メモリ総容量＝ a×b ＋ c×d**
注意事項	複数のユーザが同時に実行する場合，及び複数個の作業表を使用する SQL 文などを実行する場合には，指定したサイズのバッファが，個々の作業表ごとに確保されます。そのため，大きな値を指定するとメモリを圧迫する原因となります。	SQL 文単位に使用する作業表の数に比べて作業表用バッファ長に指定した値が小さいと，each を指定した場合に比べて処理時間が増加することがあります。具体的には，「SQL 文単位の作業表最大数×128」以上の値を指定してください。SQL 文単位の作業表最大数の見積もり式は次のとおりです。 **SQL 文単位の作業表最大数＝ b ＋ d**
オペランドの規則	指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。	• 指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。 • 384 以上の値を指定してください。384 未満の値が指定されると，自動的に 384 に切り上げられます。
省略値	このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，128 が仮定されます。	このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。 • 32 ビットモードの場合：384 • 64 ビットモードの場合：5120

a：↑｛作業表（列情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

b：作業表（列情報格納用）の最大数※

c：↑｛作業表（位置情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

d：作業表（位置情報格納用）の最大数※

注※

これらの項目の見積もり方法については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

7）　pd_work_buff_expand_limit = 作業表用バッファの増分上限値

～＜符号なし整数＞（単位：キロバイト）

- 32 ビットモードの場合：((128～1000000))
- 64 ビットモードの場合：((128～4000000000))

HiRDB が作成する作業表用バッファ長は pd_work_buff_size オペランドで指定します。この作業表用バッファの不足時，作業表用バッファを自動増分する場合にこのオペランドを指定します。このオペランドで指定した値になるまで作業表用バッファを増分します。

例えば，オペランドに次の指定をした場合，通常 1,024 キロバイトの作業表用バッファを確保します。作業表用バッファが 1,024 キロバイトで不足すると，2,048 キロバイトまで作業表用バッファを増分します。

- pd_work_buff_size = 1024
- pd_work_buff_expand_limit = 2048

次に示すときに HiRDB は作業表用バッファを増分します。

- 結合方式がハッシュジョイン，又は副問合せの実行方式にハッシュ実行が適用されている場合に，必要な作業表用バッファが確保できないとき
- 複数の作業表を同時に使用する場合，各作業表に作業表用バッファを 128 キロバイト割り当てて作業表用バッファが不足するとき

《前提条件》

pd_work_buff_mode オペランドを省略するか，又は pool を指定している必要があります。

《利点》

作業表用バッファ不足（pd_work_buff_size オペランドの指定値不足）で UAP がエラーになるのを防げます。

《注意事項》

次に示すどちらかの条件を満たす場合は作業表用バッファを増分しません。

- pd_work_buff_expand_limit オペランドを指定しない
- pd_work_buff_expand_limit オペランドの値≦pd_work_buff_size オペランドの値

《オペランドの規則》

指定値は 128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。

《ほかのオペランドとの関連》

シングルサーバプロセスでの作業表用バッファの初回増分時に KFPH29008-I メッセージを出力します。なお，pd_work_table_option オペランドで KFPH29008-I メッセージの出力を抑止できます。

《留意事項》

作業表用バッファを増分した場合，該当するサーバプロセスで使用中の作業表数が 0 になると，増分した作業表用バッファを解放します。作業表の使用数が 0 になるのは次に示すときです。

- 使用中のカーソルをすべてクローズしたとき（この場合，作業表の使用数が 0 にならないことがあります）
- ホールダブルカーソルを使用していない場合は，トランザクションを正常終了するか，又は取り消したとき
- ホールダブルカーソルを使用している場合は，UAP を HiRDB から切り離したとき（DISCONNECT したとき）

《備考》

「ハッシュジョイン，副問合せのハッシュ実行」は，次に示す場合に適用されます。

- pd_additional_optimize_level オペランド，クライアント環境定義の PDADDITIONALOPTLVL オペランド，又は SQL コンパイルオプションの ADD OPTIMIZE LEVEL オペランドで，「コストベース最適化モード 2 の適用」と一緒に「ハッシュジョイン，副問合せのハッシュ実行」を指定する
- SQL 文中の結合方式の SQL 最適化指定に HASH を指定する
- SQL 文中の副問合せ実行方式の SQL 最適化指定に HASH を指定する

5.2.3　システム監視に関するオペランド

8）pd_watch_pc_client_time = クライアントからの要求間隔監視時間

～<符号なし整数>((0～65535))（単位：秒）

Windows 版の HiRDB クライアントからの要求に対する応答を HiRDB サーバが返してから，次に HiRDB クライアントからの要求があるまでの最大待ち時間（サーバ側の最大待ち時間）を秒単位で指定します。

指定した時間内に HiRDB クライアントから全く要求がない場合には，クライアントに異常が発生したものとして，強制的にサーバとクライアントとの接続を切り離します。この時に HiRDB クライアントに対しては，接続の切り離しについての通知はしません。

時間監視の対象となるのは，CONNECT から DISCONNECT までの間で，非トランザクション状態（SQL の実行を開始してから COMMIT 又は ROLLBACK までの間以外）の間です。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，3600 が仮定されます。

《注意事項》

- 0 を指定した場合には，HiRDB クライアントからの次の要求を無限に待ち続けます。
- このオペランドの指定値を小さい値（1～600 程度）にした場合には，HiRDB クライアントが，SQL 実行時にサーバダウンを検知するなど，不当に異常終了することがあります。
- UNIX 版の HiRDB クライアント（Linux 版の HiRDB クライアントも含む）の場合は，このオペランドの指定に関係なく時間を監視しません。UNIX 版の HiRDB クライアントに対して時間監視をする場合は，HiRDB クライアントのクライアント環境定義 PDSWATCHTIME を指定してください。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSWATCHTIME オペランドを指定してください。

PDSWATCHTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

9）pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0，2～100000))

UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。

この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。

このオペランドでは，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。シンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，対象トランザクションを強制的に中断してロールバックをします。これを**シンクポイントダンプ有効化のスキップ回数監視機能**といいます。

シンクポイントダンプ有効化のスキップ回数監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

このオペランドを指定すると，UAP の無限ループなどを防止できます。

《指定値の目安》

通常は 0 を指定してください。0 を指定すると，スキップ回数の上限値を HiRDB が計算します。0 を指定して不都合が発生した場合，又は KFPS02101-I メッセージが出力された場合に，このオペランドの値を変更してください。そのときの指定値の目安については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，シンクポイントダンプ有効化のスキップ回数監視機能を使用しません。

《注意事項》

- このオペランドを指定すると，ログレスモードで実行している UAP も監視対象になります。ログレスモードで実行している UAP の処理が中断すると，データベースを自動回復できないため RD エリアを障害閉塞します。このため，上限値の設定に際しては，ログレスモードで実行する UAP のトランザクション処理中に，該当するサーバ内のほかのトランザクションから出力されるシステムログ量も考慮に入れてください。
- pdload，pdmod，pdrorg，pdexp，pddbst，pdgetcst，pdrbal，pdvrup，及び pdmemdb コマンドはこの機能の監視対象外になります。

10）　pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0～100000))

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになると，シンクポイントダンプの有効化処理ができずにスキップされます。これは，デファードライト処理によってシンクポイントダンプの取得が遅れることで，シンクポイントで出力される更新バッファ数がシンクポイントダンプの取得間隔内で出力できる更新バッファ数を超えるためです。

シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

このオペランドでは，デファードライト処理によってシンクポイントダンプの取得が遅れた場合に，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。

デファードライト処理によってシンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，シンクポイントダンプの取得間隔内でシンクポイントダンプの取得が完了するように，HiRDB が更新バッファ数の上限値を決定します。そして，更新バッファ数が上限値を超えた時点で最も古い更新バッファを出力して，シンクポイント時の更新バッファ総量を制限します。これを**更新バッファ量抑制機能**といいます。

《利点》

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生する，ユニット異常終了を回避できます。

《指定値の目安》

通常はこのオペランドを省略してください。更新処理のスループットを低下させてでも，シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生するユニット異常終了を回避したい場合に，1 を指定します。

許容できるシンクポイントダンプの有効化処理のスキップ回数がログ容量などから分かる場合は，その回数を指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《オペランドの規則》

このオペランドに 0 を指定すると，更新バッファ量抑制機能を使用しません。

《注意事項》

更新バッファ量抑制機能の有効期間について，注意事項を次に示します。なお，更新バッファ量抑制機能の有効期間とは，KFPH23035-I メッセージが出力されてから KFPH23036-I メッセージが出力されるまでの間です。

- 更新バッファ数が，HiRDB が決定した更新バッファの上限値を超えている場合，更新処理の実行後に更新バッファを出力するため，更新処理のスループットが低下します。HiRDB が決定する更新バッファの上限値は，次の計算式で求められます。

（シンクポイントインターバル時間÷WRITE 単価※）
×（1－（前回のシンクポイントからプレシンクまでのログ出力量÷シンクポイント間のログ出力量）
×（バッファプールのバッファ面数÷シンクポイントで更新があったバッファプールの総バッファ面数）

注※　WRITE 単価についての詳細は，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

- pd_dbbuff_rate_updpage オペランド，又は pdbuffer -y オペランドによってデファードライトトリガのトリガ契機を指定している場合，それぞれの指定値が，HiRDB が決定した更新バッファの上限値より大きくなると，HiRDB が決定した更新バッファの上限値をデファードライトトリガのトリガ契機となる更新バッファ数に変更します。

  また，pdbuffer -w オペランドの値は，各バッファの更新バッファ上限値まで出力されるように自動調整されます。

- シンクポイントダンプの有効化処理がスキップした場合，シンクポイント時の更新バッファ出力処理中に検知されます。そのため，更新バッファ量抑制機能が有効となる時期は，シンクポイントダンプの有効化処理がスキップされ，エラーメッセージが出力されるときよりも後になる場合があります。

- 通常，並列 write 機能使用時のシンクポイント処理では各デファードライト処理用並列 WRITE プロセスに対する出力要求はシンクポイントごとに 1 回ですが，更新バッファ量抑制機能使用時はシンクポイントスキップの検出を早めるため複数回に分割して行われます。

5.2.4　SQL 実行時間警告出力機能に関するオペランド

11）　pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜SQL 実行時間警告情報の出力条件（時間指定）

SQL 実行時間警告出力機能を使用する場合にこのオペランドを指定します。このオペランドの指定方法は次に示す二つの方法があります。

- 比率で指定する方法
- 時間で指定する方法

SQL 実行時間警告出力機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

SQL 実行時間警告情報の出力条件（比率指定）：～<符号なし整数>((0～99))又は<符号なし 10 進数>((0～99.999999))（単位：%）

SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）に対する比率で指定します。SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間がこのオペランドの指定値によって設定される時間以上になった場合，次に示す警告情報を出力します。これを SQL 実行時間警告出力機能といいます。

- SQL 実行時間警告情報ファイル
- 警告メッセージ（KFPA20009-W）

《オペランドの指定方法》

PDCWAITTIME オペランドの値に対する比率（%単位）で指定します。例えば，PDCWAITTIME オペランドに 100（秒）を指定し，このオペランドに 90（%）を指定すると，SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間が 90 秒以上 100 秒未満の場合に警告情報が出力されます。

（例）

PDCWAITTIME = 100

pd_cwaittime_wrn_pnt = 90

SQL 実行時間警告情報の出力条件（時間指定）：～<符号なし 10 進数>((0～65534.999999))sec（単位：秒）

SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を時間で指定します。SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間がこのオペランドの指定値によって設定される時間以上になった場合，次に示す警告情報を出力します。これを SQL 実行時間警告出力機能といいます。

- SQL 実行時間警告情報ファイル
- 警告メッセージ（KFPA20009-W）

《オペランドの指定方法》

出力契機とする時間を秒単位で指定します（小数点以下 6 けたまで指定できます）。指定値に「sec」を付けてください。

（例）

pd_cwaittime_wrn_pnt = 0.001sec

これ以降の説明は，比率指定及び時間指定の両方共通の説明です。

《オペランドの規則》

このオペランドに 0 又は 0sec を指定すると，警告情報を出力しません（SQL 実行時間警告出力機能を使用しません）。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- SQL 実行時間警告情報の出力条件（比率指定）：0
- SQL 実行時間警告情報の出力条件（時間指定）：0sec

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDCWAITTIMEWRNPNT オペランドを指定してください。
PDCWAITTIMEWRNPNT オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_cwaittime_report_dir
- pd_cwaittime_report_size

5.2.5　拡張 SQL エラー情報出力機能に関するオペランド

12）　pd_uap_exerror_log_use = YES | NO

拡張 SQL エラー情報出力機能を使用するかどうかを指定します。拡張 SQL エラー情報出力機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

YES：

拡張 SQL エラー情報出力機能を使用します。エラーログファイル及び SQL エラーレポートファイルに SQL エラー情報を出力します。

NO：

拡張 SQL エラー情報出力機能を使用しません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，NO が仮定されます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDUAPEXERLOGUSE オペランドを指定してください。このオペランドと，クライアント環境定義 PDUAPEXERLOGUSE の両方を指定した場合は，クライアント環境定義 PDUAPEXERLOGUSE が優先されます。

PDUAPEXERLOGUSE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

5.2.6　排他制御に関するオペランド

13）　pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～2000000))

- 64 ビットモードの場合：((1～2000000000))

シングルサーバの排他制御で使用する共用メモリ領域（排他制御用プール）の大きさをキロバイト単位で指定します。

このオペランドの指定対象の領域は，排他資源管理テーブルとして使用されます。排他資源管理テーブル数，排他要求数，及び排他制御用プールサイズの間には，次の関係があります。

「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数

係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。

《指定値の目安》

- 32 ビットモードの場合，6 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 64 ビットモードの場合，4 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 次に示す計算式でこのオペランドの指定値を見積もってください。

HiRDB の種類	計算式（単位：キロバイト）
32 ビットモードの HiRDB/シングルサーバの場合	↑↑a ÷pd_lck_pool_partition の値↑÷ 6 ↑×pd_lck_pool_partition の値
64 ビットモードの HiRDB/シングルサーバの場合	↑↑a ÷pd_lck_pool_partition の値↑÷ 4 ↑×pd_lck_pool_partition の値

a：シングルサーバで同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については，「付録 D　排他資源数の見積もり」を参照してください。

注

特に，定義系 SQL の DROP TABLE 又は DROP SCHEMA を実行する場合は，このオペランドの指定値を正確に見積もって指定してください。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）の「排他資源管理テーブル使用率（%OF USE LOCK TABLE)」を参照してください。使用率の最大値が 80%以上の場合は，これからのデータベース拡張に備えて，このオペランドの値を大きくすることをお勧めします。使用率の最大値が 10%以下の場合は，共用メモリを節約するために，このオペランドの値を小さくすることをお勧めします。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- 32 ビットモードの場合：16000
- 64 ビットモードの場合：32000

ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます。

《注意事項》

- このオペランドの指定値が小さ過ぎると，SQL がエラーリターンすることがあります。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは，pd_lck_pool_partition オペランドと関連があります。

14）　pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数

～<符号なし整数>((1～5000))

排他制御処理の分散をするときにシングルサーバの排他制御で使用する排他制御用プールパーティション数を指定します。

排他制御処理の分散については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値のチューニング方法》

排他制御用プールパーティション数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

《注意事項》

- このオペランドの指定値を大きくし過ぎると，各排他制御用プールパーティションに割り当てられる共用メモリのサイズが小さくなるため，排他制御用プールパーティションが容量不足となり SQL がエラーリターンすることがあります。また，排他制御の処理に時間が掛かるため，システムのパフォーマンスが下がります。この場合，このオペランドの指定値を小さくしてください。
- 排他制御用プールサイズは 1 キロバイト以上必要であるため，pd_lck_pool_size の値よりも大きな値を指定した場合，このオペランドには pd_lck_pool_size の値が仮定されます。また，このとき，KFPS00421-W メッセージが出力されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_pool_size
- pd_lck_deadlock_check_interval

15）　pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和

～<符号なし整数>((0～140000))

トランザクションを越えて保持する表，及び RD エリアの排他資源数を指定します。このオペランドの値に従って，表，RD エリアの UNTIL DISCONNECT 指定の排他を管理するブロックを共用メモリに確保します。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。次に示す場合に指定値を変更するかどうかを検討してください。

- 同時に実行するユティリティの数を大量に増やす場合
- ホールダブルカーソルを使用する場合
- pdlbuffer オペランドで指定したローカルバッファを使用する場合

このオペランドの指定値の見積もり方法については，「付録 C.5　1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式」を参照してください。

《指定値のチューニング方法》

このオペランドの指定値が小さいと，トランザクションがロールバックしたり，リターンコード 8 でユティリティが異常終了したりします。このとき, KFPA11914-E, 又は KFPH28001-E メッセージが出力されます。この現象が発生したら，このオペランドの値を大きくしてください。

なお，このオペランドの値を大きくすると，それに比例して必要なメモリも増加します。必要なメモリは「このオペランドの値×48（64 ビットモードの場合は 64)」バイトです。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，256 が仮定されます。

16）　pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数

～<符号なし整数>((16～1024))

UNTIL DISCONNECT 指定の LOCK 文を実行しない表に対してホールダブルカーソルを使用する場合に，そのホールダブルカーソルのトランザクション当たりの最大同時オープン数を指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，16 が仮定されます。

《注意事項》

このオペランドに省略時解釈値以外の値を指定すると，共用メモリ使用量が増加します。

《ほかのオペランドとの関連》

このオペランドと次に示すオペランドの指定値は，ロックサーバの共用メモリサイズ計算に使用されます。32 ビットモードの HiRDB の場合，各オペランドの指定値が大き過ぎると，ロックサーバの共用メモリサイズが 2 ギガバイトを超えるため，HiRDB を開始できなくなることがあります。ロックサーバの共用メモリサイズが 2 ギガバイトを超えないように，このオペランドと次に示すオペランドの指定値を調整してください。

- pd_max_access_tables
- pd_max_users
- pd_lck_hash_entry
- pd_lck_pool_size

共用メモリについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

17）　pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数

～<符号なし整数>((0～2147483647))

排他制御用プールで使用するハッシュテーブルのエントリ数を指定します。HiRDB はここで指定された値に従ってユニットコントローラ用の共用メモリに排他制御用プールを確保します。

《指定値の目安》

通常はこのオペランドを省略してください。

次に示す条件に該当する場合は指定値を検討してください。

- バージョン 06-02 以降にバージョンアップする場合に共用メモリサイズをできるだけ変更したくないときは 11261 を指定してください。この場合，バージョンアップ前と同数のハッシュエントリ数を確保するため，排他制御用プール中のハッシュテーブルサイズがバージョンアップ前と同じになります。

- このオペランドに推奨値より大きい値を指定すると，性能が向上することがあります。ただし，《推奨値の求め方》の変数 a より大きい値を指定しても，a を指定したときよりも性能が向上することはありません。

《オペランドの規則》

- このオペランド及びサーバ共通定義の pd_lck_hash_entry オペランドを省略するか，又はこのオペランドに 0 を指定すると，HiRDB はサーバごとに推奨値を計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 11261 が仮定されます）。推奨値については《推奨値の求め方》を参照してください。
- このオペランドに 0 でも素数でもない値を指定した場合，HiRDB はその指定値を超えない最大の素数が指定されたと仮定します。

《注意事項》

このオペランドの指定値が小さ過ぎると，ハッシュエントリ不足が発生して性能が低下することがあります。このオペランドを省略すれば，ハッシュエントリ不足及びハッシュエントリ不足による性能低下は発生しません。

《推奨値の求め方》

推奨値を次に示します。

推奨値＝ MAX（↑a÷10↑，11261）を超えない最大の素数

変数	変数の算出式
a	（pd_max_users の値＋ 3）×（pd_max_access_tables の値＋ 14）＋ pd_lck_pool_size×c
c	32 ビットモードの場合は 6，64 ビットモードの場合は 4

18）pd_dbsync_lck_release_count ＝ シンクポイント時のグローバルバッファ排他解除間隔～<符号なし整数>((0，100～1073741824))

シンクポイント時に発生するグローバルバッファへの排他を解除する間隔を指定します。

シンクポイント時には，ディスクに反映する必要があるバッファ（更新バッファ）のサーチ処理が発生します。通常は，更新バッファのサーチ処理中に，一定間隔でグローバルバッファへの排他を解除します。

例えば 100 を指定した場合，100 面（グローバルバッファの面数）のサーチ処理が完了すると，一度排他を解除します。その後，再度排他を掛けてサーチ処理を続行します。このように 100 面ごとに排他の解除をします。

《利点》

このオペランドを指定すると，シンクポイント時のグローバルバッファの排他占有時間を調整できます。このオペランドの値を小さくすると，グローバルバッファの排他占有時間が短くなり，シンクポイント時のトラザクション性能が向上することがあります。

グローバルバッファプールの排他占有時間については，統計解析ユティリティを実行して，グローバルバッファプールに関する統計情報の「シンクポイント時のバッファプール排他占有時間（SYNCL）」で確認できます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。次に示す条件をすべて満たす場合にこのオペランドの指定を検討してください。

- シンクポイント時にトランザクション性能が低下する
- pdbuffer オペランドの-n オプションに指定したバッファ面数が非常に大きい

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10000 が仮定されます。

《オペランドの規則》

- 1 から 99 までの値を指定した場合，自動的に 100 が設定されます。
- 0 を指定した場合，更新バッファのサーチ処理が完了するまで，グローバルバッファに対して排他が掛かります。

《注意事項》

このオペランドの値を小さくすると，ほかのトランザクションの割り込みによって，更新バッファのサーチ時間が長くなります。その間に更新されたグローバルバッファもシンクポイント時の出力対象になります。このため，シンクポイント時に出力される更新バッファ数が増加します。シンクポイント時に出力対象となる更新バッファ数については，統計解析ユティリティの，グローバルバッファプールに関する統計情報の「シンクポイント出力ページ数（SYNCW）」で確認できます。

5.2.7　バッファに関するオペランド

19）　pd_sql_object_cache_size = SQL オブジェクト用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((22～256000))
- 64 ビットモードの場合：((22～2000000))

SQL オブジェクトを格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- SQL オブジェクトは，ユーザごとにトランザクションが終了するまでバッファに保存されます。したがって，同時に実行するトランザクションの SQL オブジェクトがバッファに格納できる大きさが必要です。
- 静的 SQL の SQL オブジェクトはトランザクション終了後もバッファに空きがなくなるまで保存して，同一 UAP を実行する複数のユーザで共用することで，SQL 解析処理の削減を図ります。バッファを有効に利用するためには，使用頻度の高い UAP の SQL オブジェクトがバッファに常駐するようにバッファを確保してください。
- バッファ長の見積もり時は，まず UAP で発行する SQL 文の SQL オブジェクト長から UAP 実行に必要なバッファ長を見積もります。次に，同時に実行される UAP と同時実行ユーザ数を考慮して必要なバッファ長を算出してください。
- 1SQL 文の SQL オブジェクト長の見積もり方法については，「付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義とシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

（pd_max_users の値＋ 3）×22

《指定値のチューニング方法》

SQL オブジェクト用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

20）　pd_table_def_cache_size = 表定義情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((100～65535))
- 64 ビットモードの場合：((100～2000000))

一度使用した表と順序数生成子の定義情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。表と順序数生成子の定義情報は，SQL 文の前処理時に使用します。また，このバッファ内の定義情報は，LRU で管理されます。

《利点》

- 一度使用した表と順序数生成子の定義情報を共用メモリ内に格納して，次回使用するときは入出力なしで使用できます。
- 動的 SQL を多く使用する場合，性能が向上します。

《指定値の目安》

- 使用頻度が高い表と順序数生成子の定義情報長の合計値を指定してください。
- 1 順序数生成子当たりの表定義情報バッファサイズは，8 キロバイトです。
- 1 表当たりの表定義情報用バッファ長の求め方については，「付録 C.4　表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- HiRDB/シングルサーバの場合

《↑ $\sqrt{\text{pd_max_users の値}+3}$ ↑×100》

- HiRDB/パラレルサーバの場合

《↑ $\sqrt{(\text{pd_max_bes_process 又は pd_max_dic_process の指定値の最大値})+3}$ ↑×100》

《指定値のチューニング方法》

表定義情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

21）　pd_auth_cache_size = ユーザ権限情報用バッファ長

～<符号なし整数>((1～100))（単位：キロバイト）

ユーザ権限情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- ユーザ権限情報用バッファには CONNECT 権限，DBA 権限，及び監査権限の情報を格納します。このバッファに情報がない場合は HiRDB 接続時にディクショナリ表から情報を取得するため，応答時間が遅くなります。したがって，常時接続しているユーザ数分の情報を格納できるようにバッファ長を指定してください。
- 1 ユーザ当たりのユーザ権限情報を格納するのに 68 バイト必要になります。それを基に計算してください。

《指定値のチューニング方法》

ユーザ権限情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

22)　pd_view_def_cache_size = ビュー解析情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((0～32000))
- 64 ビットモードの場合：((0～2000000))

ビュー解析情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《利点》

一度使用したビュー解析情報を共用メモリ内に格納して，次回使用するときは入出力なしで使用できます。

《指定値の目安》

使用頻度が高いビュー表のビュー解析情報用バッファ長の合計値を指定してください。ビュー解析情報用バッファ長の見積もりについては，「付録 C.3　ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

ビュー解析情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

《↑ $\sqrt{(\text{pd_max_users の値}+3)}$ ↑×8》

23)　pd_type_def_cache_size = ユーザ定義型情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((100～65536))
- 64 ビットモードの場合：((100～2000000))

このオペランドは，ユーザ定義型に関するオペランドです。ユーザ定義型を使用する場合は，このオペランドを指定することをお勧めします。

ユーザ定義型の情報を格納するバッファ（共用メモリ）の大きさを，キロバイト単位で指定します。ユーザ定義型の情報を使用すると，その情報がこのバッファ内に格納されます。この情報は SQL 文の前処理時に使用します。

《利点》

- このバッファにユーザ定義型の情報を格納すると，このユーザ定義型の情報を次回使用するときに，ディクショナリ表をアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- 動的 SQL を多く使用する場合，このオペランドを指定すると性能が向上します。

《指定値の目安》

使用頻度が高い表に定義してあるユーザ定義型の定義情報長を合計します。一つ当たりのユーザ定義型の定義情報長は，次に示す計算式から求めてください。

↑ {((0.3 + 0.2× a + 0.1× b）＋3）÷ 4} ↑× 4

（単位：キロバイト）

a：ユーザ定義型の属性数

b：スーパタイプを継承したサブタイプ数

《指定値のチューニング方法》

ユーザ定義型情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

24）pd_routine_def_cache_size = ルーチン定義情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((0，20～65536))
- 64 ビットモードの場合：((0，20～2000000))

次に示す定義情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。これらの定義情報は SQL 文の前処理時に使用します。

- プラグイン関数の定義情報
- システム定義スカラ関数の定義情報
- ルーチンの定義情報

《利点》

このバッファに前記の定義情報を格納すると，これらの情報を次回使用するときに，ディクショナリ表をアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。

《適用基準》

次に示す SQL を多く使用する場合に指定します。

- プラグインを使用する SQL
- システム定義スカラ関数を使用する SQL
- ルーチンを使用する SQL

《指定値の目安》

このオペランドの指定値の求め方については，「付録 C.6　ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

ルーチン定義情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，100 が仮定されます。

《注意事項》

このオペランドの指定値が全プラグイン提供関数の定義情報の合計値より小さい場合，プラグイン提供関数の定義情報はバッファ中に確保されません。

25）pd_registry_cache_size = レジストリ情報用バッファ長

～<符号なし整数>((0～65536))（単位：キロバイト）

このオペランドは，プラグインに関するオペランドです。レジストリ情報を使用するプラグインを使用する場合は，このオペランドを指定することをお勧めします。HiRDB Text Search Plug-in を使用する場合は，指定することをお勧めします。

レジストリ情報を格納するバッファ（共用メモリ）の大きさを，キロバイト単位で指定します。レジストリ情報を使用すると，レジストリ情報がバッファ内に格納されます。レジストリ情報は SQL 文の実行時に使用します。

《利点》

- このバッファにレジストリ情報を格納すると，レジストリ情報を次回使用するときに，レジストリをアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- レジストリ情報を多く使用するプラグインを使用する場合，このオペランドを指定すると性能が向上します。

《指定値の目安》

レジストリ情報用バッファ長は，次に示す計算式から求めてください。

↑（0.3＋a）↑×b　　（単位：キロバイト）

a：レジストリキー長の平均長（単位：キロバイト）

レジストリキー値の平均長は次に示す SQL で求められます。

この SQL の結果の単位はバイトのため，キロバイトに変換してください。

```
SELECT AVG(KEY_LENGTH) FROM MASTER.SQL_REGISTRY
```

b：レジストリキーの登録数

レジストリキーの登録数は次に示す SQL で求められます。

```
SELECT COUNT(*) FROM MASTER.SQL_REGISTRY
```

《指定値のチューニング方法》

レジストリ情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10 が仮定されます。

5.2.8　共用メモリに関するオペランド

26)　pd_sds_shmpool_size = シングルサーバ用共用メモリサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～200000))
- 64 ビットモードの場合：((1～4000000000))

ユニットコントローラ用の共用メモリの中で，シングルサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。サーバ共通定義及びシングルサーバ定義の pd_sds_shmpool_size オペランドを省略すると，このオペランドの値を HiRDB が計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます）。HiRDB はマニュアル「HiRDB Version 8 システム導入・設計ガイド」に記載されている「シングルサーバが使用する共用メモリの計算式」を基に計算をします。この計算式中の変数の説明に出てくるオペランドの値を変更した場合，このオペランドの値を HiRDB が自動的に再計算しま

す。なお，計算式中の変数「サーバ内のインデクス総数」は pd_assurance_index_no オペランドの値が仮定されます。

また，計算式中の変数「インデクス用のグローバルバッファプール数」，及び「グローバルバッファ総数（pdbuffer オペランドの指定数）」は，32 ビットモードの場合は 500，64 ビットモードの場合は 1000 が仮定されます。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

- 必要以上にこのオペランドの値が大きいと，共用メモリをむだに確保してしまいます。また，このオペランド値が小さ過ぎると，次に示す現象が発生します。
  ・HiRDB を開始できなくなります。
  ・UAP 又はユティリティが実行できなくなります。
- このオペランドを省略する場合は，必ず pd_assurance_table_no，及び pd_assurance_index_no オペランドに適切な値を指定してください。

5.2.9　RPC トレース情報に関するオペランド

27）　pd_rpc_trace = Y | N

RPC トレース情報を取得するかどうかを指定します。RPC トレース情報には, HiRDB の保守情報が出力されます。

通常，このオペランドは省略してください。

Y：RPC トレースを取得します。

N：RPC トレースを取得しません。

《注意事項》

このオペランドに Y を指定すると，通信性能が低下します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，N が仮定されます。

28）　pd_rpc_trace_name = "RPC トレースを取得するファイル名称"

～< 254 文字以内のパス名>

RPC トレースファイルの名称を絶対パス名で指定します。RPC トレースファイルとして，指定したファイル名の後に 1，2，l（エル）を付けた三つのファイルが作成されます。

《注意事項》

このオペランドで指定したディレクトリ下に，最大「pd_rpc_trace_size の値× 2」の大きさのファイルが作成されます。ファイル容量に注意してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

%PDDIR%¥spool¥rpctr

29）　pd_rpc_trace_size = RPC トレースを取得するファイルの容量

～<符号なし整数>((1024～2147483648))（単位：バイト）

RPC トレースファイルの容量をバイト単位で指定します。

このオペランド指定値は，"RPC トレースファイル"1 及び"RPC トレースファイル"2 のそれぞれの容量となります。

《指定値の目安》

RPC トレースは，すべてのプロセス間の通信電文が一つのファイルに時系列に蓄積されます。その量がこの指定値を超えると，もう一方のファイルにトレースデータの出力を切り替えます。古いトレースは上書きされて消去されます。このため，指定値が小さいと十分なトレースを取得できなくなり，トラブルシュートが困難になることがあります。したがって，このオペランドには 1,000,000 以上の値を指定してください。

《注意事項》

このオペランドで指定した値は RPC トレースファイル l（エル）の容量に影響を与えません。RPC トレースファイル l（エル）の容量は 0 バイト固定です。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，4096 が仮定されます。

5.2.10　トラブルシュート情報に関するオペランド

30）　pd_module_trace_max = モジュールトレース格納最大数

～<符号なし整数>((126～16383))

HiRDB のプロセスは，実行した関数やマクロの履歴をプロセス固有メモリ中に記録しています。この履歴をモジュールトレースといい，このオペランドではモジュールトレースのレコード数を指定します。この履歴の内容は core ファイルに取り込まれ，プロセス異常終了時に出力されます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，126 が仮定されます。

《注意事項》

各プロセスでは，以下のサイズのプロセス固有メモリが確保されます。

32 ビットモードの場合：64 ＋ 48×pd_module_trace_max オペランドの値（バイト）

64 ビットモードの場合：64 ＋ 64×pd_module_trace_max オペランドの値（バイト）

31）　pd_module_trace_timer_level = 0 | 10 | 20

モジュールトレースに出力する時刻の取得方法を指定します。このオペランドの指定値の意味を以下に示します。

指定値	時刻取得方法
0	すべてのモジュールトレース出力箇所で，秒単位で取得します。
10	すべてのモジュールトレース出力箇所のうち，入出力処理前後など性能調査で問題となりやすい箇所だけマイクロ秒単位で取得し，そのほかの箇所では秒単位で取得します。
20	すべてのモジュールトレース出力箇所で，マイクロ秒単位で取得します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

このオペランドに 0 以外の値を指定すると，時刻をマイクロ秒で取得する関数を発行するため，性能が低下することがあります。

32）　pd_pth_trace_max = 通信トレース格納最大数

～<符号なし整数>((1024～8388608))

トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。

《指定値の目安》

通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。

《注意事項》

通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

5.2.11　グローバルバッファに関するオペランド

33）　pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値

～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加するグローバルバッファ数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

- pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，その数に余裕を持たせた値を指定してください。
- 次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_max_add_dbbuff_no の値≦2000000－HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

条件		省略値
32 ビットモードの場合	a ≧500 の場合	256
	a ＜ 500 の場合	500－ a
64 ビットモードの場合	a ≧1000 の場合	256
	a ＜ 1000 の場合	1000－ a

a：HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《注意事項》

このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_shm_no

34)　pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加したときに割り当てられる，共用メモリセグメント数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

条件		省略値
pd_max_add_dbbuff_no オペランドを省略した場合	32 ビットモードの場合	（16－ａ）＋ 500
	64 ビットモードの場合	（16－ａ）＋ 1000
pd_max_add_dbbuff_no オペランドを指定した場合	32 ビットモードの場合	↓pd_max_add_dbbuff_no の値×1.5 ＋ 16↓ （計算値が 32752 以上だった場合は自動的に 32752 が設定されます）
	64 ビットモードの場合	

ａ：HiRDB 開始時に各サーバに割り当てられている共用メモリセグメント数

《注意事項》

- 次に示す条件式を満たす場合は，このオペランドに pd_max_add_dbbuff_no オペランドの値が仮定されます。

  pd_max_add_dbbuff_shm_no の値＜ pd_max_add_dbbuff_no の値

  省略値が上記の条件を満たす場合も，pd_max_add_dbbuff_no オペランドの値が仮定されます。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- 追加する共用メモリセグメントサイズが SHMMAX オペランドの値を超える場合，SHMMAX オペランドの値を上限値とした複数の共用メモリセグメントに分割されます。追加する共用メモリセグメントサイズを想定して SHMMAX オペランドの値を増やすか，又は分割されたときに不足しないように pd_max_add_dbbuff_shm_no オペランドの値を増やしてください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_no

5.2.12　セキュリティに関するオペランド

35）　pd_audit_def_buffer_size ＝ セキュリティ監査情報用バッファ長

～＜符号なし整数＞((1～2000000))（単位：キロバイト）

セキュリティ監査機能の情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

セキュリティ監査情報用バッファ長は，次に示す計算式から求めてください。

↑0.3 ＋ a ×0.25↑　　（単位：キロバイト）

ａ：セキュリティ監査機能の監査証跡に指定した絞り込み条件のオブジェクト数

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示すサイズのバッファ長が HiRDB の開始時に取得されます。このメモリが確保できない場合，HiRDB は開始できないため注意してください。

↑0.3 ＋ MAX｛(a ＋ 100)，(a×1.2)｝×0.25↑

（単位：キロバイト）

a：セキュリティ監査機能の監査証跡に指定した絞り込み条件のオブジェクト数

《注意事項》

このオペランドで指定した大きさのメモリが確保できない場合，HiRDB は開始できません。

5.2.13　プラグインインデクスの遅延一括作成に関するオペランド

36）　pd_plugin_ixmk_dir = "インデクス情報ファイルを作成するディレクトリ名"又は "インデクス情報ファイルを作成する HiRDB ファイルシステム領域名"

～<パス名>

プラグインインデクスの遅延一括作成をする場合に，インデクス情報ファイルを作成するディレクトリ名を指定します。インデクス情報ファイルを HiRDB ファイルシステム領域に作成する場合は，HiRDB ファイルシステム領域名を指定します。ディレクトリ名及び HiRDB ファイルシステム領域名は絶対パス名で指定してください。

プラグインインデクスの遅延一括作成については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- ここで指定するディレクトリ（又は HiRDB ファイルシステム領域）を必ず作成しておいてください。存在しないディレクトリ名（又は HiRDB ファイルシステム領域名）を指定すると，プラグインインデクスの遅延一括作成を指定した UAP（クライアント環境定義に PDPLGIXMK = YES を指定した環境で実行した UAP）の実行時にエラーとなります。
- UAP 実行後，データベース再編成ユティリティでプラグインインデクスの遅延一括作成をする前にこのオペランドの指定値を変更しないでください。変更すると，プラグインインデクスを遅延一括作成できなくなります。

5.2.14　Java に関するオペランド

Java ストアドプロシジャ又は Java ストアドファンクションを使用する場合に，Java に関するオペランドを指定します。Java ストアドプロシジャ及び Java ストアドファンクションについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

37）　pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"

～<パス名>

Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合，そのファイル名を絶対パス名で指定します。

《指定値の目安》

このオペランドで指定するファイルの容量が非常に大きくなるため，通常はこのオペランドを指定しないでください。Java ストアドプロシジャ又は Java ストアドファンクションのデバッグ時に指定することをお勧めします。なお，このオペランドで指定するファイルの容量制限はありません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，Java 仮想マシンでの標準出力又は標準エラー出力が無視されます。

《注意事項》

複数プロセスからの同時書き込みがあった場合は，その出力内容を保証しません。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

38)　pd_java_castoff = Y | N

HiRDB のシングルサーバプロセスで，Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。

Y：Java 仮想マシンを起動した場合，UAP の切り離し時にサーバプロセスを終了します。

N：Java 仮想マシンを起動してもサーバプロセスが終了しません。

省略した場合は，サーバ共通定義の pd_java_castoff オペランドが解釈値となります。

《指定値のチューニング方法》

Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。

Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に，pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。

#	懸案
1	Java 仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。
2	Java 仮想マシンを使用しないコネクションで実行する SQL でも，別のコネクションで Java 仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多い SQL を実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。 なお，Java 仮想マシンの機能については Java 仮想マシンのドキュメントのオプションを参照してください。

《注意事項》

Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- pd_process_count

5.2.15　システムログファイルに関するオペランド

39)　pd_log_dual = Y | N

システムログファイルを二重化するかどうかを指定します。

Y：システムログファイルを二重化します。

N：システムログファイルを二重化しません。

《利点》

システムログファイルを二重化すると，HiRDB はＡ系及びＢ系の両方に同じシステムログを取得します。取得したシステムログを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシステムログを読み込めるため，システムの信頼性を向上できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

システムログファイルを二重化する場合は pdlogadpf オペランドで B 系のシステムログファイル名を指定してください。

40） pd_log_remain_space_check = warn | safe

システムログファイルの空き率が**警告値**未満になったときの HiRDB の処理方式を指定します。この機能を**システムログファイルの空き容量監視機能**といいます。システムログファイルの空き容量監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

warn：

システムログファイルの空き率が警告値未満になった場合，警告メッセージ KFPS01162-W を出力します。

safe：

システムログファイルの空き率が警告値未満になった場合，新規トランザクションのスケジューリングを抑止して，サーバ内の全トランザクションを強制終了します。このとき，KFPS01160-E メッセージを出力します。

《指定値の目安》

システムログファイルの空き容量不足によるユニットの異常終了の可能性を低くできるため，safe の指定を推奨します。ただし，safe を指定すると，システムログファイルの空き容量が不足したときに，サーバ内の全トランザクションが強制終了されます。このため，システムログファイルの設計をより正確に行う必要があります。システムログファイルの設計については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，warn が仮定されます。

41） pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕…

～<パス名>((1～136 文字))

システムログの**自動ログアンロード機能**を使用する場合に，アンロードログファイルの出力先ディレクトリを絶対パス名で指定します。アンロードログファイルを HiRDB ファイルシステム領域に作成する場合は，その HiRDB ファイルシステム領域名を指定してください。なお，**このオペランドに指定するディレクトリ又は HiRDB ファイルシステム領域は，HiRDB の開始前に作成しておいてください。**

自動ログアンロード機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値の目安》

空きディスク容量を確認してからディレクトリを指定してください。作成されるアンロードログファイルによって，ディスク容量が圧迫される原因になります。

また，ディスク容量の不足によって，指定したディレクトリにアンロードログファイルが作成できない場合，自動ログアンロード機能が停止します。このようなことが発生する可能性がある場合は，ディレクトリを複数指定することをお勧めします。

ただし，ディレクトリは一つだけの方がデータベース回復時の運用（回復に必要なアンロードログファイルの選別）が若干簡単になります。

ディレクトリを複数指定する場合の補足事項を次に示します。

- ディスク障害などに備えて，異なるパーティションのディレクトリを指定することをお勧めします。

- ディスク容量満杯又はディスク障害などで一つのディレクトリ下にアンロードログファイルが作成できない場合，ほかのディレクトリ下にアンロードログファイルを作成します。このとき，HiRDB はこのオペランドに指定した順番にディレクトリを使用します。

《オペランドの規則》

- ディレクトリは最大 128 個指定できます。
- 複数のディレクトリを指定した場合，同じパス名を指定できません。

《注意事項》

次の場合は，自動ログアンロード機能を使用できません。

- pd_log_unload_check オペランドに N を指定している場合

なお，マルチ HiRDB を利用している場合は，HiRDB ごとに異なるディレクトリを作成してください。同じディレクトリを指定すると，どのアンロードログファイルがどの HiRDB に対応しているか分からなくなる可能性があります。

42）pd_log_singleoperation = Y | N

このオペランドは，システムログファイルを二重化している場合に指定してください。二重化していない場合は指定する必要はありません。

システムログファイルの片系運転をするかどうかを指定します。システムログファイルに障害が発生して，両系ともに使用できるシステムログファイルがない場合でも，HiRDB を異常終了しないで正常な片方の系だけで処理を続行します。これを**システムログファイルの片系運転**といいます。

また，両方のシステムログファイルで処理を続行すること(通常の処理形態)を**システムログファイルの両系運転**といいます。

Y：システムログファイルの片系運転をします。

N：システムログファイルの片系運転をしません。常に両系運転とします。

《前提条件》

pd_log_dual オペランドに Y を指定している必要があります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

43）pd_log_rerun_reserved_file_open = Y | N

システムログファイルの自動オープンをするかどうかを指定します。

HiRDB を再開始するときに，「上書きできる状態」のシステムログファイルがない場合，予約のファイルがあれば HiRDB が予約のファイルをオープンして「上書きできる状態」にし，処理を続行します。これを**システムログファイルの自動オープン**といいます。

予約ファイルを使用するのは，次の場合です。

- 再開始後，最初のシンクポイントダンプを取得するまでの間
- オープンされたファイルグループすべてが「上書きできない状態」の場合

Y：システムログファイルの自動オープンをします（予約ファイルをオープンして使用します）。

N：システムログファイルの自動オープンをしません（予約ファイルを使用しません）。

《利点》

このオペランドに Y を指定すると，ユニットの再開始時に「スワップ先にできる状態」のファイルがなくても，予約のファイルがあればユニットを再開始できます。

ただし，「アンロード待ち状態」のファイルがあるときは，ユニットを停止します。「アンロード待ち状態」のファイルのアンロードを実行した後，再度ユニットを開始してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

44）　pd_log_rerun_swap = Y｜N

HiRDB の再開始時に，システムログファイルをスワップするかどうかを指定します。

Y：システムログファイルをスワップします。

N：システムログファイルをスワップしません。

《利点》

このオペランドに Y を指定すると，再開始前後でシステムログファイルを物理的に分けられます。このため，再開始前のシステムログファイルをサーバ稼働中に使用できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

45）　pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間

～<符号なし整数>((1～32580))（単位：秒）

システムログファイルのスワップが完了するまでの待ち時間を秒単位で指定します。指定した時間を過ぎてもシステムログファイルのスワップが完了しない場合は，ユニットが異常終了します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。マシンの性能が低いなどの理由によってシステムログファイルのスワップに時間が掛かる場合は，このオペランドの値を大きくしてください。また，ディスク障害などによってシステムログファイルのスワップに遅延が発生した場合に，より短い時間で異常を検知し，ユニットを異常終了させたいときは，このオペランドの値を小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，180 が仮定されます。

46）　pd_log_unload_check = Y｜N

システムログファイルのアンロード状態を，HiRDB がチェックするかどうかを指定します。

Y：

アンロード状態をチェックします（通常の運用となります）。

N：

アンロード状態をチェックしません。アンロードの状態に関係なく，次に示す条件がすべて満たされれば，システムログファイルを「スワップ先にできる状態」にします。

- 上書きできる状態
- 抽出完了状態（HiRDB Datareplicator）

このときシステムログファイルの運用方法が，アンロード状態のチェックを解除する運用になります。アンロード状態のチェックを解除する運用については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

N を指定したときの利点を次に示します。

- システムログファイルのアンロード操作がなくなるため，運用方法が簡単になります。

- アンロードログファイルを保管するためのファイル容量が不要になります。

《指定値の目安》

データベースを回復するときにシステムログを使用しない場合（バックアップ取得時点に回復できればよい場合）に N を指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Y が仮定されます。

《注意事項》

N を指定したときの注意事項を次に示します。

- バックアップ取得時点にしかデータベースを回復できません。
- データベースの回復にシステムログが必要なのにこの運用をした場合はデータベースの回復手段がなくなります。

47）pd_log_max_data_size = ログ入出力バッファ長

～<符号なし整数>((32000～523000))（単位：バイト）

システムログの入出力に使用するバッファの大きさをバイト単位で指定します。

《指定値の目安》

次に示す指定値のチューニング方法に従って指定値を変更してください。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）の次に示す情報を調べて，このオペランドの指定値を変更してください。

- 入出力待ちバッファ面数（# OF BUFFER FOR WAIT I/O）

  入出力待ちバッファ面数の平均値が 100 を大きく超える場合は，このオペランドの値を大きくして，平均値が 100 に近付くようにしてください。

- カレントバッファなしによる待ち回数（# OF WAIT THREAD）

  カレントバッファなしによる待ち回数が 0 以外の値になる場合は，このオペランドの値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，400000 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 32000 が仮定されます）。

《ほかのオペランドとの関連》

pd_log_write_buff_count オペランドとともに，ログ入出力バッファ長を決定してください。

48）pd_log_write_buff_count = ログ出力バッファ面数

～<符号なし整数>((3～65000))

システムログの出力に使用するバッファの面数を指定します。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）のカレントバッファなしによる待ち回数（# OF WAIT THREAD）を調べて，待ち回数が多ければ，スループットが向上するよう指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 3 が仮定されます）。

《ほかのオペランドとの関連》

pd_log_max_data_size オペランドとともに，ログ出力バッファ面数を決定してください。

49）　pd_log_rec_leng = システムログファイルのレコード長

～<符号なし整数>((1024，2048，4096))（単位：バイト）

システムログファイルのレコード長を指定します。1024，2048，4096 のどれかを指定します。

レコード長は pdloginit コマンドの-l オプションで設定します。-l オプションで設定するレコード長をこのオペランドに指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，4096 が仮定されます。

《注意事項》

- pdloginit コマンドの-l オプションで設定するレコード長と異なるレコード長をこのオペランドに指定すると，そのシステムログファイルはオープンできません。
- システムログファイルのレコード長を変更する方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

50）　pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数

～<符号なし整数>((0～256))

ロールバック処理でシステムログの入力に使用するバッファの面数を指定します。このオペランドに 0 を指定した場合，HiRDB がロールバック用ログ入力バッファ面数を決定します。

《指定値のチューニング方法》

このオペランドに 0 を指定してください。0 を指定してメモリ不足が発生する場合は，このオペランドの指定を省略してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

ユニット内サーバ数×2

51）　pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕

～<符号なし整数>((0～104857600))《0，0》（単位：レコード）

このオペランドは，システムログファイルの自動拡張機能を使用する場合に指定します。

一つのシステムログファイルが拡張契機 1 回当たりに拡張するサイズ，及び拡張できるファイルサイズの上限をレコード数で指定します。

1 回当たりに拡張するサイズを省略するか，又は 0 を指定した場合，システムログファイルの自動拡張を行いません。拡張上限サイズを省略するか，又は 0 を指定した場合，ファイルシステム領域のあるディスクが満杯になるか，システムログファイルの容量が上限に達するまでシステムログファイルの自動拡張を行います。また，1 回当たりに拡張するサイズに拡張上限サイズより大きい値を指定した場合，拡張上限サイズに指定した値まで自動拡張を行います。

システムログファイルの自動拡張機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《前提条件》

pd_large_file_use オペランドに Y を指定しているか，又は指定を省略している必要があります。

《指定値の目安》

- 1 回当たりに拡張するサイズには，pdloginit コマンドの-n オプションでシステムログファイルを作成した際に指定したレコード数を基に指定します。全システムログファイルのレコード数の平均値の 1/10 を計算し，指定してください。
- 通常，拡張上限サイズは省略してください。

《指定値のチューニング方法》

システムログの出力量が自動拡張で拡張するサイズを超えると，システムログファイルが満杯になってユニットダウンする場合があります。その場合は，1 回当たりに拡張するサイズを大きくしてください。また，拡張処理に時間が掛かり，トランザクション性能に影響を及ぼす場合は，1 回当たりに拡張するサイズを小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

5.2.16　シンクポイントダンプファイルに関するオペランド

52)　pd_spd_dual = Y | N

シンクポイントダンプファイルを二重化するかどうかを指定します。

Y：シンクポイントダンプファイルを二重化します。

N：シンクポイントダンプファイルを二重化しません。

《利点》

シンクポイントダンプファイルを二重化すると，HiRDB はA系及びB系の両方に同じシンクポイントダンプを取得します。取得したシンクポイントダンプを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシンクポイントダンプを読み込めるため，システムの信頼性を向上できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

シンクポイントダンプファイルを二重化する場合は，pdlogadpf オペランドで B 系のシンクポイントダンプファイル名を指定してください。

53)　pd_spd_assurance_msg = Y | N

シンクポイントダンプが有効化されたとき，メッセージ KFPS02183-I を出力するかどうかを指定します。

Y：出力します。

N：出力しません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Y が仮定されます。

54)　pd_spd_assurance_count = 有効保証世代数

～<符号なし整数>((1～2))

システム回復時のシンクポイントダンプファイルの入力障害などに備え，HiRDB 稼働中に保存しておくシステムログファイルの範囲を，シンクポイントダンプファイルの世代数によって指定します。この世代数のことを**有効保証世代数**といいます。ここで指定した世代数分のシンクポイントダンプファイルは，上書きできない状態になります。

《利点》

有効保証世代数を 2 にすると，最新世代のシンクポイントダンプファイルに障害が発生しても，1 世代前のシンクポイントからシステムを回復できるため，信頼性が向上します。

《指定値の目安》

- 信頼性を向上させる場合は有効保証世代数を 2 にしてください。ただし，有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます（二つになります）。
- シンクポイントダンプファイルを二重化して信頼性を向上している場合は，このオペランドを省略するか，又は 1 を指定することをお勧めします。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

《注意事項》

- シンクポイントダンプファイルの必要最低数は**有効保証世代数＋ 1** となります。
- 有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます。上書きできない状態のシンクポイントダンプファイルに対応するシステムログファイルは上書きできない状態になります。したがって，有効保証世代数を 2 にすると，上書きできない状態のシステムログファイルが増えるため，スワップ先にできる状態のシステムログファイル数が不足することがあります。これを防ぐためにはシステムログファイルの容量を考慮してください。

55)　pd_spd_reduced_mode = 縮退運転オプション

～<符号なし整数>((0～2))

シンクポイントダンプファイルの縮退運転をするかどうかを指定します。

縮退運転とは，HiRDB 稼働中又は再開始処理中のファイル障害などによって，シンクポイントダンプファイルの数が運用に必要なファイル数（**有効保証世代数**※＋ 1）以下になった場合でも，最低二つのファイルが使用できれば処理を続行する機能です。

注※　pd_spd_assurance_count オペランドの指定値です。

0：縮退運転を使用しません。

1：縮退運転を使用します。

2：縮退運転を使用し，縮退運転でシンクポイントダンプ取得の契機ごとに警告メッセージを出力します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

56)　pd_spd_reserved_file_auto_open = Y | N

シンクポイントダンプファイルの自動オープンをするかどうかを指定します。

シンクポイントダンプファイルに障害が発生して，運用に必要なファイル数（**有効保証世代数**※＋1）を下回った場合，予約のファイルがあればHiRDBが予約のファイルをオープンして上書きできる状態にし，処理を続行します。これを**シンクポイントダンプファイルの自動オープン**といいます。

注※　pd_spd_assurance_countオペランドの指定値です。

Y：

シンクポイントダンプファイルの自動オープンをします。運用に必要なファイル数（有効保証世代数＋1）を下回った場合，予約ファイルを自動的にオープンします。

N：

シンクポイントダンプファイルの自動オープンをしません。運用に必要なファイル数（有効保証世代数＋1）を下回っても，予約ファイルを自動的にオープンしません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Nが仮定されます。

《ほかのオペランドとの関連》

このオペランドの指定はpd_spd_reduced_modeオペランドより優先されます。

57）pd_spd_max_data_size＝シンクポイントダンプファイルのバッファ長

～<符号なし整数>((32000～4000000))（単位：バイト）

シンクポイントダンプをシンクポイントダンプファイルに入出力するときに使用するバッファ（共用メモリ）の大きさをバイト単位で指定します。

ここで指定した値によってシンクポイントダンプファイルの入出力回数を制御します。

《指定値の目安》

- 通常はこのオペランドを指定する必要はありません。
- 指定値を大きくするほど，シンクポイントダンプファイルに対する入出力回数が少なくなります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，32768が仮定されます。

58）pd_log_sdinterval＝システムログ出力量〔，経過時間〕

シンクポイントダンプの取得間隔を指定します。次に示す条件で指定できます。

- システムログの出力量
- 前回シンクポイントダンプを取得してからの経過時間

システムログ出力量：～<符号なし整数>((100～100000))（単位：ログブロック数）

シンクポイントダンプの取得間隔を指定します。ここで指定したログブロック数分のシステムログが出力されるたびに，シンクポイントダンプを取得します。

経過時間：～<符号なし整数>((0,10～1440))（単位：分）

シンクポイントダンプの取得間隔を指定します。前回シンクポイントダンプを取得してから，ここで指定した経過時間が過ぎると，シンクポイントダンプを取得します。

- 経過時間に0を指定すると，HiRDBは経過時間でのシンクポイントダンプを取得しません。
- 前回シンクポイントダンプを取得してからトランザクションが一度も発生しない場合，ここで指定した経過時間が過ぎてもシンクポイントダンプを取得しません。

《指定値の目安》

- HiRDB を再開始するときの所要時間に特に規定のない場合は，このオペランドを指定する必要はありません。
- このオペランドの指定値によって，HiRDB を再開始するときの所要時間が決まります。

  このオペランドの指定値を小さくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が短くなります。ただし，シンクポイントダンプの取得回数が増えるため，オンライン中の性能が低下する場合があります。

  逆に，このオペランドの指定値を大きくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が長くなります。ただし，シンクポイントダンプの取得回数が減るため，オンライン中の性能が向上する場合があります。

《指定値のチューニング方法》

シンクポイントダンプがどの程度の間隔で取得されているかは，統計解析ユティリティのシステムの稼働に関する統計情報の「シンクポイントダンプ取得間隔時間（SYNC POINT GET INTERVAL）」で確認できます。SYNC POINT GET INTERVAL の平均値を見て，シンクポイントダンプを取得する間隔が長いと判断したら，ここでの指定値を小さくしてください。また，反対に短いと判断したら，ここでの指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- システムログ出力量：5000
- 経過時間：60

ただし, pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，システムログ出力量の省略値は 1000 が仮定されます。

《注意事項》

シンクポイントダンプの取得間隔はシステムログの出力量で決まります。このため，更新系のトランザクションがほとんどない時間帯は，メモリからデータベースに反映するまでの時間が長くなります。このような状態で障害が発生すると，その間に発生したトランザクションの回復時間が長くなります。このようなケースが考えられる場合は,「経過時間」でもシンクポイントダンプの取得間隔を設定してください。

5.2.17　サーバ用ステータスファイルに関するオペランド

59）　pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

:

pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

サーバ用ステータスファイルを定義します。pd_sts_file_name_2～7 オペランドは省略できますが，pd_sts_file_name_1 オペランドは省略できません。

"論理ファイル名"：～<識別子>((1～8 文字))

サーバ用ステータスファイルの論理ファイル名を指定します。ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系ステータスファイル名"：～<パス名>((167 文字以内))

A 系のステータスファイル名を絶対パス名で指定します。

"B 系ステータスファイル名"：～<パス名>((167 文字以内))

B 系のステータスファイル名を絶対パス名で指定します。

《指定値の目安》

- A 系及び B 系のステータスファイル名には，pdstsinit コマンドで作成したステータスファイルを指定してください。pdstsinit コマンドで作成していないステータスファイルを指定すると，そのステータスファイルは**実体のないステータスファイル**となります。
- ステータスファイルに障害が発生すると，HiRDB はステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要となります。
- A 系と B 系には同じレコード長，及び同じ容量のステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- ステータスファイルは，A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系ステータスファイル名，及び B 系ステータスファイル名には同じ名称を重複して指定できません。
- HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥sts01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。sts01 については大文字と小文字の区別をします。

《留意事項》

- HiRDB を正常開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，全ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときは pd_sts_file_name_1～7 の中で最初に指定したステータスファイルが現用ファイルになります。残りのファイルのうちオープンできたファイルが予備ファイルになり，オープンできないファイルは予約ファイルになります。
- HiRDB を再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。

●実体のないステータスファイルの使い方

実体のないステータスファイルを指定すると，HiRDB の稼働中にステータスファイルを新規追加できます。例えば，ステータスファイルの障害などで予備ファイルが少なくなったときに，実体のないステータスファイルを予備ファイルにします。実体のないステータスファイルを予備ファイルにする手順を次に示します。

〈手順〉

1. pdstsinit コマンドで，システムファイル用の HiRDB ファイルシステム領域にステータスファイルを作成します。
2. pdstsopen コマンドで，ステータスファイルをオープンします。

この操作は HiRDB 稼働中に実行できます。HiRDB を一度停止する必要はありません。

- **メリット及びデメリット**

  実体のないステータスファイルを定義すると，HiRDB ファイルシステム領域の占有量は小さくなります。しかし，実体のないステータスファイルを予備ファイルとして追加するときに，システムファイル用の HiRDB ファイルシステム領域に十分な空き（ファイルを追加するだけの空き領域）がないと追加できないため，システムの信頼性は低くなります。

実体のないステータスファイルを定義しないと，HiRDB ファイルシステム領域の占有量は大きくなるが，ファイル障害によるスワップ先が保証されるため，信頼性は高くなります。

- **注意事項**

  実体のないステータスファイルを定義した場合，HiRDB は HiRDB の開始時にステータスファイルに異常があると認識します。このため，pd_sts_initial_error オペランドに stop（省略値）を指定していると，HiRDB を開始できないので注意してください。実体のないステータスファイルを定義する場合は，pd_sts_initial_error オペランドに continue 又は excontinue を指定してください。また，HiRDB を開始する前に現用ファイルを pd_sts_last_active_file オペランドに指定する必要があります。

5.2.18　サーバ用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

60）pd_sts_initial_error = stop | continue | excontinue

サーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，又はバックエンドサーバ）の開始時，HiRDB はサーバ用ステータスファイルの現用ファイルを特定する処理を実行します。この特定処理で，次に示す異常を検知したときの HiRDB の処置を指定します。

- サーバ用ステータスファイルの実体がない
- サーバ用ステータスファイルの障害を検知した

現用ファイルの特定処理は pd_sts_file_name_1～7 オペランドに指定したサーバ用ステータスファイルを対象とします。

stop：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。この場合，異常を検知したステータスファイルの対策を実施した後に HiRDB を開始してください。

continue 又は excontinue：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知しても，現用ファイルが正常な場合はサーバの開始処理を続行します。ただし，pd_sts_singleoperation オペランドの値（ステータスファイルの片系運転をするかどうか）によっては開始処理を中止することがあります。pd_sts_singleoperation オペランドとの関係を次に示します。

●pd_sts_singleoperation オペランドとの関係

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
continue	サーバ用ステータスファイルの異常を検知した場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。
stop（省略値）	サーバ用ステータスファイルの異常を検知した場合，HiRDB が現用ファイルを特定してサーバの開始処理を続行します。ただし，A 系と B 系のファイルが次に示	HiRDB が現用ファイルを特定できない場合は，HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
	す表（**HiRDB が現用ファイルを特定できないケース**）の条件のどれかに当てはまる場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	定します。その後，HiRDB を開始してください。

●HiRDB が現用ファイルを特定できないケース

pd_sts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
continue	障害閉塞	障害閉塞
	障害閉塞	オープン（初期状態）
	障害閉塞	ファイル実体なし
	オープン（初期状態）	障害閉塞
	オープン（初期状態）	ファイル実体なし
	ファイル実体なし	オープン（初期状態）
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし
excontinue	障害閉塞	障害閉塞
	障害閉塞	ファイル実体なし
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし

したがって，HiRDB 管理者の操作をなるべく少なくしたい場合（pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定するケースを減らしたい場合）は，次に示すようにオペランドを指定してください。

- pd_sts_initial_error オペランドに excontinue を指定
- pd_sts_singleoperation オペランドに stop を指定

《指定値の目安》

各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
サーバ開始時の HiRDB の処理	サーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	一部のサーバ用ステータスファイルに異常があっても，現用ファイルが正常な場合はサーバの開始処理を続行します。
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB 開始時の障害対策を簡易化する場合は continue 又は excontinue を指定します。

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
メリット	サーバの開始時に該当するサーバの全サーバ用ステータスファイルの正常が保証されます。このため，HiRDB の開始後に現用ファイルの障害が発生した場合, 予備ファイルにスワップできます。	サーバの開始時に一部のサーバ用ステータスファイルの障害を検知しても，残りの正常なファイルだけで HiRDB を開始できます。このため，HiRDB の停止時間を短くできます。この場合，予備ファイルが少なくなっているため，障害状態のステータスファイルをすぐに修復する必要があります。
デメリット	サーバ用ステータスファイルの障害によって HiRDB の開始処理が中止される可能性が高くなります。	予備ファイルが少ない状態で HiRDB を稼働することがあるため，システムの信頼性が低下します。予備ファイルの数によってはサーバ用ステータスファイルをスワップできないことがあります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，stop が仮定されます。

《注意事項》

- 現用ファイルが両系とも異常の場合は，このオペランドの指定値に関係なくサーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。
- HiRDB を開始する前に現用ファイルを pdstsinit コマンドで初期化しないでください。初期化すると HiRDB を再開始できません。
- このオペランドに excontinue を指定し，すべての世代のステータスファイルが片系運転でかつオープン（初期状態）の場合，HiRDB は起動しません。

《備考》

オペランドの指定値と HiRDB の処理及び HiRDB 管理者の処置を次に示します。

●オペランドの指定値と HiRDB の処理

pd_sts_initial_errorオペランドの値	ステータスファイル障害	pd_sts_singleoperationオペランドの値	HiRDBが現用ファイルを特定できるか	pd_sts_last_active_fileの指定	pd_sts_last_active_fileオペランドの値と，オープンできたステータスのうち，最新のファイルが一致するか	現用ファイル障害	pd_sts_last_active_sideオペランドの指定	pd_sts_last_active_sideオペランドに指定されたファイルが使用できるか	表「HiRDBの処理及びHiRDB管理者の処置」の対応番号
stop（省略値）	なし	－	－	－	－	－	－	－	[1]
	あり	－	－	－	－	－	－	－	[5]
continue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
excontinue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]

（凡例）－：該当しません（条件がHiRDBの処理に影響を与えません）。

●HiRDB の処理及び HiRDB 管理者の処置

対応番号	HiRDB の処理	HiRDB 管理者の処置
[1]	HiRDB の開始処理を続行します。	なし
[2]	HiRDB が最新の現用ファイルを特定して，開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[3]	pd_sts_last_active_file オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[4]	pd_sts_last_active_file 及び pd_sts_last_active_side オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[5]	pd_sts_initial_error オペランドに stop が指定されているため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000010 が示す対策をしてください。

対応番号	HiRDB の処理	HiRDB 管理者の処置
[6]	前回稼働時の現用ファイルが特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000016 が示す対策をしてください。
[7]	HiRDB が特定した現用ファイルの正常な系と，pd_sts_last_active_side オペランドに指定された系が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000017 が示す対策をしてください。
[8]	HiRDB が特定した現用ファイル名と，pd_sts_last_active_file オペランドに指定したファイル名が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000015 が示す対策をしてください。
[9]	前回稼働時の現用ファイルの正常な系が特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000018 が示す対策をしてください。

61）　pd_sts_singleoperation = stop | continue

サーバ用ステータスファイルの片系運転をするかどうかを指定します。予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，正常な系（片方の系）だけで処理を続行することを**ステータスファイルの片系運転**といいます。ステータスファイルの片系運転については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

なお，予備ファイルがある状況で現用ファイルの片系に障害が発生した場合は，このオペランドの指定値に関係なく，ステータスファイルをスワップして処理を続行します（片系運転にはなりません）。

stop：

片系運転をしません。片系運転になる場合は HiRDB を異常終了します。HiRDB が異常終了した場合，予備ファイルを準備した後に HiRDB を開始してください。

continue：

片系運転をします。片系運転になると KFPS01044-I メッセージが出力されます。片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，HiRDB を再開始できなくなるため，片系運転になった場合はすぐに予備ファイルを準備してください。

《指定値の目安》

- stop を指定してシステムの信頼性を上げることを推奨します。また，現用ファイルの障害に対しては，予備ファイルの数を増やして備えることを推奨します。
- 各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB を停止させないことを重視する場合は continue を指定します。
メリット	予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，片系運転をしないで HiRDB	予備ファイルがない状況で現用ファイルの片系に障害が発生しても，処理を続行できます。この

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
	を異常終了します。このため，現用ファイルの内容が失われる可能性が低くなります。	ため，ステータスファイルの障害によってHiRDB が停止する可能性が低くなります。
デメリット	ステータスファイルの障害によって HiRDB が停止する可能性が高くなります。ただし，予備ファイルの数を増やすことで可能性を低くできます。	片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，現用ファイルの内容が失われるため，HiRDB を再開始できなくなります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，stop が仮定されます。

《ほかのオペランドとの関連》

pd_sts_singleoperation 及び pd_sts_initial_error オペランドの指定値の組み合わせによって，ステータスファイル障害時の HiRDB の処理が決定します。したがって，これら二つのオペランドの指定値は一緒に考えるようにしてください。

62)　pd_sts_last_active_file = "論理ファイル名"

～<識別子>((1～8 文字))

HiRDB の開始時に現用にするステータスファイルの論理ファイル名を指定します。HiRDB は，このオペランドに指定されたファイルと HiRDB が特定した現用ファイルを比較します。ファイルが一致する場合は HiRDB を開始しますが，一致しない場合は開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **全ステータスファイルを初期化した直後に HiRDB を開始する場合**

   pd_sts_file_name1～7 オペランドに指定した最も小さい番号の正常な論理ファイル名を指定してください。この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，全ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_sts_last_active_side オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   全ステータスファイルを初期化してください。その後，1 の方法を実施してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

5. 実体のないステータスファイルを指定している場合

現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I
- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージのうちで，最も直前に出力されたメッセージに表示されているステータスファイルが現用ファイルになります。

63）　pd_sts_last_active_side = A | B

現用ファイルの片系に障害が発生している状態で HiRDB を開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB を開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- pd_sts_last_active_file オペランドを指定している

5.2.19　作業表用ファイルに関するオペランド

64）　pdwork -v "HiRDB ファイルシステム領域名称"〔，"HiRDB ファイルシステム領域名称"〕…

～< 141 文字以内のパス名>

作業表用ファイル用の HiRDB ファイルシステム領域の名称を指定します。作業表用ファイルとは，SQL 文を実行するときに一時的に情報を格納するファイルのことで HiRDB が自動的に作成します。作業表用ファイルを必要とする SQL 文については，マニュアル「HiRDB Version 8 解説」を参照してください。

なお，このオペランドは省略しないでください。省略した場合，作業表用ファイルを必要とする SQL 文を実行できないことがあります。

《注意事項》

- このオペランドに指定する HiRDB ファイルシステム領域は pdfmkfs コマンドで初期設定した HiRDB ファイルシステム領域を指定してください。
- 作業表用ファイルの容量が大きい場合は，領域長が大きい HiRDB ファイルシステム領域を指定してください。作業表用ファイルの容量見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

  ただし, HiRDB ファイルシステム領域を pdfmkfs -a コマンドで初期設定した場合, -n オプションで指定した値を使い切ると，HiRDB は領域サイズを自動で拡張します。pdfmkfs コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。
- 作業表用ファイル用の HiRDB ファイルシステム領域は，RD エリア及びシステムファイル用の HiRDB ファイルシステム領域と同じものを指定できません。

- このオペランドに HiRDB ファイルシステム領域を二つ以上指定している場合，作業表用ファイルの作成時にエラーが発生した HiRDB ファイルシステム領域は，それ以降，原則使用しません。ほかの HiRDB ファイルシステム領域だけを使用します。

  例外として，容量不足やファイル数オーバーなどによって，ほかのすべての HiRDB ファイルシステム領域で作業表用ファイルの作成が失敗した場合に，使用されなくなった HiRDB ファイルシステム領域で作業表用ファイルを作成しようとします。その結果，作業表用ファイルが正常に作成された場合，その HiRDB ファイルシステム領域を再使用するようになります。

  なお，HiRDB を一度終了して開始すると，作業表用ファイルの作成時にエラーが発生して使用されなくなった HiRDB ファイルシステム領域が使用できるようになります。

《オペランドの規則》

- HiRDB ファイルシステム領域名称は必ず 1 個以上指定してください。
- HiRDB ファイルシステム領域名称は最大 16 個指定できます。
- このオペランドはシングルサーバ定義内に一つだけ指定できます。複数個指定した場合は最初に指定したものが有効となります。

5.2.20　システムログファイルの構成に関するオペランド

65)　pdlogadfg -d sys -g ファイルグループ名〔ONL〕

システムログファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシステムログファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～200 個指定できます。

《オペランドの規則》

このオペランドは 2～200 個指定できます。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

66)　pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名"〔-b "システムログファイル名"〕

ファイルグループを構成するシステムログファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "システムログファイル名"：～<パス名>((167 文字以内))

ファイルグループを構成するシステムログファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

-b "システムログファイル名"：～<パス名>((167 文字以内))

システムログファイルを二重化（pd_log_dual オペランドに Y を指定）する場合に，B 系のシステムログファイル名を絶対パス名で指定します。pd_log_dua オペランドに Y を指定しない場合は，システムログファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

《オペランドの規則》

HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥log01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。log01 については大文字と小文字の区別をします。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

5.2.21　シンクポイントダンプファイルの構成に関するオペランド

67）　pdlogadfg -d spd -g ファイルグループ名〔ONL〕

シンクポイントダンプファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシンクポイントダンプファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～30 個指定できます。

《オペランドの規則》

このオペランドは 2～60 個指定できます。

68）　pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕

ファイルグループを構成するシンクポイントダンプファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "シンクポイントダンプファイル名"：～<パス名>((167 文字以内))

ファイルグループを構成するシンクポイントダンプファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

-b "シンクポイントダンプファイル名"：～<パス名>((167 文字以内))

シンクポイントダンプファイルを二重化（pd_spd_dual = Y を指定）する場合に，B 系のシンクポイントダンプファイル名を絶対パス名で指定します。pd_spd_dual = Y の指定がない場合は，シンクポイントダンプファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

《オペランドの規則》

HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥sync01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。sync01 については大文字と小文字の区別をします。

5.2.22　プラグインに関するオペランド

69)　pdplgprm -n プラグイン名称〔 -s 共用メモリサイズ〕

プラグインの名称及びプラグインが使用する共用メモリの大きさを指定します。プラグインを使用しない場合は，このオペランドを省略してください。

《前提条件》

ここで指定するプラグインは，あらかじめ pdplgrgst コマンドで HiRDB に登録しておいてください。

-n プラグイン名称：～<識別子>((1～30 文字))

ここで指定するプラグイン名称については，プラグインに付いているマニュアルを参照してください。

-s 共用メモリサイズ：～<符号なし整数>((1～2000000))《0》（単位：キロバイト）

プラグインが使用する共用メモリの大きさをキロバイト単位で指定します。プラグインが使用する共用メモリの大きさについては，プラグインに付いているマニュアルを参照してください。

6 フロントエンドサーバ定義

この章では, フロントエンドサーバ定義の各オペランドの内容について説明します。

6.1　オペランドの形式

フロントエンドサーバ定義ではフロントエンドサーバの実行環境を定義します。ここでは，フロントエンドサーバ定義の各オペランドの形式について説明します。なお，表中の「番号」は，「6.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

番号	形式
1	〔set pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕〕※
2	〔set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔〕※

(2)　SQL の最適化

番号	形式
3	〔set pd_optimize_level = SQL 最適化オプション〔，SQL 最適化オプション〕…〕※
4	〔set pd_additional_optimize_level = SQL 拡張最適化オプション〔，SQL 拡張最適化オプション〕…〕※
5	〔set pd_floatable_bes = "バックエンドサーバ名"〔，"バックエンドサーバ名"〕…〕
6	〔set pd_non_floatable_bes = "バックエンドサーバ名"〔，"バックエンドサーバ名"〕…〕

(3)　システム監視

番号	形式
7	〔set pd_watch_pc_client_time = クライアントからの要求間隔監視時間〕※
8	〔set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〕※

(4)　SQL 実行時間警告出力機能

番号	形式
9	〔set pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜SQL 実行時間警告情報の出力条件（時間指定）〕※

(5)　拡張 SQL エラー情報出力機能

番号	形式
10	〔set pd_uap_exerror_log_use = YES｜NO〕※

(6)　排他制御

番号	形式
11	[set pd_fes_lck_pool_size = フロントエンドサーバの排他制御用プールサイズ] ※
12	[set pd_fes_lck_pool_partition = フロントエンドサーバの排他制御用プール分割数] ※
13	[set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数] ※

(7)　バッファ

番号	形式
14	[set pd_sql_object_cache_size = SQL オブジェクト用バッファ長] ※
15	[set pd_table_def_cache_size = 表定義情報用バッファ長] ※
16	[set pd_auth_cache_size = ユーザ権限情報用バッファ長] ※
17	[set pd_view_def_cache_size = ビュー解析情報用バッファ長] ※
18	[set pd_type_def_cache_size = ユーザ定義型情報用バッファ長] ※
19	[set pd_routine_def_cache_size = ルーチン定義情報用バッファ長] ※
20	[set pd_registry_cache_size = レジストリ情報用バッファ長] ※

(8)　RPC トレース情報

番号	形式
21	[set pd_rpc_trace = Y | N] ※
22	[set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"] ※
23	[set pd_rpc_trace_size = RPC トレースを取得するファイルの容量] ※

(9)　トラブルシュート情報

番号	形式
24	[set pd_module_trace_max = モジュールトレース格納最大数] ※
25	[set pd_module_trace_timer_level = モジュールトレース出力時刻取得方法] ※
26	[set pd_pth_trace_max = 通信トレース格納最大数]

(10)　セキュリティ

番号	形式
27	[set pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長] ※

(11)　Java

番号	形式
28	〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※
29	〔set pd_java_castoff = Y ｜ N〕

(12)　システムログファイル

番号	形式
30	〔set pd_log_dual = Y｜N〕※
31	〔set pd_log_remain_space_check = warn｜safe〕※
32	〔set pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕…〕
33	〔set pd_log_singleoperation = Y｜N〕※
34	〔set pd_log_rerun_reserved_file_open = Y｜N〕※
35	〔set pd_log_rerun_swap = Y｜N〕※
36	〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕※
37	〔set pd_log_unload_check = Y｜N〕※
38	〔set pd_log_max_data_size = ログ入出力バッファ長〕※
39	〔set pd_log_write_buff_count = ログ出力バッファ面数〕※
40	〔set pd_log_rec_leng = システムログファイルのレコード長〕※
41	〔set pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数〕※
42	〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕※

(13)　シンクポイントダンプファイル

番号	形式
43	〔set pd_spd_dual = Y｜N〕※
44	〔set pd_spd_assurance_msg = Y｜N〕※
45	〔set pd_spd_assurance_count = 有効保証世代数〕※
46	〔set pd_spd_reduced_mode = 縮退運転オプション〕※
47	〔set pd_spd_reserved_file_auto_open = Y｜N〕※
48	〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕※
49	〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕※

(14)　サーバ用ステータスファイル

番号	形式
50	set pd_sts_file_name_1 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"
	〔set pd_sts_file_name_2 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_3 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_4 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_5 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_6 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_7 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕

(15)　サーバ用ステータスファイル（障害発生時）

番号	形式
51	〔set pd_sts_initial_error = stop | continue | excontinue〕※
52	〔set pd_sts_singleoperation = stop | continue〕※
53	〔set pd_sts_last_active_file = "論理ファイル名"〕
54	〔set pd_sts_last_active_side = A | B〕

(16)　システムログファイルの構成

番号	形式
55	{{pdlogadfg -d sys -g ファイルグループ名〔ONL〕}}
56	{{pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名" 〔-b "システムログファイル名"〕}}

(17)　シンクポイントダンプファイルの構成

番号	形式
57	{{pdlogadfg -d spd -g ファイルグループ名〔ONL〕}}
58	{{pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕}}

(18)　プラグイン

番号	形式
59	{{〔 pdplgprm -n プラグイン名称 〔 -s 共用メモリサイズ〕〕}}

(19)　HiRDB External Data Access 機能

番号	形式

60	{
	〔pdhubopt -s 外部サーバ名 -f Hub 最適化情報定義ファイル名〕
	|
	〔pdhubopt -d 外部サーバ種別 〔-v 外部サーバのバージョン〕 -f Hub 最適化情報定義ファイル名〕
	}

注※

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

ただし，次に示すオペランドについては，サーバ共通定義ではなく，システム共通定義の同じオペランドの指定値が有効となります。

- pd_optimize_level
- pd_additional_optimize_level
- pd_cwaittime_wrn_pnt
- pd_uap_exerror_log_use

6.2　オペランドの説明

6.2.1　プロセスに関するオペランド

1）　pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕

～<符号なし整数>((0～2000))

常駐プロセス数：

フロントエンドサーバの常駐プロセス数を指定します。常駐プロセスとは，サーバ開始時からあらかじめ起動しておくプロセスのことです。

《利点》

フロントエンドサーバで並行処理できるトランザクションが使用するプロセスを，あらかじめシステム開始時に起動して常駐しておくことで，新たにトランザクションが入力されてもプロセス起動時間を削減できます。

ただし，HiRDB の開始に時間が掛かります。

《指定値の目安》

- フロントエンドサーバのサーバプロセスのプロセス固有領域とプロセサの実メモリ量から値を求めてください。サーバプロセスのプロセス固有領域については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。
- マルチフロントエンドサーバ構成で，更に pd_max_bes_process 又は pd_max_dic_process オペランドを指定している場合は，次に示す条件を満たすようにこのオペランドを指定してください。

  pd_process_count の値≦pd_max_bes_process 又は pd_max_dic_process の値
- このオペランドには，フロントエンドサーバの**最大起動プロセス数**の範囲内で値を指定します。

サーバの種類	最大起動プロセス数
マルチフロントエンドサーバの場合	pd_max_users の値＋ 1
マルチフロントエンドサーバでない場合	pd_max_users の値

《指定値のチューニング方法》

常駐プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- 常駐プロセス数はメモリと密接に関連しているため，必要以上に大きい値を指定すると，HiRDB を開始できなくなることがあります。
- 常駐プロセス数を超えたプロセスが必要になると，最大起動プロセス数までプロセスを動的に起動します。ただし，pd_max_server_process オペランドの指定値によっては，最大起動プロセス数分のプロセスを起動できないことがあります。

《オペランドの省略値》

このオペランドを省略（又は 0 を指定）すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

↑最大起動プロセス数÷2↑

サーバ開始時の常駐プロセス数：

HiRDB の開始処理時に常駐化する常駐プロセスの数を指定します。

通常，常駐プロセスの起動処理は HiRDB の開始処理中に行われます。このため，常駐プロセス数が多くなると，それに比例して HiRDB の開始処理時間が長くなります。目安として，100MIPS 程度のサーバマシンでプロセスを一つ起動するのに約 1 秒掛かります。

サーバ開始時の常駐プロセス数の指定有無による処理の違いを次に示します。

●サーバ開始時の常駐プロセス数を指定しない場合（pd_process_count = 500 と指定した場合）

すべての常駐プロセスの起動処理を HiRDB の開始処理中に実行します。常駐プロセス（この場合は 500 個）がすべて起動されないと，HiRDB を開始できません。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 500 秒掛かります。

●サーバ開始時の常駐プロセス数を指定する場合（pd_process_count = 500,50 と指定した場合）

一部の常駐プロセスの起動処理を HiRDB の開始処理中に実行し，残りの常駐プロセスの起動処理を HiRDB の開始処理終了後に実行します。一部の常駐プロセス（この場合は 50 個）が起動されれば，HiRDB を開始できます。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 50 秒掛かります。残りの常駐プロセス（この場合は 450 個）は，HiRDB の開始処理終了後に起動されます。

《利点》

HiRDB の開始処理時間を短縮できます。系切り替え機能を使用している場合など，HiRDB の開始処理時間をなるべく短くしたいときに適用します。

《指定値の目安》

HiRDB の開始処理終了直後に必要なプロセス数を指定します。

《注意事項》

サーバ開始時の常駐プロセス数を指定する場合は，クライアント環境定義の PDCWAITTIME オペランドの値を見直してください。

HiRDB の開始直後に，サーバ開始時の常駐プロセス数の値を超える UAP が実行されると，残りの常駐プロセスを起動した後にトランザクション処理が実行されます。したがって，クライアント環境定義の PDCWAITTIME オペランドの値が小さいと，一部の UAP がタイムアウトで処理できない場合があります。

PDCWAITTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

2) pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔 ～<符号なし整数>((0～1440))（単位：分）

HiRDB の非常駐サーバプロセスの停止処理をする間隔を分単位で指定します。この機能は，稼働中のサーバプロセス数が常駐プロセス数（pd_process_count オペランドの値）より多くなった場合に適用されます。この機能によって停止するサーバプロセス数は HiRDB が自動的に算出して決定します。

《利点》

業務（稼働中のサーバプロセス数）がピークのときに再利用できる非常駐サーバプロセスが増えるため，メモリなどのプロセス資源の利用効率が向上します。

《指定値の目安》

- 例えば，一日のうち 1 時間だけ業務のピーク時間があり，その時間内でピークとなる間隔が 2 分ぐらいの場合，このオペランドに 2 を指定します。
- ピーク時に同時実行されるサーバプロセス数が常駐プロセス数以下の場合，この機能を使用しても効果は期待できないため，このオペランドを省略します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《指定値のチューニング方法》

システムの稼働に関する統計情報をサーバ単位に1週間取得してください。「サービス実行中のサーバプロセス数（# OF PROCESS ON SERVICE)」の値から業務のピークを求めてください。そのピークが現在設定されている常駐プロセス数（pd_process_count オペランドの値）を超えている場合は，ピークごとの間隔を求めてその時間を設定します。

ただし，サーバマシンのメモリ，CPU などの資源に余裕がある場合は，不足している分のプロセス数を常駐プロセス数に加算する方（pd_process_count オペランドの値を大きくする方）がこのオペランドを指定するより性能が向上します。

《注意事項》

このオペランドを省略するか，又は0を指定した場合は10秒間隔でサービス待ち状態の非常駐サーバプロセスを調査し，サービス待ち状態の非常駐サーバプロセスがあれば，その非常駐サーバプロセスを停止します。

6.2.2　SQL の最適化に関するオペランド

3)　pd_optimize_level = SQL 最適化オプション〔，SQL 最適化オプション〕…

～＜識別子又は符号なし整数＞

SQL 最適化オプションを指定します。SQL 最適化オプションの機能を次に示します。SQL 最適化オプションの各機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

SQL 最適化オプションの機能	識別子	符号なし整数	S	P
ネストループジョイン強制	"FORCE_NEST_JOIN"	4	○	○
複数の SQL オブジェクト作成	"SELECT_APSL"	10	○	○
フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）	"FLTS_INC_DATA_BES"	16	×	○
ネストループジョイン優先	"PRIOR_NEST_JOIN"	32	○	○
フロータブルサーバ候補数の拡大	"FLTS_MAX_NUMBER"	64	×	○
OR の複数インデクス利用の優先	"PRIOR_OR_INDEXES"	128	○	○
自バックエンドサーバでのグループ化，ORDER BY，DISTINCT 集合関数処理	"SORT_DATA_BES"	256	×	○
AND の複数インデクス利用の抑止	"DETER_AND_INDEXES"	512	○	○
グループ分け高速化処理	"RAPID_GROUPING"	1024	○	○
フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）	"FLTS_ONLY_DATA_BES"	2048	×	○
データ収集用サーバの分離機能	"FLTS_SEPARATE_COLLECT_SVR"	2064	×	○
インデクス利用の抑止（テーブルスキャン強制）	"FORCE_TABLE_SCAN"	4096	○	○
複数インデクス利用の強制	"FORCE_PLURAL_INDEXES"	32768	○	○
更新 SQL の作業表作成抑止	"DETER_WORK_TABLE_FOR_UPDATE"	131072	○	○

SQL 最適化オプションの機能	識別子	符号なし整数	S	P
探索高速化条件の導出	"DERIVATIVE_COND"	262144	○	○
スカラ演算を含むキー条件の適用	"APPLY_ENHANCED_KEY_COND"	524288	○	○
プラグイン提供関数からの一括取得機能	"PICKUP_MULTIPLE_ROWS_PLUGIN"	1048576	○	○
導出表の条件繰り込み機能	"MOVE_UP_DERIVED_COND"	2097152	○	○

（凡例）

S：HiRDB/シングルサーバ

P：HiRDB/パラレルサーバ

○：指定が有効になります。

×：指定しても無効になります。

《オペランドの指定方法》

適用する SQL 最適化オプションを選択して，その識別子又は符号なし整数を指定します。識別子で指定する方法と，符号なし整数（計算値）で指定する方法の二つの方法がありますが，通常は識別子で指定してください。

- **識別子で指定する場合**

  「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用する場合は，次のように指定します。

  pd_optimize_level = "FORCE_NEST_JOIN","SELECT_APSL"

- **符号なし整数で指定する場合**

  「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用する場合は，次のように指定します。

  pd_optimize_level = 4,10

●HiRDB Version 5.0 以前からバージョンアップした場合

HiRDB Version 5.0 以前の合計値指定も有効です。最適化オプションを変更する必要がない場合は，HiRDB Version 6 以降にバージョンアップしたときにこのオペランドの指定値を変更する必要はありません。また，最適化オプションを追加する場合は次のように指定できます。

（例）

HiRDB Version 5.0 で「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用していて，今回「OR の複数インデクス利用の優先」を追加します。

pd_optimize_level = 14,128

ただし，この指定方法はどの機能を適用しているか分かりにくいため，識別子指定に変更することをお勧めします。

《オペランドの規則》

- 識別子と符号なし整数の混在指定はできません。

●識別子指定の場合

- SQL 最適化オプションを引用符（"）で囲んでください。
- ここで説明した SQL 最適化オプションを使用しない場合は"NONE"を指定してください。ただし，NONE と NONE 以外の識別子を指定した場合，NONE の指定は無効になります。
- 識別子は大文字でも小文字でも指定できます。

- 同じ識別子を 2 回以上指定しても，1 回指定した場合と同じになります。

●符号なし整数指定の場合

- ここで説明した SQL 最適化オプションを使用しない場合は 0 を指定してください。ただし，0 と 0 以外の符号なし整数を指定した場合，0 の指定は無効になります。
- 同じ符号なし整数を 2 回以上指定しても，1 回指定した場合と同じになります。

《指定値の目安》

指定値の目安については，マニュアル「HiRDB Version 8 UAP 開発ガイド」の「SQL 最適化オプション」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- **HiRDB/シングルサーバの場合**

  "PRIOR_NEST_JOIN"，"PRIOR_OR_INDEXES"，"DETER_AND_INDEXES"，"RAPID_GROUPING"，"DETER_WORK_TABLE_FOR_UPDATE"，"APPLY_ENHANCED_KEY_COND"

- **HiRDB/パラレルサーバの場合**

  "PRIOR_NEST_JOIN"，"PRIOR_OR_INDEXES"，"SORT_DATA_BES"，"DETER_AND_INDEXES"，"RAPID_GROUPING"，"DETER_WORK_TABLE_FOR_UPDATE"，"APPLY_ENHANCED_KEY_COND"

ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合は "SELECT_APSL"が仮定されます。

《注意事項》

- SQL 文中に SQL 最適化指定を指定している場合は，このオペランドの指定よりも SQL 最適化指定が優先されます。SQL 最適化指定については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。
- ストアドルーチン及びトリガ中の SQL 文（CREATE PROCEDURE, CREATE TYPE, ALTER PROCEDURE，CREATE TRIGGER，ALTER ROUTINE，又は ALTER TRIGGER）に SQL 最適化オプションを指定している場合は，このオペランドの指定よりも SQL 文中の SQL 最適化オプションが優先されます。

《ほかのオペランドとの関連》

- フロントエンドサーバ定義に pd_floatable_bes，又は pd_non_floatable_bes オペランドを指定している場合には，「フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）」及び「フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）」の指定は無効になります。
- pd_indexlock_mode オペランドに KEY を指定している場合，「更新 SQL の作業表作成抑止」の指定は無効になります。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSQLOPTLVL オペランドを指定してください。PDSQLOPTLVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

4)　pd_additional_optimize_level = SQL 拡張最適化オプション〔，SQL 拡張最適化オプション〕…～＜識別子又は符号なし整数＞

SQL 拡張最適化オプションを指定します。SQL 拡張最適化オプションの機能を次に示します。SQL 拡張最適化オプションの各機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

SQL 拡張最適化オプションの機能	識別子	符号なし整数
コストベース最適化モード 2 の適用	"COST_BASE_2"	1
ハッシュジョイン，副問合せのハッシュ実行※1	"APPLY_HASH_JOIN"	2
値式に対する結合条件適用機能	"APPLY_JOIN_COND_FOR_VALUE_EXP"	32
ジョインを含む SQL 文の外部サーバ実行の抑止※1※2	"DETER_JOIN_SQL"	67108864
直積を含む SQL 文の外部サーバ実行の強制※1※2	"FORCE_CROSS_JOIN_SQL"	134217728
無条件に生成する，外部サーバで実行できる探索高速化条件の導出の抑止※1※2	"DETER_FSVR_DERIVATIVE_COND"	1073741824

注※1　これらの項目は「コストベース最適化モード 2 の適用」を指定した場合に有効になります。

注※2　これらの項目は外部表を検索する場合に有効になります。それ以外の場合は無効になります。

《オペランドの指定方法》

適用する SQL 拡張最適化オプションを選択して，その識別子又は符号なし整数を指定します。通常は識別子で指定してください。

- **識別子で指定する場合**

  「コストベース最適化モード 2 の適用」及び「ハッシュジョイン，副問合せのハッシュ実行」を適用する場合は，次のように指定します。

  pd_additional_optimize_level = "COST_BASE_2","APPLY_HASH_JOIN"

- **符号なし整数で指定する場合**

  「コストベース最適化モード 2 の適用」及び「ハッシュジョイン，副問合せのハッシュ実行」を適用する場合は，次のように指定します。

  pd_additional_optimize_level = 1,2

《オペランドの規則》

- 識別子と符号なし整数の混在指定はできません。

●識別子指定の場合

- SQL 拡張最適化オプションを引用符（"）で囲んでください。
- ここで説明した SQL 拡張最適化オプションを使用しない場合は"NONE"を指定してください。ただし，NONE と NONE 以外の識別子を指定した場合，NONE の指定は無効になります。
- 識別子は大文字でも小文字でも指定できます。
- 同じ識別子を 2 回以上指定しても，1 回指定した場合と同じになります。

●符号なし整数指定の場合

- ここで説明した SQL 拡張最適化オプションを使用しない場合は 0 を指定してください。ただし，0 と 0 以外の符号なし整数を指定した場合，0 の指定は無効になります。
- 同じ符号なし整数を 2 回以上指定しても，1 回指定した場合と同じになります。

《指定値の目安》

指定値の目安については，マニュアル「HiRDB Version 8 UAP 開発ガイド」の「SQL 拡張最適化オプション」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，COST_BASE_2 が仮定されます。

《注意事項》

- SQL 文中に SQL 最適化指定を指定している場合は，このオペランドの指定よりも SQL 最適化指定が優先されます。SQL 最適化指定については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。
- ストアドルーチン及びトリガ中の SQL 文（CREATE PROCEDURE，CREATE TYPE，CREATE TRIGGER，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER）に SQL 拡張最適化オプションを指定している場合は，このオペランドの指定よりも SQL 文中の SQL 拡張最適化オプションが優先されます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDADDITIONALOPTLVL オペランドを指定してください。
PDADDITIONALOPTLVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

5）pd_floatable_bes = "バックエンドサーバ名"〔，"バックエンドサーバ名"〕…
～<識別子>((1～8 文字))

フロータブルサーバとして使用するバックエンドサーバを指定します。

ただし，指定したバックエンドサーバをすべてフロータブルサーバとして使用するとは限りません。

通常はデータ取り出しに使用するサーバは，フロータブルサーバとして使用しませんが，このオペランドを指定した場合はデータ取り出しに使用するサーバもフロータブルサーバの候補とします。

《オペランドの規則》

- 定義されていないバックエンドサーバ名を指定した場合は無視します。
- すべてが定義されていないバックエンドサーバ名の場合は，このオペランドを無視します。

《ほかのオペランドとの関連》

- pd_non_floatable_bes オペランドを同時に指定している場合，pd_non_floatable_bes オペランドを無効とします。
- pd_floatable_bes オペランドを指定すると，pd_optimize_level オペランドの「フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）」は適用されません。
- pd_floatable_bes オペランドを指定すると，pd_optimize_level オペランドの「フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）」は適用されません。

6）pd_non_floatable_bes = "バックエンドサーバ名"〔，"バックエンドサーバ名"〕…
～<識別子>((1～8 文字))

フロータブルサーバとして使用しないバックエンドサーバを指定します。

すべてのバックエンドサーバを指定している場合は，このオペランドを無効とします。

《オペランドの規則》

定義されていないバックエンドサーバ名を指定した場合は無視します。

《ほかのオペランドとの関連》

- pd_floatable_bes オペランドを同時に指定している場合，このオペランドを無効とします。
- pd_non_floatable_bes オペランドを指定すると，pd_optimize_level オペランドの「フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）」は適用されません。
- pd_non_floatable_bes オペランドを指定すると，pd_optimize_level オペランドの「フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）」は適用されません。

6.2.3　システム監視に関するオペランド

7）pd_watch_pc_client_time = クライアントからの要求間隔監視時間

～<符号なし整数>((0～65535))（単位：秒）

Windows 版の HiRDB クライアントからの要求に対する応答を HiRDB サーバが返してから，次に HiRDB クライアントからの要求があるまでの最大待ち時間（サーバ側の最大待ち時間）を秒単位で指定します。

指定した時間内に HiRDB クライアントから全く要求がない場合には，クライアントに異常が発生したものとして，強制的にサーバとクライアントとの接続を切り離します。この時に HiRDB クライアントに対しては，接続の切り離しについての通知はしません。

時間監視の対象となるのは，CONNECT から DISCONNECT までの間で，非トランザクション状態（SQL の実行を開始してから COMMIT 又は ROLLBACK までの間以外）の間です。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，3600 が仮定されます。

《注意事項》

- 0 を指定した場合には，HiRDB クライアントからの次の要求を無限に待ち続けます。
- このオペランドの指定値を小さい値（1～600 程度）にした場合には，HiRDB クライアントが，SQL 実行時にサーバダウンを検知するなど，不当に異常終了することがあります。
- UNIX 版の HiRDB クライアント（Linux 版の HiRDB クライアントも含む）の場合は，このオペランドの指定に関係なく時間を監視しません。UNIX 版の HiRDB クライアントに対して時間監視をする場合は，HiRDB クライアントのクライアント環境定義 PDSWATCHTIME を指定してください。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSWATCHTIME オペランドを指定してください。

PDSWATCHTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

8）pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0，2～100000))

UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。

この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。

このオペランドでは，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。シンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，対象トランザクションを強制的に中断してロールバックをします。これを**シンクポイントダンプ有効化のスキップ回数監視機能**といいます。

シンクポイントダンプ有効化のスキップ回数監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

このオペランドを指定すると，UAP の無限ループなどを防止できます。

《指定値の目安》

通常は 0 を指定してください。0 を指定すると，スキップ回数の上限値を HiRDB が計算します。0 を指定して不都合が発生した場合，又は KFPS02101-I メッセージが出力された場合に，このオペランドの値を変更してください。そのときの指定値の目安については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，シンクポイントダンプ有効化のスキップ回数監視機能を使用しません。

《注意事項》

- このオペランドを指定すると，ログレスモードで実行している UAP も監視対象になります。ログレスモードで実行している UAP の処理が中断すると，データベースを自動回復できないため RD エリアを障害閉塞します。このため，上限値の設定に際しては，ログレスモードで実行する UAP のトランザクション処理中に，該当するサーバ内のほかのトランザクションから出力されるシステムログ量も考慮に入れてください。
- pdload，pdmod，pdrorg，pdexp，pddbst，pdgetcst，pdrbal，pdvrup，及び pdmemdb コマンドはこの機能の監視対象外になります。

6.2.4　SQL 実行時間警告出力機能に関するオペランド

9）　pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜SQL 実行時間警告情報の出力条件（時間指定）

SQL 実行時間警告出力機能を使用する場合にこのオペランドを指定します。このオペランドの指定方法は次に示す二つの方法があります。

- 比率で指定する方法
- 時間で指定する方法

SQL 実行時間警告出力機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

SQL 実行時間警告情報の出力条件（比率指定）：～<符号なし整数>((0～99))又は<符号なし 10 進数>((0～99.999999))（単位：%）

SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）に対する比率で指定します。SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間がこのオペランドの指定値によって設定される時間以上になった場合，次に示す警告情報を出力します。これを SQL 実行時間警告出力機能といいます。

- SQL 実行時間警告情報ファイル
- 警告メッセージ（KFPA20009-W）

《オペランドの指定方法》

PDCWAITTIME オペランドの値に対する比率（%単位）で指定します。例えば，PDCWAITTIME オペランドに 100（秒）を指定し，このオペランドに 90（%）を指定すると，SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間が 90 秒以上 100 秒未満の場合に警告情報が出力されます。

（例）

PDCWAITTIME = 100

pd_cwaittime_wrn_pnt = 90

SQL 実行時間警告情報の出力条件（時間指定）：～<符号なし 10 進数>((0～65534.999999))sec（単位：秒）

SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を時間で指定します。SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間がこのオペランドの指定値によって設定される時間以上になった場合，次に示す警告情報を出力します。これを SQL 実行時間警告出力機能といいます。

- SQL 実行時間警告情報ファイル
- 警告メッセージ（KFPA20009-W）

《オペランドの指定方法》

出力契機とする時間を秒単位で指定します（小数点以下 6 けたまで指定できます）。指定値に「sec」を付けてください。

（例）

pd_cwaittime_wrn_pnt = 0.001sec

これ以降の説明は，比率指定及び時間指定の両方共通の説明です。

《オペランドの規則》

このオペランドに 0 又は 0sec を指定すると，警告情報を出力しません（SQL 実行時間警告出力機能を使用しません）。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- SQL 実行時間警告情報の出力条件（比率指定）：0
- SQL 実行時間警告情報の出力条件（時間指定）：0sec

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDCWAITTIMEWRNPNT オペランドを指定してください。
PDCWAITTIMEWRNPNT オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_cwaittime_report_dir
- pd_cwaittime_report_size

6.2.5　拡張 SQL エラー情報出力機能に関するオペランド

10）　pd_uap_exerror_log_use = YES | NO

拡張 SQL エラー情報出力機能を使用するかどうかを指定します。拡張 SQL エラー情報出力機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

YES：

拡張 SQL エラー情報出力機能を使用します。エラーログファイル及び SQL エラーレポートファイルに SQL エラー情報を出力します。

NO：

拡張 SQL エラー情報出力機能を使用しません。

《オペランドの省略値》

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が有効となります。システム共通定義の同じオペランドも省略すると，NO が仮定されます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDUAPEXERLOGUSE オペランドを指定してください。このオペランドと，クライアント環境定義 PDUAPEXERLOGUSE の両方を指定した場合は，クライアント環境定義 PDUAPEXERLOGUSE が優先されます。

PDUAPEXERLOGUSE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

6.2.6　排他制御に関するオペランド

11）　pd_fes_lck_pool_size = フロントエンドサーバの排他制御用プールサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～2000000))
- 64 ビットモードの場合：((1～2000000000))

フロントエンドサーバの排他制御で使用する共用メモリ領域（排他制御用プール）の大きさをキロバイト単位で指定します。

このオペランドの指定対象の領域は，排他資源管理テーブルとして使用されます。排他資源管理テーブル数，排他要求数，及び排他制御用プールサイズの間には，次の関係があります。

「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数

係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。

《指定値の目安》

- 32 ビットモードの場合，6 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 64 ビットモードの場合，4 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 次に示す計算式を基に，このオペランドの指定値を見積もってください。

  ↑↑（a＋b）÷pd_fes_lck_pool_partition の値↑÷ c ↑×pd_fes_lck_pool_partition の値　（キロバイト）

  a：フロントエンドサーバで同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については,「付録 D　排他資源数の見積もり」を参照してください。

  b：（pd_max_users の値＋ 3）×（pd_max_access_tables の値＋ 4）

c：32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入してください。

《指定値のチューニング方法》

統計解析ユティリティ（サーバごとのシステムの稼働に関する統計情報）で，フロントエンドサーバの「排他資源管理テーブル使用率（%OF USE LOCK TABLE)」を参照してください。使用率の最大値が 80%以上の場合は，これからのデータベース拡張に備えて，このオペランドの値を大きくすることをお勧めします。使用率の最大値が 10%以下の場合は，共用メモリを節約するために，このオペランドの値を小さくすることをお勧めします。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

・32 ビットモードの場合

{(pd_max_users の値＋ 3）×（pd_max_access_tables の値＋ 4)}　÷ 6

・64 ビットモードの場合

{(pd_max_users の値＋ 3）×（pd_max_access_tables の値＋ 4)}　÷ 4

《注意事項》

- このオペランドの指定値が小さ過ぎると，SQL がエラーリターンすることがあります。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは，pd_fes_lck_pool_partition オペランドと関連があります。

12）　pd_fes_lck_pool_partition = フロントエンドサーバの排他制御用プール分割数～<符号なし整数>((1～5000))

排他制御処理の分散をするときにフロントエンドサーバの排他制御で使用する排他制御用プールパーティション数を指定します。

排他制御処理の分散については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値のチューニング方法》

排他制御用プールパーティション数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

《注意事項》

- このオペランドの指定値を大きくし過ぎると，各排他制御用プールパーティションに割り当てられる共用メモリのサイズが小さくなるため，排他制御用プールパーティションが容量不足となり SQL がエラーリターンすることがあります。また，排他制御の処理に時間が掛かるため，システムのパフォーマンスが下がります。この場合，このオペランドの指定値を小さくしてください。
- 排他制御用プールサイズは 1 キロバイト以上必要であるため，pd_lck_pool_size の値よりも大きな値を指定した場合，このオペランドには pd_lck_pool_size の値が仮定されます。また，このとき，KFPS00421-W メッセージが出力されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_fes_lck_pool_size
- pd_lck_deadlock_check_interval

13)　pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数～<符号なし整数>((0～2147483647))

排他制御用プールで使用するハッシュテーブルのエントリ数を指定します。HiRDB はここで指定された値に従ってユニットコントローラ用の共用メモリに排他制御用プールを確保します。

《指定値の目安》

通常はこのオペランドを省略してください。

次に示す条件に該当する場合は指定値を検討してください。

- バージョン 06-02 以降にバージョンアップする場合に共用メモリサイズをできるだけ変更したくないときは 11261 を指定してください。この場合，バージョンアップ前と同数のハッシュエントリ数を確保するため，排他制御用プール中のハッシュテーブルサイズがバージョンアップ前と同じになります。
- このオペランドに推奨値より大きい値を指定すると，性能が向上することがあります。ただし，《推奨値の求め方》の変数 a より大きい値を指定しても，a を指定したときよりも性能が向上することはありません。

《オペランドの規則》

- このオペランド及びサーバ共通定義の pd_lck_hash_entry オペランドを省略するか，又はこのオペランドに 0 を指定すると，HiRDB はサーバごとに推奨値を計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 11261 が仮定されます）。推奨値については《推奨値の求め方》を参照してください。
- このオペランドに 0 でも素数でもない値を指定した場合，HiRDB はその指定値を超えない最大の素数が指定されたと仮定します。

《注意事項》

このオペランドの指定値が小さ過ぎると，ハッシュエントリ不足が発生して性能が低下することがあります。このオペランドを省略すれば，ハッシュエントリ不足及びハッシュエントリ不足による性能低下は発生しません。

《推奨値の求め方》

推奨値を次に示します。

推奨値＝ MAX（↑a÷10↑，11261）を超えない最大の素数

変数	変数の算出式	
a	pd_fes_lck_pool_size が省略されている場合	（b ＋ 3）×（pd_max_access_tables の値＋ 4）
	pd_fes_lck_pool_size が指定されている場合	pd_fes_lck_pool_size 値×c
b	マルチフロントエンドサーバの場合	pd_max_users の値＋ 1
	マルチフロントエンドサーバでない場合	pd_max_users の値
c	32 ビットモードの場合は 6，64 ビットモードの場合は 4	

6.2.7　バッファに関するオペランド

14）　pd_sql_object_cache_size = SQL オブジェクト用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((22～256000))
- 64 ビットモードの場合：((22～2000000))

SQL オブジェクトを格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- SQL オブジェクトは，ユーザごとにトランザクションが終了するまでバッファに保存されます。したがって，同時に実行するトランザクションの SQL オブジェクトがバッファに格納できる大きさが必要です。
- 静的 SQL の SQL オブジェクトはトランザクション終了後もバッファに空きがなくなるまで保存して，同一 UAP を実行する複数のユーザで共用することで，SQL 解析処理の削減を図ります。バッファを有効に利用するためには，使用頻度の高い UAP の SQL オブジェクトがバッファに常駐するようにバッファを確保してください。
- バッファ長の見積もり時は，まず UAP で発行する SQL 文の SQL オブジェクト長から UAP 実行に必要なバッファ長を見積もります。次に，同時に実行される UAP と同時実行ユーザ数を考慮して必要なバッファ長を算出してください。
- 1SQL 文の SQL オブジェクト長の見積もり方法については，「付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義とシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

（pd_max_users の値＋ 3）×22

《指定値のチューニング方法》

SQL オブジェクト用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

15）　pd_table_def_cache_size = 表定義情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((100～65535))
- 64 ビットモードの場合：((100～2000000))

一度使用した表と順序数生成子の定義情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。表と順序数生成子の定義情報は，SQL 文の前処理時に使用します。また，このバッファ内の定義情報は，LRU で管理されます。

《利点》

- 一度使用した表と順序数生成子の定義情報を共用メモリ内に格納して，次回使用するときは入出力なしで使用できます。
- 動的 SQL を多く使用する場合，性能が向上します。
- ディクショナリサーバとの通信回数を削減できます。

《指定値の目安》

- 使用頻度が高い表と順序数生成子の定義情報長の合計値を指定してください。

- 1 順序数生成子当たりの表定義情報バッファサイズは，8 キロバイトです。
- 1 表当たりの表定義情報用バッファ長の求め方については，「付録 C.4　表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- HiRDB/シングルサーバの場合

《↑ $\sqrt{\text{pd_max_users の値}+3}$ ↑×100》

- HiRDB/パラレルサーバの場合

《↑ $\sqrt{(\text{pd_max_bes_process 又は pd_max_dic_process の指定値の最大値})+3}$ ↑×100》

《指定値のチューニング方法》

表定義情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

16）pd_auth_cache_size = ユーザ権限情報用バッファ長

～<符号なし整数>((1～100))（単位：キロバイト）

ユーザ権限情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- ユーザ権限情報用バッファには CONNECT 権限，DBA 権限，及び監査権限の情報を格納します。このバッファに情報がない場合は HiRDB 接続時にディクショナリ表から情報を取得するため，応答時間が遅くなります。したがって，常時接続しているユーザ数分の情報を格納できるようにバッファ長を指定してください。
- 1 ユーザ当たりのユーザ権限情報を格納するのに 68 バイト必要になります。それを基に計算してください。

《指定値のチューニング方法》

ユーザ権限情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

17）pd_view_def_cache_size = ビュー解析情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((0～32000))
- 64 ビットモードの場合：((0～2000000))

ビュー解析情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《利点》

一度使用したビュー解析情報を共用メモリ内に格納して，次回使用するときは入出力なしで使用できます。

《指定値の目安》

使用頻度が高いビュー表のビュー解析情報用バッファ長の合計値を指定してください。ビュー解析情報用バッファ長の見積もりについては，「付録 C.3　ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

ビュー解析情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

《↑ $\sqrt{(\text{pd_max_users の値}+3)}$ ↑×8》

18)　pd_type_def_cache_size = ユーザ定義型情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((100～65536))
- 64 ビットモードの場合：((100～2000000))

このオペランドは，ユーザ定義型に関するオペランドです。ユーザ定義型を使用する場合は，このオペランドを指定することをお勧めします。

ユーザ定義型の情報を格納するバッファ（共用メモリ）の大きさを，キロバイト単位で指定します。ユーザ定義型の情報を使用すると，その情報がこのバッファ内に格納されます。この情報は SQL 文の前処理時に使用します。

《利点》

- このバッファにユーザ定義型の情報を格納すると，このユーザ定義型の情報を次回使用するときに，ディクショナリ表をアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- ディクショナリサーバとの通信回数が削減できます。
- 動的 SQL を多く使用する場合，このオペランドを指定すると性能が向上します。

《指定値の目安》

使用頻度が高い表に定義してあるユーザ定義型の定義情報長を合計します。一つ当たりのユーザ定義型の定義情報長は，次に示す計算式から求めてください。

↑ {((0.3 + 0.2× a + 0.1× b）＋3）÷4} ↑× 4

（単位：キロバイト）

a：ユーザ定義型の属性数

b：スーパタイプを継承したサブタイプ数

《指定値のチューニング方法》

ユーザ定義型情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

19)　pd_routine_def_cache_size = ルーチン定義情報用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((0，20～65536))
- 64 ビットモードの場合：((0，20～2000000))

次に示す定義情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。これらの定義情報は SQL 文の前処理時に使用します。

- プラグイン関数の定義情報
- システム定義スカラ関数の定義情報
- ルーチンの定義情報

《利点》

- このバッファに前記の定義情報を格納すると，これらの情報を次回使用するときに，ディクショナリ表をアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- ディクショナリサーバとの通信回数が削減できます。

《適用基準》

次に示す SQL を多く使用する場合に指定します。

- プラグインを使用する SQL
- システム定義スカラ関数を使用する SQL
- ルーチンを使用する SQL

《指定値の目安》

このオペランドの指定値の求め方については，「付録 C.6　ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

ルーチン定義情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，100 が仮定されます。

《注意事項》

このオペランドの指定値が全プラグイン提供関数の定義情報の合計値より小さい場合，プラグイン提供関数の定義情報はバッファ中に確保されません。

20）pd_registry_cache_size = レジストリ情報用バッファ長

～<符号なし整数>((0～65536))（単位：キロバイト）

このオペランドは，プラグインに関するオペランドです。レジストリ情報を使用するプラグインを使用する場合は，このオペランドを指定することをお勧めします。HiRDB Text Search Plug-in を使用する場合は，指定することをお勧めします。

レジストリ情報を格納するバッファ（共用メモリ）の大きさを，キロバイト単位で指定します。レジストリ情報を使用すると，レジストリ情報がバッファ内に格納されます。レジストリ情報は SQL 文の実行時に使用します。

《利点》

- このバッファにレジストリ情報を格納すると，レジストリ情報を次回使用するときに，レジストリをアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。

- ディクショナリサーバとの通信回数が削減できます。
- レジストリ情報を多く使用するプラグインを使用する場合，このオペランドを指定すると性能が向上します。

《指定値の目安》

レジストリ情報用バッファ長は，次に示す計算式から求めてください。

↑（0.3＋a）↑×b　　（単位：キロバイト）

a：レジストリキー長の平均長（単位：キロバイト）

レジストリキー値の平均長は次に示す SQL で求められます。

この SQL の結果の単位はバイトのため，キロバイトに変換してください。

```
SELECT AVG(KEY_LENGTH) FROM MASTER.SQL_REGISTRY
```

b：レジストリキーの登録数

レジストリキーの登録数は次に示す SQL で求められます。

```
SELECT COUNT(*) FROM MASTER.SQL_REGISTRY
```

《指定値のチューニング方法》

レジストリ情報用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10 が仮定されます。

6.2.8　RPC トレース情報に関するオペランド

21）　pd_rpc_trace = Y | N

RPC トレース情報を取得するかどうかを指定します。RPC トレース情報には，HiRDB の保守情報が出力されます。

通常，このオペランドは省略してください。

Y：RPC トレースを取得します。

N：RPC トレースを取得しません。

《注意事項》

このオペランドに Y を指定すると，通信性能が低下します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，N が仮定されます。

22）　pd_rpc_trace_name = "RPC トレースを取得するファイル名称"

～＜ 254 文字以内のパス名＞

RPC トレースファイルの名称を絶対パス名で指定します。RPC トレースファイルとして，指定したファイル名の後に 1，2，l（エル）を付けた三つのファイルが作成されます。

《注意事項》

このオペランドで指定したディレクトリ下に，最大「pd_rpc_trace_size の値× 2」の大きさのファイルが作成されます。ファイル容量に注意してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

%PDDIR%¥spool¥rpctr

23)　pd_rpc_trace_size = RPC トレースを取得するファイルの容量

～<符号なし整数>((1024～2147483648))（単位：バイト）

RPC トレースファイルの容量をバイト単位で指定します。

このオペランド指定値は，"RPC トレースファイル"1 及び"RPC トレースファイル"2 のそれぞれの容量となります。

《指定値の目安》

RPC トレースは，すべてのプロセス間の通信電文が一つのファイルに時系列に蓄積されます。その量がこの指定値を超えると，もう一方のファイルにトレースデータの出力を切り替えます。古いトレースは上書きされて消去されます。このため，指定値が小さいと十分なトレースを取得できなくなり，トラブルシュートが困難になることがあります。したがって，このオペランドには 1,000,000 以上の値を指定してください。

《注意事項》

このオペランドで指定した値は RPC トレースファイル l（エル）の容量に影響を与えません。RPC トレースファイル l（エル）の容量は 0 バイト固定です。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，4096 が仮定されます。

6.2.9　トラブルシュート情報に関するオペランド

24)　pd_module_trace_max = モジュールトレース格納最大数

～<符号なし整数>((126～16383))

HiRDB のプロセスは，実行した関数やマクロの履歴をプロセス固有メモリ中に記録しています。この履歴をモジュールトレースといい，このオペランドではモジュールトレースのレコード数を指定します。この履歴の内容は core ファイルに取り込まれ，プロセス異常終了時に出力されます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，126 が仮定されます。

《注意事項》

各プロセスでは，以下のサイズのプロセス固有メモリが確保されます。

32 ビットモードの場合：64 ＋ 48×pd_module_trace_max オペランドの値（バイト）

64 ビットモードの場合：64 ＋ 64×pd_module_trace_max オペランドの値（バイト）

25)　pd_module_trace_timer_level = 0 | 10 | 20

モジュールトレースに出力する時刻の取得方法を指定します。このオペランドの指定値の意味を以下に示します。

指定値	時刻取得方法
0	すべてのモジュールトレース出力箇所で，秒単位で取得します。
10	すべてのモジュールトレース出力箇所のうち，入出力処理前後など性能調査で問題となりやすい箇所だけマイクロ秒単位で取得し，そのほかの箇所では秒単位で取得します。
20	すべてのモジュールトレース出力箇所で，マイクロ秒単位で取得します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

このオペランドに 0 以外の値を指定すると，時刻をマイクロ秒で取得する関数を発行するため，性能が低下することがあります。

26)　pd_pth_trace_max = 通信トレース格納最大数

～<符号なし整数>((1024～8388608))

トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。

《指定値の目安》

通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。

《注意事項》

通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

6.2.10　セキュリティに関するオペランド

27)　pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長

～<符号なし整数>((1～2000000))（単位：キロバイト）

セキュリティ監査機能の情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

セキュリティ監査情報用バッファ長は，次に示す計算式から求めてください。

↑0.3 ＋ a ×0.25↑　　（単位：キロバイト）

a ：セキュリティ監査機能の監査証跡に指定した絞り込み条件のオブジェクト数

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示すサイズのバッファ長が HiRDB の開始時に取得されます。このメモリが確保できない場合，HiRDB は開始できないため注意してください。

↑0.3 ＋ MAX｛(a ＋ 100)，(a×1.2)｝×0.25↑

（単位：キロバイト）

a ：セキュリティ監査機能の監査証跡に指定した絞り込み条件のオブジェクト数

《注意事項》

このオペランドで指定した大きさのメモリが確保できない場合，HiRDB は開始できません。

6.2.11　Java に関するオペランド

Java ストアドプロシジャ又は Java ストアドファンクションを使用する場合に，Java に関するオペランドを指定します。Java ストアドプロシジャ及び Java ストアドファンクションについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

28）　pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"

～<パス名>

Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合，そのファイル名を絶対パス名で指定します。

《指定値の目安》

このオペランドで指定するファイルの容量が非常に大きくなるため，通常はこのオペランドを指定しないでください。Java ストアドプロシジャ又は Java ストアドファンクションのデバッグ時に指定することをお勧めします。なお，このオペランドで指定するファイルの容量制限はありません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，Java 仮想マシンでの標準出力又は標準エラー出力が無視されます。

《注意事項》

複数プロセスからの同時書き込みがあった場合は，その出力内容を保証しません。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

29）　pd_java_castoff = Y | N

HiRDB のフロントエンドサーバプロセスで，Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。

Y：Java 仮想マシンを起動した場合，UAP の切り離し時にサーバプロセスを終了します。

N：Java 仮想マシンを起動してもサーバプロセスが終了しません。

省略した場合は，サーバ共通定義の pd_java_castoff オペランドが解釈値となります。

《指定値のチューニング方法》

Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。

Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に，pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。

#	懸案
1	Java 仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。
2	Java 仮想マシンを使用しないコネクションで実行する SQL でも，別のコネクションで Java 仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多い SQL を実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。 なお，Java 仮想マシンの機能については Java 仮想マシンのドキュメントのオプションを参照してください。

《注意事項》

Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- pd_process_count

6.2.12　システムログファイルに関するオペランド

30）pd_log_dual = Y | N

システムログファイルを二重化するかどうかを指定します。

Y：システムログファイルを二重化します。

N：システムログファイルを二重化しません。

《利点》

システムログファイルを二重化すると，HiRDB はA系及びB系の両方に同じシステムログを取得します。取得したシステムログを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシステムログを読み込めるため，システムの信頼性を向上できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

システムログファイルを二重化する場合は pdlogadpf オペランドで B 系のシステムログファイル名を指定してください。

31）pd_log_remain_space_check = warn | safe

システムログファイルの空き率が**警告値**未満になったときの HiRDB の処理方式を指定します。この機能を**システムログファイルの空き容量監視機能**といいます。システムログファイルの空き容量監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

warn：

システムログファイルの空き率が警告値未満になった場合，警告メッセージ KFPS01162-W を出力します。

safe：

システムログファイルの空き率が警告値未満になった場合，新規トランザクションのスケジューリングを抑止して，サーバ内の全トランザクションを強制終了します。このとき，KFPS01160-E メッセージを出力します。

《指定値の目安》

システムログファイルの空き容量不足によるユニットの異常終了の可能性を低くできるため，safe の指定を推奨します。ただし，safe を指定すると，システムログファイルの空き容量が不足したときに，サーバ内の全トランザクションが強制終了されます。このため，システムログファイルの設計をより正確に行う必要があります。システムログファイルの設計については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，warn が仮定されます。

32）　pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕…

～<パス名>((1～136 文字))

システムログの**自動ログアンロード機能**を使用する場合に，アンロードログファイルの出力先ディレクトリを絶対パス名で指定します。アンロードログファイルを HiRDB ファイルシステム領域に作成する場合は，その HiRDB ファイルシステム領域名を指定してください。なお，**このオペランドに指定するディレクトリ又は HiRDB ファイルシステム領域は，HiRDB の開始前に作成しておいてください。**

また，このオペランドには，サーバごとに異なるディレクトリ又は HiRDB ファイルシステム領域を指定してください。

自動ログアンロード機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値の目安》

空きディスク容量を確認してからディレクトリを指定してください。作成されるアンロードログファイルによって，ディスク容量が圧迫される原因になります。

また，ディスク容量の不足によって，指定したディレクトリにアンロードログファイルが作成できない場合，自動ログアンロード機能が停止します。このようなことが発生する可能性がある場合は，ディレクトリを複数指定することをお勧めします。

ただし，ディレクトリは一つだけの方がデータベース回復時の運用（回復に必要なアンロードログファイルの選別）が若干簡単になります。

ディレクトリを複数指定する場合の補足事項を次に示します。

- ディスク障害などに備えて，異なるパーティションのディレクトリを指定することをお勧めします。
- ディスク容量満杯又はディスク障害などで一つのディレクトリ下にアンロードログファイルが作成できない場合，ほかのディレクトリ下にアンロードログファイルを作成します。このとき，HiRDB はこのオペランドに指定した順番にディレクトリを使用します。

《オペランドの規則》

- ディレクトリは最大 128 個指定できます。
- 複数のディレクトリを指定した場合，同じパス名を指定できません。

《注意事項》

次の場合は，自動ログアンロード機能を使用できません。

- pd_log_unload_check オペランドに N を指定している場合

なお，マルチ HiRDB を利用している場合は，HiRDB ごとに異なるディレクトリを作成してください。同じディレクトリを指定すると，どのアンロードログファイルがどの HiRDB に対応しているか分からなくなる可能性があります。

33） pd_log_singleoperation = Y | N

このオペランドは，システムログファイルを二重化している場合に指定してください。二重化していない場合は指定する必要はありません。

システムログファイルの片系運転をするかどうかを指定します。システムログファイルに障害が発生して，両系ともに使用できるシステムログファイルがない場合でも，HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了しないで正常な片方の系だけで処理を続行します。これを**システムログファイルの片系運転**といいます。

また，両方のシステムログファイルで処理を続行すること(通常の処理形態)を**システムログファイルの両系運転**といいます。

Y：システムログファイルの片系運転をします。

N：システムログファイルの片系運転をしません。常に両系運転とします。

《前提条件》

pd_log_dual オペランドに Y を指定している必要があります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

34） pd_log_rerun_reserved_file_open = Y | N

システムログファイルの自動オープンをするかどうかを指定します。

ユニットを再開始するときに，「上書きできる状態」のシステムログファイルがない場合，予約のファイルがあれば HiRDB が予約のファイルをオープンして「上書きできる状態」にし，処理を続行します。これを**システムログファイルの自動オープン**といいます。

予約ファイルを使用するのは，次の場合です。

- 再開始後，最初のシンクポイントダンプを取得するまでの間
- オープンされたファイルグループすべてが「上書きできない状態」の場合

Y：システムログファイルの自動オープンをします（予約ファイルをオープンして使用します）。

N：システムログファイルの自動オープンをしません（予約ファイルを使用しません）。

《利点》

このオペランドに Y を指定すると，ユニットの再開始時に「スワップ先にできる状態」のファイルがなくても，予約のファイルがあればユニットを再開始できます。

ただし，「アンロード待ち状態」のファイルがあるときは，ユニットを停止します。「アンロード待ち状態」のファイルのアンロードを実行した後，再度ユニットを開始してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

35） pd_log_rerun_swap = Y | N

ユニットの再開始時に，システムログファイルをスワップするかどうかを指定します。

Y：システムログファイルをスワップします。

N：システムログファイルをスワップしません。

《利点》

このオペランドにYを指定すると，再開始前後でシステムログファイルを物理的に分けられます。このため，再開始前のシステムログファイルをサーバ稼働中に使用できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Nが仮定されます。

36）pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間～<符号なし整数>((1～32580))（単位：秒）

システムログファイルのスワップが完了するまでの待ち時間を秒単位で指定します。指定した時間を過ぎてもシステムログファイルのスワップが完了しない場合は，ユニットが異常終了します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。マシンの性能が低いなどの理由によってシステムログファイルのスワップに時間が掛かる場合は，このオペランドの値を大きくしてください。また，ディスク障害などによってシステムログファイルのスワップに遅延が発生した場合に，より短い時間で異常を検知し，ユニットを異常終了させたいときは，このオペランドの値を小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，180が仮定されます。

37）pd_log_unload_check = Y｜N

システムログファイルのアンロード状態を，HiRDBがチェックするかどうかを指定します。

Y：

アンロード状態をチェックします（通常の運用となります）。

N：

アンロード状態をチェックしません。アンロードの状態に関係なく，次に示す条件がすべて満たされれば，システムログファイルを「スワップ先にできる状態」にします。

- 上書きできる状態
- 抽出完了状態（HiRDB Datareplicator）

このときシステムログファイルの運用方法が，アンロード状態のチェックを解除する運用になります。アンロード状態のチェックを解除する運用については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

Nを指定したときの利点を次に示します。

- システムログファイルのアンロード操作がなくなるため，運用方法が簡単になります。
- アンロードログファイルを保管するためのファイル容量が不要になります。

《指定値の目安》

データベースを回復するときにシステムログを使用しない場合（バックアップ取得時点に回復できればよい場合）にNを指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Yが仮定されます。

《注意事項》

N を指定したときの注意事項を次に示します。

- バックアップ取得時点にしかデータベースを回復できません。
- データベースの回復にシステムログが必要なのにこの運用をした場合はデータベースの回復手段がなくなります。

38)　pd_log_max_data_size = ログ入出力バッファ長

～<符号なし整数>((32000～523000))（単位：バイト）

システムログの入出力に使用するバッファの大きさをバイト単位で指定します。

《指定値の目安》

条件式 1 を満たす値を指定してください。pd_rpl_reflect_mode オペランドに uap を指定している場合，又は回復不要 FES を使用している場合は，条件式 1 と条件式 2 の両方を満たす値を指定してください。指定値を最適化する場合は，指定値のチューニング方法に従って指定値を変更してください。

条件式 1：ログ入出力バッファ長≧a

a：72×（一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数）＋ 1344

条件式 2：ログ入出力バッファ長≧b

b：(一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数＋ 1）×128 ＋ 64

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）の次に示す情報を調べて，このオペランドの指定値を変更してください。

- 入出力待ちバッファ面数（# OF BUFFER FOR WAIT I/O）

  入出力待ちバッファ面数の平均値が 100 を大きく超える場合は，このオペランドの値を大きくして，平均値が 100 に近付くようにしてください。

- カレントバッファなしによる待ち回数（# OF WAIT THREAD）

  カレントバッファなしによる待ち回数が 0 以外の値になる場合は，このオペランドの値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，400000 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 32000 が仮定されます）。

《ほかのオペランドとの関連》

pd_log_write_buff_count オペランドとともに，ログ入出力バッファ長を決定してください。

39)　pd_log_write_buff_count = ログ出力バッファ面数

～<符号なし整数>((3～65000))

システムログの出力に使用するバッファの面数を指定します。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）のカレントバッファなしによる待ち回数（# OF WAIT THREAD）を調べて，待ち回数が多ければ，スループットが向上するよう指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 3 が仮定されます）。

《ほかのオペランドとの関連》

pd_log_max_data_size オペランドとともに，ログ出力バッファ面数を決定してください。

40）pd_log_rec_leng = システムログファイルのレコード長

～<符号なし整数>((1024，2048，4096))（単位：バイト）

システムログファイルのレコード長を指定します。1024，2048，4096 のどれかを指定します。

レコード長は pdloginit コマンドの-l オプションで設定します。-l オプションで設定するレコード長をこのオペランドに指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，4096 が仮定されます。

《注意事項》

- pdloginit コマンドの-l オプションで設定するレコード長と異なるレコード長をこのオペランドに指定すると，そのシステムログファイルはオープンできません。
- システムログファイルのレコード長を変更する方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

41）pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数

～<符号なし整数>((0～256))

ロールバック処理でシステムログの入力に使用するバッファの面数を指定します。このオペランドに 0 を指定した場合，HiRDB がロールバック用ログ入力バッファ面数を決定します。

《指定値のチューニング方法》

このオペランドに 0 を指定してください。0 を指定してメモリ不足が発生する場合は，このオペランドの指定を省略してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

ユニット内サーバ数×2

42）pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕

～<符号なし整数>((0～104857600))《0，0》（単位：レコード）

このオペランドは，システムログファイルの自動拡張機能を使用する場合に指定します。

一つのシステムログファイルが拡張契機 1 回当たりに拡張するサイズ，及び拡張できるファイルサイズの上限をレコード数で指定します。

1 回当たりに拡張するサイズを省略するか，又は 0 を指定した場合，システムログファイルの自動拡張を行いません。拡張上限サイズを省略するか，又は 0 を指定した場合，ファイルシステム領域のあるディスクが満杯になるか，システムログファイルの容量が上限に達するまでシステムログファイルの自動拡張を行います。また，1 回当たりに拡張するサイズに拡張上限サイズより大きい値を指定した場合，拡張上限サイズに指定した値まで自動拡張を行います。

システムログファイルの自動拡張機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《前提条件》

pd_large_file_use オペランドに Y を指定しているか，又は指定を省略している必要があります。

《指定値の目安》

- 1 回当たりに拡張するサイズには，pdloginit コマンドの-n オプションでシステムログファイルを作成した際に指定したレコード数を基に指定します。全システムログファイルのレコード数の平均値の 1/10 を計算し，指定してください。
- 通常，拡張上限サイズは省略してください。

《指定値のチューニング方法》

システムログの出力量が自動拡張で拡張するサイズを超えると，システムログファイルが満杯になってユニットダウンする場合があります。その場合は，1 回当たりに拡張するサイズを大きくしてください。また，拡張処理に時間が掛かり，トランザクション性能に影響を及ぼす場合は，1 回当たりに拡張するサイズを小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

6.2.13　シンクポイントダンプファイルに関するオペランド

43)　pd_spd_dual = Y | N

シンクポイントダンプファイルを二重化するかどうかを指定します。

Y：シンクポイントダンプファイルを二重化します。

N：シンクポイントダンプファイルを二重化しません。

《利点》

シンクポイントダンプファイルを二重化すると，HiRDB はA系及びB系の両方に同じシンクポイントダンプを取得します。取得したシンクポイントダンプを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシンクポイントダンプを読み込めるため，システムの信頼性を向上できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

シンクポイントダンプファイルを二重化する場合は，pdlogadpf オペランドで B 系のシンクポイントダンプファイル名を指定してください。

44)　pd_spd_assurance_msg = Y | N

シンクポイントダンプが有効化されたとき，メッセージ KFPS02183-I を出力するかどうかを指定します。

Y：出力します。

N：出力しません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Y が仮定されます。

45)　pd_spd_assurance_count = 有効保証世代数

～<符号なし整数>((1～2))

システム回復時のシンクポイントダンプファイルの入力障害などに備え，HiRDB 稼働中に保存しておくシステムログファイルの範囲を，シンクポイントダンプファイルの世代数によって指定します。この世代数のことを**有効保証世代数**といいます。ここで指定した世代数分のシンクポイントダンプファイルは，上書きできない状態になります。

《利点》

有効保証世代数を 2 にすると，最新世代のシンクポイントダンプファイルに障害が発生しても，1 世代前のシンクポイントからシステムを回復できるため，信頼性が向上します。

《指定値の目安》

- 信頼性を向上させる場合は有効保証世代数を 2 にしてください。ただし，有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます（二つになります）。
- シンクポイントダンプファイルを二重化して信頼性を向上している場合は，このオペランドを省略するか，又は 1 を指定することをお勧めします。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

《注意事項》

- シンクポイントダンプファイルの必要最低数は**有効保証世代数＋ 1** となります。
- 有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます。上書きできない状態のシンクポイントダンプファイルに対応するシステムログファイルは上書きできない状態になります。したがって，有効保証世代数を 2 にすると，上書きできない状態のシステムログファイルが増えるため，スワップ先にできる状態のシステムログファイル数が不足することがあります。これを防ぐためにはシステムログファイルの容量を考慮してください。

46）pd_spd_reduced_mode = 縮退運転オプション

～＜符号なし整数＞((0～2))

シンクポイントダンプファイルの縮退運転をするかどうかを指定します。

縮退運転とは，HiRDB 稼働中又は再開始処理中のファイル障害などによって，シンクポイントダンプファイルの数が運用に必要なファイル数（**有効保証世代数**※＋ 1）以下になった場合でも，最低二つのファイルが使用できれば処理を続行する機能です。

注※　pd_spd_assurance_count オペランドの指定値です。

0：縮退運転を使用しません。

1：縮退運転を使用します。

2：縮退運転を使用し，縮退運転でシンクポイントダンプ取得の契機ごとに警告メッセージを出力します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

47）pd_spd_reserved_file_auto_open = Y｜N

シンクポイントダンプファイルの自動オープンをするかどうかを指定します。

シンクポイントダンプファイルに障害が発生して，運用に必要なファイル数（**有効保証世代数**※＋ 1）を下回った場合，予約のファイルがあれば HiRDB が予約のファイルをオープンして上書きできる状態にし，処理を続行します。これを**シンクポイントダンプファイルの自動オープン**といいます。

注※　pd_spd_assurance_count オペランドの指定値です。

Y：

シンクポイントダンプファイルの自動オープンをします。運用に必要なファイル数（有効保証世代数＋1）を下回った場合，予約ファイルを自動的にオープンします。

N：

シンクポイントダンプファイルの自動オープンをしません。運用に必要なファイル数（有効保証世代数＋1）を下回っても，予約ファイルを自動的にオープンしません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

このオペランドの指定は pd_spd_reduced_mode オペランドより優先されます。

48）pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長

～<符号なし整数>((32000～4000000))（単位：バイト）

シンクポイントダンプをシンクポイントダンプファイルに入出力するときに使用するバッファ（共用メモリ）の大きさをバイト単位で指定します。

ここで指定した値によってシンクポイントダンプファイルの入出力回数を制御します。

《指定値の目安》

- 通常はこのオペランドを指定する必要はありません。
- 指定値を大きくするほど，シンクポイントダンプファイルに対する入出力回数が少なくなります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，32768 が仮定されます。

49）pd_log_sdinterval = システムログ出力量〔，経過時間〕

シンクポイントダンプの取得間隔を指定します。次に示す条件で指定できます。

- システムログの出力量
- 前回シンクポイントダンプを取得してからの経過時間

システムログ出力量：～<符号なし整数>((100～100000))（単位：ログブロック数）

シンクポイントダンプの取得間隔を指定します。ここで指定したログブロック数分のシステムログが出力されるたびに，シンクポイントダンプを取得します。

経過時間：～<符号なし整数>((0,10～1440))（単位：分）

シンクポイントダンプの取得間隔を指定します。前回シンクポイントダンプを取得してから，ここで指定した経過時間が過ぎると，シンクポイントダンプを取得します。

- 経過時間に 0 を指定すると，HiRDB は経過時間でのシンクポイントダンプを取得しません。
- 前回シンクポイントダンプを取得してからトランザクションが一度も発生しない場合，ここで指定した経過時間が過ぎてもシンクポイントダンプを取得しません。

《指定値の目安》

- HiRDB を再開始するときの所要時間に特に規定のない場合は，このオペランドを指定する必要はありません。
- このオペランドの指定値によって，HiRDB を再開始するときの所要時間が決まります。

このオペランドの指定値を小さくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が短くなります。ただし，シンクポイントダンプの取得回数が増えるため，オンライン中の性能が低下する場合があります。

逆に，このオペランドの指定値を大きくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が長くなります。ただし，シンクポイントダンプの取得回数が減るため，オンライン中の性能が向上する場合があります。

《指定値のチューニング方法》

シンクポイントダンプがどの程度の間隔で取得されているかは，統計解析ユティリティのシステムの稼働に関する統計情報の「シンクポイントダンプ取得間隔時間（SYNC POINT GET INTERVAL）」で確認できます。SYNC POINT GET INTERVAL の平均値を見て，シンクポイントダンプを取得する間隔が長いと判断したら，ここでの指定値を小さくしてください。また，反対に短いと判断したら，ここでの指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- システムログ出力量：5000
- 経過時間：60

ただし，pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，システムログ出力量の省略値は 1000 が仮定されます。

《注意事項》

シンクポイントダンプの取得間隔はシステムログの出力量で決まります。このため，更新系のトランザクションがほとんどない時間帯は，メモリからデータベースに反映するまでの時間が長くなります。このような状態で障害が発生すると，その間に発生したトランザクションの回復時間が長くなります。このようなケースが考えられる場合は，「経過時間」でもシンクポイントダンプの取得間隔を設定してください。

6.2.14　サーバ用ステータスファイルに関するオペランド

50）　pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

：

pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

サーバ用ステータスファイルを定義します。pd_sts_file_name_2～7 オペランドは省略できますが，pd_sts_file_name_1 オペランドは省略できません。

"論理ファイル名"：～<識別子>((1～8 文字))

サーバ用ステータスファイルの論理ファイル名を指定します。ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系ステータスファイル名"：～<パス名>((167 文字以内))

A 系のステータスファイル名を絶対パス名で指定します。

"B 系ステータスファイル名"：～<パス名>((167 文字以内))

B 系のステータスファイル名を絶対パス名で指定します。

《指定値の目安》

- A 系及び B 系のステータスファイル名には，pdstsinit コマンドで作成したステータスファイルを指定してください。pdstsinit コマンドで作成していないステータスファイルを指定すると，そのステータスファイルは**実体のないステータスファイル**となります。
- ステータスファイルに障害が発生すると，HiRDB はステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要となります。
- A 系と B 系には同じレコード長，及び同じ容量のステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- ステータスファイルは，A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系ステータスファイル名，及び B 系ステータスファイル名には同じ名称を重複して指定できません。
- HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥sts01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。sts01 については大文字と小文字の区別をします。

《留意事項》

- HiRDB を正常開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，全ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときは pd_sts_file_name_1～7 の中で最初に指定したステータスファイルが現用ファイルになります。残りのファイルのうちオープンできたファイルが予備ファイルになり，オープンできないファイルは予約ファイルになります。
- HiRDB を再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。

●実体のないステータスファイルの使い方

実体のないステータスファイルを指定すると，HiRDB の稼働中にステータスファイルを新規追加できます。例えば，ステータスファイルの障害などで予備ファイルが少なくなったときに，実体のないステータスファイルを予備ファイルにします。実体のないステータスファイルを予備ファイルにする手順を次に示します。

〈手順〉

1. pdstsinit コマンドで，システムファイル用の HiRDB ファイルシステム領域にステータスファイルを作成します。
2. pdstsopen コマンドで，ステータスファイルをオープンします。

この操作は HiRDB 稼働中に実行できます。HiRDB を一度停止する必要はありません。

- **メリット及びデメリット**

  実体のないステータスファイルを定義すると，HiRDB ファイルシステム領域の占有量は小さくなります。しかし，実体のないステータスファイルを予備ファイルとして追加するときに，システムファイル用の HiRDB ファイルシステム領域に十分な空き（ファイルを追加するだけの空き領域）がないと追加できないため，システムの信頼性は低くなります。

  実体のないステータスファイルを定義しないと，HiRDB ファイルシステム領域の占有量は大きくなるが，ファイル障害によるスワップ先が保証されるため，信頼性は高くなります。

- **注意事項**

実体のないステータスファイルを定義した場合，HiRDB は HiRDB の開始時にステータスファイルに異常があると認識します。このため，pd_sts_initial_error オペランドに stop（省略値）を指定していると，HiRDB を開始できないので注意してください。実体のないステータスファイルを定義する場合は，pd_sts_initial_error オペランドに continue 又は excontinue を指定してください。また，HiRDB を開始する前に現用ファイルを pd_sts_last_active_file オペランドに指定する必要があります。

6.2.15　サーバ用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

51）pd_sts_initial_error = stop | continue | excontinue

サーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，又はバックエンドサーバ）の開始時，HiRDB はサーバ用ステータスファイルの現用ファイルを特定する処理を実行します。この特定処理で，次に示す異常を検知したときの HiRDB の処置を指定します。

- サーバ用ステータスファイルの実体がない
- サーバ用ステータスファイルの障害を検知した

現用ファイルの特定処理は pd_sts_file_name_1～7 オペランドに指定したサーバ用ステータスファイルを対象とします。

stop：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。この場合，異常を検知したステータスファイルの対策を実施した後に HiRDB を開始してください。

continue 又は excontinue：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知しても，現用ファイルが正常な場合はサーバの開始処理を続行します。ただし，pd_sts_singleoperation オペランドの値（ステータスファイルの片系運転をするかどうか）によっては開始処理を中止することがあります。pd_sts_singleoperation オペランドとの関係を次に示します。

●pd_sts_singleoperation オペランドとの関係

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
continue	サーバ用ステータスファイルの異常を検知した場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。
stop（省略値）	サーバ用ステータスファイルの異常を検知した場合，HiRDB が現用ファイルを特定してサーバの開始処理を続行します。ただし，A 系と B 系のファイルが次に示す表（**HiRDB が現用ファイルを特定できないケース**）の条件のどれかに当てはまる場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，	HiRDB が現用ファイルを特定できない場合は，HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
	HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	

●HiRDB が現用ファイルを特定できないケース

pd_sts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
continue	障害閉塞	障害閉塞
	障害閉塞	オープン（初期状態）
	障害閉塞	ファイル実体なし
	オープン（初期状態）	障害閉塞
	オープン（初期状態）	ファイル実体なし
	ファイル実体なし	オープン（初期状態）
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし
excontinue	障害閉塞	障害閉塞
	障害閉塞	ファイル実体なし
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし

したがって，HiRDB 管理者の操作をなるべく少なくしたい場合（pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定するケースを減らしたい場合）は，次に示すようにオペランドを指定してください。

- pd_sts_initial_error オペランドに excontinue を指定
- pd_sts_singleoperation オペランドに stop を指定

《指定値の目安》

各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
サーバ開始時の HiRDB の処理	サーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	一部のサーバ用ステータスファイルに異常があっても，現用ファイルが正常な場合はサーバの開始処理を続行します。
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB 開始時の障害対策を簡易化する場合は continue 又は excontinue を指定します。
メリット	サーバの開始時に該当するサーバの全サーバ用ステータスファイルの正常が保証されます。このため，HiRDB の開始後に現用ファイルの障害	サーバの開始時に一部のサーバ用ステータスファイルの障害を検知しても，残りの正常なファイルだけで HiRDB を開始できます。このため，

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
	が発生した場合, 予備ファイルにスワップできます。	HiRDB の停止時間を短くできます。この場合，予備ファイルが少なくなっているため，障害状態のステータスファイルをすぐに修復する必要があります。
デメリット	サーバ用ステータスファイルの障害によって HiRDB の開始処理が中止される可能性が高くなります。	予備ファイルが少ない状態で HiRDB を稼働することがあるため，システムの信頼性が低下します。予備ファイルの数によってはサーバ用ステータスファイルをスワップできないことがあります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，stop が仮定されます。

《注意事項》

- 現用ファイルが両系とも異常の場合は，このオペランドの指定値に関係なくサーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。
- HiRDB を開始する前に現用ファイルを pdstsinit コマンドで初期化しないでください。初期化すると HiRDB を再開始できません。
- このオペランドに excontinue を指定し，すべての世代のステータスファイルが片系運転でかつオープン（初期状態）の場合，HiRDB は起動しません。

《備考》

オペランドの指定値と HiRDB の処理及び HiRDB 管理者の処置を次に示します。

●オペランドの指定値と HiRDB の処理

pd_sts_initial_errorオペランドの値	ステータスファイル障害	pd_sts_singleoperationオペランドの値	HiRDBが現用ファイルを特定できるか	pd_sts_last_active_fileの指定	pd_sts_last_active_fileオペランドの値と，オープンできたステータスのうち，最新のファイルが一致するか	現用ファイル障害	pd_sts_last_active_sideオペランドの指定	pd_sts_last_active_sideオペランドに指定されたファイルが使用できるか	表「HiRDBの処理及びHiRDB管理者の処置」の対応番号
stop（省略値）	なし	－	－	－	－	－	－	－	[1]
	あり	－	－	－	－	－	－	－	[5]
continue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
excontinue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]

（凡例）－：該当しません（条件がHiRDBの処理に影響を与えません）。

●HiRDB の処理及び HiRDB 管理者の処置

対応番号	HiRDB の処理	HiRDB 管理者の処置
[1]	HiRDB の開始処理を続行します。	なし
[2]	HiRDB が最新の現用ファイルを特定して，開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[3]	pd_sts_last_active_file オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[4]	pd_sts_last_active_file 及び pd_sts_last_active_side オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[5]	pd_sts_initial_error オペランドに stop が指定されているため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000010 が示す対策をしてください。

対応番号	HiRDB の処理	HiRDB 管理者の処置
[6]	前回稼働時の現用ファイルが特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000016 が示す対策をしてください。
[7]	HiRDB が特定した現用ファイルの正常な系と，pd_sts_last_active_side オペランドに指定された系が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000017 が示す対策をしてください。
[8]	HiRDB が特定した現用ファイル名と，pd_sts_last_active_file オペランドに指定したファイル名が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000015 が示す対策をしてください。
[9]	前回稼働時の現用ファイルの正常な系が特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000018 が示す対策をしてください。

52）pd_sts_singleoperation = stop | continue

サーバ用ステータスファイルの片系運転をするかどうかを指定します。予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，正常な系（片方の系）だけで処理を続行することを**ステータスファイルの片系運転**といいます。ステータスファイルの片系運転については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

なお，予備ファイルがある状況で現用ファイルの片系に障害が発生した場合は，このオペランドの指定値に関係なく，ステータスファイルをスワップして処理を続行します（片系運転にはなりません）。

stop：

片系運転をしません。片系運転になる場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。HiRDB が異常終了した場合，予備ファイルを準備した後に HiRDB を開始してください。

continue：

片系運転をします。片系運転になると KFPS01044-I メッセージが出力されます。片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了したりすると，HiRDB を再開始できなくなるため，片系運転になった場合はすぐに予備ファイルを準備してください。

《指定値の目安》

- stop を指定してシステムの信頼性を上げることを推奨します。また，現用ファイルの障害に対しては，予備ファイルの数を増やして備えることを推奨します。
- 各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB を停止させないことを重視する場合は continue を指定します。
メリット	予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，片系運転をしないで HiRDB	予備ファイルがない状況で現用ファイルの片系に障害が発生しても，処理を続行できます。この

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
	を異常終了します。このため，現用ファイルの内容が失われる可能性が低くなります。	ため，ステータスファイルの障害によってHiRDB が停止する可能性が低くなります。
デメリット	ステータスファイルの障害によって HiRDB が停止する可能性が高くなります。ただし，予備ファイルの数を増やすことで可能性を低くできます。	片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，現用ファイルの内容が失われるため，HiRDB を再開始できなくなります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，stop が仮定されます。

《ほかのオペランドとの関連》

pd_sts_singleoperation 及び pd_sts_initial_error オペランドの指定値の組み合わせによって，ステータスファイル障害時の HiRDB の処理が決定します。したがって，これら二つのオペランドの指定値は一緒に考えるようにしてください。

53)　pd_sts_last_active_file = "論理ファイル名"

～<識別子>((1～8 文字))

フロントエンドサーバの開始時に現用にするステータスファイルの論理ファイル名を指定します。HiRDB は，このオペランドに指定されたファイルと HiRDB が特定した現用ファイルを比較します。ファイルが一致する場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しますが，一致しない場合は開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **全ステータスファイルを初期化した直後に HiRDB を開始する場合**

   pd_sts_file_name1～7 オペランドに指定した最も小さい番号の正常な論理ファイル名を指定してください。この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，全ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_sts_last_active_side オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   全ステータスファイルを初期化してください。その後，1 の方法を実施してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

5. 実体のないステータスファイルを指定している場合

現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I
- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージのうちで，最も直前に出力されたメッセージに表示されているステータスファイルが現用ファイルになります。

54）　pd_sts_last_active_side = A | B

現用ファイルの片系に障害が発生している状態でフロントエンドサーバを開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- pd_sts_last_active_file オペランドを指定している

6.2.16　システムログファイルの構成に関するオペランド

55）　pdlogadfg -d sys -g ファイルグループ名〔ONL〕

システムログファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシステムログファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～200 個指定できます。

《オペランドの規則》

このオペランドは 2～200 個指定できます。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

56)　pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名"〔-b "システムログファイル名"〕

ファイルグループを構成するシステムログファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "システムログファイル名"：～<パス名>((167 文字以内))

ファイルグループを構成するシステムログファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

-b "システムログファイル名"：～<パス名>((167 文字以内))

システムログファイルを二重化（pd_log_dual オペランドに Y を指定）する場合に，B 系のシステムログファイル名を絶対パス名で指定します。pd_log_dua オペランドに Y を指定しない場合は，システムログファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

《オペランドの規則》

HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥log01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。log01 については大文字と小文字の区別をします。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

6.2.17　シンクポイントダンプファイルの構成に関するオペランド

57)　pdlogadfg -d spd -g ファイルグループ名〔ONL〕

シンクポイントダンプファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシンクポイントダンプファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～30 個指定できます。

《オペランドの規則》

このオペランドは 2～60 個指定できます。

58)　pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕

ファイルグループを構成するシンクポイントダンプファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "シンクポイントダンプファイル名"：～<パス名>((167 文字以内))

ファイルグループを構成するシンクポイントダンプファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

-b "シンクポイントダンプファイル名"：～<パス名>((167 文字以内))

シンクポイントダンプファイルを二重化（pd_spd_dual = Y を指定）する場合に，B 系のシンクポイントダンプファイル名を絶対パス名で指定します。pd_spd_dual = Y の指定がない場合は，シンクポイントダンプファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

《オペランドの規則》

HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥sync01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。sync01 については大文字と小文字の区別をします。

6.2.18　プラグインに関するオペランド

59)　pdplgprm -n プラグイン名称〔 -s 共用メモリサイズ〕

プラグインの名称及びプラグインが使用する共用メモリの大きさを指定します。プラグインを使用しない場合，又はプラグインがフロントエンドサーバで稼働しない場合は，このオペランドを省略してください。

《前提条件》

ここで指定するプラグインは，あらかじめ pdplgrgst コマンドで HiRDB に登録しておいてください。

-n プラグイン名称：～<識別子>((1～30 文字))

ここで指定するプラグイン名称については，プラグインに付いているマニュアルを参照してください。

-s 共用メモリサイズ：～<符号なし整数>((1～2000000))《0》(単位：キロバイト)

プラグインが使用する共用メモリの大きさをキロバイト単位で指定します。プラグインが使用する共用メモリの大きさについては，プラグインに付いているマニュアルを参照してください。

6.2.19　HiRDB External Data Access 機能に関するオペランド

HiRDB External Data Access 機能については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

なお，HiRDB の稼働プラットフォームによっては HiRDB External Data Access 機能を使用できません。詳細については，マニュアル「HiRDB Version 8 解説」の「HiRDB External Data Access」を参照してください。

60）　pdhubopt

Hub 最適化情報定義を適用する外部サーバを指定します。このオペランドは次に示すどちらかの形式で指定します。

- pdhubopt -s 外部サーバ名 -f Hub 最適化情報定義ファイル名
- pdhubopt -d 外部サーバ種別 〔-v 外部サーバのバージョン〕 -f Hub 最適化情報定義ファイル名

●-s 外部サーバ名 ～＜ 1～30 文字の識別子＞

CREATE SERVER で指定する外部サーバ名を指定します。Hub の最適化を外部サーバごとに行う場合に指定します。このオプションを指定した場合は，-d 及び-v オプションを指定できません。

このオプションに指定した外部サーバ名が登録されていない場合，%PDCONFPATH%下に「外部サーバ.opt」ファイルがあると，次に示すようにこのオペランドが指定されていると仮定されます。

pdhubopt -s 外部サーバ名 -f 外部サーバ名.opt

●-d 外部サーバ種別〔-v 外部サーバのバージョン〕

Hub の最適化を外部サーバの種別ごと，及び外部サーバのバージョンごとに行う場合に指定します。外部サーバの種別ごと，外部サーバのバージョンごとに次のように指定します。なお，このオプションを指定した場合は，-s オプションを指定できません。

外部サーバの種別	-d オプションの指定値※1	-v オプションの指定値※2
HiRDB Version 6 以降	HIRDB	6.0
HiRDB Version 5.0		5.0
XDM/RD E2	XDMRD	6.0
Oracle8i	ORACLE	8.1.5
Oracle9i		9.2
Oracle 10g		10.1

注※1　CREATE SERVER の TYPE 句のサーバ種別と同じです。

注※2　CREATE SERVER の VERSION 句のサーババージョンと同じです。

●-f Hub 最適化情報定義ファイル名

Hub 最適化情報定義ファイルの名称を指定します。パス名は指定しないでファイル名称だけ指定してください。また，Hub 最適化情報定義ファイルは%PDDIR%¥conf 下に格納してください。

《前提条件》

HiRDB External Data Access が必要です。

《オペランドの指定方法》

- このオペランドは複数個指定できます。
- 特定の外部サーバの Hub 最適化情報定義ファイルを個別に管理する場合は，-s（外部サーバ名）を指定してください。
- 複数の外部サーバを利用する場合，同一の DBMS で Hub 最適化情報定義ファイルを共用化したいときは-d（外部サーバ種別）及び-v（外部サーバのバージョン）を指定してください。

《注意事項》

- -s（外部サーバ名）を指定する場合の注意事項

  以下に示す文字を含む外部サーバ名は指定できません。指定した場合，表識別子に外部表を指定したSQLの動作を保証できません。

  ・半角空白

  ・半角ハイフン（-）

  ・¥

  ・@

  ・#

  上記文字を含む外部サーバ名をCREATE SERVERに指定し，Hubの最適化を外部サーバごとに行う場合は，-sオプションに上記文字を含む外部サーバ名を指定したpdhuboptを省略し，%PDCONFPATH%下に「外部サーバ名.opt」のファイル名称でHub最適化情報定義ファイルを格納してください。

- Hub最適化情報定義ファイルが複数ある場合の注意事項ここで指定した条件が一致するすべてのHub最適化情報定義ファイルが適用されます。優先順位は-sの指定，-d及び-vの指定，-dだけの指定の順になります。

（例）

外部サーバにHiRDB Version 6（サーバ名：hirdb_v6_01）とHiRDB Version 5.0を使用しています。pdhuboptオペランドの指定は次のようになっています。

```
pdhubopt -d HIRDB -f hubopt1           1
pdhubopt -d HIRDB -v 6.0 -f hubopt2    2
pdhubopt -s hirdb_v6_01 -f hubopt3     3
```

- HiRDB Version 6には1～3のHub最適化情報定義ファイルが適用されます。同じオペランドが指定されている場合は，3の指定値が一番優先されます。1と2で同じオペランドが指定されている場合は，2の指定値が優先されます。
- HiRDB Version 5.0には1のHub最適化情報定義ファイルが適用されます。

7 ディクショナリサーバ定義

この章では, ディクショナリサーバ定義の各オペランドの内容について説明します。

7.1　オペランドの形式

ディクショナリサーバ定義ではディクショナリサーバの実行環境を定義します。ここでは，ディクショナリサーバ定義の各オペランドの形式について説明します。なお，表中の「番号」は，「7.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

番号	形式
1	[set pd_max_dic_process = 1 ディクショナリサーバ当たりの最大起動プロセス数] ※
2	[set pd_process_count = 常駐プロセス数 [，サーバ開始時の常駐プロセス数]] ※
3	[set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔] ※
4	[set pd_max_ard_process = 非同期 READ プロセス数] ※
5	[set pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数]

(2)　作業表

番号	形式
6	[set pd_work_buff_mode = each | pool] ※
7	[set pd_work_buff_size = 作業表用バッファ長] ※
8	[set pd_work_buff_expand_limit = 作業表用バッファの増分上限値] ※

(3)　システム監視

番号	形式
9	[set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値] ※
10	[set pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値] ※

(4)　排他制御

番号	形式
11	[set pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ] ※
12	[set pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数] ※
13	[set pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和] ※

番号	形式
14	[set pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数] ※
15	[set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数] ※
16	[set pd_dbsync_lck_release_count = シンクポイント時のグローバルバッファ排他解除間隔] ※

(5)　バッファ

番号	形式
17	[set pd_sql_object_cache_size = SQL オブジェクト用バッファ長] ※

(6)　共用メモリ

番号	形式
18	[set pd_dic_shmpool_size = ディクショナリサーバ用共用メモリサイズ] ※

(7)　RPC トレース情報

番号	形式
19	[set pd_rpc_trace = Y | N] ※
20	[set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"] ※
21	[set pd_rpc_trace_size = RPC トレースを取得するファイルの容量] ※

(8)　トラブルシュート情報

番号	形式
22	[set pd_module_trace_max = モジュールトレース格納最大数] ※
23	[set pd_module_trace_timer_level = モジュールトレース出力時刻取得方法] ※
24	[set pd_pth_trace_max = 通信トレース格納最大数]

(9)　グローバルバッファ

番号	形式
25	[set pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値] ※
26	[set pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値] ※

(10)　Java

番号	形式
27	[set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"] ※

番号	形式
28	〔set pd_java_castoff = Y ｜ N〕

(11)　システムログファイル

番号	形式
29	〔set pd_log_dual = Y｜N〕※
30	〔set pd_log_remain_space_check = warn｜safe〕※
31	〔set pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔,"アンロードログファイルの出力先ディレクトリ"〕…〕
32	〔set pd_log_singleoperation = Y｜N〕※
33	〔set pd_log_rerun_reserved_file_open = Y｜N〕※
34	〔set pd_log_rerun_swap = Y｜N〕※
35	〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕※
36	〔set pd_log_unload_check = Y｜N〕※
37	〔set pd_log_max_data_size = ログ入出力バッファ長〕※
38	〔set pd_log_write_buff_count = ログ出力バッファ面数〕※
39	〔set pd_log_rec_leng = システムログファイルのレコード長〕※
40	〔set pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数〕※
41	〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔,拡張上限サイズ〕〕※

(12)　シンクポイントダンプファイル

番号	形式
42	〔set pd_spd_dual = Y｜N〕※
43	〔set pd_spd_assurance_msg = Y｜N〕※
44	〔set pd_spd_assurance_count = 有効保証世代数〕※
45	〔set pd_spd_reduced_mode = 縮退運転オプション〕※
46	〔set pd_spd_reserved_file_auto_open = Y｜N〕※
47	〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕※
48	〔set pd_log_sdinterval = システムログ出力量〔,経過時間〕〕※

(13)　サーバ用ステータスファイル

番号	形式
49	set pd_sts_file_name_1 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"
	〔set pd_sts_file_name_2 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_3 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_4 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_5 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_6 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕
	〔set pd_sts_file_name_7 = "論理ファイル名", "A 系ステータスファイル名", "B 系ステータスファイル名"〕

(14)　サーバ用ステータスファイル（障害発生時）

番号	形式
50	〔set pd_sts_initial_error = stop｜continue｜excontinue〕※
51	〔set pd_sts_singleoperation = stop｜continue〕※
52	〔set pd_sts_last_active_file = "論理ファイル名"〕
53	〔set pd_sts_last_active_side = A｜B〕

(15)　作業表用ファイル

番号	形式
54	pdwork -v "HiRDB ファイルシステム領域名称"〔, "HiRDB ファイルシステム領域名称"〕…

(16)　システムログファイルの構成

番号	形式
55	{{pdlogadfg -d sys -g ファイルグループ名〔ONL〕}}
56	{{pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名" 〔-b "システムログファイル名"〕}}

(17)　シンクポイントダンプファイルの構成

番号	形式
57	{{pdlogadfg -d spd -g ファイルグループ名〔ONL〕}}
58	{{pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕}}

(18)　プラグイン

番号	形式
59	{{[pdplgprm -n プラグイン名称 〔 -s 共用メモリサイズ]]}}

注※

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

7.2　オペランドの説明

7.2.1　プロセスに関するオペランド

1）　pd_max_dic_process = 1 ディクショナリサーバ当たりの最大起動プロセス数

～<符号なし整数>((1～2048))

1 ディクショナリサーバ当たりの**最大起動プロセス数**を指定します。マルチフロントエンドサーバの場合，ディクショナリサーバに pd_max_users の値を超える処理が集中することがあります。また，マルチフロントエンドサーバでない場合も，RD エリアやグローバルバッファに関連する運用コマンド（pdbufls，pddbls，pdopen，pdclose，pdhold，pdrels）を pd_max_users の値以上同時に実行すると，ディクショナリサーバに pd_max_users の値を超える処理が集中することがあります。

このオペランドには，1 ディクショナリサーバのプロセスを pd_max_users の値を超えて起動する場合の最大起動プロセス数を指定します。

《指定値の目安》

- 1 ディクショナリサーバに集中する可能性がある処理の最大値を次に示します。

  pd_max_users オペランドの値×フロントエンドサーバ数

  ここで求めた値を最大値として，ディクショナリサーバへの処理の集中度を考慮してこのオペランドの値を指定してください。必要以上に大きな値を指定すると，メモリを圧迫する原因となります。

- RD エリアやグローバルバッファに関する運用コマンドの実行で，ディクショナリサーバに処理が集中する場合は，同時に実行する運用コマンドの数が最大値になります。

- pd_max_users の値よりも小さな値を指定した場合は pd_max_users の値を仮定します。このとき，警告メッセージ（KFPS01888-W）を出力します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，pd_max_users の値が仮定されます。

《指定値のチューニング方法》

最大起動プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

2）　pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕

～<符号なし整数>((0～2048))

常駐プロセス数：

ディクショナリサーバの常駐プロセス数を指定します。常駐プロセスとは，サーバ開始時からあらかじめ起動しておくプロセスのことです。

《利点》

ディクショナリサーバで並行処理できるトランザクションが使用するプロセスを，あらかじめシステム開始時に起動して常駐しておくことで，新たにトランザクションが入力されてもプロセス起動時間を削減できます。ただし，HiRDB の開始に時間が掛かります。

《指定値の目安》

- ディクショナリサーバのサーバプロセスのプロセス固有領域とプロセサの実メモリ量から値を求めてください。サーバプロセスのプロセス固有領域については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

- マルチフロントエンドサーバ構成で，更に pd_max_bes_process 又は pd_max_dic_process オペランドを指定している場合は，次に示す条件を満たすようにこのオペランドを指定してください。

  pd_process_count の値≦pd_max_bes_process 又は pd_max_dic_process の値

- このオペランドには，ディクショナリサーバの**最大起動プロセス数**（pd_max_dic_process※の値）の範囲内で値を指定します。

  注※

  pd_max_dic_process 又は pd_max_bes_process オペランドを省略している場合は，pd_max_users の値が仮定されます。

《指定値のチューニング方法》

常駐プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- 常駐プロセス数はメモリと密接に関連しているため，必要以上に大きい値を指定すると，HiRDB を開始できなくなることがあります。
- 常駐プロセス数を超えたプロセスが必要になると，最大起動プロセス数までプロセスを動的に起動します。ただし，pd_max_server_process オペランドの指定値によっては，最大起動プロセス数分のプロセスを起動できないことがあります。

《オペランドの省略値》

このオペランドを省略（又は 0 を指定）すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

↑最大起動プロセス数÷2↑

サーバ開始時の常駐プロセス数：

HiRDB の開始処理時に常駐化する常駐プロセスの数を指定します。

通常，常駐プロセスの起動処理は HiRDB の開始処理中に行われます。このため，常駐プロセス数が多くなると，それに比例して HiRDB の開始処理時間が長くなります。目安として，100MIPS 程度のサーバマシンでプロセスを一つ起動するのに約 1 秒掛かります。

サーバ開始時の常駐プロセス数の指定有無による処理の違いを次に示します。

●サーバ開始時の常駐プロセス数を指定しない場合（pd_process_count = 500 と指定した場合）

すべての常駐プロセスの起動処理を HiRDB の開始処理中に実行します。常駐プロセス（この場合は 500 個）がすべて起動されないと，HiRDB を開始できません。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 500 秒掛かります。

●サーバ開始時の常駐プロセス数を指定する場合（pd_process_count = 500,50 と指定した場合）

一部の常駐プロセスの起動処理を HiRDB の開始処理中に実行し，残りの常駐プロセスの起動処理を HiRDB の開始処理終了後に実行します。一部の常駐プロセス（この場合は 50 個）が起動されれば，HiRDB を開始できます。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 50 秒掛かります。残りの常駐プロセス（この場合は 450 個）は，HiRDB の開始処理終了後に起動されます。

《利点》

HiRDB の開始処理時間を短縮できます。系切り替え機能を使用している場合など，HiRDB の開始処理時間をなるべく短くしたいときに適用します。

《指定値の目安》

HiRDB の開始処理終了直後に必要なプロセス数を指定します。

《注意事項》

サーバ開始時の常駐プロセス数を指定する場合は，クライアント環境定義の PDCWAITTIME オペランドの値を見直してください。

HiRDB の開始直後に，サーバ開始時の常駐プロセス数の値を超える UAP が実行されると，残りの常駐プロセスを起動した後にトランザクション処理が実行されます。したがって，クライアント環境定義の PDCWAITTIME オペランドの値が小さいと，一部の UAP がタイムアウトで処理できない場合があります。

PDCWAITTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

3）pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔

～<符号なし整数>((0～1440))（単位：分）

HiRDB の非常駐サーバプロセスの停止処理をする間隔を分単位で指定します。この機能は，稼働中のサーバプロセス数が常駐プロセス数（pd_process_count オペランドの値）より多くなった場合に適用されます。この機能によって停止するサーバプロセス数は HiRDB が自動的に算出して決定します。

《利点》

業務（稼働中のサーバプロセス数）がピークのときに再利用できる非常駐サーバプロセスが増えるため，メモリなどのプロセス資源の利用効率が向上します。

《指定値の目安》

- 例えば，一日のうち 1 時間だけ業務のピーク時間があり，その時間内でピークとなる間隔が 2 分ぐらいの場合，このオペランドに 2 を指定します。
- ピーク時に同時実行されるサーバプロセス数が常駐プロセス数以下の場合，この機能を使用しても効果は期待できないため，このオペランドを省略します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《指定値のチューニング方法》

システムの稼働に関する統計情報をサーバ単位に 1 週間取得してください。「サービス実行中のサーバプロセス数（# OF PROCESS ON SERVICE)」の値から業務のピークを求めてください。そのピークが現在設定されている常駐プロセス数（pd_process_count オペランドの値）を超えている場合は，ピークごとの間隔を求めてその時間を設定します。

ただし，サーバマシンのメモリ，CPU などの資源に余裕がある場合は，不足している分のプロセス数を常駐プロセス数に加算する方（pd_process_count オペランドの値を大きくする方）がこのオペランドを指定するより性能が向上します。

《注意事項》

このオペランドを省略するか，又は 0 を指定した場合は 10 秒間隔でサービス待ち状態の非常駐サーバプロセスを調査し，サービス待ち状態の非常駐サーバプロセスがあれば，その非常駐サーバプロセスを停止します。

4）pd_max_ard_process = 非同期 READ プロセス数

～<符号なし整数>((0～256))

非同期 READ 機能を使用する場合にこのオペランドを指定します。このオペランドには非同期 READ 処理に必要なプロセス数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（バックエンドサーバ又はディクショナリサーバ）当たりのプロセス数を指定します。非同期 READ 機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《前提条件》

pdbuffer オペランドの-m オプションで 1 以上を指定している必要があります。

《指定値の目安》

- 0 又は 1 を指定してください。ただし，pdbuffer オペランドの-m オプションの値が 2～256 の場合は，-m オプションと同じ値を指定してください。-m オプションの値が 257 以上の場合は，RD エリアやシステムファイルを格納するディスク装置の数（HiRDB/パラレルサーバの場合は 1 サーバ当たりの数）と同じ値を指定するか，又は 256 を指定してください。
- このオペランドの値を大きくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に処理時間を短縮できます。このオペランドの値を小さくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に，非同期 READ プロセスの処理完了待ちが発生する分，処理時間が余計に掛かることがあります。
- このオペランドの値×サーバ数分のプロセスが起動されるため，リソース（共用メモリ及びメッセージキュー）を考慮してこのオペランドの値を決定してください。共用メモリ及びメッセージキューの見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値のチューニング方法》

指定値（非同期 READ プロセス数）のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《ほかのオペランドとの関連》

このオペランドの値を変更した場合は pd_max_server_process オペランドの値を見直してください。

《オペランドの規則》

このオペランドに 0 を指定すると，非同期 READ 機能を使用しません。

5)　pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数～<符号なし整数>((2～255))

すべてのバッファプールに対して，**デファードライト処理の並列 WRITE 機能**を使用する場合にこのオペランドを指定します。このオペランドには並列処理するプロセス数を指定します。プロセス数を増加させることで書き込み処理時間を短縮できます。デファードライト処理の並列 WRITE 機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値の目安》

デファードライト処理の並列 WRITE 機能が有効になる最小値の 2 を指定してください。また，マニュアル「HiRDB Version 8 システム運用ガイド」の「デファードライト処理のチューニング」を参照してオペランドの値を決定してください。

《注意事項》

デファードライト処理の並列 WRITE 機能を指定すると，プロセス数が増加するため，CPU 利用率が上がります。

7.2.2　作業表に関するオペランド

6)　pd_work_buff_mode = each | pool

HiRDB が作業表を作成するときのバッファの確保方式を指定します。

each：個々の作業表ごとにバッファを確保します。

pool：サーバプロセス単位にバッファプールとして一括して確保します。

《指定値の目安》

- 通常は pool を指定してください。特に大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合には pool を指定してください。
- あらかじめ，作業表用バッファに使用できるプロセス固有領域サイズが決まっている場合，pool を指定してください。pool を指定すると，HiRDB が効率良く作業表用バッファを各作業表に配分します。

  この場合，pd_work_buff_size に指定した値でプロセス固有領域を占有し，そのプール内で作業表の入出力をバッファリングします。したがって，pd_work_buff_size に指定した値以上にプロセス固有メモリを占有することはありません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，pool が仮定されます（pd_sysdef_default_option オペランドに v6compatible を指定している場合は each が仮定されます）。

7）pd_work_buff_size = 作業表用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((128～1000000))
- 64 ビットモードの場合：((128～4000000000))

HiRDB が作成する作業表のバッファの大きさをキロバイト単位で指定します。

項目	pd_work_buff_mode = each の場合	pd_work_buff_mode = pool の場合
利点	作業表バッファ長が大きいと，データ操作時に発生する作業表用ファイルへの入出力回数を削減できるため，作業表を使用する SQL の実行時間を削減できます。ただし，各サーバのプロセス固有メモリを使用するので，システム全体のメモリ量（実メモリ，仮想メモリ）を考慮して指定してください。 特に pd_work_buff_mode=each の場合，メモリは「pd_work_buff_size の値×必要作業表数」分確保されます。したがって，必要以上に大きな値を指定すると，ほかのプロセスで仮想メモリ不足が生じることがあります。	
適用基準	大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合に指定します。このとき，pd_work_buff_mode = pool を指定してください。	
指定値の目安	• 個々の作業表ごとに確保するバッファの大きさを指定します。 • 作業表用バッファ長に，作業表メモリ容量より大きな値を指定すると，作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。作業表メモリ容量の見積もり式は次のとおりです。 **作業表メモリ容量＝対象となる作業表容量※÷2**	• サーバプロセス単位に一括して確保する作業表用バッファプールの大きさを指定します。 • 大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合は，4352～5120 を指定してください。指定すると，ソートでの入出力単位が大きくなるため，ソート処理時間が短くなります。 • 作業表用バッファ長に，SQL 文単位の作業表メモリ総容量を上限として，大きな値を指定するほど作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。SQL 文単位の作業表メモリ総容量の見積もり式は次のとおりです。 **SQL 文単位の作業表メモリ総容量＝ a×b ＋ c×d**

項目	pd_work_buff_mode = each の場合	pd_work_buff_mode = pool の場合
注意事項	複数のユーザが同時に実行する場合，及び複数個の作業表を使用する SQL 文などを実行する場合には，指定したサイズのバッファが，個々の作業表ごとに確保されます。そのため，大きな値を指定するとメモリを圧迫する原因となります。	SQL 文単位に使用する作業表の数に比べて作業表用バッファ長に指定した値が小さいと，each を指定した場合に比べて処理時間が増加することがあります。具体的には，「SQL 文単位の作業表最大数×128」以上の値を指定してください。SQL 文単位の作業表最大数の見積もり式は次のとおりです。 SQL 文単位の作業表最大数＝ b ＋ d
オペランドの規則	指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。	• 指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。 • 384 以上の値を指定してください。384 未満の値が指定されると，自動的に 384 に切り上げられます。
省略値	このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，512 が仮定されます。	このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。 • 32 ビットモードの場合：1024 • 64 ビットモードの場合：5120

a：↑｛作業表（列情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

b：作業表（列情報格納用）の最大数※

c：↑｛作業表（位置情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

d：作業表（位置情報格納用）の最大数※

注※

これらの項目の見積もり方法については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

8)　pd_work_buff_expand_limit = 作業表用バッファの増分上限値

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((128～1000000))
- 64 ビットモードの場合：((128～4000000000))

HiRDB が作成する作業表用バッファ長は pd_work_buff_size オペランドで指定します。この作業表用バッファの不足時，作業表用バッファを自動増分する場合にこのオペランドを指定します。このオペランドで指定した値になるまで作業表用バッファを増分します。

例えば，オペランドに次の指定をした場合，通常 1,024 キロバイトの作業表用バッファを確保します。作業表用バッファが 1,024 キロバイトで不足すると，2,048 キロバイトまで作業表用バッファを増分します。

- pd_work_buff_size = 1024
- pd_work_buff_expand_limit = 2048

次に示すときに HiRDB は作業表用バッファを増分します。

- 結合方式がハッシュジョイン，又は副問合せの実行方式にハッシュ実行が適用されている場合に，必要な作業表用バッファが確保できないとき

- 複数の作業表を同時に使用する場合，各作業表に作業表用バッファを 128 キロバイト割り当てて作業表用バッファが不足するとき

《前提条件》

pd_work_buff_mode オペランドを省略するか，又は pool を指定している必要があります。

《利点》

作業表用バッファ不足（pd_work_buff_size オペランドの指定値不足）で UAP がエラーになるのを防げます。

《注意事項》

次に示すどちらかの条件を満たす場合は作業表用バッファを増分しません。

- pd_work_buff_expand_limit オペランドを指定しない
- pd_work_buff_expand_limit オペランドの値≦pd_work_buff_size オペランドの値

《オペランドの規則》

指定値は 128 の倍数を指定してください。128 の倍数でない場合, 自動的に 128 の倍数に切り上げられます。

《ほかのオペランドとの関連》

ディクショナリサーバプロセスでの作業表用バッファの初回増分時に KFPH29008-I メッセージを出力します。なお，pd_work_table_option オペランドで KFPH29008-I メッセージの出力を抑止できます。

《留意事項》

作業表用バッファを増分した場合，該当するサーバプロセスで使用中の作業表数が 0 になると，増分した作業表用バッファを解放します。作業表の使用数が 0 になるのは次に示すときです。

- 使用中のカーソルをすべてクローズしたとき（この場合，作業表の使用数が 0 にならないことがあります）
- ホールダブルカーソルを使用していない場合は，トランザクションを正常終了するか，又は取り消したとき
- ホールダブルカーソルを使用している場合は，UAP を HiRDB から切り離したとき（DISCONNECT したとき）

《備考》

「ハッシュジョイン，副問合せのハッシュ実行」は，次に示す場合に適用されます。

- pd_additional_optimize_level オペランド，クライアント環境定義の PDADDITIONALOPTLVL オペランド, 又は SQL コンパイルオプションの ADD OPTIMIZE LEVEL オペランドで,「コストベース最適化モード 2 の適用」と一緒に「ハッシュジョイン，副問合せのハッシュ実行」を指定する
- SQL 文中の結合方式の SQL 最適化指定に HASH を指定する
- SQL 文中の副問合せ実行方式の SQL 最適化指定に HASH を指定する

7.2.3　システム監視に関するオペランド

9)　pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0，2～100000))

UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状

態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると, HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。

この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。

このオペランドでは，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。シンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，対象トランザクションを強制的に中断してロールバックをします。これを**シンクポイントダンプ有効化のスキップ回数監視機能**といいます。

シンクポイントダンプ有効化のスキップ回数監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

このオペランドを指定すると，UAP の無限ループなどを防止できます。

《指定値の目安》

通常は 0 を指定してください。0 を指定すると，スキップ回数の上限値を HiRDB が計算します。0 を指定して不都合が発生した場合，又は KFPS02101-I メッセージが出力された場合に，このオペランドの値を変更してください。そのときの指定値の目安については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，シンクポイントダンプ有効化のスキップ回数監視機能を使用しません。

《注意事項》

- このオペランドを指定すると，ログレスモードで実行している UAP も監視対象になります。ログレスモードで実行している UAP の処理が中断すると，データベースを自動回復できないため RD エリアを障害閉塞します。このため，上限値の設定に際しては，ログレスモードで実行する UAP のトランザクション処理中に，該当するサーバ内のほかのトランザクションから出力されるシステムログ量も考慮に入れてください。
- pdload，pdmod，pdrorg，pdexp，pddbst，pdgetcst，pdrbal，pdvrup，及び pdmemdb コマンドはこの機能の監視対象外になります。

10）　pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0～100000))

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになると，シンクポイントダンプの有効化処理ができずにスキップされます。これは，デファードライト処理によってシンクポイントダンプの取得が遅れることで，シンクポイントで出力される更新バッファ数がシンクポイントダンプの取得間隔内で出力できる更新バッファ数を超えるためです。

シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

このオペランドでは，デファードライト処理によってシンクポイントダンプの取得が遅れた場合に，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。

デファードライト処理によってシンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，シンクポイントダンプの取得間隔内でシンクポイントダンプの取得が完了するように，HiRDB が更新バッファ数の上限値を決定します。そして，更新バッファ数が上限値を超えた時点で最も古い更新バッファを出力して，シンクポイント時の更新バッファ総量を制限します。これを**更新バッファ量抑制機能**といいます。

《利点》

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生する，ユニット異常終了を回避できます。

《指定値の目安》

通常はこのオペランドを省略してください。更新処理のスループットを低下させてでも，シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生するユニット異常終了を回避したい場合に，1 を指定します。

許容できるシンクポイントダンプの有効化処理のスキップ回数がログ容量などから分かる場合は，その回数を指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《オペランドの規則》

このオペランドに 0 を指定すると，更新バッファ量抑制機能を使用しません。

《注意事項》

更新バッファ量抑制機能の有効期間について，注意事項を次に示します。なお，更新バッファ量抑制機能の有効期間とは，KFPH23035-I メッセージが出力されてから KFPH23036-I メッセージが出力されるまでの間です。

- 更新バッファ数が，HiRDB が決定した更新バッファの上限値を超えている場合，更新処理の実行後に更新バッファを出力するため，更新処理のスループットが低下します。HiRDB が決定する更新バッファの上限値は，次の計算式で求められます。

（シンクポイントインターバル時間÷WRITE 単価※）
×（1－（前回のシンクポイントからプレシンクまでのログ出力量÷シンクポイント間のログ出力量）
×（バッファプールのバッファ面数÷シンクポイントで更新があったバッファプールの総バッファ面数）

注※　WRITE 単価についての詳細は，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

- pd_dbbuff_rate_updpage オペランド，又は pdbuffer -y オペランドによってデファードライトトリガのトリガ契機を指定している場合，それぞれの指定値が，HiRDB が決定した更新バッファの上限値より大きくなると，HiRDB が決定した更新バッファの上限値をデファードライトトリガのトリガ契機となる更新バッファ数に変更します。

  また，pdbuffer -w オペランドの値は，各バッファの更新バッファ上限値まで出力されるように自動調整されます。

- シンクポイントダンプの有効化処理がスキップした場合，シンクポイント時の更新バッファ出力処理中に検知されます。そのため，更新バッファ量抑制機能が有効となる時期は，シンクポイントダンプの有効化処理がスキップされ，エラーメッセージが出力されるときよりも後になる場合があります。

- 通常，並列 write 機能使用時のシンクポイント処理では各デファードライト処理用並列 WRITE プロセスに対する出力要求はシンクポイントごとに 1 回ですが，更新バッファ量抑制機能使用時はシンクポイントスキップの検出を早めるため複数回に分割して行われます。

7.2.4　排他制御に関するオペランド

11）　pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～2000000))
- 64 ビットモードの場合：((1～2000000000))

ディクショナリサーバの排他制御で使用する共用メモリ領域（排他制御用プール）の大きさをキロバイト単位で指定します。

このオペランドの指定対象の領域は，排他資源管理テーブルとして使用されます。排他資源管理テーブル数，排他要求数，及び排他制御用プールサイズの間には，次の関係があります。

「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数

係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。

《指定値の目安》

- 32 ビットモードの場合，6 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 64 ビットモードの場合，4 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 次に示す計算式でこのオペランドの指定値を見積もってください。

HiRDB の種類	計算式（単位：キロバイト）
32 ビットモードの HiRDB/パラレルサーバの場合	↑↑b ÷pd_lck_pool_partition の値↑÷ 6 ↑×pd_lck_pool_partition の値
64 ビットモードの HiRDB/パラレルサーバの場合	↑↑b ÷pd_lck_pool_partition の値↑÷ 4 ↑×pd_lck_pool_partition の値

b：ディクショナリサーバで同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については，「付録 D　排他資源数の見積もり」を参照してください。

注

特に，定義系 SQL の DROP TABLE 又は DROP SCHEMA を実行する場合は，このオペランドの指定値を正確に見積もって指定してください。

《指定値のチューニング方法》

統計解析ユティリティ（サーバごとのシステムの稼働に関する統計情報）で，ディクショナリサーバの「排他資源管理テーブル使用率（%OF USE LOCK TABLE)」を参照してください。使用率の最大値が 80%以上の場合は，これからのデータベース拡張に備えて，このオペランドの値を大きくすることをお勧めします。使用率の最大値が 10%以下の場合は，共用メモリを節約するために，このオペランドの値を小さくすることをお勧めします。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- 32 ビットモードの場合：16000
- 64 ビットモードの場合：32000

ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます。

《注意事項》

- このオペランドの指定値が小さ過ぎると，SQL がエラーリターンすることがあります。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは，pd_lck_pool_partition オペランドと関連があります。

12）pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数

～<符号なし整数>((1～5000))

排他制御処理の分散をするときにディクショナリサーバの排他制御で使用する排他制御用プールパーティション数を指定します。

排他制御処理の分散については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値のチューニング方法》

排他制御用プールパーティション数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

《注意事項》

- このオペランドの指定値を大きくし過ぎると，各排他制御用プールパーティションに割り当てられる共用メモリのサイズが小さくなるため，排他制御用プールパーティションが容量不足となり SQL がエラーリターンすることがあります。また，排他制御の処理に時間が掛かるため，システムのパフォーマンスが下がります。この場合，このオペランドの指定値を小さくしてください。
- 排他制御用プールサイズは 1 キロバイト以上必要であるため，pd_lck_pool_size の値よりも大きな値を指定した場合，このオペランドには pd_lck_pool_size の値が仮定されます。また，このとき，KFPS00421-W メッセージが出力されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_pool_size
- pd_lck_deadlock_check_interval

13）pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和

～<符号なし整数>((0～140000))

トランザクションを越えて保持する表，及び RD エリアの排他資源数を指定します。このオペランドの値に従って，表，RD エリアの UNTIL DISCONNECT 指定の排他を管理するブロックを共用メモリに確保します。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。次に示す場合に指定値を変更するかどうかを検討してください。

- 同時に実行するユティリティの数を大量に増やす場合

- ホールダブルカーソルを使用する場合

このオペランドの指定値の見積もり方法については，「付録 C.5　1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式」を参照してください。

《指定値のチューニング方法》

このオペランドの指定値が小さいと，トランザクションがロールバックしたり，リターンコード 8 でユティリティが異常終了したりします。このとき, KFPA11914-E, 又は KFPH28001-E メッセージが出力されます。この現象が発生したら，このオペランドの値を大きくしてください。

なお，このオペランドの値を大きくすると，それに比例して必要なメモリも増加します。必要なメモリは「このオペランドの値×48（64 ビットモードの場合は 64)」バイトです。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，256 が仮定されます。

14）　pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数

～<符号なし整数>((16～1024))

UNTIL DISCONNECT 指定の LOCK 文を実行しない表に対してホールダブルカーソルを使用する場合に，そのホールダブルカーソルのトランザクション当たりの最大同時オープン数を指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，16 が仮定されます。

《注意事項》

このオペランドに省略時解釈値以外の値を指定すると，共用メモリ使用量が増加します。

《ほかのオペランドとの関連》

このオペランドと次に示すオペランドの指定値は，ロックサーバの共用メモリサイズ計算に使用されます。32 ビットモードの HiRDB の場合，各オペランドの指定値が大き過ぎると，ロックサーバの共用メモリサイズが 2 ギガバイトを超えるため，HiRDB を開始できなくなることがあります。ロックサーバの共用メモリサイズが 2 ギガバイトを超えないように，このオペランドと次に示すオペランドの指定値を調整してください。

- pd_max_access_tables
- pd_max_users
- pd_max_dic_process
- pd_lck_hash_entry
- pd_lck_pool_size

共用メモリについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

15）　pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数

～<符号なし整数>((0～2147483647))

排他制御用プールで使用するハッシュテーブルのエントリ数を指定します。HiRDB はここで指定された値に従ってユニットコントローラ用の共用メモリに排他制御用プールを確保します。

《指定値の目安》

通常はこのオペランドを省略してください。

次に示す条件に該当する場合は指定値を検討してください。

- バージョン 06-02 以降にバージョンアップする場合に共用メモリサイズをできるだけ変更したくないときは 11261 を指定してください。この場合，バージョンアップ前と同数のハッシュエントリ数を確保するため，排他制御用プール中のハッシュテーブルサイズがバージョンアップ前と同じになります。
- このオペランドに推奨値より大きい値を指定すると，性能が向上することがあります。ただし，《推奨値の求め方》の変数 a より大きい値を指定しても，a を指定したときよりも性能が向上することはありません。

《オペランドの規則》

- このオペランド及びサーバ共通定義の pd_lck_hash_entry オペランドを省略するか，又はこのオペランドに 0 を指定すると，HiRDB はサーバごとに推奨値を計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 11261 が仮定されます）。推奨値については《推奨値の求め方》を参照してください。
- このオペランドに 0 でも素数でもない値を指定した場合，HiRDB はその指定値を超えない最大の素数が指定されたと仮定します。

《注意事項》

このオペランドの指定値が小さ過ぎると，ハッシュエントリ不足が発生して性能が低下することがあります。このオペランドを省略すれば，ハッシュエントリ不足及びハッシュエントリ不足による性能低下は発生しません。

《推奨値の求め方》

推奨値を次に示します。

推奨値＝ MAX（↑a÷10↑，11261）を超えない最大の素数

変数	変数の算出式	
a	（b ＋ 3）×10 ＋ pd_lck_pool_size の値×c	
b	pd_max_users の値＞ pd_max_dic_process の値の場合	pd_max_users の値
	pd_max_users の値≦pd_max_dic_process の値の場合	pd_max_dic_process の値
c	32 ビットモードの場合は 6，64 ビットモードの場合は 4	

16）　pd_dbsync_lck_release_count ＝ シンクポイント時のグローバルバッファ排他解除間隔

～＜符号なし整数＞((0，100～1073741824))

シンクポイント時に発生するグローバルバッファへの排他を解除する間隔を指定します。

シンクポイント時には，ディスクに反映する必要があるバッファ（更新バッファ）のサーチ処理が発生します。通常は，更新バッファのサーチ処理中に，一定間隔でグローバルバッファへの排他を解除します。

例えば 100 を指定した場合，100 面（グローバルバッファの面数）のサーチ処理が完了すると，一度排他を解除します。その後，再度排他を掛けてサーチ処理を続行します。このように 100 面ごとに排他の解除をします。

《利点》

このオペランドを指定すると，シンクポイント時のグローバルバッファの排他占有時間を調整できます。このオペランドの値を小さくすると，グローバルバッファの排他占有時間が短くなり，シンクポイント時のトラザクション性能が向上することがあります。

グローバルバッファプールの排他占有時間については，統計解析ユティリティを実行して，グローバルバッファプールに関する統計情報の「シンクポイント時のバッファプール排他占有時間（SYNCL）」で確認できます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。次に示す条件をすべて満たす場合にこのオペランドの指定を検討してください。

- シンクポイント時にトランザクション性能が低下する
- pdbuffer オペランドの-n オプションに指定したバッファ面数が非常に大きい

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10000 が仮定されます。

《オペランドの規則》

- 1 から 99 までの値を指定した場合，自動的に 100 が設定されます。
- 0 を指定した場合，更新バッファのサーチ処理が完了するまで，グローバルバッファに対して排他が掛かります。

《注意事項》

このオペランドの値を小さくすると，ほかのトランザクションの割り込みによって，更新バッファのサーチ時間が長くなります。その間に更新されたグローバルバッファもシンクポイント時の出力対象になります。このため，シンクポイント時に出力される更新バッファ数が増加します。シンクポイント時に出力対象となる更新バッファ数については，統計解析ユティリティの，グローバルバッファプールに関する統計情報の「シンクポイント出力ページ数（SYNCW）」で確認できます。

7.2.5　バッファに関するオペランド

17）　pd_sql_object_cache_size = SQL オブジェクト用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((22～256000))
- 64 ビットモードの場合：((22～2000000))

SQL オブジェクトを格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- SQL オブジェクトは，ユーザごとにトランザクションが終了するまでバッファに保存されます。したがって，同時に実行するトランザクションの SQL オブジェクトがバッファに格納できる大きさが必要です。
- 静的 SQL の SQL オブジェクトはトランザクション終了後もバッファに空きがなくなるまで保存して，同一 UAP を実行する複数のユーザで共用することで，SQL 解析処理の削減を図ります。バッファを有効に利用するためには，使用頻度の高い UAP の SQL オブジェクトがバッファに常駐するようにバッファを確保してください。
- バッファ長の見積もり時は，まず UAP で発行する SQL 文の SQL オブジェクト長から UAP 実行に必要なバッファ長を見積もります。次に，同時に実行される UAP と同時実行ユーザ数を考慮して必要なバッファ長を算出してください。
- 1SQL 文の SQL オブジェクト長の見積もり方法については，「付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義とシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

（pd_max_users の値＋ 3）×22

《指定値のチューニング方法》

SQL オブジェクト用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

7.2.6　共用メモリに関するオペランド

18）pd_dic_shmpool_size = ディクショナリサーバ用共用メモリサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～200000))
- 64 ビットモードの場合：((1～4000000000))

ユニットコントローラ用の共用メモリの中で，ディクショナリサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。サーバ共通定義及びディクショナリサーバ定義の pd_dic_shmpool_size オペランドを省略すると，このオペランドの値を HiRDB が計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます）。HiRDB はマニュアル「HiRDB Version 8 システム導入・設計ガイド」に記載されている「ディクショナリサーバが使用する共用メモリの計算式」を基に計算をします。この計算式中の変数の説明に出てくるオペランドの値を変更した場合，このオペランドの値を HiRDB が自動的に再計算します。なお，計算式中の変数「サーバ内のインデクス総数」はデータディクショナリ表のインデクス数が仮定されます。また，計算式中の変数「インデクス用のグローバルバッファプール数」，及び「グローバルバッファ総数（pdbuffer オペランドの指定数）」は，32 ビットモードの場合は 500，64 ビットモードの場合は 1000 が仮定されます。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

- 必要以上にこのオペランドの値が大きいと，共用メモリをむだに確保してしまいます。また，このオペランド値が小さ過ぎると，次に示す現象が発生します。
  - ・ユニットを開始できなくなります。
  - ・UAP 又はユティリティが実行できなくなります。
- このオペランドを省略する場合は，必ず pd_assurance_index_no オペランドに適切な値を指定してください。

7.2.7　RPC トレース情報に関するオペランド

19）　pd_rpc_trace = Y｜N

RPC トレース情報を取得するかどうかを指定します。RPC トレース情報には, HiRDB の保守情報が出力されます。

通常，このオペランドは省略してください。

Y：RPC トレースを取得します。

N：RPC トレースを取得しません。

《注意事項》

このオペランドに Y を指定すると，通信性能が低下します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，N が仮定されます。

20）　pd_rpc_trace_name = "RPC トレースを取得するファイル名称"

～< 254 文字以内のパス名>

RPC トレースファイルの名称を絶対パス名で指定します。RPC トレースファイルとして，指定したファイル名の後に 1，2，l（エル）を付けた三つのファイルが作成されます。

《注意事項》

このオペランドで指定したディレクトリ下に，最大「pd_rpc_trace_size の値× 2」の大きさのファイルが作成されます。ファイル容量に注意してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

%PDDIR%¥spool¥rpctr

21）　pd_rpc_trace_size = RPC トレースを取得するファイルの容量

～<符号なし整数>((1024～2147483648))（単位：バイト）

RPC トレースファイルの容量をバイト単位で指定します。

このオペランド指定値は，"RPC トレースファイル"1 及び"RPC トレースファイル"2 のそれぞれの容量となります。

《指定値の目安》

RPC トレースは，すべてのプロセス間の通信電文が一つのファイルに時系列に蓄積されます。その量がこの指定値を超えると，もう一方のファイルにトレースデータの出力を切り替えます。古いトレースは上書きされて消去されます。このため，指定値が小さいと十分なトレースを取得できなくなり, トラブルシュートが困難になることがあります。したがって, このオペランドには 1,000,000 以上の値を指定してください。

《注意事項》

このオペランドで指定した値は RPC トレースファイル l（エル）の容量に影響を与えません。RPC トレースファイル l（エル）の容量は 0 バイト固定です。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，4096 が仮定されます。

7.2.8　トラブルシュート情報に関するオペランド

22)　pd_module_trace_max = モジュールトレース格納最大数

～<符号なし整数>((126～16383))

HiRDB のプロセスは，実行した関数やマクロの履歴をプロセス固有メモリ中に記録しています。この履歴をモジュールトレースといい，このオペランドではモジュールトレースのレコード数を指定します。この履歴の内容は core ファイルに取り込まれ，プロセス異常終了時に出力されます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，126 が仮定されます。

《注意事項》

各プロセスでは，以下のサイズのプロセス固有メモリが確保されます。

32 ビットモードの場合：64 + 48×pd_module_trace_max オペランドの値（バイト）

64 ビットモードの場合：64 + 64×pd_module_trace_max オペランドの値（バイト）

23)　pd_module_trace_timer_level = 0 | 10 | 20

モジュールトレースに出力する時刻の取得方法を指定します。このオペランドの指定値の意味を以下に示します。

指定値	時刻取得方法
0	すべてのモジュールトレース出力箇所で，秒単位で取得します。
10	すべてのモジュールトレース出力箇所のうち，入出力処理前後など性能調査で問題となりやすい箇所だけマイクロ秒単位で取得し，そのほかの箇所では秒単位で取得します。
20	すべてのモジュールトレース出力箇所で，マイクロ秒単位で取得します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

このオペランドに 0 以外の値を指定すると，時刻をマイクロ秒で取得する関数を発行するため，性能が低下することがあります。

24)　pd_pth_trace_max = 通信トレース格納最大数

～<符号なし整数>((1024～8388608))

トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。

《指定値の目安》

通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。

《注意事項》

通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

7.2.9　グローバルバッファに関するオペランド

25)　pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値

～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加するグローバルバッファ数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

- pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，その数に余裕を持たせた値を指定してください。
- 次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_max_add_dbbuff_no の値≦2000000－HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

条件		省略値
32 ビットモードの場合	a ≧500 の場合	256
	a ＜ 500 の場合	500－ a
64 ビットモードの場合	a ≧1000 の場合	256
	a ＜ 1000 の場合	1000－ a

a：HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《注意事項》

このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_shm_no

26）pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加したときに割り当てられる，共用メモリセグメント数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

条件		省略値
pd_max_add_dbbuff_no オペランドを省略した場合	32 ビットモードの場合	（16－a）＋ 500
	64 ビットモードの場合	（16－a）＋ 1000
pd_max_add_dbbuff_no オペランドを指定した場合	32 ビットモードの場合	↓pd_max_add_dbbuff_no の値×1.5 ＋ 16↓ （計算値が 32752 以上だった場合は自動的に 32752 が設定されます）
	64 ビットモードの場合	

a：HiRDB 開始時に各サーバに割り当てられている共用メモリセグメント数

《注意事項》

- 次に示す条件式を満たす場合は，このオペランドに pd_max_add_dbbuff_no オペランドの値が仮定されます。

  pd_max_add_dbbuff_shm_no の値＜ pd_max_add_dbbuff_no の値

  省略値が上記の条件を満たす場合も，pd_max_add_dbbuff_no オペランドの値が仮定されます。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- 追加する共用メモリセグメントサイズが SHMMAX オペランドの値を超える場合，SHMMAX オペランドの値を上限値とした複数の共用メモリセグメントに分割されます。追加する共用メモリセグメントサイズを想定して SHMMAX オペランドの値を増やすか，又は分割されたときに不足しないように pd_max_add_dbbuff_shm_no オペランドの値を増やしてください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_no

7.2.10　Java に関するオペランド

Java ストアドプロシジャ又は Java ストアドファンクションを使用する場合に，Java に関するオペランドを指定します。Java ストアドプロシジャ及び Java ストアドファンクションについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

27）　pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル" ～<パス名>

Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合, そのファイル名を絶対パス名で指定します。

《指定値の目安》

このオペランドで指定するファイルの容量が非常に大きくなるため，通常はこのオペランドを指定しないでください。Java ストアドプロシジャ又は Java ストアドファンクションのデバッグ時に指定することをお勧めします。なお，このオペランドで指定するファイルの容量制限はありません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，Java 仮想マシンでの標準出力又は標準エラー出力が無視されます。

《注意事項》

複数プロセスからの同時書き込みがあった場合は，その出力内容を保証しません。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

28）　pd_java_castoff = Y｜N

HiRDB のディクショナリサーバプロセスで，Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。

Y：Java 仮想マシンを起動した場合，トランザクション決着時，又は UAP の切り離し時にサーバプロセスを終了します。

N：Java 仮想マシンを起動してもサーバプロセスが終了しません。

省略した場合は，サーバ共通定義の pd_java_castoff オペランドが解釈値となります。

《指定値のチューニング方法》

Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。

Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に, pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。

#	懸案
1	Java 仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。
2	Java 仮想マシンを使用しないコネクションで実行する SQL でも，別のコネクションで Java 仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多い SQL を実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。 なお，Java 仮想マシンの機能については Java 仮想マシンのドキュメントのオプションを参照してください。

《注意事項》

Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- pd_process_count

7.2.11　システムログファイルに関するオペランド

29）pd_log_dual = Y | N

システムログファイルを二重化するかどうかを指定します。

Y：システムログファイルを二重化します。

N：システムログファイルを二重化しません。

《利点》

システムログファイルを二重化すると，HiRDB はA系及びB系の両方に同じシステムログを取得します。取得したシステムログを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシステムログを読み込めるため，システムの信頼性を向上できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

システムログファイルを二重化する場合は pdlogadpf オペランドで B 系のシステムログファイル名を指定してください。

30）pd_log_remain_space_check = warn | safe

システムログファイルの空き率が**警告値**未満になったときの HiRDB の処理方式を指定します。この機能を**システムログファイルの空き容量監視機能**といいます。システムログファイルの空き容量監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

warn：

システムログファイルの空き率が警告値未満になった場合，警告メッセージ KFPS01162-W を出力します。

safe：

システムログファイルの空き率が警告値未満になった場合，新規トランザクションのスケジューリングを抑止して，サーバ内の全トランザクションを強制終了します。このとき，KFPS01160-E メッセージを出力します。

《指定値の目安》

システムログファイルの空き容量不足によるユニットの異常終了の可能性を低くできるため，safe の指定を推奨します。ただし，safe を指定すると，システムログファイルの空き容量が不足したときに，サーバ内の全トランザクションが強制終了されます。このため，システムログファイルの設計をより正確に行う必要があります。システムログファイルの設計については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，warn が仮定されます。

31）　pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕…

～<パス名>((1～136 文字))

システムログの**自動ログアンロード機能**を使用する場合に，アンロードログファイルの出力先ディレクトリを絶対パス名で指定します。アンロードログファイルを HiRDB ファイルシステム領域に作成する場合は，その HiRDB ファイルシステム領域名を指定してください。なお，**このオペランドに指定するディレクトリ又は HiRDB ファイルシステム領域は，HiRDB の開始前に作成しておいてください。**

また，このオペランドには，サーバごとに異なるディレクトリ又は HiRDB ファイルシステム領域を指定してください。

自動ログアンロード機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値の目安》

空きディスク容量を確認してからディレクトリを指定してください。作成されるアンロードログファイルによって，ディスク容量が圧迫される原因になります。

また，ディスク容量の不足によって，指定したディレクトリにアンロードログファイルが作成できない場合，自動ログアンロード機能が停止します。このようなことが発生する可能性がある場合は，ディレクトリを複数指定することをお勧めします。

ただし，ディレクトリは一つだけの方がデータベース回復時の運用（回復に必要なアンロードログファイルの選別）が若干簡単になります。

ディレクトリを複数指定する場合の補足事項を次に示します。

- ディスク障害などに備えて，異なるパーティションのディレクトリを指定することをお勧めします。
- ディスク容量満杯又はディスク障害などで一つのディレクトリ下にアンロードログファイルが作成できない場合，ほかのディレクトリ下にアンロードログファイルを作成します。このとき，HiRDB はこのオペランドに指定した順番にディレクトリを使用します。

《オペランドの規則》

- ディレクトリは最大 128 個指定できます。
- 複数のディレクトリを指定した場合，同じパス名を指定できません。

《注意事項》

次の場合は，自動ログアンロード機能を使用できません。

- pd_log_unload_check オペランドに N を指定している場合

なお，マルチ HiRDB を利用している場合は，HiRDB ごとに異なるディレクトリを作成してください。同じディレクトリを指定すると，どのアンロードログファイルがどの HiRDB に対応しているか分からなくなる可能性があります。

32)　pd_log_singleoperation = Y | N

このオペランドは，システムログファイルを二重化している場合に指定してください。二重化していない場合は指定する必要はありません。

システムログファイルの片系運転をするかどうかを指定します。システムログファイルに障害が発生して，両系ともに使用できるシステムログファイルがない場合でも，HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了しないで正常な片方の系だけで処理を続行します。これを**システムログファイルの片系運転**といいます。

また，両方のシステムログファイルで処理を続行すること(通常の処理形態)を**システムログファイルの両系運転**といいます。

Y：システムログファイルの片系運転をします。

N：システムログファイルの片系運転をしません。常に両系運転とします。

《前提条件》

pd_log_dual オペランドに Y を指定している必要があります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

33)　pd_log_rerun_reserved_file_open = Y | N

システムログファイルの自動オープンをするかどうかを指定します。

ユニットを再開始するときに，「上書きできる状態」のシステムログファイルがない場合，予約のファイルがあれば HiRDB が予約のファイルをオープンして「上書きできる状態」にし，処理を続行します。これを**システムログファイルの自動オープン**といいます。

予約ファイルを使用するのは，次の場合です。

- 再開始後，最初のシンクポイントダンプを取得するまでの間
- オープンされたファイルグループすべてが「上書きできない状態」の場合

Y：システムログファイルの自動オープンをします（予約ファイルをオープンして使用します）。

N：システムログファイルの自動オープンをしません（予約ファイルを使用しません）。

《利点》

このオペランドに Y を指定すると，ユニットの再開始時に「スワップ先にできる状態」のファイルがなくても，予約のファイルがあればユニットを再開始できます。

ただし，「アンロード待ち状態」のファイルがあるときは，ユニットを停止します。「アンロード待ち状態」のファイルのアンロードを実行した後，再度ユニットを開始してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

34)　pd_log_rerun_swap = Y | N

ユニットの再開始時に，システムログファイルをスワップするかどうかを指定します。

Y：システムログファイルをスワップします。

N：システムログファイルをスワップしません。

《利点》

このオペランドに Y を指定すると，再開始前後でシステムログファイルを物理的に分けられます。このため，再開始前のシステムログファイルをサーバ稼働中に使用できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

35）pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間～<符号なし整数>((1～32580))（単位：秒）

システムログファイルのスワップが完了するまでの待ち時間を秒単位で指定します。指定した時間を過ぎてもシステムログファイルのスワップが完了しない場合は，ユニットが異常終了します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。マシンの性能が低いなどの理由によってシステムログファイルのスワップに時間が掛かる場合は，このオペランドの値を大きくしてください。また，ディスク障害などによってシステムログファイルのスワップに遅延が発生した場合に，より短い時間で異常を検知し，ユニットを異常終了させたいときは，このオペランドの値を小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，180 が仮定されます。

36）pd_log_unload_check = Y | N

システムログファイルのアンロード状態を，HiRDB がチェックするかどうかを指定します。

Y：

アンロード状態をチェックします（通常の運用となります）。

N：

アンロード状態をチェックしません。アンロードの状態に関係なく，次に示す条件がすべて満たされれば，システムログファイルを「スワップ先にできる状態」にします。

- 上書きできる状態
- 抽出完了状態（HiRDB Datareplicator）

このときシステムログファイルの運用方法が，アンロード状態のチェックを解除する運用になります。アンロード状態のチェックを解除する運用については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

N を指定したときの利点を次に示します。

- システムログファイルのアンロード操作がなくなるため，運用方法が簡単になります。
- アンロードログファイルを保管するためのファイル容量が不要になります。

《指定値の目安》

データベースを回復するときにシステムログを使用しない場合（バックアップ取得時点に回復できればよい場合）に N を指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Y が仮定されます。

《注意事項》

N を指定したときの注意事項を次に示します。

- バックアップ取得時点にしかデータベースを回復できません。

- データベースの回復にシステムログが必要なのにこの運用をした場合はデータベースの回復手段がなくなります。

37)　pd_log_max_data_size = ログ入出力バッファ長

～<符号なし整数>((32000～523000))（単位：バイト）

システムログの入出力に使用するバッファの大きさをバイト単位で指定します。

《指定値の目安》

条件式 1 を満たす値を指定してください。pd_rpl_reflect_mode オペランドに uap を指定している場合，又は回復不要 FES を使用している場合は，条件式 1 と条件式 2 の両方を満たす値を指定してください。指定値を最適化する場合は，指定値のチューニング方法に従って指定値を変更してください。

条件式 1：ログ入出力バッファ長≧a

a：72×（一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数）＋ 1344

条件式 2：ログ入出力バッファ長≧b

b：(一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数＋ 1）×128 ＋ 64

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）の次に示す情報を調べて，このオペランドの指定値を変更してください。

- 入出力待ちバッファ面数（# OF BUFFER FOR WAIT I/O）

  入出力待ちバッファ面数の平均値が 100 を大きく超える場合は，このオペランドの値を大きくして，平均値が 100 に近付くようにしてください。

- カレントバッファなしによる待ち回数（# OF WAIT THREAD）

  カレントバッファなしによる待ち回数が 0 以外の値になる場合は，このオペランドの値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，400000 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 32000 が仮定されます）。

《ほかのオペランドとの関連》

pd_log_write_buff_count オペランドとともに，ログ入出力バッファ長を決定してください。

38)　pd_log_write_buff_count = ログ出力バッファ面数

～<符号なし整数>((3～65000))

システムログの出力に使用するバッファの面数を指定します。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）のカレントバッファなしによる待ち回数（# OF WAIT THREAD）を調べて，待ち回数が多ければ，スループットが向上するよう指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 3 が仮定されます）。

《ほかのオペランドとの関連》

pd_log_max_data_size オペランドとともに，ログ出力バッファ面数を決定してください。

39）　pd_log_rec_leng = システムログファイルのレコード長

～<符号なし整数>((1024，2048，4096))（単位：バイト）

システムログファイルのレコード長を指定します。1024，2048，4096 のどれかを指定します。

レコード長は pdloginit コマンドの-l オプションで設定します。-l オプションで設定するレコード長をこのオペランドに指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，4096 が仮定されます。

《注意事項》

- pdloginit コマンドの-l オプションで設定するレコード長と異なるレコード長をこのオペランドに指定すると，そのシステムログファイルはオープンできません。
- システムログファイルのレコード長を変更する方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

40）　pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数

～<符号なし整数>((0～256))

ロールバック処理でシステムログの入力に使用するバッファの面数を指定します。このオペランドに 0 を指定した場合，HiRDB がロールバック用ログ入力バッファ面数を決定します。

《指定値のチューニング方法》

このオペランドに 0 を指定してください。0 を指定してメモリ不足が発生する場合は，このオペランドの指定を省略してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

ユニット内サーバ数×2

41）　pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕

～<符号なし整数>((0～104857600))《0，0》（単位：レコード）

このオペランドは，システムログファイルの自動拡張機能を使用する場合に指定します。

一つのシステムログファイルが拡張契機 1 回当たりに拡張するサイズ，及び拡張できるファイルサイズの上限をレコード数で指定します。

1 回当たりに拡張するサイズを省略するか，又は 0 を指定した場合，システムログファイルの自動拡張を行いません。拡張上限サイズを省略するか，又は 0 を指定した場合，ファイルシステム領域のあるディスクが満杯になるか，システムログファイルの容量が上限に達するまでシステムログファイルの自動拡張を行います。また，1 回当たりに拡張するサイズに拡張上限サイズより大きい値を指定した場合，拡張上限サイズに指定した値まで自動拡張を行います。

システムログファイルの自動拡張機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《前提条件》

pd_large_file_use オペランドに Y を指定しているか，又は指定を省略している必要があります。

《指定値の目安》

- 1 回当たりに拡張するサイズには，pdloginit コマンドの-n オプションでシステムログファイルを作成した際に指定したレコード数を基に指定します。全システムログファイルのレコード数の平均値の 1/10 を計算し，指定してください。
- 通常，拡張上限サイズは省略してください。

《指定値のチューニング方法》

システムログの出力量が自動拡張で拡張するサイズを超えると，システムログファイルが満杯になってユニットダウンする場合があります。その場合は，1 回当たりに拡張するサイズを大きくしてください。また，拡張処理に時間が掛かり，トランザクション性能に影響を及ぼす場合は，1 回当たりに拡張するサイズを小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

7.2.12　シンクポイントダンプファイルに関するオペランド

42)　pd_spd_dual = Y | N

シンクポイントダンプファイルを二重化するかどうかを指定します。

Y：シンクポイントダンプファイルを二重化します。

N：シンクポイントダンプファイルを二重化しません。

《利点》

シンクポイントダンプファイルを二重化すると，HiRDB はA系及びB系の両方に同じシンクポイントダンプを取得します。取得したシンクポイントダンプを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシンクポイントダンプを読み込めるため，システムの信頼性を向上できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

シンクポイントダンプファイルを二重化する場合は，pdlogadpf オペランドで B 系のシンクポイントダンプファイル名を指定してください。

43)　pd_spd_assurance_msg = Y | N

シンクポイントダンプが有効化されたとき，メッセージ KFPS02183-I を出力するかどうかを指定します。

Y：出力します。

N：出力しません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Y が仮定されます。

44)　pd_spd_assurance_count = 有効保証世代数

～<符号なし整数>((1～2))

システム回復時のシンクポイントダンプファイルの入力障害などに備え，HiRDB 稼働中に保存しておくシステムログファイルの範囲を，シンクポイントダンプファイルの世代数によって指定します。この

世代数のことを**有効保証世代数**といいます。ここで指定した世代数分のシンクポイントダンプファイルは，上書きできない状態になります。

《利点》

有効保証世代数を 2 にすると，最新世代のシンクポイントダンプファイルに障害が発生しても，1 世代前のシンクポイントからシステムを回復できるため，信頼性が向上します。

《指定値の目安》

- 信頼性を向上させる場合は有効保証世代数を 2 にしてください。ただし，有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます（二つになります）。
- シンクポイントダンプファイルを二重化して信頼性を向上している場合は，このオペランドを省略するか，又は 1 を指定することをお勧めします。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

《注意事項》

- シンクポイントダンプファイルの必要最低数は**有効保証世代数＋ 1** となります。
- 有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます。上書きできない状態のシンクポイントダンプファイルに対応するシステムログファイルは上書きできない状態になります。したがって，有効保証世代数を 2 にすると，上書きできない状態のシステムログファイルが増えるため，スワップ先にできる状態のシステムログファイル数が不足することがあります。これを防ぐためにはシステムログファイルの容量を考慮してください。

45）　pd_spd_reduced_mode = **縮退運転オプション**

～＜符号なし整数＞((0～2))

シンクポイントダンプファイルの縮退運転をするかどうかを指定します。

縮退運転とは，HiRDB 稼働中又は再開始処理中のファイル障害などによって，シンクポイントダンプファイルの数が運用に必要なファイル数（**有効保証世代数**※＋ 1）以下になった場合でも，最低二つのファイルが使用できれば処理を続行する機能です。

注※　pd_spd_assurance_count オペランドの指定値です。

0：縮退運転を使用しません。

1：縮退運転を使用します。

2：縮退運転を使用し，縮退運転でシンクポイントダンプ取得の契機ごとに警告メッセージを出力します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

46）　pd_spd_reserved_file_auto_open = Y｜N

シンクポイントダンプファイルの自動オープンをするかどうかを指定します。

シンクポイントダンプファイルに障害が発生して，運用に必要なファイル数（**有効保証世代数**※＋ 1）を下回った場合，予約のファイルがあれば HiRDB が予約のファイルをオープンして上書きできる状態にし，処理を続行します。これを**シンクポイントダンプファイルの自動オープン**といいます。

注※　pd_spd_assurance_count オペランドの指定値です。

Y：

シンクポイントダンプファイルの自動オープンをします。運用に必要なファイル数（有効保証世代数＋1）を下回った場合，予約ファイルを自動的にオープンします。

N：

シンクポイントダンプファイルの自動オープンをしません。運用に必要なファイル数（有効保証世代数＋1）を下回っても，予約ファイルを自動的にオープンしません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

このオペランドの指定は pd_spd_reduced_mode オペランドより優先されます。

47）pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長

～<符号なし整数>((32000～4000000))（単位：バイト）

シンクポイントダンプをシンクポイントダンプファイルに入出力するときに使用するバッファ（共用メモリ）の大きさをバイト単位で指定します。

ここで指定した値によってシンクポイントダンプファイルの入出力回数を制御します。

《指定値の目安》

- 通常はこのオペランドを指定する必要はありません。
- 指定値を大きくするほど，シンクポイントダンプファイルに対する入出力回数が少なくなります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，32768 が仮定されます。

48）pd_log_sdinterval = システムログ出力量〔，経過時間〕

シンクポイントダンプの取得間隔を指定します。次に示す条件で指定できます。

- システムログの出力量
- 前回シンクポイントダンプを取得してからの経過時間

システムログ出力量：～<符号なし整数>((100～100000))（単位：ログブロック数）

シンクポイントダンプの取得間隔を指定します。ここで指定したログブロック数分のシステムログが出力されるたびに，シンクポイントダンプを取得します。

経過時間：～<符号なし整数>((0,10～1440))（単位：分）

シンクポイントダンプの取得間隔を指定します。前回シンクポイントダンプを取得してから，ここで指定した経過時間が過ぎると，シンクポイントダンプを取得します。

- 経過時間に 0 を指定すると，HiRDB は経過時間でのシンクポイントダンプを取得しません。
- 前回シンクポイントダンプを取得してからトランザクションが一度も発生しない場合，ここで指定した経過時間が過ぎてもシンクポイントダンプを取得しません。

《指定値の目安》

- HiRDB を再開始するときの所要時間に特に規定のない場合は，このオペランドを指定する必要はありません。
- このオペランドの指定値によって，HiRDB を再開始するときの所要時間が決まります。

このオペランドの指定値を小さくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が短くなります。ただし，シンクポイントダンプの取得回数が増えるため，オンライン中の性能が低下する場合があります。

逆に，このオペランドの指定値を大きくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が長くなります。ただし，シンクポイントダンプの取得回数が減るため，オンライン中の性能が向上する場合があります。

《指定値のチューニング方法》

シンクポイントダンプがどの程度の間隔で取得されているかは，統計解析ユティリティのシステムの稼働に関する統計情報の「シンクポイントダンプ取得間隔時間（SYNC POINT GET INTERVAL）」で確認できます。SYNC POINT GET INTERVAL の平均値を見て，シンクポイントダンプを取得する間隔が長いと判断したら，ここでの指定値を小さくしてください。また，反対に短いと判断したら，ここでの指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- システムログ出力量：5000
- 経過時間：60

ただし，pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，システムログ出力量の省略値は 1000 が仮定されます。

《注意事項》

シンクポイントダンプの取得間隔はシステムログの出力量で決まります。このため，更新系のトランザクションがほとんどない時間帯は，メモリからデータベースに反映するまでの時間が長くなります。このような状態で障害が発生すると，その間に発生したトランザクションの回復時間が長くなります。このようなケースが考えられる場合は，「経過時間」でもシンクポイントダンプの取得間隔を設定してください。

7.2.13　サーバ用ステータスファイルに関するオペランド

49）　pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

：

pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

サーバ用ステータスファイルを定義します。pd_sts_file_name_2～7 オペランドは省略できますが，pd_sts_file_name_1 オペランドは省略できません。

"論理ファイル名"：～<識別子>((1～8 文字))

サーバ用ステータスファイルの論理ファイル名を指定します。ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系ステータスファイル名"：～<パス名>((167 文字以内))

A 系のステータスファイル名を絶対パス名で指定します。

"B 系ステータスファイル名"：～<パス名>((167 文字以内))

B 系のステータスファイル名を絶対パス名で指定します。

《指定値の目安》

- A 系及び B 系のステータスファイル名には，pdstsinit コマンドで作成したステータスファイルを指定してください。pdstsinit コマンドで作成していないステータスファイルを指定すると，そのステータスファイルは**実体のないステータスファイル**となります。
- ステータスファイルに障害が発生すると，HiRDB はステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要となります。
- A 系と B 系には同じレコード長，及び同じ容量のステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- ステータスファイルは，A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系ステータスファイル名，及び B 系ステータスファイル名には同じ名称を重複して指定できません。
- HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥sts01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。sts01 については大文字と小文字の区別をします。

《留意事項》

- HiRDB を正常開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，全ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときは pd_sts_file_name_1～7 の中で最初に指定したステータスファイルが現用ファイルになります。残りのファイルのうちオープンできたファイルが予備ファイルになり，オープンできないファイルは予約ファイルになります。
- HiRDB を再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。

●実体のないステータスファイルの使い方

実体のないステータスファイルを指定すると，HiRDB の稼働中にステータスファイルを新規追加できます。例えば，ステータスファイルの障害などで予備ファイルが少なくなったときに，実体のないステータスファイルを予備ファイルにします。実体のないステータスファイルを予備ファイルにする手順を次に示します。

〈手順〉

1. pdstsinit コマンドで，システムファイル用の HiRDB ファイルシステム領域にステータスファイルを作成します。
2. pdstsopen コマンドで，ステータスファイルをオープンします。

この操作は HiRDB 稼働中に実行できます。HiRDB を一度停止する必要はありません。

- **メリット及びデメリット**

  実体のないステータスファイルを定義すると，HiRDB ファイルシステム領域の占有量は小さくなります。しかし，実体のないステータスファイルを予備ファイルとして追加するときに，システムファイル用の HiRDB ファイルシステム領域に十分な空き（ファイルを追加するだけの空き領域）がないと追加できないため，システムの信頼性は低くなります。

  実体のないステータスファイルを定義しないと，HiRDB ファイルシステム領域の占有量は大きくなるが，ファイル障害によるスワップ先が保証されるため，信頼性は高くなります。

- **注意事項**

実体のないステータスファイルを定義した場合，HiRDB は HiRDB の開始時にステータスファイルに異常があると認識します。このため，pd_sts_initial_error オペランドに stop（省略値）を指定していると，HiRDB を開始できないので注意してください。実体のないステータスファイルを定義する場合は，pd_sts_initial_error オペランドに continue 又は excontinue を指定してください。また，HiRDB を開始する前に現用ファイルを pd_sts_last_active_file オペランドに指定する必要があります。

7.2.14　サーバ用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

50）pd_sts_initial_error = stop | continue | excontinue

サーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，又はバックエンドサーバ）の開始時，HiRDB はサーバ用ステータスファイルの現用ファイルを特定する処理を実行します。この特定処理で，次に示す異常を検知したときの HiRDB の処置を指定します。

- サーバ用ステータスファイルの実体がない
- サーバ用ステータスファイルの障害を検知した

現用ファイルの特定処理は pd_sts_file_name_1～7 オペランドに指定したサーバ用ステータスファイルを対象とします。

stop：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。この場合，異常を検知したステータスファイルの対策を実施した後に HiRDB を開始してください。

continue 又は excontinue：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知しても，現用ファイルが正常な場合はサーバの開始処理を続行します。ただし，pd_sts_singleoperation オペランドの値（ステータスファイルの片系運転をするかどうか）によっては開始処理を中止することがあります。pd_sts_singleoperation オペランドとの関係を次に示します。

●pd_sts_singleoperation オペランドとの関係

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
continue	サーバ用ステータスファイルの異常を検知した場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。
stop（省略値）	サーバ用ステータスファイルの異常を検知した場合，HiRDB が現用ファイルを特定してサーバの開始処理を続行します。ただし，A 系と B 系のファイルが次に示す表（**HiRDB が現用ファイルを特定できないケース**）の条件のどれかに当てはまる場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，	HiRDB が現用ファイルを特定できない場合は，HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
	HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	

●HiRDB が現用ファイルを特定できないケース

pd_sts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
continue	障害閉塞	障害閉塞
	障害閉塞	オープン（初期状態）
	障害閉塞	ファイル実体なし
	オープン（初期状態）	障害閉塞
	オープン（初期状態）	ファイル実体なし
	ファイル実体なし	オープン（初期状態）
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし
excontinue	障害閉塞	障害閉塞
	障害閉塞	ファイル実体なし
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし

したがって，HiRDB 管理者の操作をなるべく少なくしたい場合（pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定するケースを減らしたい場合）は，次に示すようにオペランドを指定してください。

- pd_sts_initial_error オペランドに excontinue を指定
- pd_sts_singleoperation オペランドに stop を指定

《指定値の目安》

各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
サーバ開始時の HiRDB の処理	サーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	一部のサーバ用ステータスファイルに異常があっても，現用ファイルが正常な場合はサーバの開始処理を続行します。
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB 開始時の障害対策を簡易化する場合は continue 又は excontinue を指定します。
メリット	サーバの開始時に該当するサーバの全サーバ用ステータスファイルの正常が保証されます。このため，HiRDB の開始後に現用ファイルの障害	サーバの開始時に一部のサーバ用ステータスファイルの障害を検知しても，残りの正常なファイルだけで HiRDB を開始できます。このため，

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
	が発生した場合, 予備ファイルにスワップできます。	HiRDB の停止時間を短くできます。この場合，予備ファイルが少なくなっているため，障害状態のステータスファイルをすぐに修復する必要があります。
デメリット	サーバ用ステータスファイルの障害によって HiRDB の開始処理が中止される可能性が高くなります。	予備ファイルが少ない状態で HiRDB を稼働することがあるため，システムの信頼性が低下します。予備ファイルの数によってはサーバ用ステータスファイルをスワップできないことがあります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，stop が仮定されます。

《注意事項》

- 現用ファイルが両系とも異常の場合は，このオペランドの指定値に関係なくサーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。
- HiRDB を開始する前に現用ファイルを pdstsinit コマンドで初期化しないでください。初期化すると HiRDB を再開始できません。
- このオペランドに excontinue を指定し，すべての世代のステータスファイルが片系運転でかつオープン（初期状態）の場合，HiRDB は起動しません。

《備考》

オペランドの指定値と HiRDB の処理及び HiRDB 管理者の処置を次に示します。

●オペランドの指定値と HiRDB の処理

pd_sts_initial_errorオペランドの値	ステータスファイル障害	pd_sts_singleoperationオペランドの値	HiRDBが現用ファイルを特定できるか	pd_sts_last_active_fileの指定	pd_sts_last_active_fileオペランドの値と，オープンできたステータスのうち，最新のファイルが一致するか	現用ファイル障害	pd_sts_last_active_sideオペランドの指定	pd_sts_last_active_sideオペランドに指定されたファイルが使用できるか	表「HiRDBの処理及びHiRDB管理者の処置」の対応番号
stop（省略値）	なし	－	－	－	－	－	－	－	[1]
	あり	－	－	－	－	－	－	－	[5]
continue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
excontinue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]

（凡例）－：該当しません（条件がHiRDBの処理に影響を与えません）。

●HiRDB の処理及び HiRDB 管理者の処置

対応番号	HiRDB の処理	HiRDB 管理者の処置
[1]	HiRDB の開始処理を続行します。	なし
[2]	HiRDB が最新の現用ファイルを特定して，開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[3]	pd_sts_last_active_file オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[4]	pd_sts_last_active_file 及び pd_sts_last_active_side オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[5]	pd_sts_initial_error オペランドに stop が指定されているため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000010 が示す対策をしてください。

対応番号	HiRDB の処理	HiRDB 管理者の処置
[6]	前回稼働時の現用ファイルが特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000016 が示す対策をしてください。
[7]	HiRDB が特定した現用ファイルの正常な系と，pd_sts_last_active_side オペランドに指定された系が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000017 が示す対策をしてください。
[8]	HiRDB が特定した現用ファイル名と，pd_sts_last_active_file オペランドに指定したファイル名が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000015 が示す対策をしてください。
[9]	前回稼働時の現用ファイルの正常な系が特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000018 が示す対策をしてください。

51）　pd_sts_singleoperation = stop | continue

サーバ用ステータスファイルの片系運転をするかどうかを指定します。予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，正常な系（片方の系）だけで処理を続行することを**ステータスファイルの片系運転**といいます。ステータスファイルの片系運転については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

なお，予備ファイルがある状況で現用ファイルの片系に障害が発生した場合は，このオペランドの指定値に関係なく，ステータスファイルをスワップして処理を続行します（片系運転にはなりません）。

stop：

片系運転をしません。片系運転になる場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。HiRDB が異常終了した場合，予備ファイルを準備した後に HiRDB を開始してください。

continue：

片系運転をします。片系運転になると KFPS01044-I メッセージが出力されます。片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了したりすると，HiRDB を再開始できなくなるため，片系運転になった場合はすぐに予備ファイルを準備してください。

《指定値の目安》

- stop を指定してシステムの信頼性を上げることを推奨します。また，現用ファイルの障害に対しては，予備ファイルの数を増やして備えることを推奨します。
- 各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB を停止させないことを重視する場合は continue を指定します。
メリット	予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，片系運転をしないで HiRDB	予備ファイルがない状況で現用ファイルの片系に障害が発生しても，処理を続行できます。この

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
	を異常終了します。このため，現用ファイルの内容が失われる可能性が低くなります。	ため，ステータスファイルの障害によってHiRDB が停止する可能性が低くなります。
デメリット	ステータスファイルの障害によって HiRDB が停止する可能性が高くなります。ただし，予備ファイルの数を増やすことで可能性を低くできます。	片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，現用ファイルの内容が失われるため，HiRDB を再開始できなくなります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，stop が仮定されます。

《ほかのオペランドとの関連》

pd_sts_singleoperation 及び pd_sts_initial_error オペランドの指定値の組み合わせによって，ステータスファイル障害時の HiRDB の処理が決定します。したがって，これら二つのオペランドの指定値は一緒に考えるようにしてください。

52)　pd_sts_last_active_file = "論理ファイル名"

～<識別子>((1～8 文字))

ディクショナリサーバの開始時に現用にするステータスファイルの論理ファイル名を指定します。HiRDB は，このオペランドに指定されたファイルと HiRDB が特定した現用ファイルを比較します。ファイルが一致する場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しますが，一致しない場合は開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **全ステータスファイルを初期化した直後に HiRDB を開始する場合**

   pd_sts_file_name1～7 オペランドに指定した最も小さい番号の正常な論理ファイル名を指定してください。この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，全ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_sts_last_active_side オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   全ステータスファイルを初期化してください。その後，1 の方法を実施してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

5. 実体のないステータスファイルを指定している場合

現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I
- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージのうちで，最も直前に出力されたメッセージに表示されているステータスファイルが現用ファイルになります。

53）　pd_sts_last_active_side = A | B

現用ファイルの片系に障害が発生している状態でディクショナリサーバを開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- pd_sts_last_active_file オペランドを指定している

7.2.15　作業表用ファイルに関するオペランド

54）　pdwork -v "HiRDB ファイルシステム領域名称"〔，"HiRDB ファイルシステム領域名称"〕…

～< 141 文字以内のパス名>

作業表用ファイル用の HiRDB ファイルシステム領域の名称を指定します。作業表用ファイルとは，SQL 文を実行するときに一時的に情報を格納するファイルのことで HiRDB が自動的に作成します。作業表用ファイルを必要とする SQL 文については，マニュアル「HiRDB Version 8 解説」を参照してください。

なお，このオペランドは省略しないでください。省略した場合，作業表用ファイルを必要とする SQL 文を実行できないことがあります。

《注意事項》

- このオペランドに指定する HiRDB ファイルシステム領域は pdfmkfs コマンドで初期設定した HiRDB ファイルシステム領域を指定してください。
- 作業表用ファイルの容量が大きい場合は，領域長が大きい HiRDB ファイルシステム領域を指定してください。作業表用ファイルの容量見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

  ただし, HiRDB ファイルシステム領域を pdfmkfs -a コマンドで初期設定した場合, -n オプションで指定した値を使い切ると，HiRDB は領域サイズを自動で拡張します。pdfmkfs コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。
- 作業表用ファイル用の HiRDB ファイルシステム領域は，RD エリア及びシステムファイル用の HiRDB ファイルシステム領域と同じものを指定できません。

- このオペランドに HiRDB ファイルシステム領域を二つ以上指定している場合，作業表用ファイルの作成時にエラーが発生した HiRDB ファイルシステム領域は，それ以降，原則使用しません。ほかの HiRDB ファイルシステム領域だけを使用します。

  例外として，容量不足やファイル数オーバーなどによって，ほかのすべての HiRDB ファイルシステム領域で作業表用ファイルの作成が失敗した場合に，使用されなくなった HiRDB ファイルシステム領域で作業表用ファイルを作成しようとします。その結果，作業表用ファイルが正常に作成された場合，その HiRDB ファイルシステム領域を再使用するようになります。

  なお，ディクショナリサーバを一度終了して開始すると，作業表用ファイルの作成時にエラーが発生して使用されなくなった HiRDB ファイルシステム領域が使用できるようになります。

《オペランドの規則》

- HiRDB ファイルシステム領域名称は必ず 1 個以上指定してください。
- HiRDB ファイルシステム領域名称は最大 16 個指定できます。
- このオペランドはディクショナリサーバ定義内に一つだけ指定できます。複数個指定した場合は最初に指定したものが有効となります。
- バックエンドサーバと同じ HiRDB ファイルシステム領域を指定しないでください。

7.2.16　システムログファイルの構成に関するオペランド

55)　pdlogadfg -d sys -g ファイルグループ名〔ONL〕

システムログファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシステムログファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～200 個指定できます。

《オペランドの規則》

このオペランドは 2～200 個指定できます。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

56)　pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名"〔-b "システムログファイル名"〕

ファイルグループを構成するシステムログファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "システムログファイル名"：～<パス名>((167 文字以内))

ファイルグループを構成するシステムログファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

-b "システムログファイル名"：～<パス名>((167 文字以内))

システムログファイルを二重化（pd_log_dual オペランドに Y を指定）する場合に，B 系のシステムログファイル名を絶対パス名で指定します。pd_log_dua オペランドに Y を指定しない場合は，システムログファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

《オペランドの規則》

HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥log01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。log01 については大文字と小文字の区別をします。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

7.2.17　シンクポイントダンプファイルの構成に関するオペランド

57）　pdlogadfg -d spd -g ファイルグループ名〔ONL〕

シンクポイントダンプファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシンクポイントダンプファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～30 個指定できます。

《オペランドの規則》

このオペランドは 2～60 個指定できます。

58）　pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕

ファイルグループを構成するシンクポイントダンプファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "シンクポイントダンプファイル名"：～<パス名>((167 文字以内))

ファイルグループを構成するシンクポイントダンプファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

-b "シンクポイントダンプファイル名"：～<パス名>((167 文字以内))

シンクポイントダンプファイルを二重化（pd_spd_dual = Y を指定）する場合に，B 系のシンクポイントダンプファイル名を絶対パス名で指定します。pd_spd_dual = Y の指定がない場合は，シンクポイントダンプファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

《オペランドの規則》

HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥sync01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。sync01 については大文字と小文字の区別をします。

7.2.18　プラグインに関するオペランド

59)　pdplgprm -n プラグイン名称〔 -s 共用メモリサイズ〕

プラグインの名称及びプラグインが使用する共用メモリの大きさを指定します。プラグインを使用しない場合，又はプラグインがディクショナリサーバで稼働しない場合は，このオペランドを省略してください。

《前提条件》

ここで指定するプラグインは，あらかじめ pdplgrgst コマンドで HiRDB に登録しておいてください。

-n プラグイン名称：～<識別子>((1～30 文字))

ここで指定するプラグイン名称については，プラグインに付いているマニュアルを参照してください。

-s 共用メモリサイズ：～<符号なし整数>((1～2000000))《0》(単位：キロバイト)

プラグインが使用する共用メモリの大きさをキロバイト単位で指定します。プラグインが使用する共用メモリの大きさについては，プラグインに付いているマニュアルを参照してください。

8 バックエンドサーバ定義

この章では, バックエンドサーバ定義の各オペランドの内容について説明します。

8.1　オペランドの形式

バックエンドサーバ定義ではバックエンドサーバの実行環境を定義します。ここでは，バックエンドサーバ定義の各オペランドの形式について説明します。なお，表中の「番号」は，「8.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

番号	形式
1	[set pd_max_bes_process = 1 バックエンドサーバ当たりの最大起動プロセス数] ※
2	[set pd_process_count = 常駐プロセス数 [，サーバ開始時の常駐プロセス数]] ※
3	[set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔] ※
4	[set pd_max_ard_process = 非同期 READ プロセス数] ※
5	[set pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数]

(2)　作業表

番号	形式
6	[set pd_work_buff_mode = each | pool] ※
7	[set pd_work_buff_size = 作業表用バッファ長] ※
8	[set pd_work_buff_expand_limit = 作業表用バッファの増分上限値] ※

(3)　システム監視

番号	形式
9	[set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値] ※
10	[set pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値] ※

(4)　排他制御

番号	形式
11	[set pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ] ※
12	[set pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数] ※
13	[set pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和] ※

番号	形式
14	[set pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数] ※
15	[set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数] ※
16	[set pd_dbsync_lck_release_count = シンクポイント時のグローバルバッファ排他解除間隔] ※

(5)　バッファ

番号	形式
17	[set pd_sql_object_cache_size = SQL オブジェクト用バッファ長] ※

(6)　共用メモリ

番号	形式
18	[set pd_bes_shmpool_size = バックエンドサーバ用共用メモリサイズ] ※

(7)　RPC トレース情報

番号	形式
19	[set pd_rpc_trace = Y | N] ※
20	[set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"] ※
21	[set pd_rpc_trace_size = RPC トレースを取得するファイルの容量] ※

(8)　トラブルシュート情報

番号	形式
22	[set pd_module_trace_max = モジュールトレース格納最大数] ※
23	[set pd_module_trace_timer_level = モジュールトレース出力時刻取得方法] ※
24	[set pd_pth_trace_max = 通信トレース格納最大数]

(9)　グローバルバッファ

番号	形式
25	[set pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値] ※
26	[set pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値] ※

(10)　プラグインインデクスの遅延一括作成

番号	形式
27	〔set pd_plugin_ixmk_dir = "インデクス情報ファイルを作成するディレクトリ名"又は "インデクス情報ファイルを作成する HiRDB ファイルシステム領域名"〕

(11)　Java

番号	形式
28	〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※
29	〔set pd_java_castoff = Y ｜ N〕

(12)　HiRDB External Data Access 機能

番号	形式
30	〔set pd_foreign_server_libpath = "外部サーバのクライアントライブラリのパス名"〔，"外部サーバのクライアントライブラリのパス名"〕…〕

(13)　システムログファイル

番号	形式
31	〔set pd_log_dual = Y ｜ N〕※
32	〔set pd_log_remain_space_check = warn ｜ safe〕※
33	〔set pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕…〕
34	〔set pd_log_singleoperation = Y ｜ N〕※
35	〔set pd_log_rerun_reserved_file_open = Y ｜ N〕※
36	〔set pd_log_rerun_swap = Y ｜ N〕※
37	〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕※
38	〔set pd_log_unload_check = Y ｜ N〕※
39	〔set pd_log_max_data_size = ログ入出力バッファ長〕※
40	〔set pd_log_write_buff_count = ログ出力バッファ面数〕※
41	〔set pd_log_rec_leng = システムログファイルのレコード長〕※
42	〔set pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数〕※
43	〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕※

(14)　シンクポイントダンプファイル

番号	形式
44	〔set pd_spd_dual = Y｜N〕※
45	〔set pd_spd_assurance_msg = Y｜N〕※
46	〔set pd_spd_assurance_count = 有効保証世代数〕※
47	〔set pd_spd_reduced_mode = 縮退運転オプション〕※
48	〔set pd_spd_reserved_file_auto_open = Y｜N〕※
49	〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕※
50	〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕※

(15)　サーバ用ステータスファイル

番号	形式
51	set pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"
	〔set pd_sts_file_name_2 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_3 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_4 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_5 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_6 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕
	〔set pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

(16)　サーバ用ステータスファイル（障害発生時）

番号	形式
52	〔set pd_sts_initial_error = stop｜continue｜excontinue〕※
53	〔set pd_sts_singleoperation = stop｜continue〕※
54	〔set pd_sts_last_active_file = "論理ファイル名"〕
55	〔set pd_sts_last_active_side = A｜B〕

(17)　バックエンドサーバ接続保持機能

番号	形式
56	〔set pd_bes_connection_hold = Y｜N〕※
57	〔set pd_bes_conn_hold_trn_interval = バックエンドサーバ接続保持時間〕※

(18)　作業表用ファイル

番号	形式
58	pdwork -v "HiRDB ファイルシステム領域名称"〔，"HiRDB ファイルシステム領域名称"〕…

(19)　システムログファイルの構成

番号	形式
59	{{pdlogadfg -d sys -g ファイルグループ名〔ONL〕}}
60	{{pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名" 〔-b "システムログファイル名"〕}}

(20)　シンクポイントダンプファイルの構成

番号	形式
61	{{pdlogadfg -d spd -g ファイルグループ名〔ONL〕}}
62	{{pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕}}

(21)　プラグイン

番号	形式
63	{{〔 pdplgprm -n プラグイン名称 〔 -s 共用メモリサイズ〕〕}}

(22)　HiRDB External Data Access 機能（環境変数）

番号	形式
64	〔putenv 外部サーバで必要な環境変数〕

注※

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

8.2　オペランドの説明

8.2.1　プロセスに関するオペランド

1）　pd_max_bes_process = 1 バックエンドサーバ当たりの最大起動プロセス数

～<符号なし整数>((1～2048))

1 バックエンドサーバ当たりの**最大起動プロセス数**を指定します。マルチフロントエンドサーバの場合，1 バックエンドサーバに pd_max_users の値を超える処理が集中することがあります。このオペランドには，1 バックエンドサーバのプロセスを pd_max_users の値を超えて起動する場合の最大起動プロセス数を指定します。

《前提条件》

このオペランドはマルチフロントエンドサーバのときに指定します。

《指定値の目安》

- 1 バックエンドサーバに集中する可能性がある処理の最大値を次に示します。

  pd_max_users オペランドの値×フロントエンドサーバ数

  ここで求めた値を最大値として，バックエンドサーバへの処理の集中度を考慮してこのオペランドの値を指定してください。必要以上に大きな値を指定すると，メモリを圧迫する原因となります。

- pd_max_users の値よりも小さな値を指定した場合は pd_max_users の値を仮定します。このとき，警告メッセージ（KFPS01888-W）を出力します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，pd_max_users の値が仮定されます。

《指定値のチューニング方法》

最大起動プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

2）　pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕

～<符号なし整数>((0～2048))

常駐プロセス数：

バックエンドサーバの常駐プロセス数を指定します。常駐プロセスとは，サーバ開始時からあらかじめ起動しておくプロセスのことです。

《利点》

バックエンドサーバで並行処理できるトランザクションが使用するプロセスを，あらかじめシステム開始時に起動して常駐しておくことで，新たにトランザクションが入力されてもプロセス起動時間を削減できます。

ただし，HiRDB の開始に時間が掛かります。

《指定値の目安》

- バックエンドサーバのサーバプロセスのプロセス固有領域とプロセサの実メモリ量から値を求めてください。サーバプロセスのプロセス固有領域については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

- マルチフロントエンドサーバ構成で，更に pd_max_bes_process 又は pd_max_dic_process オペランドを指定している場合は，次に示す条件を満たすようにこのオペランドを指定してください。

  pd_process_count の値≦pd_max_bes_process 又は pd_max_dic_process の値

- このオペランドには，バックエンドサーバの**最大起動プロセス数**（pd_max_bes_process※の値）の範囲内で値を指定します。

  注※

  pd_max_dic_process 又は pd_max_bes_process オペランドを省略している場合は，pd_max_users の値が仮定されます。

《指定値のチューニング方法》

常駐プロセス数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- 常駐プロセス数はメモリと密接に関連しているため，必要以上に大きい値を指定すると，HiRDB を開始できなくなることがあります。
- 常駐プロセス数を超えたプロセスが必要になると，最大起動プロセス数までプロセスを動的に起動します。ただし，pd_max_server_process オペランドの指定値によっては，最大起動プロセス数分のプロセスを起動できないことがあります。

《オペランドの省略値》

このオペランドを省略（又は 0 を指定）すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，最大起動プロセス数が仮定されます。

サーバ開始時の常駐プロセス数：

HiRDB の開始処理時に常駐化する常駐プロセスの数を指定します。

通常，常駐プロセスの起動処理は HiRDB の開始処理中に行われます。このため，常駐プロセス数が多くなると，それに比例して HiRDB の開始処理時間が長くなります。目安として，100MIPS 程度のサーバマシンでプロセスを一つ起動するのに約 1 秒掛かります。

サーバ開始時の常駐プロセス数の指定有無による処理の違いを次に示します。

●サーバ開始時の常駐プロセス数を指定しない場合（pd_process_count = 500 と指定した場合）

すべての常駐プロセスの起動処理を HiRDB の開始処理中に実行します。常駐プロセス（この場合は 500 個）がすべて起動されないと，HiRDB を開始できません。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 500 秒掛かります。

●サーバ開始時の常駐プロセス数を指定する場合（pd_process_count = 500,50 と指定した場合）

一部の常駐プロセスの起動処理を HiRDB の開始処理中に実行し，残りの常駐プロセスの起動処理を HiRDB の開始処理終了後に実行します。一部の常駐プロセス（この場合は 50 個）が起動されれば，HiRDB を開始できます。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 50 秒掛かります。残りの常駐プロセス（この場合は 450 個）は，HiRDB の開始処理終了後に起動されます。

《利点》

HiRDB の開始処理時間を短縮できます。系切り替え機能を使用している場合など，HiRDB の開始処理時間をなるべく短くしたいときに適用します。

《指定値の目安》

HiRDB の開始処理終了直後に必要なプロセス数を指定します。

《注意事項》

サーバ開始時の常駐プロセス数を指定する場合は，クライアント環境定義の PDCWAITTIME オペランドの値を見直してください。

HiRDB の開始直後に，サーバ開始時の常駐プロセス数の値を超える UAP が実行されると，残りの常駐プロセスを起動した後にトランザクション処理が実行されます。したがって，クライアント環境定義の PDCWAITTIME オペランドの値が小さいと，一部の UAP がタイムアウトで処理できない場合があります。

PDCWAITTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

3）pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔

～<符号なし整数>((0～1440))（単位：分）

HiRDB の非常駐サーバプロセスの停止処理をする間隔を分単位で指定します。この機能は，稼働中のサーバプロセス数が常駐プロセス数（pd_process_count オペランドの値）より多くなった場合に適用されます。この機能によって停止するサーバプロセス数は HiRDB が自動的に算出して決定します。

《利点》

業務（稼働中のサーバプロセス数）がピークのときに再利用できる非常駐サーバプロセスが増えるため，メモリなどのプロセス資源の利用効率が向上します。

《指定値の目安》

- 例えば，一日のうち 1 時間だけ業務のピーク時間があり，その時間内でピークとなる間隔が 2 分ぐらいの場合，このオペランドに 2 を指定します。
- ピーク時に同時実行されるサーバプロセス数が常駐プロセス数以下の場合，この機能を使用しても効果は期待できないため，このオペランドを省略します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《指定値のチューニング方法》

システムの稼働に関する統計情報をサーバ単位に 1 週間取得してください。「サービス実行中のサーバプロセス数（# OF PROCESS ON SERVICE)」の値から業務のピークを求めてください。そのピークが現在設定されている常駐プロセス数（pd_process_count オペランドの値）を超えている場合は，ピークごとの間隔を求めてその時間を設定します。

ただし，サーバマシンのメモリ，CPU などの資源に余裕がある場合は，不足している分のプロセス数を常駐プロセス数に加算する方（pd_process_count オペランドの値を大きくする方）がこのオペランドを指定するより性能が向上します。

《注意事項》

このオペランドを省略するか，又は 0 を指定した場合は 10 秒間隔でサービス待ち状態の非常駐サーバプロセスを調査し，サービス待ち状態の非常駐サーバプロセスがあれば，その非常駐サーバプロセスを停止します。

4）pd_max_ard_process = 非同期 READ プロセス数

～<符号なし整数>((0～256))

非同期 READ 機能を使用する場合にこのオペランドを指定します。このオペランドには非同期 READ 処理に必要なプロセス数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（バックエンドサーバ又はディクショナリサーバ）当たりのプロセス数を指定します。非同期 READ 機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《前提条件》

pdbuffer オペランドの-m オプションで 1 以上を指定している必要があります。

《指定値の目安》

- 0 又は 1 を指定してください。ただし，pdbuffer オペランドの-m オプションの値が 2～256 の場合は，-m オプションと同じ値を指定してください。-m オプションの値が 257 以上の場合は，RD エリアやシステムファイルを格納するディスク装置の数（HiRDB/パラレルサーバの場合は 1 サーバ当たりの数）と同じ値を指定するか，又は 256 を指定してください。
- このオペランドの値を大きくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に処理時間を短縮できます。このオペランドの値を小さくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に，非同期 READ プロセスの処理完了待ちが発生する分，処理時間が余計に掛かることがあります。
- このオペランドの値×サーバ数分のプロセスが起動されるため，リソース（共用メモリ及びメッセージキュー）を考慮してこのオペランドの値を決定してください。共用メモリ及びメッセージキューの見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値のチューニング方法》

指定値（非同期 READ プロセス数）のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《オペランドの規則》

このオペランドに 0 を指定すると，非同期 READ 機能を使用しません。

1：1 スタンバイレス型系切り替え機能を使用している場合，正規 BES とそれに対応する代替 BES に同じ値を指定してください。

《ほかのオペランドとの関連》

このオペランドの値を変更した場合は pd_max_server_process オペランドの値を見直してください。

5）pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数

～<符号なし整数>((2～255))

すべてのバッファプールに対して，**デファードライト処理の並列 WRITE 機能**を使用する場合にこのオペランドを指定します。このオペランドには並列処理するプロセス数を指定します。プロセス数を増加させることで書き込み処理時間を短縮できます。デファードライト処理の並列 WRITE 機能については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《指定値の目安》

デファードライト処理の並列 WRITE 機能が有効になる最小値の 2 を指定してください。また，マニュアル「HiRDB Version 8 システム運用ガイド」の「デファードライト処理のチューニング」を参照してオペランドの値を決定してください。

《注意事項》

デファードライト処理の並列 WRITE 機能を指定すると，プロセス数が増加するため，CPU 利用率が上がります。

8.2.2　作業表に関するオペランド

6）pd_work_buff_mode = each | pool

HiRDB が作業表を作成するときのバッファの確保方式を指定します。

each：個々の作業表ごとにバッファを確保します。

pool：サーバプロセス単位にバッファプールとして一括して確保します。

《指定値の目安》

- 通常は pool を指定してください。特に大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合には pool を指定してください。
- あらかじめ，作業表用バッファに使用できるプロセス固有領域サイズが決まっている場合，pool を指定してください。pool を指定すると，HiRDB が効率良く作業表用バッファを各作業表に配分します。

  この場合，pd_work_buff_size に指定した値でプロセス固有領域を占有し，そのプール内で作業表の入出力をバッファリングします。したがって，pd_work_buff_size に指定した値以上にプロセス固有メモリを占有することはありません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，pool が仮定されます（pd_sysdef_default_option オペランドに v6compatible を指定している場合は each が仮定されます）。

7）pd_work_buff_size = 作業表用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((128～1000000))
- 64 ビットモードの場合：((128～4000000000))

HiRDB が作成する作業表のバッファの大きさをキロバイト単位で指定します。

項目	pd_work_buff_mode = each の場合	pd_work_buff_mode = pool の場合
利点	作業表バッファ長が大きいと，データ操作時に発生する作業表用ファイルへの入出力回数を削減できるため，作業表を使用する SQL の実行時間を削減できます。ただし，各サーバのプロセス固有メモリを使用するので，システム全体のメモリ量（実メモリ，仮想メモリ）を考慮して指定してください。 特に pd_work_buff_mode=each の場合，メモリは「pd_work_buff_size の値×必要作業表数」分確保されます。したがって，必要以上に大きな値を指定すると，ほかのプロセスで仮想メモリ不足が生じることがあります。	
適用基準	大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合に指定します。このとき，pd_work_buff_mode = pool を指定してください。	
指定値の目安	• 個々の作業表ごとに確保するバッファの大きさを指定します。 • 作業表用バッファ長に，作業表メモリ容量より大きな値を指定すると，作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。作業表メモリ容量の見積もり式は次のとおりです。 **作業表メモリ容量＝対象となる作業表容量※÷2**	• サーバプロセス単位に一括して確保する作業表用バッファプールの大きさを指定します。 • 大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合は，4352～5120 を指定してください。指定すると，ソートでの入出力単位が大きくなるため，ソート処理時間が短くなります。 • 作業表用バッファ長に，SQL 文単位の作業表メモリ総容量を上限として，大きな値を指定するほど作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間

項目	pd_work_buff_mode = each の場合	pd_work_buff_mode = pool の場合
		が短くなります。SQL 文単位の作業表メモリ総容量の見積もり式は次のとおりです。 **SQL 文単位の作業表メモリ総容量＝ a×b ＋ c×d**
注意事項	複数のユーザが同時に実行する場合，及び複数個の作業表を使用する SQL 文などを実行する場合には，指定したサイズのバッファが，個々の作業表ごとに確保されます。そのため，大きな値を指定するとメモリを圧迫する原因となります。	SQL 文単位に使用する作業表の数に比べて作業表用バッファ長に指定した値が小さいと，each を指定した場合に比べて処理時間が増加することがあります。具体的には，「SQL 文単位の作業表最大数×128」以上の値を指定してください。SQL 文単位の作業表最大数の見積もり式は次のとおりです。 **SQL 文単位の作業表最大数＝ b ＋ d**
オペランドの規則	指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。	• 指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。 • 384 以上の値を指定してください。384 未満の値が指定されると，自動的に 384 に切り上げられます。
省略値	このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，512 が仮定されます。	このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。 • 32 ビットモードの場合：1024 • 64 ビットモードの場合：5120

a：↑｛作業表（列情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

b：作業表（列情報格納用）の最大数※

c：↑｛作業表（位置情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

d：作業表（位置情報格納用）の最大数※

注※

これらの項目の見積もり方法については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

8）　pd_work_buff_expand_limit = 作業表用バッファの増分上限値

～＜符号なし整数＞（単位：キロバイト）

- 32 ビットモードの場合：((128～1000000))
- 64 ビットモードの場合：((128～4000000000))

HiRDB が作成する作業表用バッファ長は pd_work_buff_size オペランドで指定します。この作業表用バッファの不足時，作業表用バッファを自動増分する場合にこのオペランドを指定します。このオペランドで指定した値になるまで作業表用バッファを増分します。

例えば，オペランドに次の指定をした場合，通常 1,024 キロバイトの作業表用バッファを確保します。作業表用バッファが 1,024 キロバイトで不足すると，2,048 キロバイトまで作業表用バッファを増分します。

- pd_work_buff_size = 1024

- pd_work_buff_expand_limit = 2048

次に示すときに HiRDB は作業表用バッファを増分します。

- 結合方式がハッシュジョイン，又は副問合せの実行方式にハッシュ実行が適用されている場合に，必要な作業表用バッファが確保できないとき
- 複数の作業表を同時に使用する場合，各作業表に作業表用バッファを 128 キロバイト割り当てて作業表用バッファが不足するとき

《前提条件》

pd_work_buff_mode オペランドを省略するか，又は pool を指定している必要があります。

《利点》

作業表用バッファ不足（pd_work_buff_size オペランドの指定値不足）で UAP がエラーになるのを防げます。

《注意事項》

次に示すどちらかの条件を満たす場合は作業表用バッファを増分しません。

- pd_work_buff_expand_limit オペランドを指定しない
- pd_work_buff_expand_limit オペランドの値≦pd_work_buff_size オペランドの値

《オペランドの規則》

指定値は 128 の倍数を指定してください。128 の倍数でない場合, 自動的に 128 の倍数に切り上げられます。

《ほかのオペランドとの関連》

バックエンドサーバプロセスでの作業表用バッファの初回増分時に KFPH29008-I メッセージを出力します。なお，pd_work_table_option オペランドで KFPH29008-I メッセージの出力を抑止できます。

《留意事項》

作業表用バッファを増分した場合，該当するサーバプロセスで使用中の作業表数が 0 になると，増分した作業表用バッファを解放します。作業表の使用数が 0 になるのは次に示すときです。

- 使用中のカーソルをすべてクローズしたとき（この場合，作業表の使用数が 0 にならないことがあります）
- ホールダブルカーソルを使用していない場合は，トランザクションを正常終了するか，又は取り消したとき
- ホールダブルカーソルを使用している場合は，UAP を HiRDB から切り離したとき（DISCONNECT したとき）

《備考》

「ハッシュジョイン，副問合せのハッシュ実行」は，次に示す場合に適用されます。

- pd_additional_optimize_level オペランド，クライアント環境定義の PDADDITIONALOPTLVL オペランド, 又は SQL コンパイルオプションの ADD OPTIMIZE LEVEL オペランドで,「コストベース最適化モード 2 の適用」と一緒に「ハッシュジョイン，副問合せのハッシュ実行」を指定する
- SQL 文中の結合方式の SQL 最適化指定に HASH を指定する
- SQL 文中の副問合せ実行方式の SQL 最適化指定に HASH を指定する

8.2.3　システム監視に関するオペランド

9)　pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0，2～100000))

UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

また，上書きできない状態のシステムログファイルが，全システムログファイルの半分以上になったときに HiRDB が異常終了又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足します。

この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。

このオペランドでは，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。シンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，対象トランザクションを強制的に中断してロールバックをします。これを**シンクポイントダンプ有効化のスキップ回数監視機能**といいます。

シンクポイントダンプ有効化のスキップ回数監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

このオペランドを指定すると，UAP の無限ループなどを防止できます。

《指定値の目安》

通常は 0 を指定してください。0 を指定すると，スキップ回数の上限値を HiRDB が計算します。0 を指定して不都合が発生した場合，又は KFPS02101-I メッセージが出力された場合に，このオペランドの値を変更してください。そのときの指定値の目安については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，シンクポイントダンプ有効化のスキップ回数監視機能を使用しません。

《注意事項》

- このオペランドを指定すると，ログレスモードで実行している UAP も監視対象になります。ログレスモードで実行している UAP の処理が中断すると，データベースを自動回復できないため RD エリアを障害閉塞します。このため，上限値の設定に際しては，ログレスモードで実行する UAP のトランザクション処理中に，該当するサーバ内のほかのトランザクションから出力されるシステムログ量も考慮に入れてください。
- pdload，pdmod，pdrorg，pdexp，pddbst，pdgetcst，pdrbal，pdvrup，及び pdmemdb コマンドはこの機能の監視対象外になります。

10)　pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値

～<符号なし整数>((0～100000))

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになると，シンクポイントダンプの有効化処理ができずにスキップされます。これは，デファードライト処理によってシンクポイントダンプの取得が遅れることで，シンクポイントで出力される更新バッファ数がシンクポイントダンプの取得間隔内で出力できる更新バッファ数を超えるためです。

シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

このオペランドでは，デファードライト処理によってシンクポイントダンプの取得が遅れた場合に，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。

デファードライト処理によってシンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，シンクポイントダンプの取得間隔内でシンクポイントダンプの取得が完了するように，HiRDB が更新バッファ数の上限値を決定します。そして，更新バッファ数が上限値を超えた時点で最も古い更新バッファを出力して，シンクポイント時の更新バッファ総量を制限します。これを**更新バッファ量抑制機能**といいます。

《利点》

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生する，ユニット異常終了を回避できます。

《指定値の目安》

通常はこのオペランドを省略してください。更新処理のスループットを低下させてでも，シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生するユニット異常終了を回避したい場合に，1 を指定します。

許容できるシンクポイントダンプの有効化処理のスキップ回数がログ容量などから分かる場合は，その回数を指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

《オペランドの規則》

このオペランドに 0 を指定すると，更新バッファ量抑制機能を使用しません。

《注意事項》

更新バッファ量抑制機能の有効期間について，注意事項を次に示します。なお，更新バッファ量抑制機能の有効期間とは，KFPH23035-I メッセージが出力されてから KFPH23036-I メッセージが出力されるまでの間です。

- 更新バッファ数が，HiRDB が決定した更新バッファの上限値を超えている場合，更新処理の実行後に更新バッファを出力するため，更新処理のスループットが低下します。HiRDB が決定する更新バッファの上限値は，次の計算式で求められます。

（シンクポイントインターバル時間÷WRITE 単価※）
×（1－（前回のシンクポイントからプレシンクまでのログ出力量÷シンクポイント間のログ出力量）
×（バッファプールのバッファ面数÷シンクポイントで更新があったバッファプールの総バッファ面数）

注※　WRITE 単価についての詳細は，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

- pd_dbbuff_rate_updpage オペランド，又は pdbuffer -y オペランドによってデファードライトトリガのトリガ契機を指定している場合，それぞれの指定値が，HiRDB が決定した更新バッファの上限値より大きくなると，HiRDB が決定した更新バッファの上限値をデファードライトトリガのトリガ契機となる更新バッファ数に変更します。

  また，pdbuffer -w オペランドの値は，各バッファの更新バッファ上限値まで出力されるように自動調整されます。

- シンクポイントダンプの有効化処理がスキップした場合，シンクポイント時の更新バッファ出力処理中に検知されます。そのため，更新バッファ量抑制機能が有効となる時期は，シンクポイントダンプの有効化処理がスキップされ，エラーメッセージが出力されるときよりも後になる場合があります。
- 通常，並列 write 機能使用時のシンクポイント処理では各デファードライト処理用並列 WRITE プロセスに対する出力要求はシンクポイントごとに 1 回ですが，更新バッファ量抑制機能使用時はシンクポイントスキップの検出を早めるため複数回に分割して行われます。

8.2.4　排他制御に関するオペランド

11）　pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～2000000))
- 64 ビットモードの場合：((1～2000000000))

バックエンドサーバの排他制御で使用する共用メモリ領域（排他制御用プール）の大きさをキロバイト単位で指定します。

このオペランドの指定対象の領域は，排他資源管理テーブルとして使用されます。排他資源管理テーブル数，排他要求数，及び排他制御用プールサイズの間には，次の関係があります。

「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数

係数には，32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入します。

《指定値の目安》

- 32 ビットモードの場合，6 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 64 ビットモードの場合，4 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 次に示す計算式でこのオペランドの指定値を見積もってください。

HiRDB の種類	計算式（単位：キロバイト）
32 ビットモードの HiRDB/パラレルサーバの場合	↑↑b÷pd_lck_pool_partition の値↑÷6↑×pd_lck_pool_partition の値
64 ビットモードの HiRDB/パラレルサーバの場合	↑↑b÷pd_lck_pool_partition の値↑÷4↑×pd_lck_pool_partition の値

b：バックエンドサーバで同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については，「付録 D　排他資源数の見積もり」を参照してください。

注

特に，定義系 SQL の DROP TABLE 又は DROP SCHEMA を実行する場合は，このオペランドの指定値を正確に見積もって指定してください。

《指定値のチューニング方法》

統計解析ユティリティ（サーバごとのシステムの稼働に関する統計情報）で，バックエンドサーバの「排他資源管理テーブル使用率（%OF USE LOCK TABLE)」を参照してください。使用率の最大値が 80%以上の場合は，これからのデータベース拡張に備えて，このオペランドの値を大きくすることをお勧めします。使用率の最大値が 10%以下の場合は，共用メモリを節約するために，このオペランドの値を小さくすることをお勧めします。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- 32 ビットモードの場合：16000
- 64 ビットモードの場合：32000

ただし，pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます。

《注意事項》

- このオペランドの指定値が小さ過ぎると，SQL がエラーリターンすることがあります。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは，pd_lck_pool_partition オペランドと関連があります。

12）　pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数

～<符号なし整数>((1～5000))

排他制御処理の分散をするときにバックエンドサーバの排他制御で使用する排他制御用プールパーティション数を指定します。

排他制御処理の分散については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値のチューニング方法》

排他制御用プールパーティション数のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

《注意事項》

- このオペランドの指定値を大きくし過ぎると，各排他制御用プールパーティションに割り当てられる共用メモリのサイズが小さくなるため，排他制御用プールパーティションが容量不足となり SQL がエラーリターンすることがあります。また，排他制御の処理に時間が掛かるため，システムのパフォーマンスが下がります。この場合，このオペランドの指定値を小さくしてください。
- 排他制御用プールサイズは 1 キロバイト以上必要であるため，pd_lck_pool_size の値よりも大きな値を指定した場合，このオペランドには pd_lck_pool_size の値が仮定されます。また，このとき，KFPS00421-W メッセージが出力されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_pool_size
- pd_lck_deadlock_check_interval

13）　pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和

～<符号なし整数>((0～140000))

トランザクションを越えて保持する表，及び RD エリアの排他資源数を指定します。このオペランドの値に従って，表，RD エリアの UNTIL DISCONNECT 指定の排他を管理するブロックを共用メモリに確保します。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。次に示す場合に指定値を変更するかどうかを検討してください。

- 同時に実行するユティリティの数を大量に増やす場合
- ホールダブルカーソルを使用する場合
- pdlbuffer オペランドで指定したローカルバッファを使用する場合
- 共用 RD エリアを使用する場合

このオペランドの指定値の見積もり方法については，「付録 C.5　1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式」を参照してください。

《指定値のチューニング方法》

このオペランドの指定値が小さいと，トランザクションがロールバックしたり，リターンコード 8 でユティリティが異常終了したりします。このとき, KFPA11914-E, 又は KFPH28001-E メッセージが出力されます。この現象が発生したら，このオペランドの値を大きくしてください。

なお，このオペランドの値を大きくすると，それに比例して必要なメモリも増加します。必要なメモリは「このオペランドの値×48（64 ビットモードの場合は 64)」バイトです。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，256 が仮定されます。

14）　pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数

～<符号なし整数>((16～1024))

UNTIL DISCONNECT 指定の LOCK 文を実行しない表に対してホールダブルカーソルを使用する場合に，そのホールダブルカーソルのトランザクション当たりの最大同時オープン数を指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，16 が仮定されます。

《注意事項》

このオペランドに省略時解釈値以外の値を指定すると，共用メモリ使用量が増加します。

《ほかのオペランドとの関連》

このオペランドと次に示すオペランドの指定値は，ロックサーバの共用メモリサイズ計算に使用されます。32 ビットモードの HiRDB の場合，各オペランドの指定値が大き過ぎると，ロックサーバの共用メモリサイズが 2 ギガバイトを超えるため，HiRDB を開始できなくなることがあります。ロックサーバの共用メモリサイズが 2 ギガバイトを超えないように，このオペランドと次に示すオペランドの指定値を調整してください。

- pd_max_access_tables
- pd_max_users
- pd_max_bes_process
- pd_lck_hash_entry

- pd_lck_pool_size

共用メモリについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

15)　pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数

～<符号なし整数>((0～2147483647))

排他制御用プールで使用するハッシュテーブルのエントリ数を指定します。HiRDB はここで指定された値に従ってユニットコントローラ用の共用メモリに排他制御用プールを確保します。

《指定値の目安》

通常はこのオペランドを省略してください。

次に示す条件に該当する場合は指定値を検討してください。

- バージョン 06-02 以降にバージョンアップする場合に共用メモリサイズをできるだけ変更したくないときは 11261 を指定してください。この場合，バージョンアップ前と同数のハッシュエントリ数を確保するため，排他制御用プール中のハッシュテーブルサイズがバージョンアップ前と同じになります。
- このオペランドに推奨値より大きい値を指定すると，性能が向上することがあります。ただし，《推奨値の求め方》の変数 a より大きい値を指定しても，a を指定したときよりも性能が向上することはありません。

《オペランドの規則》

- このオペランド及びサーバ共通定義の pd_lck_hash_entry オペランドを省略するか，又はこのオペランドに 0 を指定すると，HiRDB はサーバごとに推奨値を計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 11261 が仮定されます）。推奨値については《推奨値の求め方》を参照してください。
- このオペランドに 0 でも素数でもない値を指定した場合，HiRDB はその指定値を超えない最大の素数が指定されたと仮定します。

《注意事項》

このオペランドの指定値が小さ過ぎると，ハッシュエントリ不足が発生して性能が低下することがあります。このオペランドを省略すれば，ハッシュエントリ不足及びハッシュエントリ不足による性能低下は発生しません。

《推奨値の求め方》

推奨値を次に示します。

推奨値＝ MAX（↑a÷10↑，11261）を超えない最大の素数

変数	変数の算出式	
a	（b ＋ 3）×10 ＋ pd_lck_pool_size の値×c	
b	pd_max_users の値＞ pd_max_bes_process の値の場合	pd_max_users の値
	pd_max_users の値≦pd_max_bes_process の値の場合	pd_max_bes_process の値
c	32 ビットモードの場合は 6，64 ビットモードの場合は 4	

16)　pd_dbsync_lck_release_count = シンクポイント時のグローバルバッファ排他解除間隔

～<符号なし整数>((0，100～1073741824))

シンクポイント時に発生するグローバルバッファへの排他を解除する間隔を指定します。

シンクポイント時には，ディスクに反映する必要があるバッファ（更新バッファ）のサーチ処理が発生します。通常は，更新バッファのサーチ処理中に，一定間隔でグローバルバッファへの排他を解除します。

例えば 100 を指定した場合，100 面（グローバルバッファの面数）のサーチ処理が完了すると，一度排他を解除します。その後, 再度排他を掛けてサーチ処理を続行します。このように 100 面ごとに排他の解除をします。

《利点》

このオペランドを指定すると，シンクポイント時のグローバルバッファの排他占有時間を調整できます。このオペランドの値を小さくすると，グローバルバッファの排他占有時間が短くなり，シンクポイント時のトラザクション性能が向上することがあります。

グローバルバッファプールの排他占有時間については，統計解析ユティリティを実行して，グローバルバッファプールに関する統計情報の「シンクポイント時のバッファプール排他占有時間（SYNCL）」で確認できます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。次に示す条件をすべて満たす場合にこのオペランドの指定を検討してください。

- シンクポイント時にトランザクション性能が低下する
- pdbuffer オペランドの-n オプションに指定したバッファ面数が非常に大きい

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10000 が仮定されます。

《オペランドの規則》

- 1 から 99 までの値を指定した場合，自動的に 100 が設定されます。
- 0 を指定した場合，更新バッファのサーチ処理が完了するまで，グローバルバッファに対して排他が掛かります。

《注意事項》

このオペランドの値を小さくすると，ほかのトランザクションの割り込みによって，更新バッファのサーチ時間が長くなります。その間に更新されたグローバルバッファもシンクポイント時の出力対象になります。このため，シンクポイント時に出力される更新バッファ数が増加します。シンクポイント時に出力対象となる更新バッファ数については, 統計解析ユティリティの, グローバルバッファプールに関する統計情報の「シンクポイント出力ページ数（SYNCW）」で確認できます。

8.2.5　バッファに関するオペランド

17）　pd_sql_object_cache_size = SQL オブジェクト用バッファ長

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((22～256000))
- 64 ビットモードの場合：((22～2000000))

SQL オブジェクトを格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- SQL オブジェクトは，ユーザごとにトランザクションが終了するまでバッファに保存されます。したがって，同時に実行するトランザクションの SQL オブジェクトがバッファに格納できる大きさが必要です。

- 静的 SQL の SQL オブジェクトはトランザクション終了後もバッファに空きがなくなるまで保存して，同一 UAP を実行する複数のユーザで共用することで，SQL 解析処理の削減を図ります。バッファを有効に利用するためには，使用頻度の高い UAP の SQL オブジェクトがバッファに常駐するようにバッファを確保してください。
- バッファ長の見積もり時は，まず UAP で発行する SQL 文の SQL オブジェクト長から UAP 実行に必要なバッファ長を見積もります。次に，同時に実行される UAP と同時実行ユーザ数を考慮して必要なバッファ長を算出してください。
- 1SQL 文の SQL オブジェクト長の見積もり方法については，「付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義とシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

（pd_max_users の値＋ 3）×22

《指定値のチューニング方法》

SQL オブジェクト用バッファ長のチューニング方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

8.2.6　共用メモリに関するオペランド

18）　pd_bes_shmpool_size = バックエンドサーバ用共用メモリサイズ

～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：((1～200000))
- 64 ビットモードの場合：((1～4000000000))

ユニットコントローラ用の共用メモリの中で，バックエンドサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。サーバ共通定義及びバックエンドサーバ定義の pd_bes_shmpool_size オペランドを省略すると，このオペランドの値を HiRDB が計算します（pd_sysdef_default_option オペランドに v6compatible を指定している場合は 1024 が仮定されます）。HiRDB はマニュアル「HiRDB Version 8 システム導入・設計ガイド」に記載されている「バックエンドサーバが使用する共用メモリの計算式」を基に計算をします。この計算式中の変数の説明に出てくるオペランドの値を変更した場合，このオペランドの値を HiRDB が自動的に再計算します。なお，計算式中の変数「サーバ内のインデクス総数」は pd_assurance_index_no オペランドの値が仮定されます。また，計算式中の変数「インデクス用のグローバルバッファプール数」，及び「グローバルバッファ総数（pdbuffer オペランドの指定数）」は，32 ビットモードの場合は 500，64 ビットモードの場合は 1000 が仮定されます。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

- 必要以上にこのオペランドの値が大きいと，共用メモリをむだに確保してしまいます。また，このオペランド値が小さ過ぎると，次に示す現象が発生します。
  ・ユニットを開始できなくなります。
  ・UAP 又はユティリティが実行できなくなります。
- このオペランドを省略する場合は，必ず pd_assurance_table_no，及び pd_assurance_index_no オペランドに適切な値を指定してください。

8.2.7　RPC トレース情報に関するオペランド

19）　pd_rpc_trace = Y｜N

RPC トレース情報を取得するかどうかを指定します。RPC トレース情報には, HiRDB の保守情報が出力されます。

通常，このオペランドは省略してください。

Y：RPC トレースを取得します。

N：RPC トレースを取得しません。

《注意事項》

このオペランドに Y を指定すると，通信性能が低下します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，N が仮定されます。

20）　pd_rpc_trace_name = "RPC トレースを取得するファイル名称"

～< 254 文字以内のパス名>

RPC トレースファイルの名称を絶対パス名で指定します。RPC トレースファイルとして，指定したファイル名の後に 1，2，l（エル）を付けた三つのファイルが作成されます。

《注意事項》

このオペランドで指定したディレクトリ下に，最大「pd_rpc_trace_size の値× 2」の大きさのファイルが作成されます。ファイル容量に注意してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

%PDDIR%¥spool¥rpctr

21）　pd_rpc_trace_size = RPC トレースを取得するファイルの容量

～<符号なし整数>((1024～2147483648))（単位：バイト）

RPC トレースファイルの容量をバイト単位で指定します。

このオペランド指定値は，"RPC トレースファイル"1 及び"RPC トレースファイル"2 のそれぞれの容量となります。

《指定値の目安》

RPC トレースは，すべてのプロセス間の通信電文が一つのファイルに時系列に蓄積されます。その量がこの指定値を超えると，もう一方のファイルにトレースデータの出力を切り替えます。古いトレースは上書きされて消去されます。このため，指定値が小さいと十分なトレースを取得できなく

なり，トラブルシュートが困難になることがあります。したがって，このオペランドには 1,000,000 以上の値を指定してください。

《注意事項》

このオペランドで指定した値は RPC トレースファイル l（エル）の容量に影響を与えません。RPC トレースファイル l（エル）の容量は 0 バイト固定です。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，4096 が仮定されます。

8.2.8　トラブルシュート情報に関するオペランド

22)　pd_module_trace_max = モジュールトレース格納最大数

～<符号なし整数>((126～16383))

HiRDB のプロセスは，実行した関数やマクロの履歴をプロセス固有メモリ中に記録しています。この履歴をモジュールトレースといい，このオペランドではモジュールトレースのレコード数を指定します。この履歴の内容は core ファイルに取り込まれ，プロセス異常終了時に出力されます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，126 が仮定されます。

《注意事項》

各プロセスでは，以下のサイズのプロセス固有メモリが確保されます。

32 ビットモードの場合：64 ＋ 48×pd_module_trace_max オペランドの値（バイト）

64 ビットモードの場合：64 ＋ 64×pd_module_trace_max オペランドの値（バイト）

23)　pd_module_trace_timer_level = 0 | 10 | 20

モジュールトレースに出力する時刻の取得方法を指定します。このオペランドの指定値の意味を以下に示します。

指定値	時刻取得方法
0	すべてのモジュールトレース出力箇所で，秒単位で取得します。
10	すべてのモジュールトレース出力箇所のうち，入出力処理前後など性能調査で問題となりやすい箇所だけマイクロ秒単位で取得し，そのほかの箇所では秒単位で取得します。
20	すべてのモジュールトレース出力箇所で，マイクロ秒単位で取得します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，0 が仮定されます。

《注意事項》

このオペランドに 0 以外の値を指定すると，時刻をマイクロ秒で取得する関数を発行するため，性能が低下することがあります。

24）　pd_pth_trace_max = 通信トレース格納最大数

～<符号なし整数>((1024～8388608))

トラブルシュートの際に保守情報として使用する通信トレースのレコード数を指定します。

《指定値の目安》

通常は，本オペランドを指定する必要はありません。性能調査などにおいて，保守員が本オペランドの指定を依頼する場合があります。その場合に，保守員に指定された値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，1024 が仮定されます。

《注意事項》

通信トレース格納最大数の指定値を大きくすると，HiRDB プロセスが確保するプロセス固有メモリが増加します。また，通信トレースのプロセス固有メモリは，オペランド指定値を 2 のべき乗の値に切り上げた値を元に計算します。メモリ所要量についての詳細は，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

8.2.9　グローバルバッファに関するオペランド

25）　pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値

～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加するグローバルバッファ数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

- pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，その数に余裕を持たせた値を指定してください。
- 次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_max_add_dbbuff_no の値≦2000000－HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

条件		省略値
32 ビットモードの場合	a ≧500 の場合	256
	a ＜ 500 の場合	500－ a
64 ビットモードの場合	a ≧1000 の場合	256
	a ＜ 1000 の場合	1000－ a

a：HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《注意事項》

このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_shm_no

26)　pd_max_add_dbbuff_shm_no ＝ 動的追加用共用メモリセグメント数の上限値 ～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加したときに割り当てられる，共用メモリセグメント数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，値を指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

条件		省略値
pd_max_add_dbbuff_no オペランドを省略した場合	32 ビットモードの場合	（16－ a ）＋ 500
	64 ビットモードの場合	（16－ a ）＋ 1000
pd_max_add_dbbuff_no オペランドを指定した場合	32 ビットモードの場合	↓pd_max_add_dbbuff_no の値×1.5 ＋ 16↓ （計算値が 32752 以上だった場合は自動的に 32752 が設定されます）
	64 ビットモードの場合	

a：HiRDB 開始時に各サーバに割り当てられている共用メモリセグメント数

《注意事項》

- 次に示す条件式を満たす場合は，このオペランドに pd_max_add_dbbuff_no オペランドの値が仮定されます。

  pd_max_add_dbbuff_shm_no の値＜ pd_max_add_dbbuff_no の値

省略値が上記の条件を満たす場合も，pd_max_add_dbbuff_no オペランドの値が仮定されます。

- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- 追加する共用メモリセグメントサイズが SHMMAX オペランドの値を超える場合，SHMMAX オペランドの値を上限値とした複数の共用メモリセグメントに分割されます。追加する共用メモリセグメントサイズを想定して SHMMAX オペランドの値を増やすか，又は分割されたときに不足しないように pd_max_add_dbbuff_shm_no オペランドの値を増やしてください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_no

8.2.10　プラグインインデクスの遅延一括作成に関するオペランド

27）　pd_plugin_ixmk_dir = "インデクス情報ファイルを作成するディレクトリ名"又は "インデクス情報ファイルを作成する HiRDB ファイルシステム領域名"

～<パス名>

プラグインインデクスの遅延一括作成をする場合に，インデクス情報ファイルを作成するディレクトリ名を指定します。インデクス情報ファイルを HiRDB ファイルシステム領域に作成する場合は，HiRDB ファイルシステム領域名を指定します。ディレクトリ名及び HiRDB ファイルシステム領域名は絶対パス名で指定してください。

プラグインインデクスの遅延一括作成については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《注意事項》

- ここで指定するディレクトリ（又は HiRDB ファイルシステム領域）を必ず作成しておいてください。存在しないディレクトリ名（又は HiRDB ファイルシステム領域名）を指定すると，プラグインインデクスの遅延一括作成を指定した UAP（クライアント環境定義に PDPLGIXMK = YES を指定した環境で実行した UAP）の実行時にエラーとなります。
- UAP 実行後，データベース再編成ユティリティでプラグインインデクスの遅延一括作成をする前にこのオペランドの指定値を変更しないでください。変更すると，プラグインインデクスを遅延一括作成できなくなります。

8.2.11　Java に関するオペランド

Java ストアドプロシジャ又は Java ストアドファンクションを使用する場合に，Java に関するオペランドを指定します。Java ストアドプロシジャ及び Java ストアドファンクションについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

28）　pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"

～<パス名>

Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合，そのファイル名を絶対パス名で指定します。

《指定値の目安》

このオペランドで指定するファイルの容量が非常に大きくなるため，通常はこのオペランドを指定しないでください。Java ストアドプロシジャ又は Java ストアドファンクションのデバッグ時に指定することをお勧めします。なお，このオペランドで指定するファイルの容量制限はありません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，システム共通定義の優先順位で，同じオペランドの指定値が有効となります。サーバ共通定義，ユニット制御情報定義，及びシステム共通定義の同じオペランドも省略すると，Java 仮想マシンでの標準出力又は標準エラー出力が無視されます。

《注意事項》

複数プロセスからの同時書き込みがあった場合は，その出力内容を保証しません。

《オペランドの規則》

- パス名は 255 文字以内とします。
- パス名の大文字と小文字の区別をしません。

29)　pd_java_castoff = Y | N

HiRDB のバックエンドサーバプロセスで，Java 仮想マシンを起動したサーバプロセスの終了方法を指定します。

Y：Java 仮想マシンを起動した場合，トランザクション決着時，又は UAP の切り離し時にサーバプロセスを終了します。

N：Java 仮想マシンを起動してもサーバプロセスが終了しません。

省略した場合は，サーバ共通定義の pd_java_castoff オペランドが解釈値となります。

《指定値のチューニング方法》

Java ストアドルーチン（Java ストアドプロシジャ，又は Java ストアドファンクション）を実行すると HiRDB のサーバプロセスで Java 仮想マシンが起動します。

Java 仮想マシンを使用したアプリケーションの実行が限定的な場合に，pd_java_castoff=Y を指定することによって Java 仮想マシンが起動したプロセスが再利用されたときの以下の懸案を解決することが出来ます。

#	懸案
1	Java 仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリを圧迫している。
2	Java 仮想マシンを使用しないコネクションで実行する SQL でも，別のコネクションで Java 仮想マシンが既に設定したスタックサイズの上限によって，探索条件の多い SQL を実行するとスタックの拡張が出来ずにサーバプロセスがセグメンテーションフォルトでアボートする。 なお，Java 仮想マシンの機能については Java 仮想マシンのドキュメントのオプションを参照してください。

《注意事項》

Java ストアドルーチンを頻繁に実行するシステムで Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- pd_process_count

8.2.12　HiRDB External Data Access 機能に関するオペランド

HiRDB External Data Access 機能については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

なお，HiRDB の稼働プラットフォームによっては HiRDB External Data Access 機能を使用できません。詳細については，マニュアル「HiRDB Version 8 解説」の「HiRDB External Data Access」を参照してください。

30）　pd_foreign_server_libpath = "外部サーバのクライアントライブラリのパス名" 〔, "外部サーバのクライアントライブラリのパス名"〕…

～<パス名>

外部サーバのクライアントライブラリのパス名を絶対パス名で指定します。外部サーバのクライアントライブラリのパス名については，接続する外部サーバの DBMS のマニュアルを参照してください。

《前提条件》

HiRDB External Data Access が必要です。

《オペランドの規則》

指定できる文字列の最大長は 512 文字です。

《注意事項》

1：1 スタンバイレス型系切り替え機能使用時の注意事項です。正規 BES 及び代替 BES に外部サーバを定義していて，外部サーバのサーバ種別（アクセスする DBMS）が同じ場合は，外部サーバのクライアントライブラリのバージョンを同じにしてください。

8.2.13　システムログファイルに関するオペランド

31）　pd_log_dual = Y | N

システムログファイルを二重化するかどうかを指定します。

Y：システムログファイルを二重化します。

N：システムログファイルを二重化しません。

《利点》

システムログファイルを二重化すると，HiRDB はA系及びB系の両方に同じシステムログを取得します。取得したシステムログを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシステムログを読み込めるため，システムの信頼性を向上できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

システムログファイルを二重化する場合は pdlogadpf オペランドで B 系のシステムログファイル名を指定してください。

32）　pd_log_remain_space_check = warn | safe

システムログファイルの空き率が**警告値**未満になったときの HiRDB の処理方式を指定します。この機能を**システムログファイルの空き容量監視機能**といいます。システムログファイルの空き容量監視機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

warn：

システムログファイルの空き率が警告値未満になった場合，警告メッセージ KFPS01162-W を出力します。

safe：

システムログファイルの空き率が警告値未満になった場合，新規トランザクションのスケジューリングを抑止して，サーバ内の全トランザクションを強制終了します。このとき，KFPS01160-E メッセージを出力します。

《指定値の目安》

システムログファイルの空き容量不足によるユニットの異常終了の可能性を低くできるため，safe の指定を推奨します。ただし，safe を指定すると，システムログファイルの空き容量が不足したときに，サーバ内の全トランザクションが強制終了されます。このため，システムログファイルの設計をより正確に行う必要があります。システムログファイルの設計については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，warn が仮定されます。

33）　pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕…

～<パス名>((1～136 文字))

システムログの**自動ログアンロード機能**を使用する場合に，アンロードログファイルの出力先ディレクトリを絶対パス名で指定します。アンロードログファイルを HiRDB ファイルシステム領域に作成する場合は，その HiRDB ファイルシステム領域名を指定してください。なお，**このオペランドに指定するディレクトリ又は HiRDB ファイルシステム領域は，HiRDB の開始前に作成しておいてください。**

また，このオペランドには，サーバごとに異なるディレクトリ又は HiRDB ファイルシステム領域を指定してください。

自動ログアンロード機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《指定値の目安》

空きディスク容量を確認してからディレクトリを指定してください。作成されるアンロードログファイルによって，ディスク容量が圧迫される原因になります。

また，ディスク容量の不足によって，指定したディレクトリにアンロードログファイルが作成できない場合，自動ログアンロード機能が停止します。このようなことが発生する可能性がある場合は，ディレクトリを複数指定することをお勧めします。

ただし，ディレクトリは一つだけの方がデータベース回復時の運用（回復に必要なアンロードログファイルの選別）が若干簡単になります。

ディレクトリを複数指定する場合の補足事項を次に示します。

- ディスク障害などに備えて，異なるパーティションのディレクトリを指定することをお勧めします。
- ディスク容量満杯又はディスク障害などで一つのディレクトリ下にアンロードログファイルが作成できない場合，ほかのディレクトリ下にアンロードログファイルを作成します。このとき，HiRDB はこのオペランドに指定した順番にディレクトリを使用します。

《オペランドの規則》

- ディレクトリは最大 128 個指定できます。
- 複数のディレクトリを指定した場合，同じパス名を指定できません。

《注意事項》

次の場合は，自動ログアンロード機能を使用できません。

- pd_log_unload_check オペランドに N を指定している場合

なお，マルチ HiRDB を利用している場合は，HiRDB ごとに異なるディレクトリを作成してください。同じディレクトリを指定すると, どのアンロードログファイルがどの HiRDB に対応しているか分からなくなる可能性があります。

34）　pd_log_singleoperation = Y | N

このオペランドは，システムログファイルを二重化している場合に指定してください。二重化していない場合は指定する必要はありません。

システムログファイルの片系運転をするかどうかを指定します。システムログファイルに障害が発生して，両系ともに使用できるシステムログファイルがない場合でも，HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了しないで正常な片方の系だけで処理を続行します。これを**システムログファイルの片系運転**といいます。

また，両方のシステムログファイルで処理を続行すること(通常の処理形態)を**システムログファイルの両系運転**といいます。

Y：システムログファイルの片系運転をします。

N：システムログファイルの片系運転をしません。常に両系運転とします。

《前提条件》

pd_log_dual オペランドに Y を指定している必要があります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

35）　pd_log_rerun_reserved_file_open = Y | N

システムログファイルの自動オープンをするかどうかを指定します。

ユニットを再開始するときに，「上書きできる状態」のシステムログファイルがない場合，予約のファイルがあれば HiRDB が予約のファイルをオープンして「上書きできる状態」にし，処理を続行します。これを**システムログファイルの自動オープン**といいます。

予約ファイルを使用するのは，次の場合です。

- 再開始後，最初のシンクポイントダンプを取得するまでの間
- オープンされたファイルグループすべてが「上書きできない状態」の場合

Y：システムログファイルの自動オープンをします（予約ファイルをオープンして使用します)。

N：システムログファイルの自動オープンをしません（予約ファイルを使用しません)。

《利点》

このオペランドに Y を指定すると，ユニットの再開始時に「スワップ先にできる状態」のファイルがなくても，予約のファイルがあればユニットを再開始できます。

ただし，「アンロード待ち状態」のファイルがあるときは，ユニットを停止します。「アンロード待ち状態」のファイルのアンロードを実行した後，再度ユニットを開始してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

36）　pd_log_rerun_swap = Y | N

ユニットの再開始時に，システムログファイルをスワップするかどうかを指定します。

Y：システムログファイルをスワップします。

N：システムログファイルをスワップしません。

《利点》

このオペランドにYを指定すると，再開始前後でシステムログファイルを物理的に分けられます。このため，再開始前のシステムログファイルをサーバ稼働中に使用できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Nが仮定されます。

37）pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間～<符号なし整数>((1～32580))（単位：秒）

システムログファイルのスワップが完了するまでの待ち時間を秒単位で指定します。指定した時間を過ぎてもシステムログファイルのスワップが完了しない場合は，ユニットが異常終了します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。マシンの性能が低いなどの理由によってシステムログファイルのスワップに時間が掛かる場合は，このオペランドの値を大きくしてください。また，ディスク障害などによってシステムログファイルのスワップに遅延が発生した場合に，より短い時間で異常を検知し，ユニットを異常終了させたいときは，このオペランドの値を小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，180が仮定されます。

38）pd_log_unload_check = Y｜N

システムログファイルのアンロード状態を，HiRDBがチェックするかどうかを指定します。

Y：

アンロード状態をチェックします（通常の運用となります）。

N：

アンロード状態をチェックしません。アンロードの状態に関係なく，次に示す条件がすべて満たされれば，システムログファイルを「スワップ先にできる状態」にします。

- 上書きできる状態
- 抽出完了状態（HiRDB Datareplicator）

このときシステムログファイルの運用方法が，アンロード状態のチェックを解除する運用になります。アンロード状態のチェックを解除する運用については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《利点》

Nを指定したときの利点を次に示します。

- システムログファイルのアンロード操作がなくなるため，運用方法が簡単になります。
- アンロードログファイルを保管するためのファイル容量が不要になります。

《指定値の目安》

データベースを回復するときにシステムログを使用しない場合（バックアップ取得時点に回復できればよい場合）にNを指定します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Y が仮定されます。

《注意事項》

N を指定したときの注意事項を次に示します。

- バックアップ取得時点にしかデータベースを回復できません。
- データベースの回復にシステムログが必要なのにこの運用をした場合はデータベースの回復手段がなくなります。

39）　pd_log_max_data_size = ログ入出力バッファ長

～<符号なし整数>((32000～523000))（単位：バイト）

システムログの入出力に使用するバッファの大きさをバイト単位で指定します。

《指定値の目安》

条件式 1 を満たす値を指定してください。pd_rpl_reflect_mode オペランドに uap を指定している場合，又は回復不要 FES を使用している場合は，条件式 1 と条件式 2 の両方を満たす値を指定してください。指定値を最適化する場合は，指定値のチューニング方法に従って指定値を変更してください。

条件式 1：ログ入出力バッファ長≧a

a：72×（一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数）＋ 1344

条件式 2：ログ入出力バッファ長≧b

b：(一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数＋ 1）×128 ＋ 64

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）の次に示す情報を調べて，このオペランドの指定値を変更してください。

- 入出力待ちバッファ面数（# OF BUFFER FOR WAIT I/O）

  入出力待ちバッファ面数の平均値が 100 を大きく超える場合は，このオペランドの値を大きくして，平均値が 100 に近付くようにしてください。

- カレントバッファなしによる待ち回数（# OF WAIT THREAD）

  カレントバッファなしによる待ち回数が 0 以外の値になる場合は，このオペランドの値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，400000 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 32000 が仮定されます）。

《ほかのオペランドとの関連》

pd_log_write_buff_count オペランドとともに，ログ入出力バッファ長を決定してください。

40）　pd_log_write_buff_count = ログ出力バッファ面数

～<符号なし整数>((3～65000))

システムログの出力に使用するバッファの面数を指定します。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）のカレントバッファなしによる待ち回数（# OF WAIT THREAD）を調べて，待ち回数が多ければ，スループットが向上するよう指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，10 が仮定されます（pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は 3 が仮定されます）。

《ほかのオペランドとの関連》

pd_log_max_data_size オペランドとともに，ログ出力バッファ面数を決定してください。

41）　pd_log_rec_leng = システムログファイルのレコード長

～<符号なし整数>((1024，2048，4096))（単位：バイト）

システムログファイルのレコード長を指定します。1024，2048，4096 のどれかを指定します。

レコード長は pdloginit コマンドの-l オプションで設定します。-l オプションで設定するレコード長をこのオペランドに指定してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，4096 が仮定されます。

《注意事項》

- pdloginit コマンドの-l オプションで設定するレコード長と異なるレコード長をこのオペランドに指定すると，そのシステムログファイルはオープンできません。
- システムログファイルのレコード長を変更する方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

42）　pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数

～<符号なし整数>((0～256))

ロールバック処理でシステムログの入力に使用するバッファの面数を指定します。このオペランドに 0 を指定した場合，HiRDB がロールバック用ログ入力バッファ面数を決定します。

《指定値のチューニング方法》

このオペランドに 0 を指定してください。0 を指定してメモリ不足が発生する場合は，このオペランドの指定を省略してください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- ユニットに 1：1 スタンバイレス型系切り替え機能を適用している場合
  代替 BES 数×2
- ユニットに影響分散スタンバイレス型系切り替えを適用している場合
  （ホスト BES 数＋ pd_ha_max_act_guest_servers の値）×2
- 上記以外の場合
  ユニット内サーバ数×2

43)　pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔,拡張上限サイズ〕

～<符号なし整数>((0～104857600))《0，0》（単位：レコード）

このオペランドは，システムログファイルの自動拡張機能を使用している場合に指定します。

一つのシステムログファイルが拡張契機 1 回当たりに拡張するサイズ，及び拡張できるファイルサイズの上限をレコード数で指定します。

1 回当たりに拡張するサイズを省略するか，又は 0 を指定した場合，システムログファイルの自動拡張を行いません。拡張上限サイズを省略するか，又は 0 を指定した場合，ファイルシステム領域のあるディスクが満杯になるか，システムログファイルの容量が上限に達するまでシステムログファイルの自動拡張を行います。また，1 回当たりに拡張するサイズに拡張上限サイズより大きい値を指定した場合，拡張上限サイズに指定した値まで自動拡張を行います。

システムログファイルの自動拡張機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

《前提条件》

pd_large_file_use オペランドに Y を指定しているか，又は指定を省略している必要があります。

《指定値の目安》

- 1 回当たりに拡張するサイズには，pdloginit コマンドの-n オプションでシステムログファイルを作成した際に指定したレコード数を基に指定します。全システムログファイルのレコード数の平均値の 1/10 を計算し，指定してください。

  ・1:1 スタンバイレス型系切り替え機能を適用している BES の場合，代替 BES のシステムログファイルを含めた平均値の 1/10 を計算してください。

  ・影響分散スタンバイレス型系切り替え機能を適用している BES の場合，ゲスト BES のシステムログファイルを含めた平均値の 1/10 を計算してください。

- 通常，拡張上限サイズは省略してください。

《指定値のチューニング方法》

システムログの出力量が自動拡張で拡張するサイズを超えると，システムログファイルが満杯になってユニットダウンする場合があります。その場合は，1 回当たりに拡張するサイズを大きくしてください。また，拡張処理に時間が掛かり，トランザクション性能に影響を及ぼす場合は，1 回当たりに拡張するサイズを小さくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

8.2.14　シンクポイントダンプファイルに関するオペランド

44)　pd_spd_dual = Y | N

シンクポイントダンプファイルを二重化するかどうかを指定します。

Y：シンクポイントダンプファイルを二重化します。

N：シンクポイントダンプファイルを二重化しません。

《利点》

シンクポイントダンプファイルを二重化すると，HiRDB はA系及びB系の両方に同じシンクポイントダンプを取得します。取得したシンクポイントダンプを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシンクポイントダンプを読み込めるため，システムの信頼性を向上できます。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

シンクポイントダンプファイルを二重化する場合は，pdlogadpf オペランドで B 系のシンクポイントダンプファイル名を指定してください。

45）　pd_spd_assurance_msg = Y | N

シンクポイントダンプが有効化されたとき，メッセージ KFPS02183-I を出力するかどうかを指定します。

Y：出力します。

N：出力しません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，Y が仮定されます。

46）　pd_spd_assurance_count = 有効保証世代数

～<符号なし整数>((1～2))

システム回復時のシンクポイントダンプファイルの入力障害などに備え，HiRDB 稼働中に保存しておくシステムログファイルの範囲を，シンクポイントダンプファイルの世代数によって指定します。この世代数のことを**有効保証世代数**といいます。ここで指定した世代数分のシンクポイントダンプファイルは，上書きできない状態になります。

《利点》

有効保証世代数を 2 にすると，最新世代のシンクポイントダンプファイルに障害が発生しても，1 世代前のシンクポイントからシステムを回復できるため，信頼性が向上します。

《指定値の目安》

- 信頼性を向上させる場合は有効保証世代数を 2 にしてください。ただし，有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます（二つになります）。
- シンクポイントダンプファイルを二重化して信頼性を向上している場合は，このオペランドを省略するか，又は 1 を指定することをお勧めします。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

《注意事項》

- シンクポイントダンプファイルの必要最低数は**有効保証世代数＋ 1** となります。
- 有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます。上書きできない状態のシンクポイントダンプファイルに対応するシステムログファイルは上書きできない状態になります。したがって，有効保証世代数を 2 にすると，上書きできない状態のシステムログファイルが増えるため，スワップ先にできる状態のシステムログファイル数が不足することがあります。これを防ぐためにはシステムログファイルの容量を考慮してください。

47）　pd_spd_reduced_mode = 縮退運転オプション

～<符号なし整数>((0～2))

シンクポイントダンプファイルの縮退運転をするかどうかを指定します。

縮退運転とは，HiRDB 稼働中又は再開始処理中のファイル障害などによって，シンクポイントダンプファイルの数が運用に必要なファイル数（**有効保証世代数**※＋1）以下になった場合でも，最低二つのファイルが使用できれば処理を続行する機能です。

注※　pd_spd_assurance_count オペランドの指定値です。

0：縮退運転を使用しません。

1：縮退運転を使用します。

2：縮退運転を使用し，縮退運転でシンクポイントダンプ取得の契機ごとに警告メッセージを出力します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，0 が仮定されます。

48）　pd_spd_reserved_file_auto_open = Y｜N

シンクポイントダンプファイルの自動オープンをするかどうかを指定します。

シンクポイントダンプファイルに障害が発生して，運用に必要なファイル数（**有効保証世代数**※＋1）を下回った場合，予約のファイルがあれば HiRDB が予約のファイルをオープンして上書きできる状態にし，処理を続行します。これを**シンクポイントダンプファイルの自動オープン**といいます。

注※　pd_spd_assurance_count オペランドの指定値です。

Y：

シンクポイントダンプファイルの自動オープンをします。運用に必要なファイル数（有効保証世代数＋1）を下回った場合，予約ファイルを自動的にオープンします。

N：

シンクポイントダンプファイルの自動オープンをしません。運用に必要なファイル数（有効保証世代数＋1）を下回っても，予約ファイルを自動的にオープンしません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《ほかのオペランドとの関連》

このオペランドの指定は pd_spd_reduced_mode オペランドより優先されます。

49）　pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長

～<符号なし整数>((32000～4000000))（単位：バイト）

シンクポイントダンプをシンクポイントダンプファイルに入出力するときに使用するバッファ（共用メモリ）の大きさをバイト単位で指定します。

ここで指定した値によってシンクポイントダンプファイルの入出力回数を制御します。

《指定値の目安》

- 通常はこのオペランドを指定する必要はありません。
- 指定値を大きくするほど，シンクポイントダンプファイルに対する入出力回数が少なくなります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，32768 が仮定されます。

50）　pd_log_sdinterval = システムログ出力量〔，経過時間〕

シンクポイントダンプの取得間隔を指定します。次に示す条件で指定できます。

- システムログの出力量
- 前回シンクポイントダンプを取得してからの経過時間

システムログ出力量：～<符号なし整数>((100～100000))（単位：ログブロック数）

シンクポイントダンプの取得間隔を指定します。ここで指定したログブロック数分のシステムログが出力されるたびに，シンクポイントダンプを取得します。

経過時間：～<符号なし整数>((0,10～1440))（単位：分）

シンクポイントダンプの取得間隔を指定します。前回シンクポイントダンプを取得してから，ここで指定した経過時間が過ぎると，シンクポイントダンプを取得します。

- 経過時間に 0 を指定すると，HiRDB は経過時間でのシンクポイントダンプを取得しません。
- 前回シンクポイントダンプを取得してからトランザクションが一度も発生しない場合，ここで指定した経過時間が過ぎてもシンクポイントダンプを取得しません。

《指定値の目安》

- HiRDB を再開始するときの所要時間に特に規定のない場合は，このオペランドを指定する必要はありません。
- このオペランドの指定値によって，HiRDB を再開始するときの所要時間が決まります。

  このオペランドの指定値を小さくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が短くなります。ただし，シンクポイントダンプの取得回数が増えるため，オンライン中の性能が低下する場合があります。

  逆に，このオペランドの指定値を大きくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が長くなります。ただし，シンクポイントダンプの取得回数が減るため，オンライン中の性能が向上する場合があります。

《指定値のチューニング方法》

シンクポイントダンプがどの程度の間隔で取得されているかは，統計解析ユティリティのシステムの稼働に関する統計情報の「シンクポイントダンプ取得間隔時間（SYNC POINT GET INTERVAL）」で確認できます。SYNC POINT GET INTERVAL の平均値を見て，シンクポイントダンプを取得する間隔が長いと判断したら，ここでの指定値を小さくしてください。また，反対に短いと判断したら，ここでの指定値を大きくしてください。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，次に示す値が仮定されます。

- システムログ出力量：5000
- 経過時間：60

ただし，pd_sysdef_default_option オペランドに v6compatible 又は v7compatible を指定している場合は，システムログ出力量の省略値は 1000 が仮定されます。

《注意事項》

シンクポイントダンプの取得間隔はシステムログの出力量で決まります。このため，更新系のトランザクションがほとんどない時間帯は，メモリからデータベースに反映するまでの時間が長くなります。このような状態で障害が発生すると，その間に発生したトランザクションの回復時間が長くなります。このようなケースが考えられる場合は，「経過時間」でもシンクポイントダンプの取得間隔を設定してください。

8.2.15　サーバ用ステータスファイルに関するオペランド

51）　pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

：

pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

サーバ用ステータスファイルを定義します。pd_sts_file_name_2～7 オペランドは省略できますが，pd_sts_file_name_1 オペランドは省略できません。

"論理ファイル名"：～<識別子>((1～8 文字))

サーバ用ステータスファイルの論理ファイル名を指定します。ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系ステータスファイル名"：～<パス名>((167 文字以内))

A 系のステータスファイル名を絶対パス名で指定します。

"B 系ステータスファイル名"：～<パス名>((167 文字以内))

B 系のステータスファイル名を絶対パス名で指定します。

《指定値の目安》

- A 系及び B 系のステータスファイル名には，pdstsinit コマンドで作成したステータスファイルを指定してください。pdstsinit コマンドで作成していないステータスファイルを指定すると，そのステータスファイルは**実体のないステータスファイル**となります。
- ステータスファイルに障害が発生すると，HiRDB はステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要となります。
- A 系と B 系には同じレコード長，及び同じ容量のステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- ステータスファイルは，A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系ステータスファイル名，及び B 系ステータスファイル名には同じ名称を重複して指定できません。
- HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥sts01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。sts01 については大文字と小文字の区別をします。

《留意事項》

- HiRDB を正常開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，全ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときは pd_sts_file_name_1～7 の中で最初に指定したステータスファイルが現用ファイルになります。残りのファイルのうちオープンできたファイルが予備ファイルになり，オープンできないファイルは予約ファイルになります。
- HiRDB を再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。

●実体のないステータスファイルの使い方

実体のないステータスファイルを指定すると，HiRDB の稼働中にステータスファイルを新規追加できます。例えば，ステータスファイルの障害などで予備ファイルが少なくなったときに，実体のな

いステータスファイルを予備ファイルにします。実体のないステータスファイルを予備ファイルにする手順を次に示します。

〈手順〉

1. pdstsinit コマンドで，システムファイル用の HiRDB ファイルシステム領域にステータスファイルを作成します。
2. pdstsopen コマンドで，ステータスファイルをオープンします。

この操作は HiRDB 稼働中に実行できます。HiRDB を一度停止する必要はありません。

- **メリット及びデメリット**

  実体のないステータスファイルを定義すると，HiRDB ファイルシステム領域の占有量は小さくなります。しかし，実体のないステータスファイルを予備ファイルとして追加するときに，システムファイル用の HiRDB ファイルシステム領域に十分な空き（ファイルを追加するだけの空き領域）がないと追加できないため，システムの信頼性は低くなります。

  実体のないステータスファイルを定義しないと，HiRDB ファイルシステム領域の占有量は大きくなるが，ファイル障害によるスワップ先が保証されるため，信頼性は高くなります。

- **注意事項**

  実体のないステータスファイルを定義した場合，HiRDB は HiRDB の開始時にステータスファイルに異常があると認識します。このため，pd_sts_initial_error オペランドに stop（省略値）を指定していると，HiRDB を開始できないので注意してください。実体のないステータスファイルを定義する場合は，pd_sts_initial_error オペランドに continue 又は excontinue を指定してください。また，HiRDB を開始する前に現用ファイルを pd_sts_last_active_file オペランドに指定する必要があります。

8.2.16　サーバ用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

52）　pd_sts_initial_error = stop | continue | excontinue

サーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，又はバックエンドサーバ）の開始時，HiRDB はサーバ用ステータスファイルの現用ファイルを特定する処理を実行します。この特定処理で，次に示す異常を検知したときの HiRDB の処置を指定します。

- サーバ用ステータスファイルの実体がない
- サーバ用ステータスファイルの障害を検知した

現用ファイルの特定処理は pd_sts_file_name_1～7 オペランドに指定したサーバ用ステータスファイルを対象とします。

stop：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。この場合，異常を検知したステータスファイルの対策を実施した後に HiRDB を開始してください。

continue 又は excontinue：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知しても，現用ファイルが正常な場合はサーバの開始処理を続行します。ただし，pd_sts_singleoperation オペランドの値（ステータスファイルの片系運転をするかどうか）によっては開始処理を中止することがあります。pd_sts_singleoperation オペランドとの関係を次に示します。

●pd_sts_singleoperation オペランドとの関係

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
continue	サーバ用ステータスファイルの異常を検知した場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。
stop（省略値）	サーバ用ステータスファイルの異常を検知した場合，HiRDB が現用ファイルを特定してサーバの開始処理を続行します。ただし，A 系と B 系のファイルが次に示す表（**HiRDB が現用ファイルを特定できないケース**）の条件のどれかに当てはまる場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB が現用ファイルを特定できない場合は，HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。

●HiRDB が現用ファイルを特定できないケース

pd_sts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
continue	障害閉塞	障害閉塞
	障害閉塞	オープン（初期状態）
	障害閉塞	ファイル実体なし
	オープン（初期状態）	障害閉塞
	オープン（初期状態）	ファイル実体なし
	ファイル実体なし	オープン（初期状態）
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし
excontinue	障害閉塞	障害閉塞
	障害閉塞	ファイル実体なし
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし

したがって，HiRDB 管理者の操作をなるべく少なくしたい場合（pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定するケースを減らしたい場合）は，次に示すようにオペランドを指定してください。

- pd_sts_initial_error オペランドに excontinue を指定
- pd_sts_singleoperation オペランドに stop を指定

《指定値の目安》

各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
サーバ開始時の HiRDB の処理	サーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	一部のサーバ用ステータスファイルに異常があっても，現用ファイルが正常な場合はサーバの開始処理を続行します。
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB 開始時の障害対策を簡易化する場合は continue 又は excontinue を指定します。
メリット	サーバの開始時に該当するサーバの全サーバ用ステータスファイルの正常が保証されます。このため，HiRDB の開始後に現用ファイルの障害が発生した場合, 予備ファイルにスワップできます。	サーバの開始時に一部のサーバ用ステータスファイルの障害を検知しても，残りの正常なファイルだけで HiRDB を開始できます。このため，HiRDB の停止時間を短くできます。この場合，予備ファイルが少なくなっているため，障害状態のステータスファイルをすぐに修復する必要があります。
デメリット	サーバ用ステータスファイルの障害によって HiRDB の開始処理が中止される可能性が高くなります。	予備ファイルが少ない状態で HiRDB を稼働することがあるため，システムの信頼性が低下します。予備ファイルの数によってはサーバ用ステータスファイルをスワップできないことがあります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，stop が仮定されます。

《注意事項》

- 現用ファイルが両系とも異常の場合は，このオペランドの指定値に関係なくサーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。
- HiRDB を開始する前に現用ファイルを pdstsinit コマンドで初期化しないでください。初期化すると HiRDB を再開始できません。
- このオペランドに excontinue を指定し，すべての世代のステータスファイルが片系運転でかつオープン（初期状態）の場合，HiRDB は起動しません。

《備考》

オペランドの指定値と HiRDB の処理及び HiRDB 管理者の処置を次に示します。

●オペランドの指定値と HiRDB の処理

pd_sts_initial_errorオペランドの値	ステータスファイル障害	pd_sts_singleoperationオペランドの値	HiRDBが現用ファイルを特定できるか	pd_sts_last_active_fileの指定	pd_sts_last_active_fileオペランドの値と，オープンできたステータスのうち，最新のファイルが一致するか	現用ファイル障害	pd_sts_last_active_sideオペランドの指定	pd_sts_last_active_sideオペランドに指定されたファイルが使用できるか	表「HiRDBの処理及びHiRDB管理者の処置」の対応番号
stop（省略値）	なし	－	－	－	－	－	－	－	[1]
	あり	－	－	－	－	－	－	－	[5]
continue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
excontinue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しません
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]

（凡例）－：該当しません（条件がHiRDBの処理に影響を与えません）。

●HiRDB の処理及び HiRDB 管理者の処置

対応番号	HiRDB の処理	HiRDB 管理者の処置
[1]	HiRDB の開始処理を続行します。	なし
[2]	HiRDB が最新の現用ファイルを特定して，開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[3]	pd_sts_last_active_file オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[4]	pd_sts_last_active_file 及び pd_sts_last_active_side オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[5]	pd_sts_initial_error オペランドに stop が指定されているため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000010 が示す対策をしてください。

対応番号	HiRDB の処理	HiRDB 管理者の処置
[6]	前回稼働時の現用ファイルが特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000016 が示す対策をしてください。
[7]	HiRDB が特定した現用ファイルの正常な系と，pd_sts_last_active_side オペランドに指定された系が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000017 が示す対策をしてください。
[8]	HiRDB が特定した現用ファイル名と，pd_sts_last_active_file オペランドに指定したファイル名が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000015 が示す対策をしてください。
[9]	前回稼働時の現用ファイルの正常な系が特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB Version 8 メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000018 が示す対策をしてください。

53）pd_sts_singleoperation = stop | continue

サーバ用ステータスファイルの片系運転をするかどうかを指定します。予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，正常な系（片方の系）だけで処理を続行することを**ステータスファイルの片系運転**といいます。ステータスファイルの片系運転については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

なお，予備ファイルがある状況で現用ファイルの片系に障害が発生した場合は，このオペランドの指定値に関係なく，ステータスファイルをスワップして処理を続行します（片系運転にはなりません）。

stop：

片系運転をしません。片系運転になる場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。HiRDB が異常終了した場合，予備ファイルを準備した後に HiRDB を開始してください。

continue：

片系運転をします。片系運転になると KFPS01044-I メッセージが出力されます。片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了したりすると，HiRDB を再開始できなくなるため，片系運転になった場合はすぐに予備ファイルを準備してください。

《指定値の目安》

- stop を指定してシステムの信頼性を上げることを推奨します。また，現用ファイルの障害に対しては，予備ファイルの数を増やして備えることを推奨します。
- 各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB を停止させないことを重視する場合は continue を指定します。
メリット	予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，片系運転をしないで HiRDB	予備ファイルがない状況で現用ファイルの片系に障害が発生しても，処理を続行できます。この

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
	を異常終了します。このため，現用ファイルの内容が失われる可能性が低くなります。	ため，ステータスファイルの障害によってHiRDB が停止する可能性が低くなります。
デメリット	ステータスファイルの障害によって HiRDB が停止する可能性が高くなります。ただし，予備ファイルの数を増やすことで可能性を低くできます。	片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，現用ファイルの内容が失われるため，HiRDB を再開始できなくなります。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，stop が仮定されます。

《ほかのオペランドとの関連》

pd_sts_singleoperation 及び pd_sts_initial_error オペランドの指定値の組み合わせによって，ステータスファイル障害時の HiRDB の処理が決定します。したがって，これら二つのオペランドの指定値は一緒に考えるようにしてください。

54)　pd_sts_last_active_file = "論理ファイル名"

～<識別子>((1～8 文字))

バックエンドサーバの開始時に現用にするステータスファイルの論理ファイル名を指定します。
HiRDB は，このオペランドに指定されたファイルと HiRDB が特定した現用ファイルを比較します。ファイルが一致する場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しますが，一致しない場合は開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **全ステータスファイルを初期化した直後に HiRDB を開始する場合**

   pd_sts_file_name1～7 オペランドに指定した最も小さい番号の正常な論理ファイル名を指定してください。この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，全ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_sts_last_active_side オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   全ステータスファイルを初期化してください。その後，1 の方法を実施してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

5. 実体のないステータスファイルを指定している場合

現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I
- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージうちで，最も直前に出力されたメッセージに表示されているステータスファイルが現用ファイルになります。

55）　pd_sts_last_active_side = A | B

現用ファイルの片系に障害が発生している状態でバックエンドサーバを開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- pd_sts_last_active_file オペランドを指定している

8.2.17　バックエンドサーバ接続保持機能に関するオペランド

バックエンドサーバ接続保持機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

56）　pd_bes_connection_hold = Y | N

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

バックエンドサーバ接続保持機能を使用するかどうかを指定します。

Y：バックエンドサーバ接続保持機能を使用します。

N：バックエンドサーバ接続保持機能を使用しません。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，N が仮定されます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDBESCONHOLD オペランドを指定します。PDBESCONHOLD オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

《注意事項》

バックエンドサーバ接続保持機能を使用する場合は，次に示す計算式を満たすようにしてください。

各バックエンドサーバプロセス数（pd_max_bes_process オペランドの値）

≧全フロントエンドサーバプロセス数（pd_max_users オペランドの値×フロントエンドサーバ数）

この計算式を満たさない場合，バックエンドサーバプロセス数が不足して SQL エラーになることがあります。また，HiRDB の稼働中にユティリティなどを実行する場合は，バックエンドサーバプロセス数にユティリティの分の余裕値を確保してください。

57）　pd_bes_conn_hold_trn_interval = バックエンドサーバ接続保持時間

～<符号なし整数>((0～3600))（単位：秒）

バックエンドサーバ接続保持時間を秒単位で指定します。

バックエンドサーバ接続保持機能を使用すると，HiRDB はトランザクションの終了後から次のトランザクションが実行されるまでの時間を監視します。次のトランザクションが実行されるまでの時間が，指定値の範囲内の場合はフロントエンドサーバとバックエンドサーバの接続を保持します。指定値を超えた場合は，トランザクション終了後にフロントエンドサーバとバックエンドサーバの接続を切断します。

なお，このオペランドに 0 を指定した場合は時間を監視しません。SQL DISCONNECT（XA ライブラリを使用時は xa_close），PDCWAITTIME オーバなどでフロントエンドサーバとクライアント間の接続が切断されたときだけ，フロントエンドサーバとバックエンドサーバの接続を切断します。

《オペランドの省略値》

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。サーバ共通定義の同じオペランドも省略すると，1 が仮定されます。

8.2.18　作業表用ファイルに関するオペランド

58）　pdwork -v "HiRDB ファイルシステム領域名称"〔，"HiRDB ファイルシステム領域名称"〕…

～< 141 文字以内のパス名>

作業表用ファイル用の HiRDB ファイルシステム領域の名称を指定します。作業表用ファイルとは，SQL 文を実行するときに一時的に情報を格納するファイルのことで HiRDB が自動的に作成します。作業表用ファイルを必要とする SQL 文については，マニュアル「HiRDB Version 8 解説」を参照してください。

なお，このオペランドは省略しないでください。省略した場合，作業表用ファイルを必要とする SQL 文を実行できないことがあります。

《注意事項》

- このオペランドに指定する HiRDB ファイルシステム領域は pdfmkfs コマンドで初期設定した HiRDB ファイルシステム領域を指定してください。
- 作業表用ファイルの容量が大きい場合は，領域長が大きい HiRDB ファイルシステム領域を指定してください。作業表用ファイルの容量見積もりについては，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

  ただし，HiRDB ファイルシステム領域を pdfmkfs -a コマンドで初期設定した場合，-n オプションで指定した値を使い切ると，HiRDB は領域サイズを自動で拡張します。pdfmkfs コマンドについては，マニュアル「HiRDB Version 8 コマンドリファレンス」を参照してください。
- 作業表用ファイル用の HiRDB ファイルシステム領域は，RD エリア及びシステムファイル用の HiRDB ファイルシステム領域と同じものを指定できません。
- このオペランドに HiRDB ファイルシステム領域を二つ以上指定している場合，作業表用ファイルの作成時にエラーが発生した HiRDB ファイルシステム領域は，それ以降，原則使用しません。ほかの HiRDB ファイルシステム領域だけを使用します。

  例外として，容量不足やファイル数オーバーなどによって，ほかのすべての HiRDB ファイルシステム領域で作業表用ファイルの作成が失敗した場合に，使用されなくなった HiRDB ファイル

システム領域で作業表用ファイルを作成しようとします。その結果，作業表用ファイルが正常に作成された場合，その HiRDB ファイルシステム領域を再使用するようになります。

なお，バックエンドサーバを一度終了して開始すると，作業表用ファイルの作成時にエラーが発生して使用されなくなった HiRDB ファイルシステム領域が使用できるようになります。

《オペランドの規則》

- HiRDB ファイルシステム領域名称は必ず 1 個以上指定してください。
- HiRDB ファイルシステム領域名称は最大 16 個指定できます。
- このオペランドはバックエンドサーバ定義内に一つだけ指定できます。複数個指定した場合は最初に指定したものが有効となります。
- ほかのバックエンドサーバと同じ HiRDB ファイルシステム領域を指定しないでください。
- ディクショナリサーバと同じ HiRDB ファイルシステム領域を指定しないでください。

8.2.19　システムログファイルの構成に関するオペランド

59)　pdlogadfg -d sys -g ファイルグループ名〔ONL〕

システムログファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシステムログファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～200 個指定できます。

《オペランドの規則》

このオペランドは 2～200 個指定できます。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

60)　pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名"〔-b "システムログファイル名"〕

ファイルグループを構成するシステムログファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "システムログファイル名"：～<パス名>((167 文字以内))

ファイルグループを構成するシステムログファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

-b "システムログファイル名"：～<パス名>((167 文字以内))

システムログファイルを二重化（pd_log_dual オペランドに Y を指定）する場合に，B 系のシステムログファイル名を絶対パス名で指定します。pd_log_dua オペランドに Y を指定しない場合は，システムログファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

《オペランドの規則》

HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥log01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。log01 については大文字と小文字の区別をします。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

8.2.20　シンクポイントダンプファイルの構成に関するオペランド

61）　pdlogadfg -d spd -g ファイルグループ名〔ONL〕

シンクポイントダンプファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシンクポイントダンプファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～30 個指定できます。

《オペランドの規則》

このオペランドは 2～60 個指定できます。

62）　pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕

ファイルグループを構成するシンクポイントダンプファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～<識別子>((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "シンクポイントダンプファイル名"：～<パス名>((167 文字以内))

ファイルグループを構成するシンクポイントダンプファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

-b "シンクポイントダンプファイル名"：～<パス名>((167 文字以内))

シンクポイントダンプファイルを二重化（pd_spd_dual = Y を指定）する場合に，B 系のシンクポイントダンプファイル名を絶対パス名で指定します。pd_spd_dual = Y の指定がない場合は，シンクポイントダンプファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

《オペランドの規則》

HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。C:¥hirdb¥sysfile¥sync01 の場合，C:¥hirdb¥sysfile については大文字と小文字の区別をしません。sync01 については大文字と小文字の区別をします。

8.2.21　プラグインに関するオペランド

63)　pdplgprm -n プラグイン名称〔 -s 共用メモリサイズ〕

プラグインの名称及びプラグインが使用する共用メモリの大きさを指定します。プラグインを使用しない場合，又はプラグインがバックエンドサーバで稼働しない場合は，このオペランドを省略してください。

《前提条件》

ここで指定するプラグインは，あらかじめ pdplgrgst コマンドで HiRDB に登録しておいてください。

-n プラグイン名称：～<識別子>((1～30 文字))

ここで指定するプラグイン名称については，プラグインに付いているマニュアルを参照してください。

-s 共用メモリサイズ：～<符号なし整数>((1～2000000))《0》(単位：キロバイト)

プラグインが使用する共用メモリの大きさをキロバイト単位で指定します。プラグインが使用する共用メモリの大きさについては，プラグインに付いているマニュアルを参照してください。

8.2.22　HiRDB External Data Access 機能（環境変数）に関するオペランド

HiRDB External Data Access 機能については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

なお，HiRDB の稼働プラットフォームによっては HiRDB External Data Access 機能を使用できません。詳細については，マニュアル「HiRDB Version 8 解説」の「HiRDB External Data Access」を参照してください。

64）　外部サーバで必要な環境変数

外部サーバで必要な環境変数を指定します。必要な環境変数については，接続する外部サーバの DBMS のマニュアルを参照してください。

必要な環境変数が指定されていないと，外部サーバへアクセスするときにエラーとなることがあります。

《前提条件》

HiRDB External Data Access が必要です。

9 UAP 環境定義

この章では，UAP 環境定義の各オペランドの内容について説明します。

9.1　オペランドの形式

UAP 環境定義では UAP の実行環境を定義します。ここでは，UAP 環境定義の各オペランドの形式について説明します。なお，表中の「番号」は，「9.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

番号	形式	強制，異常終了後の変更可否	計画停止後の変更可否
1	〔set pd_uap_wait = Y｜N〕	○	○
2	〔{{pdlbuffer -a ローカルバッファ名 {-r RD エリア名〔，RD エリア名〕… ｜-i 認可識別子．インデクス識別子} -n バッファ面数 〔-p 一括入力最大ページ数〕}}〕	○	○

（凡例）

○：指定値を変更できます。

9.2　オペランドの説明

1）　pd_uap_wait = Y｜N

ローカルバッファを使用してアクセスする RD エリア又はインデクスがほかのユーザに使用されている場合の UAP の動作を指定します。

Y：使用中のユーザの終了を待ちます。UAP は WAIT 状態になります。

N：ほかのユーザが使用中の場合は UAP をエラーリターンします。

二つ以上の pdlbuffer オペランドと一緒にこのオペランドを指定した場合，そのすべてのローカルバッファに対してこのオペランドの指定が有効になります。

《前提条件》

pdlbuffer オペランドを指定する必要があります。

2）　pdlbuffer -a ローカルバッファ名

{-r RD エリア名〔，RD エリア名〕…｜-i 認可識別子．インデクス識別子}

-n バッファ面数

〔-p 一括入力最大ページ数〕

UAP が使用するローカルバッファを定義します。ローカルバッファとはディスク上の表又はインデクスなどのデータを入出力するときに使用する領域のことで，シングルサーバ又はバックエンドサーバのプロセス固有メモリ上に確保されます。なお，ローカルバッファは 100 個まで定義できます。HiRDB/パラレルサーバの場合は 1 バックエンドサーバ当たり 100 個まで定義できます。

また，ローカルバッファを指定するときは，pdlbuffer オペランドのオプションの指定によって次のように排他が掛かります。

pdlbuffer オペランドの指定	排他資源	排他モード
pdlbuffer オペランドの-r オプションを指定した場合	指定された RD エリア	EX
pdlbuffer オペランドの-i オプションを指定した場合	インデクス格納 RD エリア	SU
	インデクス格納表	EX
	指定されたインデクス	EX

《前提条件》

ローカルバッファに指定する RD エリア及びインデクスに対して，pdbuffer オペランドでグローバルバッファを割り当てておく必要があります。

《適用基準》

次に示す条件をすべて満たす場合にローカルバッファを定義します。

- 大量のデータを検索又は更新する
- アクセス対象の RD エリアがほかの UAP からアクセスされない

また，HiRDB に常時接続する UAP はシステムへの影響（メモリの圧迫，プロセスの占有など）が大きいため，ローカルバッファを定義しないでください。

《注意事項》

HiRDB/パラレルサーバの場合の注意事項を次に示します。

- ローカルバッファの使用時にサーバプロセスが異常終了すると，アボートコード Phb3008 を出力してユニットが異常終了することがあります。

- サーバプロセスの異常終了時に更新ページがあると，ロールバックプロセスで回復処理ができないことがあります。その場合，ユニットの再開始時に回復処理を実行します。
- ローカルバッファを使用している場合に障害が発生したときの HiRDB の処理と対処方法については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

-a ローカルバッファ名

～<識別子>((1～16 文字))

ローカルバッファの名称を指定します。ほかのローカルバッファと同じ名称を指定できません。ただし，UAP 環境定義ファイルが異なる場合は同じ名称を指定できます。

-r RD エリア名〔，RD エリア名〕…

～<識別子>((1～30 文字))

データ用のローカルバッファを割り当てる RD エリアの名称を指定します。ユーザ用 RD エリアの名称だけ指定できます。

同じ名称は指定できません。ただし，UAP 環境定義ファイルが異なる場合は同じ名称を指定できます。

《指定値の目安》

一つのローカルバッファに複数の RD エリアを割り当てる場合は，RD エリアの最大ページ長がバッファ長になります。したがって，ページ長が同じか，又はページ長が近い RD エリアを同じローカルバッファに割り当てると，入出力回数を削減できます。ただし，同じページ長の RD エリアでも，次に示すような RD エリアは別々のローカルバッファに割り当てた方が入出力回数を削減できます。

- 用途の異なる表を格納する RD エリア
- ランダムアクセスが多い RD エリアと順次アクセスが多い RD エリア

《オペランドの規則》

- RD エリア名に半角英数字以外の文字を含む場合は引用符（"）で囲んでください。
- 一つのローカルバッファには最大 3200 個の RD エリアを定義できます。

-i 認可識別子．インデクス識別子

認可識別子：～<識別子>((1～8 文字))

インデクス識別子：～<識別子>((1～30 文字))

インデクス用のローカルバッファを割り当てるインデクスの名称（認可識別子．インデクス識別子）を指定します。

同じ名称は指定できません。ただし，UAP 環境定義ファイルが異なる場合は同じ名称を指定できます。

《オペランドの規則》

認可識別子又はインデクス識別子に半角英数字以外の文字を含む場合は，認可識別子及びインデクス識別子をそれぞれ引用符（"）で囲んでください。

《指定値の目安》

使用頻度の高いインデクスを指定してください。使用頻度の高いインデクスにローカルバッファを割り当てると，インデクスページのメモリ常駐度が上がり，入出力回数の削減効果が出ます。

特に，クラスタキー又はユニークキーに定義したインデクスをローカルバッファに割り当てると効果が大きくなります。なお，クラスタキーのインデクス識別子は HiRDB が決めるため，表を定義した後にディクショナリ表（SQL_INDEXES 表の INDEX_NAME 列）を検索してインデクス識別子を確認してください。

-n バッファ面数

～<符号なし整数>((4～125000))

ローカルバッファの面数を指定します。

《指定値の目安》

- ローカルバッファはプロセス固有メモリ上に確保されます。必要以上に大きくするとほかのメモリの使用時にページングが多発し，性能が低下します。
- バッファ面数が大き過ぎるとプロセス固有メモリを確保できないことがあるので注意してください。
- シーケンシャルアクセスをする UAP の場合は，バッファリングの効果が望めないため，バッファ面数に最小値 4 を指定してください。

-p 一括入力最大ページ数

～<符号なし整数>((2～256))

プリフェッチ機能の一括入力最大ページ数を指定します。このオペランドを省略するとプリフェッチ機能は動作しません。

《指定値の目安》

メモリ量と入出力時間の削減効果のコストパフォーマンスから，次に示す計算式を満足する値を指定してください。

a × b = 64～128　（単位：キロバイト）

a：プリフェッチ対象表のデータ，又はインデクスが格納されている RD エリアのページ長

b：一括入力最大ページ数

なお，RD エリアのセグメントサイズが 1 の場合は，このオプションを指定しても効果が得られないため指定しないでください。

10 外部サーバ情報定義（外部サーバがHiRDBの場合）

外部サーバがHiRDBの場合の外部サーバ情報定義について説明します。この定義はHiRDB External Data Access機能使用時に定義します。

10.1　オペランドの形式

外部サーバ情報定義では，HiRDB External Data Access 機能の実行環境を定義します。なお，表中の「番号」は，「10.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●外部サーバ情報定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB External Data Access 機能を使用できます。

●再開始前に変更できるオペランド

外部サーバ情報定義の全オペランドが HiRDB の再開始前（計画停止，強制終了，又は異常終了後）に変更できます。

(1)　外部サーバとの接続及び解除

番号	形式
1	〔set pd_hb_db_con = STARTUP ｜ ACCESS〕
2	〔set pd_hb_db_dis_con = SHUTDOWN ｜ COMMIT〕

(2)　外部サーバエラー情報

番号	形式
3	〔set pd_hb_e_mode = ALL ｜ NONE〕
4	〔set pd_hb_e_code = SQLCODE〔，SQLCODE〕…〕
5	〔set pd_hb_e_size = 外部サーバエラー情報ファイルのサイズ〕

(3)　外部サーバインタフェーストレース情報

番号	形式
6	〔set pd_hb_l_path = 外部サーバインタフェーストレース情報ファイルの出力先ディレクトリ〕
7	〔set pd_hb_l_size = 外部サーバインタフェーストレース情報ファイルのサイズ〕
8	〔set pd_hb_l_prm = Y ｜ N〕
9	〔set pd_hb_l_prm_size = 外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長〕
10	〔set pd_hb_l_mode = ALL ｜ PARTIAL〕

(4)　FETCH

番号	形式
11	〔set pd_hb_ary_fec_num = 配列を使用した FETCH 機能で一度に取得する行数〕
12	〔putenv PDCURSORLVL 0｜1｜2 〕

(5)　外部サーバとの接続

番号	形式
13	putenv PDHOST 外部 HiRDB のホスト名〔，外部 HiRDB の予備系のホスト名〕
14	〔putenv PDFESHOST 外部 HiRDB のフロントエンドサーバのホスト名〔：外部 HiRDB のフロントエンドサーバのポート番号〕〔，外部 HiRDB の予備系フロントエンドサーバのホスト名〔：外部 HiRDB の予備系フロントエンドサーバのポート番号〕〕〕
15	putenv PDNAMEPORT 外部 HiRDB のポート番号
16	〔putenv PDSERVICEPORT 外部 HiRDB の高速接続用のポート番号〔，外部 HiRDB の予備系の高速接続用のポート番号〕〕
17	〔putenv PDSERVICEGRP 外部 HiRDB のサーバ名〕
18	〔putenv PDSRVTYPE WS ｜ PC〕
19	〔putenv PDCLTRCVPORT 予約する受信ポート番号の範囲〕
20	〔putenv PDCLTRCVADDR 外部サーバ接続用のバックエンドサーバの IP アドレス又はホスト名〕
21	〔putenv PDAUTORECONNECT YES ｜ NO〕
22	〔putenv PDRCCOUNT 自動再接続機能使用時の CONNECT リトライ回数の上限〕
23	〔putenv PDRCINTERVAL 自動再接続機能使用時の CONNECT リトライ間隔〕

(6)　データベースの更新ログ

番号	形式
24	〔putenv PDDBLOG ALL ｜ NO〕

(7)　クライアントグループ

番号	形式
25	〔putenv PDCLTGRP クライアントグループ名〕

(8)　通信処理

番号	形式
26	〔putenv PDIPC MEMORY ｜ DEFAULT〕
27	〔putenv PDSENDMEMSIZE データ送信用メモリサイズ〕
28	〔putenv PDRECVMEMSIZE データ受信用メモリサイズ〕
29	〔putenv PDTCPCONOPT 0 ｜ 1〕
30	〔putenv PDKALVL 0 ｜ 1 ｜ 2〕
31	〔putenv PDKATIME 外部 HiRDB へのパケット送信間隔〕
32	〔putenv PDNODELAYACK YES ｜ NO〕

(9)　システム監視

番号	形式
33	〔putenv PDCWAITTIME 〔HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 1〕〔，HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 2〕〕
34	〔putenv PDSWAITTIME 〔外部 HiRDB の最大待ち時間 1〕〔，外部 HiRDB の最大待ち時間 2〕〕

(10)　トラブルシュート情報

番号	形式
35	〔putenv PDCLTPATH SQL トレースファイル及びエラーログファイルの出力先ディレクトリ〕
36	〔putenv PDSQLTRACE SQL トレースファイルのサイズ〕
37	〔putenv PDSQLTEXTSIZE SQL トレースファイルに出力する SQL 文のサイズ〕
38	〔putenv PDSQLEXECTIME YES ｜ NO〕
39	〔putenv PDUAPERLOG エラーログファイルのサイズ〕
40	〔putenv PDPRMTRC YES ｜ NO〕
41	〔putenv PDPRMTRCSIZE SQL トレース中の？パラメタ情報の最大データ長〕
42	〔putenv PDTRCMODE ERR ｜ NONE〕
43	〔putenv PDTRCPATH 動的 SQL トレースファイルの出力先ディレクトリ〕
44	〔putenv PDUAPREPLVL 〔s〕〔u〕〔p〕〔r〕 ｜ a〕
45	〔putenv PDREPPATH UAP 統計レポートの出力先ディレクトリ〕
46	〔putenv PDSQLTRCOPENMODE CNCT ｜ SQL〕
47	〔putenv PDVWOPTMODE アクセスパス情報の取得モード〕

(11)　排他制御

番号	形式
48	〔putenv PDLOCKLIMIT 外部 HiRDB に対する排他要求の上限値〕

(12)　インナレプリカ機能

番号	形式
49	〔putenv PDDBACCS 外部 HiRDB の RD エリア世代番号〕
50	〔putenv PDDBORGUAP YES ｜ NO〕

(13)　SQL の最適化

番号	形式
51	〔putenv PDSQLOPTLVL 外部 HiRDB の SQL 最適化オプション〔，外部 HiRDB の SQL 最適化オプション〕…〕
52	〔putenv PDADDITIONALOPTLVL 外部 HiRDB の SQL 拡張最適化オプション〔，外部 HiRDB の SQL 拡張最適化オプション〕…〕
53	〔putenv PDAGGR 外部 HiRDB のグループ化処理時に発生するグループ数〕
54	〔putenv PDHASHTBLSIZE 外部 HiRDB のハッシュジョイン，副問合せのハッシュ実行適用時のハッシュ表サイズ〕

(14)　拡張 SQL エラー情報出力機能

番号	形式
55	〔putenv PDUAPEXERLOGUSE YES｜NO〕
56	〔putenv PDUAPEXERLOGPRMSZ エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長〕

(15)　バックエンドサーバ接続保持機能

番号	形式
57	〔putenv PDBESCONHOLD YES ｜ NO〕
58	〔putenv PDBESCONHTI バックエンドサーバ接続保持時間〕

(16)　系切り替え機能

番号	形式
59	〔putenv PDHATRNQUEUING NO〕

10.2　オペランドの説明

10.2.1　外部サーバとの接続及び解除に関するオペランド

1）　pd_hb_db_con = STARTUP｜ACCESS

外部サーバに接続するタイミングを指定します。

STARTUP：

バックエンドサーバプロセスの起動時に外部サーバに接続します。

STARTUP を指定すると, 外部サーバへの処理が実行されないときも外部サーバに接続しています。外部サーバの接続数に限りがある場合は ACCESS を指定してください。

ACCESS：

バックエンドサーバプロセスから外部サーバへのアクセス要求が初めてあったときに外部サーバに接続します。

2）　pd_hb_db_dis_con = SHUTDOWN｜COMMIT

外部サーバとの接続を解除するタイミングを指定します。

SHUTDOWN：

バックエンドサーバプロセスの終了時に接続を解除します。

SHUTDOWN を指定すると，外部サーバへの処理が実行されないときも外部サーバに接続しています。外部サーバの接続数に限りがある場合は COMMIT を指定してください。

COMMIT：

外部サーバに対してコミットを発行した直後に接続を解除します。

10.2.2　外部サーバエラー情報に関するオペランド

3）　pd_hb_e_mode = ALL ｜ NONE

外部サーバエラー情報を出力するかどうかを指定します。外部サーバエラー情報については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

ALL：外部サーバエラー情報を出力します。

NONE：外部サーバエラー情報を出力しません。

4）　pd_hb_e_code = SQLCODE〔，SQLCODE〕…

～<符号あり整数>((－214748364～－1))

このオペランドには，外部サーバの SQLCODE（SQL 連絡領域 SQLCA の SQLCODE に返される値）を指定します。ここで指定した SQLCODE に対応するエラーメッセージを外部サーバエラー情報に出力しません。

このオペランドを省略すると，すべての SQLCODE に対応するエラーメッセージを外部サーバエラー情報ファイルに出力します。SQLCODE については，各 DBMS のマニュアルを参照してください。

《オペランドの規則》

- 複数の SQLCODE を指定する場合は，SQLCODE ごとにコンマ（,）で区切ってください。
- SQLCODE は 10 個まで指定できます。

5）　pd_hb_e_size = 外部サーバエラー情報ファイルのサイズ

～<符号なし整数>((0，4～4096))《32》（単位：キロバイト）

外部サーバエラー情報ファイルのサイズをキロバイト単位で指定します。サイズの上限を決めない場合は 0 を指定します。

10.2.3　外部サーバインタフェーストレース情報に関するオペランド

6）　pd_hb_l_path = 外部サーバインタフェーストレース情報ファイルの出力先ディレクトリ

～<パス名>《%PDDIR%¥spool¥》

外部サーバインタフェーストレース情報ファイルの出力先ディレクトリを指定します。このディレクトリ下に**外部サーバインタフェーストレース情報**を出力します。出力ファイルはこのディレクトリ下に二つ作成されます。ファイル名は pdhl 外部サーバ名 1，又は pdhl 外部サーバ名 2 です。

外部サーバインタフェーストレース情報については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

7）　pd_hb_l_size = 外部サーバインタフェーストレース情報ファイルのサイズ

～<符号なし整数>((0，4～4096))《1024》（単位：キロバイト）

外部サーバインタフェーストレース情報ファイルのサイズをキロバイト単位で指定します。サイズの上限を決めない場合は 0 を指定します。

ファイル容量がこのオペランドに指定した値を超えると，ファイルをスワップしてもう一方のファイルに出力します。

《指定値の目安》

1 トレース当たり平均 300 バイトの情報が出力されます（？パラメタ情報を除く）。

《注意事項》

- クライアント環境定義の PDSQLTRACE オペランドを指定して実行した UAP に対して，外部サーバ単位に外部サーバインタフェーストレース情報を取得します。複数のトランザクションが一つのファイルに書き込みをするため，書き込み時に排他を掛けます。したがって，外部サーバインタフェーストレース情報を取得すると，トランザクションの処理性能及び同時実行性が低下します。
- 外部サーバのアクセス状況によって，ファイル容量がこのオペランドに指定した値を大幅に超えてしまうことがあります。したがって，十分な空きがあるディスクに外部サーバインタフェーストレース情報を取得してください。

8）　pd_hb_l_prm = Y｜N

外部サーバインタフェーストレース情報に，？パラメタ情報を出力するかどうかを指定します。

Y：？パラメタ情報を出力します。

N：？パラメタ情報を出力しません。

9）　pd_hb_l_prm_size = 外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長

～<符号なし整数>((4～4096))《256》（単位：バイト）

外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長をバイト単位で指定します。

《前提条件》

pd_hb_l_prm オペランドに Y を指定している必要があります。

10）　pd_hb_l_mode = ALL｜PARTIAL

外部サーバインタフェーストレース情報のすべての情報を出力するかどうかを指定します。

ALL：すべての情報を出力します。

PARTIAL：外部サーバと通信を行う関数についての情報だけを出力します。

10.2.4　FETCH に関するオペランド

11）　pd_hb_ary_fec_num = 配列を使用した FETCH 機能で一度に取得する行数

～<符号なし整数>((1～30000))

配列を使用した FETCH で一度に取得する行数を指定します。

外部表からの検索結果行数がある程度予測できる場合，このオペランドを指定してください。HiRDB と外部サーバ間の通信によるオーバヘッドの削減が期待できます。外部サーバへの配列 FETCH 数による外部表への SQL 実行時間及び FETCH 回数については，外部サーバの利用状況に関する統計情報で調べてください。外部サーバの利用状況に関する統計情報については，マニュアル「HiRDB Version 8 コマンドリファレンス」の「統計解析ユティリティ」を参照してください。

このオペランドを省略すると，HiRDB が処理をするのに最適な値を HiRDB が自動計算します。

《指定値の目安》

- 指定値が小さ過ぎると, HiRDB と外部サーバ間の通信回数が多くなるため, その分のオーバヘッドが大きくなります。
- 指定値が大き過ぎると，メモリを不要に使用するため，メモリ資源を圧迫します。
- 外部表からの検索結果行数が業務によって変わる場合はこのオペランドを省略することをお勧めします。このオペランドを省略したときの利点は，外部表から取得する 1 行以上のデータが pd_sql_send_buff_size オペランドに指定した通信用バッファ長に収まるかどうかによって異なります。収まる場合は外部表から取得したデータを HiRDB は効率良く処理します。収まらない場合は外部表アクセス時のメモリ使用量を抑えます。

12）　PDCURSORLVL 0｜1｜2

カーソルを使用した検索をする場合に，HiRDB サーバに対してのカーソルオープン・クローズの要求を，どのタイミングで行うかを指定します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDCURSORLVL オペランドと同じです。PDCURSORLVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

0：

アプリケーションから，カーソルオープン・クローズの要求を受けた場合，HiRDB クライアントは HiRDB サーバに対して，そのまま実行要求をします。

1：

検索データがない場合，HiRDB サーバは SQLCODE=100 の返却と同時に，HiRDB クライアントからの要求なしでカーソルをクローズします。HiRDB クライアントは，アプリケーションからのカーソルクローズ要求を受けた場合に，既に SQLCODE=100 を検知していれば，HiRDB サーバに対してカーソルクローズ要求をしません。SQLCODE=100 を検知していない場合にだけ，カーソルクローズ要求をします。

カーソルオープン要求については，0 を指定した場合と同じです。

2：

アプリケーションからのカーソルオープン要求を受けた場合に，HiRDB サーバに対しては実行を要求しないで，初回取り出し要求と同時にカーソルオープン要求をします。

カーソルクローズ要求については，1 を指定した場合と同じです。

10.2.5　外部サーバとの接続に関するオペランド

ここで説明するオペランドを指定すると，外部 HiRDB への接続時間を短縮できます。外部 HiRDB との接続に必要なオペランドの一覧を次に示します。

オペランド名	外部サーバが HiRDB/シングルサーバの場合		外部サーバが HiRDB/パラレルサーバの場合				
			1 フロントエンドサーバの場合		マルチフロントエンドサーバの場合		
						フロントエンドサーバ指定接続	
	通常接続	高速接続	通常接続	高速接続	通常接続	FES ホストダイレクト接続	高速接続
PDHOST	○	○	○	○	○	○	○
PDFESHOST	×	×	×	○	×	○	○
PDNAMEPORT	○	○	○	○	○	○	○
PDSERVICEPORT	×	○	×	○	×	×	○
PDSERVICEGRP	×	○	×	○	×	○	○

（凡例）

○：指定が必要です。

×：指定不要です。

13）　PDHOST 外部 HiRDB のホスト名〔，外部 HiRDB の予備系のホスト名〕

～〈ホスト名〉

外部 HiRDB（外部サーバ側の HiRDB）のホスト名を指定します。

外部 HiRDB が HiRDB/シングルサーバの場合は，シングルサーバがあるサーバマシンのホスト名を指定します。外部 HiRDB が HiRDB/パラレルサーバの場合は，システムマネジャがあるサーバマシンのホスト名を指定します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDHOST オペランドと同じです。PDHOST オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

14）　PDFESHOST 外部 HiRDB のフロントエンドサーバのホスト名〔：外部 HiRDB のフロントエンドサーバのポート番号〕〔，外部 HiRDB の予備系フロントエンドサーバのホスト名〔：外部 HiRDB の予備系フロントエンドサーバのポート番号〕〕

～〈ホスト名〉

外部 HiRDB がマルチフロントエンドサーバの場合にこのオペランドを指定します。外部 HiRDB の接続先フロントエンドサーバのホスト名を指定します。このオペランドを省略すると，接続先のフロントエンドサーバを外部 HiRDB が決定します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDFESHOST オペランドと同じです。PDFESHOST オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照して

ください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

15）　PDNAMEPORT 外部 HiRDB のポート番号

～<符号なし整数>((5001～65535))

外部 HiRDB のポート番号（外部 HiRDB の pd_name_port オペランドの値）を指定します。このオペランドは省略できません。必ず指定してください。

このオペランドの機能及び指定方法は，クライアント環境定義の PDNAMEPORT オペランドと同じです。PDNAMEPORT オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

16）　PDSERVICEPORT 外部 HiRDB の高速接続用のポート番号〔，外部 HiRDB の予備系の高速接続用のポート番号〕

～<符号なし整数>((5001～65535))

高速接続用のポート番号を指定します。このオペランドを指定すると，外部 HiRDB への接続時間を短縮できます。

このオペランドの機能及び指定方法は，クライアント環境定義の PDSERVICEPORT オペランドと同じです。PDSERVICEPORT オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

17）　PDSERVICEGRP 外部 HiRDB のサーバ名

～<文字列>

外部 HiRDB のサーバ名を指定します。外部 HiRDB が HiRDB/シングルサーバの場合はシングルサーバ名を，HiRDB/パラレルサーバの場合はフロントエンドサーバ名を指定します。外部 HiRDB がマルチフロントエンドサーバの場合は，接続先のフロントエンドサーバ名を指定します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDSERVICEGRP オペランドと同じです。PDSERVICEGRP オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

18）　PDSRVTYPE WS｜PC

外部 HiRDB の種別を指定します。

WS：外部 HiRDB が UNIX 版の場合に指定します。

PC：外部 HiRDB が Windows 版の場合に指定します。

19）　PDCLTRCVPORT 予約する受信ポート番号の範囲

～<符号なし整数>((5001～65535))

外部 HiRDB との通信に使用する受信ポート番号の予約範囲を指定します。このオペランドを省略すると，自動的に OS によって空いているポート番号が割り当てられます。したがって，通常はこのオペランドを指定する必要はありません。

《オペランドの指定方法》

このオペランドの指定例を次に示します。

- ポート番号を一つ指定する場合（10000 を使用します）

  PDCLTRCVPORT 10000

- ポート番号を範囲指定する場合（10000～105000 を使用します）

PDCLTRCVPORT 10000－10500

《指定値の目安》

ポート番号の数が不足しないように範囲指定してください。必要なポート番号の数を次に示します。

必要なポート番号の数＞外部 HiRDB の pd_max_bes_process オペランドの値

《注意事項》

同一ユニット内に外部サーバ接続用のバックエンドサーバが複数ある場合，このオペランドの指定値に注意してください。各外部サーバ定義でこのオペランドの指定値が重複していると（ポート番号の予約範囲が重複していると），重複した範囲は各バックエンドサーバで使用します。そのため，ポート番号の数が不足する可能性があります。

このオペランドの機能及び指定方法は，クライアント環境定義の PDCLTRCVPORT オペランドと同じです。PDCLTRCVPORT オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

20）PDCLTRCVADDR 外部サーバ接続用のバックエンドサーバの IP アドレス又はホスト名

～＜符号なし整数＞又は＜識別子＞

外部サーバ接続用のバックエンドサーバがあるサーバマシンに複数の通信経路が設定されている場合にこのオペランドを指定します。外部サーバとの通信に使用する通信経路の IP アドレス又はホスト名を指定します。

このオペランドを省略すると，外部サーバ接続用のバックエンドサーバがあるサーバマシンの標準ホスト名の IP アドレスが使用されます。

このオペランドの機能及び指定方法は，クライアント環境定義の PDCLTRCVADDR オペランドと同じです。PDCLTRCVADDR オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

21）PDAUTORECONNECT YES | NO

外部 HiRDB に対して自動再接続機能を使用するかどうかを指定します。自動再接続機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

YES：自動再接続機能を使用します。

NO：自動再接続機能を使用しません。

再接続するときの CONNECT リトライ回数，CONNECT リトライ間隔については，外部サーバ情報定義の PDRCCOUNT 及び PDRCINTERVAL オペランドで指定します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDAUTORECONNECT オペランドと同じです。PDAUTORECONNECT オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

22）PDRCCOUNT 自動再接続機能使用時の CONNECT リトライ回数の上限

～＜符号なし整数＞((1～200))《5》

自動再接続機能を使用して再接続するときの CONNECT リトライ回数の上限を指定します。
CONNECT リトライ回数がこのオペランドの値を超えた場合，外部サーバへの接続処理を中止します。

《前提条件》

外部サーバ情報定義の PDAUTORECONNECT オペランドに YES を指定する必要があります。

23）PDRCINTERVAL 自動再接続機能使用時の CONNECT リトライ間隔

～＜符号なし整数＞((0～600))《5》（単位：秒）

自動再接続機能を使用して再接続するときの CONNECT リトライ間隔を秒単位で指定します。

《前提条件》

外部サーバ情報定義の PDAUTORECONNECT オペランドに YES を指定する必要があります。

10.2.6　データベースの更新ログに関するオペランド

24)　PDDBLOG ALL | NO

外部サーバのデータベースの更新ログを取得するかどうかを指定します。

ALL：

外部サーバに対する SQL をログ取得モードで実行します（データベースの更新ログを取得します）。

NO：

外部サーバに対する SQL をログレスモードで実行します（データベースの更新ログを取得しません）。

10.2.7　クライアントグループに関するオペランド

25)　PDCLTGRP クライアントグループ名

～<文字列>((1 文字))

外部 HiRDB で**クライアントグループの接続枠保証機能**を使用している場合にこのオペランドを指定します。このオペランドにはクライアントグループ名を指定します。外部 HiRDB の pdcltgrp オペランドに指定したクライアントグループ名を英大文字 1 文字で指定します。英小文字を指定した場合は，英大文字を指定したと仮定します。

外部 HiRDB の pdcltgrp オペランドを指定していない場合，又は pdcltgrp オペランドに指定されていないクライアントグループ名を指定した場合は，このオペランドの指定を無効にします。

このオペランドの機能及び指定方法は，クライアント環境定義の PDCLTGRP オペランドと同じです。PDCLTGRP オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。クライアントグループの接続枠保証機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は, 対応するバージョンのマニュアルを参照してください。

10.2.8　通信処理に関するオペランド

26)　PDIPC MEMORY | DEFAULT

外部サーバ接続用のバックエンドサーバと外部 HiRDB（シングルサーバ又はフロントエンドサーバがあるユニット）が同じサーバマシンにある場合のプロセス間通信の方法を指定します。

MEMORY：

プロセス間通信にメモリを使用します。これを，**プロセス間メモリ通信機能**といいます。外部 HiRDB との通信が高速になります。

DEFAULT：

プロセス間通信に各 OS でのデフォルトの通信方式（TCP/IP 又は PIPE）を使用します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDIPC オペランドと同じです。PDIPC オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

27)　PDSENDMEMSIZE データ送信用メモリサイズ

～<符号なし整数>((4～2097152))《16》（単位：キロバイト）

プロセス間メモリ通信機能を使用する場合（PDIPC オペランドに MEMORY を指定する場合）にこのオペランドを指定します。

外部サーバへデータを送信するときのデータ格納領域サイズをキロバイト単位で指定します。4 の倍数で指定してください。

このオペランドの機能及び指定方法は，クライアント環境定義の PDSENDMEMSIZE オペランドと同じです。PDSENDMEMSIZE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

28)　PDRECVMEMSIZE データ受信用メモリサイズ

～<符号なし整数>((4～2097152))《32》（単位：キロバイト）

プロセス間メモリ通信機能を使用する場合（PDIPC オペランドに MEMORY を指定する場合）にこのオペランドを指定します。

外部サーバからデータを受信するときのデータ格納領域サイズをキロバイト単位で指定します。4 の倍数で指定してください。

このオペランドの機能及び指定方法は，クライアント環境定義の PDRECVMEMSIZE オペランドと同じです。PDRECVMEMSIZE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

29)　PDTCPCONOPT 0 | 1

HiRDB External Data Access Adapter で使用する HiRDB クライアントと外部 HiRDB（バージョン 06-02 以降）間の通信で使用する TCP ポートの数を削減するかどうかを指定します。TCP プロトコルの仕様によって TCP ポートは TCP コネクション終了後の一定時間（1 分～4 分の間），新しい TCP コネクションで使用できない状態（TIME_WAIT 状態）になることがあります。TIME_WAIT 状態のポートは終了した TCP コネクションで使用されます。

0：TIME_WAIT 状態の TCP ポートの数を削減しません。

1：TIME_WAIT 状態の TCP ポートの数を削減します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDTCPCONOPT オペランドと同じです。PDTCPCONOPT オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

30)　PDKALVL 0 | 1 | 2

HiRDB External Data Access Adapter で使用する HiRDB クライアントから外部 HiRDB にパケットを送信するかどうかを指定します。

0：外部 HiRDB にパケットを送信しません。

1：外部 HiRDB との接続経路に定期的にパケットを送信します。

2：外部 HiRDB との接続経路に定期的にパケットを送信し，外部 HiRDB からの返信パケットを受信します。

このオペランドに 1 又は 2 を指定した場合，パケット送信スレッドを外部 HiRDB との接続ごとに一つ生成します。パケットの送信間隔は外部サーバ情報定義の PDKATIME オペランドで指定します。

《指定値の目安》

- ルータやファイアウォールなどのネットワーク管理アプリケーションでは，一定時間パケットが流れないと接続を切断することがあります。このため，外部 HiRDB に接続した状態でサービス

の要求を待機する Web アプリケーションなどは，待機状態が一定時間を超えると外部 HiRDB との接続を切断されます。このような場合に 1 又は 2 を指定してください。

- 外部 HiRDB の最大待ち時間を設定しない場合（外部サーバ情報定義の PDSWAITTIME オペランドの第 1 引数を省略して第 2 引数に 0 を指定した場合），HiRDB External Data Access Adapter で使用する HiRDB クライアントがあるサーバマシンの異常終了やネットワーク障害のときに，外部 HiRDB にサーバプロセスが居残ることがあります。このオペランドに 2 を指定すると，このようなサーバプロセスの居残りを防げます。

このオペランドの機能及び指定方法は，クライアント環境定義の PDKALVL オペランドと同じです。PDKALVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

31）　PDKATIME 外部 HiRDB へのパケット送信間隔

～<符号なし整数>((60～65535))《3000》（単位：秒）

HiRDB External Data Access Adapter で使用する HiRDB クライアントから外部 HiRDB にパケットを送信する間隔を秒単位で指定します。

《前提条件》

外部サーバ情報定義の PDKALVL オペランドに 1 又は 2 を指定する必要があります。

このオペランドの機能及び指定方法は，クライアント環境定義の PDKATIME オペランドと同じです。PDKATIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

32）　PDNODELAYACK YES｜NO

このオペランドは，外部 HiRDB が AIX 版のときに指定するオペランドです。

HiRDB サーバマシンと HiRDB クライアントマシン間の通信で，データを受信するときに，即時 ACK を適用するかどうかを指定します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDNODELAYACK オペランドと同じです。PDNODELAYACK オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

YES：即時 ACK を適用します。

NO：即時 ACK を適用しません。

10.2.9　システム監視に関するオペランド

33）　PDCWAITTIME〔HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 1〕〔，HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 2〕

このオペランドの機能は，クライアント環境定義の PDCWAITTIME オペランドと同じです。PDCWAITTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 1：

～<符号あり整数>((－65534～0))《0》（単位：秒）

クライアント環境定義の PDCWAITTIME オペランドの値に対して減らす秒数を指定します。HiRDB External Data Access Adapter で使用する HiRDB クライアントが，外部 HiRDB に処理要求をしてから応答があるまでの最大待ち時間がこのオペランドの指定で決まります。

《オペランドの指定方法》

最大待ち時間は次のようになります。

最大待ち時間＝クライアント環境定義の PDCWAITTIME オペランドの値＋このオペランドの値

例えば，クライアント環境定義の PDCWAITTIME オペランドの値が 100 で，このオペランドに－10 を指定した場合，最大待ち時間は 100－10 ＝ 90 秒になります。

《オペランドの規則》

- クライアント環境定義の PDCWAITTIME オペランドに 0 を指定した場合，このオペランドの指定は無効になります。
- クライアント環境定義の PDCWAITTIME オペランドとこのオペランドとの合計値が正になるようにこのオペランドを指定してください。
- クライアント環境定義の PDCWAITTIME オペランドの値とこのオペランドで指定した値の合計が 0 以下になる場合，最大待ち時間には 1 が仮定されます。

HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 2：

～<符号なし整数>((0～65535))《600》（単位：秒）

HiRDB External Data Access Adapter で使用する HiRDB クライアントが，外部 HiRDB に処理要求をしてから応答があるまでの最大待ち時間を秒単位で指定します。最大待ち時間 1 を省略して，最大待ち時間 2 に 0 を指定した場合，HiRDB External Data Access Adapter で使用する HiRDB クライアントは外部 HiRDB からの応答があるまで待ち続けます。

《オペランドの指定方法》

最大待ち時間 2 を指定する場合はコンマの指定を忘れないでください。指定形式を次に示します。

PDCWAITTIME ,1200

●最大待ち時間 1 と最大待ち時間 2 のどちらを指定するか

HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間を UAP ごとに設定する場合は，最大待ち時間 1 を指定してください。HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間を一定の値（すべての UAP が一定の値になります）にする場合は，最大待ち時間 2 を指定してください。

●最大待ち時間 1 と最大待ち時間 2 の優先度

- 最大待ち時間 1 と最大待ち時間 2 を同時に指定した場合は，最大待ち時間 1 の指定が適用されます。
- このオペランドを省略した場合は，最大待ち時間 1 の省略値（0）が仮定されます。
- 外部 HiRDB のバージョンが 06-01 以前の場合，最大待ち時間 1 の指定は無効になります。したがって，最大待ち時間 1 と最大待ち時間 2 を同時に指定した場合は，最大待ち時間 2 の指定が適用されます。このオペランドを省略した場合は，最大待ち時間 2 の省略値（600）が仮定されます。また，誤って最大待ち時間 1 だけを指定した場合は，最大待ち時間 2 の省略値（600）が仮定されます。

34）　PDSWAITTIME 〔外部 HiRDB の最大待ち時間 1〕〔，外部 HiRDB の最大待ち時間 2〕

このオペランドの機能は，クライアント環境定義の PDSWAITTIME オペランドと同じです。PDSWAITTIME オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

外部 HiRDB の最大待ち時間 1：

～<符号なし整数>((0～65534))《0》（単位：秒）

クライアント環境定義の PDSWAITTIME オペランドで指定した値に対して増やす秒数を指定します。HiRDB External Data Access Adapter で使用する HiRDB クライアントに，外部 HiRDB が応答を返してから次に要求があるまでの最大待ち時間がこのオペランドの指定で決まります。

《オペランドの指定方法》

最大待ち時間は次のようになります。

最大待ち時間＝クライアント環境定義の PDSWAITTIME オペランドの値＋このオペランドの値

例えば，クライアント環境定義の PDSWAITTIME オペランドの値が 100 で，このオペランドに 10 を指定した場合，最大待ち時間は 100 ＋ 10 ＝ 110 秒になります。

《オペランドの規則》

- クライアント環境定義の PDSWAITTIME オペランドに 0 を指定した場合，このオペランドの指定は無効になります。
- クライアント環境定義の PDSWAITTIME オペランドの値とこのオペランドの値の合計値が 65535 を超える場合は，外部 HiRDB の最大待ち時間に 65535 が仮定されます。

外部 HiRDB の最大待ち時間 2：

～<符号なし整数>((0～65535))《600》（単位：秒）

HiRDB External Data Access Adapter で使用する HiRDB クライアントに，外部 HiRDB が応答を返してから次に要求があるまでの最大待ち時間を秒単位で指定します。最大待ち時間 1 を省略して，最大待ち時間 2 に 0 を指定した場合，外部 HiRDB は HiRDB External Data Access Adapter で使用する HiRDB クライアントからの応答があるまで待ち続けます。

《オペランドの指定方法》

最大待ち時間 2 を指定する場合はコンマの指定を忘れないでください。指定形式を次に示します。

PDSWAITTIME ,1200

●最大待ち時間 1 と最大待ち時間 2 のどちらを指定するか

外部 HiRDB の最大待ち時間を UAP ごとに設定する場合は，最大待ち時間 1 を指定してください。外部 HiRDB の最大待ち時間を一定の値（すべての UAP が一定の値になります）にする場合は，最大待ち時間 2 を指定してください。

●最大待ち時間 1 と最大待ち時間 2 の優先度

- 最大待ち時間 1 と最大待ち時間 2 を同時に指定した場合は，最大待ち時間 1 の指定が適用されます。
- このオペランドを省略した場合は，最大待ち時間 1 の省略値（0）が仮定されます。
- 外部 HiRDB のバージョンが 06-01 以前の場合，最大待ち時間 1 の指定は無効になります。したがって，最大待ち時間 1 と最大待ち時間 2 を同時に指定した場合は，最大待ち時間 2 の指定が適用されます。このオペランドを省略した場合は，最大待ち時間 2 の省略値（600）が仮定されます。また，誤って最大待ち時間 1 だけを指定した場合は，最大待ち時間 2 の省略値（600）が仮定されます。

10.2.10　トラブルシュート情報に関するオペランド

35）　PDCLTPATH SQL トレースファイル及びエラーログファイルの出力先ディレクトリ

～〈パス名〉《%PDDIR%¥spool¥hubspool¥外部サーバ名¥》

SQL トレースファイル及びエラーログファイルの出力先ディレクトリを指定します。SQL トレースファイル及びエラーログファイルについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

36)　PDSQLTRACE SQL トレースファイルのサイズ

～<符号なし整数>((0，4096～2000000000))（単位：バイト）

SQL トレースファイルのサイズをバイト単位で指定します。SQL トレースファイルは PDCLTPATH オペランドで指定したディレクトリに出力されます。PDCLTPATH オペランドを省略した場合は，%PDDIR%¥spool¥hubspool¥外部サーバ名¥下に出力されます。

《オペランドの規則》

- 0 を指定した場合はファイルの最大のサイズとなります。
- このオペランドを省略した場合は SQL トレースを出力しません。

このオペランドの機能及び指定方法は，クライアント環境定義の PDSQLTRACE オペランドと同じです。PDSQLTRACE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

37)　PDSQLTEXTSIZE SQL トレースファイルに出力する SQL 文のサイズ

～<符号なし整数>((4096～2000000))《4096》（単位：バイト）

SQL トレースファイルに出力する SQL 文のサイズをバイト単位で指定します。

なお，アクセスパス取得時に省略した場合は 2000000 が仮定されます。

38)　PDSQLEXECTIME YES | NO

SQL トレースに SQL 実行時間を出力するかどうかを指定します。

YES：SQL トレースに SQL 実行時間を出力します。

NO：SQL トレースに SQL 実行時間を出力しません。

《注意事項》

- このオペランドに YES を指定した場合，UAP 統計レポートに出力される SQL 実行時間の単位はマイクロ秒になります。
- SQL トレースに出力される値は，実行時間が 24 時間を超えると正常に出力されません。

39)　PDUAPERLOG エラーログファイルのサイズ

～<符号なし整数>((0，4096～65536))《4096》（単位：バイト）

エラーログファイルのサイズをバイト単位で指定します。エラーログファイルは PDCLTPATH オペランドで指定したディレクトリに出力されます。PDCLTPATH オペランドを省略した場合は，%PDDIR%¥spool¥hubspool¥外部サーバ名¥下に出力されます。

《オペランドの規則》

0 を指定した場合はファイルの最大のサイズとなります。

このオペランドの機能及び指定方法は，クライアント環境定義の PDUAPERLOG オペランドと同じです。PDUAPERLOG オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

40)　PDPRMTRC YES | NO

SQL トレースに？パラメタ情報を出力するかどうかを指定します。

YES：SQL トレースに？パラメタ情報を出力します。

NO：SQL トレースに？パラメタ情報を出力しません。

41)　PDPRMTRCSIZE SQL トレース中の？パラメタ情報の最大データ長

～<符号なし整数>((4～4096))《256》（単位：バイト）

SQL トレースに出力する？パラメタ情報の最大データ長を指定します。

《前提条件》

PDPRMTRC オペランドに YES を指定する必要があります。

42)　PDTRCMODE ERR｜NONE

SQL トレース以外のトラブルシュート情報（ファイル名が pderrXXXX.trc の情報）を出力するかどうかを指定します。

ERR：トラブルシュート情報を出力します。

NONE：トラブルシュート情報を出力しません。

トラブルシュート情報は PDCLTPATH オペランドで指定したディレクトリ下に出力されます。PDCLTPATH オペランドを省略した場合は，%PDDIR%¥spool¥hubspool¥外部サーバ名¥下に出力されます。

43)　PDTRCPATH 動的 SQL トレースファイルの出力先ディレクトリ

～<パス名>

動的 SQL トレースファイルの出力先ディレクトリを指定します。トレース取得コマンド（pdtrcmgr）で動的 SQL トレースファイルを取得する場合は，このオペランドを必ず指定してください。

また，pdtrcmgr にこのオペランドで指定したディレクトリを指定すると，次回の CONNECT 時からこのオペランドで指定したディレクトリに SQL トレースファイルを出力します。

トレース取得コマンド（pdtrcmgr）については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

44)　PDUAPREPLVL 〔s〕〔u〕〔p〕〔r〕｜a

UAP 統計レポートの出力情報種別を指定します。UAP 統計レポート機能については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

s：SQL 単位の情報を出力します。また，SQL トレースも出力します。

u：UAP 単位の情報を出力します。

p：アクセスパス情報を出力します。

r：SQL 実行時の中間情報を出力します。

a：すべての情報を出力します。

《前提条件》

PDCLTPATH オペランドを指定する必要があります。

《オペランドの規則》

- このオペランドを省略すると，SQL トレースだけを出力します。
- s，u，p，及び r は，1 文字以上の任意の組み合わせで指定できます。supr と指定した場合は，a と同じ意味になります。s，及び a 以外を指定した場合，SQL トレースは出力されません。
- PDUAPREPLVL だけを指定して s，u，p，r，a を指定しないとエラーになります。

このオペランドの機能及び指定方法は，クライアント環境定義の PDUAPREPLVL オペランドと同じです。PDUAPREPLVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

45)　PDREPPATH UAP 統計レポートの出力先ディレクトリ

～<パス名>

通常，UAP 統計レポートは PDCLTPATH オペランドに指定したディレクトリ下に出力されます。これとは異なるディレクトリ下に UAP 統計レポートを出力する場合にこのオペランドを指定します。

CONNECT/DISCONNECT 単位にファイルが作成されて，ファイル名は pdHHMMSSmmm_xxx_＊(1or2).trc になります。

HHMMSSmmm：CONNECT した時間

xxx：コネクト通番

このオペランドの機能及び指定方法は，クライアント環境定義の PDREPPATH オペランドと同じです。PDREPPATH オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

46)　PDSQLTRCOPENMODE CNCT | SQL

SQL トレースを取得する場合の SQL トレースファイルのオープンモードを指定します。

CNCT：

CONNECT/DISCONNECT 単位に SQL トレースファイルをオープン又はクローズして SQL トレースを出力します。SQL を指定するよりオーバヘッドが削減されるため，SQL トレースの出力時間が短縮されます。ただし，SQL トレースファイルをオープンしたまま情報を書き込むため，正常に DISCONNECT しない場合は，SQL トレースが正しく出力されないことがあります。

SQL：

SQL（オペレーション）単位に SQL トレースファイルをオープン又はクローズして SQL トレースを出力します。

《前提条件》

PDREPPATH オペランドを指定する必要があります。

47)　PDVWOPTMODE アクセスパス情報の取得モード

～<符号なし整数>((0～2))《0》

アクセスパス情報を取得するかどうかを指定します。アクセスパス表示ユティリティ実行時にここで取得したアクセスパス情報を使用します。外部 HiRDB（シングルサーバ又はフロントエンドサーバがあるユニット）の%PDDIR%¥spool¥pdsqldump 下にアクセスパス情報が出力されます。

0：

アクセスパス情報を取得しません。

1：

アクセスパス情報を取得します。ただし，SQL オブジェクトがバッファ中にある SQL の情報は出力しません。

2：

アクセスパス情報を取得します。SQL オブジェクトがバッファ中にある SQL についても SQL オブジェクトを再作成して，SQL の情報を出力します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDVWOPTMODE オペランドと同じです。PDVWOPTMODE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

10.2.11　排他制御に関するオペランド

48）　PDLOCKLIMIT 外部 HiRDB に対する排他要求の上限値

～＜符号なし整数＞((0～200000000))《0》

1 外部 HiRDB に対する排他要求の上限値を指定します。0 を指定するか，このオペランドを省略した場合は，排他要求の上限値をチェックしません。

《見積もり方法》

排他資源数は SQL によって異なります。排他制御による排他資源数を概算し，その値からこの指定値を算出してください。排他資源数の見積もりについては，「付録 D　排他資源数の見積もり」を参照してください。

また，排他制御については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

10.2.12　インナレプリカ機能に関するオペランド

49）　PDDBACCS 外部 HiRDB の RD エリア世代番号

～＜符号なし整数＞((0～10))

外部 HiRDB がインナレプリカ機能を使用している場合にこのオペランドを指定します。インナレプリカグループ内でカレント RD エリアでない RD エリアをアクセスする場合に，その RD エリアの世代番号を指定します。0 はオリジナル RD エリアとなります。

このオペランドを省略した場合，外部 HiRDB が UAP 環境定義 PDDBACCS をサポートしていれば，外部の HiRDB の UAP 環境定義 PDDBACCS が仮定されます。外部の HiRDB が UAP 環境定義 PDDBACCS をサポートしていなければ，カレント RD エリアをアクセスします。PDDBACCS については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

50）　PDDBORGUAP YES ｜ NO

外部 HiRDB の，オンライン再編成閉塞状態のオリジナル RD エリアに対する UAP の実行について指定します。

YES：

外部 HiRDB のオリジナル RD エリアに対して，UAP によるデータベース操作（バッチ処理）が実行できます。

NO：

外部 HiRDB のオリジナル RD エリアに対して，UAP によるデータベース操作が制限されます。

10.2.13　SQL の最適化に関するオペランド

51）　PDSQLOPTLVL 外部 HiRDB の SQL 最適化オプション〔，外部 HiRDB の SQL 最適化オプション〕…

～＜識別子又は符号なし整数＞

外部 HiRDB の SQL 最適化オプションを指定します。SQL の最適化については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

このオペランドを省略すると，外部 HiRDB で設定している SQL 最適化オプションが適用されます。外部 HiRDB の SQL 最適化オプションを適用しない場合にこのオペランドを指定します。

《オペランドの規則》

このオペランドに指定できる文字列の合計長は 575 バイトまでです。

このオペランドの機能及び指定方法は，クライアント環境定義の PDSQLOPTLVL オペランドと同じです。PDSQLOPTLVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

52）　PDADDITIONALOPTLVL 外部 HiRDB の SQL 拡張最適化オプション〔，外部 HiRDB の SQL 拡張最適化オプション〕…

～＜識別子又は符号なし整数＞

外部 HiRDB の SQL 拡張最適化オプションを指定します。SQL の拡張最適化については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

このオペランドを省略すると，外部 HiRDB で設定している SQL 拡張最適化オプションが適用されます。外部 HiRDB の SQL 拡張最適化オプションを適用しない場合にこのオペランドを指定します。

《オペランドの規則》

このオペランドに指定できる文字列の合計長は 575 バイトまでです。

このオペランドの機能及び指定方法は，クライアント環境定義の PDADDITIONALOPTLVL オペランドと同じです。PDADDITIONALOPTLVL オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

53）　PDAGGR 外部 HiRDB のグループ化処理時に発生するグループ数

32 ビットモードの場合：～＜符号なし整数＞((0～30000000))《1024》

64 ビットモードの場合：～＜符号なし整数＞((0～2147483647))《1024》

外部 HiRDB の GROUP BY 処理に使用するメモリ量を決定するため，サーバごとに発生するグループ数の最大値を指定します。このオペランドは，外部 HiRDB で SQL 最適化オプションの「グループ分け高速化処理」を実行する場合に有効になります。

このオペランドの機能及び指定方法は，クライアント環境定義の PDAGGR オペランドと同じです。PDAGGR オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

54）　PDHASHTBLSIZE 外部 HiRDB のハッシュジョイン，副問合せのハッシュ実行適用時のハッシュ表サイズ

32 ビットモードの場合：～＜符号なし整数＞((128～524288))（単位：キロバイト）

64 ビットモードの場合：～＜符号なし整数＞((128～2097152))（単位：キロバイト）

外部 HiRDB で SQL 拡張最適化オプションの「ハッシュジョイン，副問合せのハッシュ実行」を実行する場合，ハッシュ表のサイズをキロバイト単位で指定します。このオペランドを省略すると，外部 HiRDB の pd_hash_table_size オペランドの値が仮定されます。

このオペランドの機能及び指定方法は，クライアント環境定義の PDHASHTBLSIZE オペランドと同じです。PDHASHTBLSIZE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

10.2.14　拡張 SQL エラー情報出力機能に関するオペランド

55)　PDUAPEXERLOGUSE YES｜NO

外部 HiRDB で拡張 SQL エラー情報出力機能を使用するかどうかを指定します。

YES：拡張 SQL エラー情報出力機能を使用します。

NO ：拡張 SQL エラー情報出力機能を使用しません。

このオペランドの機能及び指定方法は，クライアント環境定義の PDUAPEXERLOGUSE オペランドと同じです。PDUAPEXERLOGUSE オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

56)　PDUAPEXERLOGPRMSZ エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長

～<符号なし整数>((0～32008))《0》（単位：バイト）

外部 HiRDB で拡張 SQL エラー情報出力機能を使用する場合に，エラーログファイルに出力するパラメタ情報の最大データ長を指定します。

- 1 以上を指定した場合

  エラーログファイル及び SQL エラーレポートファイルにパラメタ情報を出力します。

- 0 を指定した場合

  エラーログファイル及び SQL エラーレポートファイルにパラメタ情報を出力しません。

可変長文字列型，BLOB 型及び BINARY 型の場合，データ長の領域もこの指定値に含まれます。エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報のデータ長が PDUAPEXERLOGPRMSZ を超えた場合，PDUAPEXERLOGPRMSZ までのパラメタ情報を出力します。この場合，超過分は切り捨てます。

このオペランドの機能及び指定方法は，クライアント環境定義の PDUAPEXERLOGPRMSZ オペランドと同じです。PDUAPEXERLOGPRMSZ オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

10.2.15　バックエンドサーバ接続保持機能に関するオペランド

バックエンドサーバ接続保持機能については，マニュアル「HiRDB Version 8 システム運用ガイド」を参照してください。

57)　PDBESCONHOLD YES ｜ NO

外部 HiRDB でバックエンドサーバ接続保持機能を使用するかどうかを指定します。なお，HiRDB/シングルサーバに接続している場合，このオペランドの指定は無視されます。

YES：バックエンドサーバ接続保持機能を使用します。

NO：バックエンドサーバ接続保持機能を使用しません。

このオペランドを省略すると，外部 HiRDB のバックエンドサーバ定義の pd_bes_connection_hold オペランドの値が適用されます。

このオペランドの機能及び指定方法は，クライアント環境定義の PDBESCONHOLD オペランドと同じです。PDBESCONHOLD オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

58)　PDBESCONHTI バックエンドサーバ接続保持時間

～<符号なし整数>((0～3600))（単位：秒）

外部 HiRDB でバックエンドサーバ接続保持機能を使用する場合，バックエンドサーバ接続保持時間を秒単位で指定します。

バックエンドサーバ接続保持機能を使用すると，HiRDB はトランザクションの終了後から次のトランザクションが実行されるまでの時間を監視します。次のトランザクションが実行されるまでの時間が，指定値の範囲内の場合はフロントエンドサーバとバックエンドサーバの接続を保持します。指定値を超えた場合は，トランザクション終了後にフロントエンドサーバとバックエンドサーバの接続を切断します。

なお，このオペランドに 0 を指定した場合は時間を監視しません。SQL DISCONNECT（XA ライブラリを使用時は xa_close），PDCWAITTIME オーバなどでフロントエンドサーバとクライアント間の接続が切断されたときだけ，フロントエンドサーバとバックエンドサーバの接続を切断します。

このオペランドの機能及び指定方法は，クライアント環境定義の PDBESCONHTI オペランドと同じです。PDBESCONHTI オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

10.2.16　系切り替え機能に関するオペランド

59）PDHATRNQUEUING NO

外部の HiRDB システムでトランザクションキューイング機能を使用する場合に，トランザクションキューイング機能をキャンセルするかどうかを指定します。このオペランドの機能及び指定方法は，クライアント環境定義の PDHATRNQUEUING オペランドと同じです。PDHATRNQUEUING オペランドについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。外部サーバの HiRDB のバージョンが異なる場合は，対応するバージョンのマニュアルを参照してください。

10.3　注意事項

(1)　外部 HiRDB に適用されるクライアント環境定義

次に示すクライアント環境定義の値は外部 HiRDB に適用されます。

- PDISLLVL

外部 HiRDB のバージョンが 06-02 以降の場合, 次に示すクライアント環境定義の値は外部 HiRDB に適用されます。

- PDCLTAPNAME
- PDEXWARN
- PDDLKPRIO
- PDLOCKSKIP
- PDCWAITTIMEWRNPNT

(2)　外部 HiRDB に対して固定値が適用されるクライアント環境定義

次に示すクライアント環境定義の値は，外部 HiRDB に対して固定値が適用されます。

- PDSTJTRNOUT：NO 固定
- PDDFLNVAL：NOUSE 固定
- PDSWATCHTIME：0 固定
- PDAUTOCONNECT：OFF 固定

(3)　外部 HiRDB のバージョンが自 HiRDB のバージョンより古い場合

外部 HiRDB のバージョンが自 HiRDB のバージョンより古い場合，外部サーバ情報定義で指定したオペランドが外部 HiRDB で適用されないことがあります。適用されるかどうかは，指定したオペランドが外部 HiRDB でサポートされているかどうかによります。

11 外部サーバ情報定義（外部サーバがXDM/RD E2の場合）

外部サーバがXDM/RD E2の場合の外部サーバ情報定義について説明します。この定義はHiRDB External Data Access機能使用時に定義します。

11.1　オペランドの形式

外部サーバ情報定義では，HiRDB External Data Access 機能の実行環境を定義します。なお，表中の「番号」は，「11.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●外部サーバ情報定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB External Data Access 機能を使用できます。

●再開始前に変更できるオペランド

外部サーバ情報定義の全オペランドが HiRDB の再開始前（計画停止，強制終了，又は異常終了後）に変更できます。

(1)　外部サーバとの接続及び解除

番号	形式
1	〔set pd_hb_db_con = STARTUP｜ACCESS〕
2	〔set pd_hb_db_dis_con = SHUTDOWN｜COMMIT〕

(2)　外部サーバエラー情報

番号	形式
3	〔set pd_hb_e_mode = ALL ｜ NONE〕
4	〔set pd_hb_e_code = SQLCODE〔，SQLCODE〕…〕
5	〔set pd_hb_e_size = 外部サーバエラー情報ファイルのサイズ〕

(3)　外部サーバインタフェーストレース情報

番号	形式
6	〔set pd_hb_l_path = 外部サーバインタフェーストレース情報ファイルの出力先ディレクトリ〕
7	〔set pd_hb_l_size = 外部サーバインタフェーストレース情報ファイルのサイズ〕
8	〔set pd_hb_l_prm = Y｜N〕
9	〔set pd_hb_l_prm_size = 外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長〕
10	〔set pd_hb_l_mode = ALL｜PARTIAL〕

(4)　外部サーバとの接続

番号	形式
11	putenv PDCLTRDNODE XDM/RD E2 のデータベース識別子
12	putenv PDHOST DB コネクションサーバのホスト名〔，予備系のホスト名〕
13	putenv PDNAMEPORT DB コネクションサーバのポート番号

(5)　システム監視

番号	形式
14	〔putenv PDCWAITTIME 〔HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 1〕〔，HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 2〕〕

(6)　トラブルシュート情報

番号	形式
15	〔putenv PDCLTPATH SQL トレースファイル及びエラーログファイルの出力先ディレクトリ〕
16	〔putenv PDSQLTRACE SQL トレースファイルのサイズ〕
17	〔putenv PDUAPERLOG エラーログファイルのサイズ〕
18	〔putenv PDPRMTRC YES｜NO 〕
19	〔putenv PDPRMTRCSIZE SQL トレース中の？パラメタ情報の最大データ長〕

(7)　データ転送

番号	形式
20	〔putenv PDBLKF XDM/RD E2 からの検索結果を転送するときのブロック転送の行数〕

11.2　オペランドの説明

11.2.1　外部サーバとの接続及び解除に関するオペランド

1）　pd_hb_db_con = STARTUP｜ACCESS

外部サーバに接続するタイミングを指定します。

STARTUP：

バックエンドサーバプロセスの起動時に外部サーバに接続します。

STARTUP を指定すると, 外部サーバへの処理が実行されないときも外部サーバに接続しています。外部サーバの接続数に限りがある場合は ACCESS を指定してください。

ACCESS：

バックエンドサーバプロセスから外部サーバへのアクセス要求が初めてあったときに外部サーバに接続します。

2）　pd_hb_db_dis_con = SHUTDOWN｜COMMIT

外部サーバとの接続を解除するタイミングを指定します。

SHUTDOWN：

バックエンドサーバプロセスの終了時に接続を解除します。

SHUTDOWN を指定すると，外部サーバへの処理が実行されないときも外部サーバに接続しています。外部サーバの接続数に限りがある場合は COMMIT を指定してください。

COMMIT：

外部サーバに対してコミットを発行した直後に接続を解除します。

11.2.2　外部サーバエラー情報に関するオペランド

3）　pd_hb_e_mode = ALL ｜ NONE

外部サーバエラー情報を出力するかどうかを指定します。外部サーバエラー情報については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

ALL：外部サーバエラー情報を出力します。

NONE：外部サーバエラー情報を出力しません。

4）　pd_hb_e_code = SQLCODE〔，SQLCODE〕…

～<符号あり整数>((- 214748364～ - 1))

このオペランドには，外部サーバの SQLCODE（SQL 連絡領域 SQLCA の SQLCODE に返される値）を指定します。ここで指定した SQLCODE に対応するエラーメッセージを外部サーバエラー情報に出力しません。

このオペランドを省略すると，すべての SQLCODE に対応するエラーメッセージを外部サーバエラー情報ファイルに出力します。SQLCODE については，各 DBMS のマニュアルを参照してください。

《オペランドの規則》

- 複数の SQLCODE を指定する場合は，SQLCODE ごとにコンマ（, ）で区切ってください。
- SQLCODE は 10 個まで指定できます。

5）　pd_hb_e_size = 外部サーバエラー情報ファイルのサイズ

～<符号なし整数>((0，4～4096))《32》（単位：キロバイト）

外部サーバエラー情報ファイルのサイズをキロバイト単位で指定します。サイズの上限を決めない場合は 0 を指定します。

11.2.3　外部サーバインタフェーストレース情報に関するオペランド

6）　pd_hb_l_path = 外部サーバインタフェーストレース情報ファイルの出力先ディレクトリ

～<パス名>《%PDDIR%¥spool¥》

外部サーバインタフェーストレース情報ファイルの出力先ディレクトリを指定します。このディレクトリ下に**外部サーバインタフェーストレース情報**を出力します。出力ファイルはこのディレクトリ下に二つ作成されます。ファイル名は pdhl 外部サーバ名 1，又は pdhl 外部サーバ名 2 です。

外部サーバインタフェーストレース情報については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

7）　pd_hb_l_size = 外部サーバインタフェーストレース情報ファイルのサイズ

～<符号なし整数>((0，4～4096))《1024》（単位：キロバイト）

外部サーバインタフェーストレース情報ファイルのサイズをキロバイト単位で指定します。サイズの上限を決めない場合は 0 を指定します。

ファイル容量がこのオペランドに指定した値を超えると，ファイルをスワップしてもう一方のファイルに出力します。

《指定値の目安》

1 トレース当たり平均 300 バイトの情報が出力されます（？パラメタ情報を除く）。

《注意事項》

- クライアント環境定義の PDSQLTRACE オペランドを指定して実行した UAP に対して，外部サーバ単位に外部サーバインタフェーストレース情報を取得します。複数のトランザクションが一つのファイルに書き込みをするため，書き込み時に排他を掛けます。したがって，外部サーバインタフェーストレース情報を取得すると，トランザクションの処理性能及び同時実行性が低下します。
- 外部サーバのアクセス状況によって，ファイル容量がこのオペランドに指定した値を大幅に超えてしまうことがあります。したがって，十分な空きがあるディスクに外部サーバインタフェーストレース情報を取得してください。

8）　pd_hb_l_prm = Y | N

外部サーバインタフェーストレース情報に，？パラメタ情報を出力するかどうかを指定します。

Y：？パラメタ情報を出力します。

N：？パラメタ情報を出力しません。

9）　pd_hb_l_prm_size = 外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長

～<符号なし整数>((4～4096))《256》（単位：バイト）

外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長をバイト単位で指定します。

《前提条件》

pd_hb_l_prm オペランドに Y を指定している必要があります。

10）　pd_hb_l_mode = ALL | PARTIAL

外部サーバインタフェーストレース情報のすべての情報を出力するかどうかを指定します。

ALL：すべての情報を出力します。

PARTIAL：外部サーバと通信を行う関数についての情報だけを出力します。

11.2.4　外部サーバとの接続に関するオペランド

11）　PDCLTRDNODE XDM/RD E2 のデータベース識別子

～<識別子>

このオペランドは省略できません。必ず指定してください。

接続する XDM/RD E2 のデータベース識別子を指定します。データベース識別子とは，XDM のサブシステム定義で指定する RD ノード名称のことです。

12）　PDHOST DB コネクションサーバのホスト名〔，予備系のホスト名〕

～<ホスト名>

このオペランドは省略できません。必ず指定してください。

接続する DB コネクションサーバのホスト名を指定します。ホスト名は IP アドレスでも指定できます。IP アドレスは，バイトごとにピリオド（.）で区切られた 10 進数で，次の形式で指定します。

IP アドレスの形式：NNN.OOO.PPP.QQQ

通常は先に指定したホスト名に接続します。先に指定したホスト名に接続できない場合に，予備系のホスト名に接続します。

13）　PDNAMEPORT DB コネクションサーバのポート番号

～<符号なし整数>((5001～65535))

このオペランドは省略できません。必ず指定してください。

接続する DB コネクションサーバのポート番号を指定します。

ポート番号には，DB コネクションサーバのコントロール空間起動制御文及びサーバ空間起動制御文の SCHEDULENO オペランドで指定するサーバスケジュール番号を指定します。SCHEDULENO オペランドを省略した場合は，サーバスケジュール番号（ポート番号）に 40179 が仮定されます。

11.2.5　システム監視に関するオペランド

14）　PDCWAITTIME〔HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 1〕〔，HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 2〕

このオペランドの機能は，クライアント環境定義の PDCWAITTIME オペランドと同じです。PDCWAITTIME オペランドについては，マニュアル「HiRDB Version 8 XDM/RD E2 接続機能」を参照してください。

このオペランドに最大待ち時間 1 を指定しても無効になります。必ず最大待ち時間 2 だけを指定してください。このオペランドを省略した場合は，最大待ち時間 2 の省略値（600）が仮定されます。

HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 1：

～〈符号あり整数〉((－65534～0))《0》（単位：秒）

クライアント環境定義の PDCWAITTIME オペランドの値に対して減らす秒数を指定します。HiRDB External Data Access Adapter で使用する HiRDB クライアントが，外部サーバに処理要求をしてから応答があるまでの最大待ち時間がこのオペランドの指定で決まります。

《オペランドの指定方法》

最大待ち時間は次のようになります。

最大待ち時間＝クライアント環境定義の PDCWAITTIME オペランドの値＋このオペランドの値

例えば，クライアント環境定義の PDCWAITTIME オペランドの値が 100 で，このオペランドに－10 を指定した場合，最大待ち時間は 100－10 ＝ 90 秒になります。

《オペランドの規則》

- クライアント環境定義の PDCWAITTIME オペランドに 0 を指定した場合，このオペランドの指定は無効になります。
- クライアント環境定義の PDCWAITTIME オペランドとこのオペランドとの合計値が正になるようにこのオペランドを指定してください。
- クライアント環境定義の PDCWAITTIME オペランドの値とこのオペランドで指定した値の合計が 0 以下になる場合，最大待ち時間には 1 が仮定されます。

HiRDB External Data Access Adapter で使用する HiRDB クライアントの最大待ち時間 2：

～<符号なし整数>((0～65535))《600》（単位：秒）

HiRDB External Data Access Adapter で使用する HiRDB クライアントが，外部サーバ（XDM/RD E2）に処理要求をしてから応答があるまでの最大待ち時間を秒単位で指定します。最大待ち時間 2 に 0 を指定した場合, HiRDB External Data Access Adapter で使用する HiRDB クライアントは外部サーバからの応答があるまで待ち続けます。

《オペランドの指定方法》

最大待ち時間 2 を指定する場合はコンマの指定を忘れないでください。指定形式を次に示します。

PDCWAITTIME ,1200

11.2.6　トラブルシュート情報に関するオペランド

15）　PDCLTPATH SQL トレースファイル及びエラーログファイルの出力先ディレクトリ

～〈パス名〉《%PDDIR%¥spool¥hubspool¥外部サーバ名¥》

SQL トレースファイル及びエラーログファイルの出力先ディレクトリを指定します。SQL トレースファイル及びエラーログファイルについては，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。

16）　PDSQLTRACE SQL トレースファイルのサイズ

～<符号なし整数>((0，4096～2000000000))（単位：バイト）

SQL トレースファイルのサイズをバイト単位で指定します。SQL トレースファイルは PDCLTPATH オペランドで指定したディレクトリに出力されます。PDCLTPATH オペランドを省略した場合は，%PDDIR%¥spool¥hubspool¥外部サーバ名¥下に出力されます。

SQL トレース機能については，マニュアル「HiRDB Version 8 XDM/RD E2 接続機能」を参照してください。

《オペランドの規則》

- 0 を指定した場合はファイルの最大のサイズとなります。
- 省略した場合は SQL トレースを出力しません。

このオペランドの機能は，クライアント環境定義の PDSQLTRACE オペランドと同じです。PDSQLTRACE オペランドについては，マニュアル「HiRDB Version 8 XDM/RD E2 接続機能」を参照してください。

17）　PDUAPERLOG エラーログファイルのサイズ

～<符号なし整数>((0，4096～65536))《4096》（単位：バイト）

エラーログファイルのサイズをバイト単位で指定します。エラーログファイルは PDCLTPATH オペランドで指定したディレクトリに出力されます。PDCLTPATH オペランドを省略した場合は，%PDDIR%¥spool¥hubspool¥外部サーバ名¥下に出力されます。

《オペランドの規則》

0 を指定した場合はファイルの最大のサイズとなります。

18）　PDPRMTRC YES｜NO

SQL トレースに？パラメタ情報を出力するかどうかを指定します。

YES：SQL トレースに？パラメタ情報を出力します。

NO：SQL トレースに？パラメタ情報を出力しません。

19）　PDPRMTRCSIZE SQL トレース中の？パラメタ情報の最大データ長

～<符号なし整数>((4～4096))《256》（単位：バイト）

SQL トレースに出力する？パラメタ情報の最大データ長を指定します。

《前提条件》

PDPRMTRC オペランドに YES を指定する必要があります。

11.2.7　データ転送に関するオペランド

20）　PDBLKF XDM/RD E2 からの検索結果を転送するときのブロック転送の行数

～<符号なし整数>((1～4096))《10》

XDM/RD E2 から DB アクセスアダプタに検索結果を転送するときの，一回の転送処理で送られる行数を指定します。

このオペランドの機能は，クライアント環境定義の PDBLKF オペランドと同じです。PDBLKF オペランドについては，マニュアル「HiRDB Version 8 XDM/RD E2 接続機能」を参照してください。

11.3　外部サーバ（XDM/RD E2）に対して固定値が適用されるクライアント環境定義

次に示すクライアント環境定義の値は，外部サーバ（XDM/RD E2）に対して固定値が適用されます。

- PDSRVTYP：VOS3 固定

12 外部サーバ情報定義（外部サーバがORACLEの場合）

外部サーバがORACLEの場合の外部サーバ情報定義について説明します。
この定義はHiRDB External Data Access機能使用時に定義します。

12.1　オペランドの形式

外部サーバ情報定義では，HiRDB External Data Access 機能の実行環境を定義します。なお，表中の「番号」は，「12.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。なお，外部サーバ情報定義のほかに，ORACLE のクライアントの定義が必要になります。詳細は ORACLE のマニュアルを参照してください。

●外部サーバ情報定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB External Data Access 機能を使用できます。

●再開始前に変更できるオペランド

外部サーバ情報定義の全オペランドが HiRDB の再開始前（計画停止，強制終了，又は異常終了後）に変更できます。

(1)　外部サーバとの接続及び解除

番号	形式
1	〔set pd_hb_db_con = STARTUP｜ACCESS〕
2	〔set pd_hb_db_dis_con = SHUTDOWN｜COMMIT〕

(2)　外部サーバエラー情報

番号	形式
3	〔set pd_hb_e_mode = ALL ｜ NONE〕
4	〔set pd_hb_e_code = SQLCODE〔，SQLCODE〕…〕
5	〔set pd_hb_e_size = 外部サーバエラー情報ファイルのサイズ〕

(3)　外部サーバインタフェーストレース情報

番号	形式
6	〔set pd_hb_l_path = 外部サーバインタフェーストレース情報ファイルの出力先ディレクトリ〕
7	〔set pd_hb_l_size = 外部サーバインタフェーストレース情報ファイルのサイズ〕
8	〔set pd_hb_l_prm = Y｜N〕
9	〔set pd_hb_l_prm_size = 外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長〕
10	〔set pd_hb_l_mode = ALL｜PARTIAL〕

(4)　FETCH

番号	形式
11	〔set pd_hb_ary_fec_num = 配列を使用した FETCH 機能で一度に取得する行数〕

(5)　トラブルシュート情報

番号	形式
12	〔set pd_hb_sql_trace = Y ｜ N 〕

(6)　排他制御

番号	形式
13	〔set pd_hb_get_lock = Y｜ N 〕

(7)　DESCRIBE 文

番号	形式
14	〔set pd_hb_use_describe = Y ｜ N 〕

(8)　外部サーバとの接続

番号	形式
15	putenv PDNETSERVICE 接続するデータベースのネット・サービス名

12.2　オペランドの説明

12.2.1　外部サーバとの接続及び解除に関するオペランド

1）　pd_hb_db_con = STARTUP｜ACCESS

外部サーバに接続するタイミングを指定します。

STARTUP：

バックエンドサーバプロセスの起動時に外部サーバに接続します。

STARTUP を指定すると, 外部サーバへの処理が実行されないときも外部サーバに接続しています。外部サーバの接続数に限りがある場合は ACCESS を指定してください。

ACCESS：

バックエンドサーバプロセスから外部サーバへのアクセス要求が初めてあったときに外部サーバに接続します。

2）　pd_hb_db_dis_con = SHUTDOWN｜COMMIT

外部サーバとの接続を解除するタイミングを指定します。

SHUTDOWN：

バックエンドサーバプロセスの終了時に接続を解除します。

SHUTDOWN を指定すると，外部サーバへの処理が実行されないときも外部サーバに接続しています。外部サーバの接続数に限りがある場合は COMMIT を指定してください。

COMMIT：

外部サーバに対してコミットを発行した直後に接続を解除します。

12.2.2　外部サーバエラー情報に関するオペランド

3）　pd_hb_e_mode = ALL ｜ NONE

外部サーバエラー情報を出力するかどうかを指定します。外部サーバエラー情報については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

ALL：外部サーバエラー情報を出力します。

NONE：外部サーバエラー情報を出力しません。

4）　pd_hb_e_code = SQLCODE〔，SQLCODE〕…

～<符号あり整数>((- 214748364～ - 1))

このオペランドには，外部サーバの SQLCODE（SQL 連絡領域 SQLCA の SQLCODE に返される値）を指定します。ここで指定した SQLCODE に対応するエラーメッセージを外部サーバエラー情報に出力しません。

このオペランドを省略すると，すべての SQLCODE に対応するエラーメッセージを外部サーバエラー情報ファイルに出力します。SQLCODE については，各 DBMS のマニュアルを参照してください。

《オペランドの規則》

- 複数の SQLCODE を指定する場合は，SQLCODE ごとにコンマ（, ）で区切ってください。
- SQLCODE は 10 個まで指定できます。

5）　pd_hb_e_size = 外部サーバエラー情報ファイルのサイズ

～<符号なし整数>((0，4～4096))《32》（単位：キロバイト）

外部サーバエラー情報ファイルのサイズをキロバイト単位で指定します。サイズの上限を決めない場合は 0 を指定します。

12.2.3　外部サーバインタフェーストレース情報に関するオペランド

6）　pd_hb_l_path = 外部サーバインタフェーストレース情報ファイルの出力先ディレクトリ

～<パス名>《%PDDIR%¥spool¥》

外部サーバインタフェーストレース情報ファイルの出力先ディレクトリを指定します。このディレクトリ下に**外部サーバインタフェーストレース情報**を出力します。出力ファイルはこのディレクトリ下に二つ作成されます。ファイル名は pdhl 外部サーバ名 1，又は pdhl 外部サーバ名 2 です。

外部サーバインタフェーストレース情報については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

7）　pd_hb_l_size = 外部サーバインタフェーストレース情報ファイルのサイズ

～<符号なし整数>((0，4～4096))《1024》（単位：キロバイト）

外部サーバインタフェーストレース情報ファイルのサイズをキロバイト単位で指定します。サイズの上限を決めない場合は 0 を指定します。

ファイル容量がこのオペランドに指定した値を超えると，ファイルをスワップしてもう一方のファイルに出力します。

《指定値の目安》

1 トレース当たり平均 300 バイトの情報が出力されます（？パラメタ情報を除く）。

《注意事項》

- クライアント環境定義の PDSQLTRACE オペランドを指定して実行した UAP に対して，外部サーバ単位に外部サーバインタフェーストレース情報を取得します。複数のトランザクションが一つのファイルに書き込みをするため，書き込み時に排他を掛けます。したがって，外部サーバインタフェーストレース情報を取得すると，トランザクションの処理性能及び同時実行性が低下します。
- 外部サーバのアクセス状況によって，ファイル容量がこのオペランドに指定した値を大幅に超えてしまうことがあります。したがって，十分な空きがあるディスクに外部サーバインタフェーストレース情報を取得してください。

8）　pd_hb_l_prm = Y | N

外部サーバインタフェーストレース情報に，？パラメタ情報を出力するかどうかを指定します。

Y：？パラメタ情報を出力します。

N：？パラメタ情報を出力しません。

9）　pd_hb_l_prm_size = 外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長

～<符号なし整数>((4～4096))《256》（単位：バイト）

外部サーバインタフェーストレース情報ファイルに出力する？パラメタ情報の最大データ長をバイト単位で指定します。

《前提条件》

pd_hb_l_prm オペランドに Y を指定している必要があります。

10）　pd_hb_l_mode = ALL | PARTIAL

外部サーバインタフェーストレース情報のすべての情報を出力するかどうかを指定します。

ALL：すべての情報を出力します。

PARTIAL：外部サーバと通信を行う関数についての情報だけを出力します。

12.2.4　FETCH に関するオペランド

11）　pd_hb_ary_fec_num = 配列を使用した FETCH 機能で一度に取得する行数

～<符号なし整数>((1～30000))

配列を使用した FETCH で一度に取得する行数を指定します。

外部表からの検索結果行数がある程度予測できる場合，このオペランドを指定してください。HiRDB と外部サーバ間の通信によるオーバヘッドの削減が期待できます。外部サーバへの配列 FETCH 数による外部表への SQL 実行時間及び FETCH 回数については，外部サーバの利用状況に関する統計情報で調べてください。外部サーバの利用状況に関する統計情報については，マニュアル「HiRDB Version 8 コマンドリファレンス」の「統計解析ユティリティ」を参照してください。

このオペランドを省略すると，HiRDB が処理を行うのに最適な値を HiRDB が自動計算します。

《指定値の目安》

- 指定値が小さ過ぎると, HiRDB と外部サーバ間の通信回数が多くなるため, その分のオーバヘッドが大きくなります。
- 指定値が大き過ぎると，メモリを不要に使用するため，メモリ資源を圧迫します。
- 外部表からの検索結果行数が業務によって変わる場合はこのオペランドを省略することをお勧めします。このオペランドを省略したときの利点は，外部表から取得する 1 行以上のデータが pd_sql_send_buff_size オペランドに指定した通信用バッファ長に収まるかどうかによって異なります。収まる場合は外部表から取得したデータを HiRDB は効率良く処理します。収まらない場合は外部表アクセス時のメモリ使用量を抑えます。

12.2.5　トラブルシュート情報に関するオペランド

12）　pd_hb_sql_trace = Y ｜ N

外部サーバ（ORACLE）に対して発行する SQL のトレース情報を出力するかどうかを指定します。出力先は ORACLE の初期化パラメタで指定します。ORACLE の初期化パラメタについては，ORACLE のマニュアルを参照してください。

Y：SQL のトレース情報を出力します。

N：SQL のトレース情報を出力しません。

12.2.6　排他制御に関するオペランド

13）　pd_hb_get_lock = Y| N

検索系 SQL で参照するデータに排他を掛けるかどうかを指定します。ORACLE では通常，検索系 SQL で排他を掛けません。このため，ORACLE のデータベースを参照してほかのデータベースを更新するトランザクションと，ほかのデータベースを参照して ORACLE のデータベースを更新するトランザクションを同時に実行すると，これらのトランザクションの連続性を保証しません。このような場合に，ORACLE から読み出したデータに排他を掛けてトランザクションの連続性を保証してください。

Y：

検索系 SQL で参照したデータに排他を掛けます。SELECT 文に FOR UPDATE 句を指定した場合に排他が掛かります。Y を指定すると，排他を掛けたデータはほかのトランザクションから参照できなくなるため，ORACLE での同時実行性能が低下します。

N：

検索系 SQL で参照したデータに排他を掛けません。

12.2.7　DESCRIBE 文に関するオペランド

14）　pd_hb_use_describe = Y | N

外部サーバ（ORACLE）からデータを受け取るときに，外部サーバに対して DESCRIBE 文の発行を許すかどうかを指定します。外部サーバに対して DESCRIBE 文の発行を許すと，外部サーバのデータ型をより多く外部表の列にマッピングできます。外部表を利用する場合のデータ型の対応関係については，マニュアル「HiRDB Version 8 SQL リファレンス」を参照してください。

Y：DESCRIBE 文の発行を許します。

N：DESCRIBE 文の発行を許しません。

《指定値の目安》

- HiRDB のデータ型へのマッピングに DESCRIBE 文が必要な場合に Y を指定します。
- このオペランドに Y を指定すると，SQL 実行ごとに DESCRIBE 文を内部的に発行するため，N を指定した場合に比べ性能が低下することがあります。

12.2.8　外部サーバとの接続に関するオペランド

15）　PDNETSERVICE 接続するデータベースのネット・サービス名

～<文字列>

このオペランドは省略できません。必ず指定してください。

接続する外部サーバ（ORACLE）のネット・サービス名を指定します。ORACLE の tnsnames.ora ファイルで指定したサーバ名を指定します。詳細については，ORACLE のマニュアルを参照してください。

13 Hub 最適化情報定義

Hub 最適化情報定義について説明します。この定義は HiRDB External Data Access 機能使用時に定義します。

13.1　オペランドの形式

Hub 最適化情報定義では，入力された問合せの各構文を外部サーバで実行するかどうかを定義します。この指定によって HiRDB と外部サーバとの機能差を吸収します。Hub 最適化情報定義の指定値を変更すると，互換性のない機能は結果不正になります。また，外部サーバで未サポートの構文は外部サーバでエラーになることがあります。そのため，「13.4　各外部サーバの推奨値」で説明する値をそのまま利用することをお勧めします。なお，表中の「番号」は，「13.2　オペランドの説明」で説明するオペランドに付けた番号と対応しています。

●再開始前に変更できるオペランド

Hub 最適化情報定義の全オペランドが HiRDB の再開始前（計画停止，強制終了，又は異常終了後）に変更できます。

番号	形式
1	[set pd_hub_opt_on_cnd = 1～2]
2	[set pd_hub_opt_joined_table = 1～3]
3	[set pd_hub_opt_set_func = 1～4]
4	[set pd_hub_opt_case = 1～2]
5	[set pd_hub_opt_like = 1～4]
6	[set pd_hub_opt_grouping = 1～4]
7	[set pd_hub_opt_data_len = 1～2147483647]
8	[set pd_hub_opt_abs = 1～3]
9	[set pd_hub_opt_date = 1～3]
10	[set pd_hub_opt_time = 1～2]
11	[set pd_hub_opt_digits = 1～2]
12	[set pd_hub_opt_length = 1～8]
13	[set pd_hub_opt_lower_upper_type = 1～8]
14	[set pd_hub_opt_mod_div_type = 1～3]
15	[set pd_hub_opt_substr = 1～3]
16	[set pd_hub_opt_num = 1～2]
17	[set pd_hub_opt_datetime = 1～2]
18	[set pd_hub_opt_datetime_op = 1～5]
19	[set pd_hub_opt_trailing_spc = 1～2]
20	[set pd_hub_opt_in_value_num = 1～30000]
21	[set pd_hub_opt_nullable = 1～2]
22	[set pd_hub_opt_use_zero_string = 1～2]
23	[set pd_hub_opt_nchar = 1～2]

番号	形式
24	〔set pd_hub_opt_nest_scalar = 0～65535〕
25	〔set pd_hub_opt_float = 1～2〕
26	〔set pd_hub_opt_table_num = 0～65535〕
27	〔set pd_hub_opt_time_24hour = 1～2〕

13.2　オペランドの説明

1）　pd_hub_opt_on_cnd = 1～2　～<符号なし整数>《2》

外結合（LEFT OUTER JOIN）を外部サーバで実行するかどうかを指定します。

1：ON 条件に関係なく外結合を外部サーバで実行します。

2：次に示す場合に外結合を外部サーバで実行します。

- ON 条件が「外表カラム＝内表カラム」の AND 連接の場合
- ON 条件が「外表カラム＝内表カラム」及び「内表カラム＝定数」の AND 連接の場合

2）　pd_hub_opt_joined_table = 1～3　～<符号なし整数>《2》

表の結合を外部サーバで実行するかどうかを指定します。

1：外結合，内結合，及び内表側が結合である結合を外部サーバで実行します。

2：内表側が結合である結合，及び外結合と（内）結合が混在した結合を外部サーバで実行しません。

3：2 の条件に加えて，外結合と WITH 句が混在した場合，それらを外部サーバで実行しません。

3）　pd_hub_opt_set_func = 1～4　～<符号なし整数>《2》

集合関数を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：DISTINCT 集合関数の引数が列指定ではない場合，この集合関数を外部サーバで実行しません。

4：ALL 集合関数の COUNT，DISTINCT 集合関数の AVG，及び 2 種類以上の DISTINCT 集合関数を一つの問合せ指定に記述した場合，これらの集合関数を外部サーバで実行しません。

4）　pd_hub_opt_case = 1～2　～<符号なし整数>《2》

CASE 式を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

COALESCE はスカラ関数 VALUE に変換するため，このオプションに 2 を指定しても外部サーバで実行されます。

5）　pd_hub_opt_like = 1～4　～<符号なし整数>《2》

LIKE 述語，NOT LIKE 述語，XLIKE 述語，NOT XLIKE 述語を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：LIKE 述語，NOT LIKE 述語だけを外部サーバで実行しません。

4：XLIKE 述語，NOT XLIKE 述語だけを外部サーバで実行しません。

《注意事項》

文字列型，可変長文字列型，混在文字列型，及び可変長混在文字列型のデータに対する LIKE 述語，NOT LIKE 述語，XLIKE 述語，NOT XLIKE 述語に全角の特殊文字を含む場合，又は各国文字列型，可変長各国文字列型のデータに対する LIKE 述語，NOT LIKE 述語，XLIKE 述語，NOT XLIKE 述語に半角の特殊文字を含む場合，これらの述語を外部サーバで実行しないでください。

6）　pd_hub_opt_grouping = 1～4　～<符号なし整数>《2》

GROUP 化処理を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：GROUP BY なしの HAVING 句を外部サーバで実行しません。

4：GROUP 化列が列指定ではない場合，GROUP 化処理を外部サーバで実行しません。

7）　pd_hub_opt_data_len = 1～2147483647　～<符号なし整数>《255》

固定長文字型，可変長文字型，固定長各国文字型，可変長各国文字型，固定長混在文字型，及び可変長混在文字型の値式の結果のデータ長がこのオペランドで指定した値（バイト数）を超える場合，値式の処理を HiRDB で実行します。

ただし，値式中以外に？パラメタ，埋込み変数，SQL パラメタ，SQL 変数，又は USER を指定した場合，？パラメタ，埋込み変数，SQL パラメタ，SQL 変数，又は USER の実長がこのオペランドの値を超えても，外部サーバで処理を実行するためエラーになることがあります。

8）　pd_hub_opt_abs = 1～3　～<符号なし整数>《2》

スカラ関数 ABS()を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：引数に日間隔型又は時間隔型を指定した場合，外部サーバで実行しません。

9）　pd_hub_opt_date = 1～3　～<符号なし整数>《2》

DATE 型列指定，スカラ関数 DATE()，及び CHAR()に日付型を指定した場合，外部サーバで実行するかどうかを指定します。ただし，次に示す場合は外部サーバで実行しません。

- 第一引数に文字列型又は時刻印型を指定し，第二引数に書式を指定した DATE()の場合

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：DATE()の引数に整数型を指定した場合，DATE()を外部サーバで実行しません。DATE 型列指定も外部サーバで実行しません。引数に日付型を指定した CHAR()は外部サーバで実行します。

10）　pd_hub_opt_time = 1～2　～<符号なし整数>《2》

スカラ関数 TIME()，及び CHAR()に時刻型を指定した場合，外部サーバで実行するかどうかを指定します。ただし，次に示す場合は外部サーバで実行しません。

- 第一引数に文字列型を指定し，第二引数に書式を指定した TIME()の場合

1：外部サーバで実行します。

2：外部サーバで実行しません。

11）　pd_hub_opt_digits = 1～2　～<符号なし整数>《2》

スカラ関数 DIGITS()を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

12）　pd_hub_opt_length = 1～8　～<符号なし整数>《2》

スカラ関数 LENGTH()を外部サーバで実行するかどうかを指定します。ただし，引数が時刻印型，又は BINARY 型の場合は外部サーバで実行しません。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：引数が混在文字列型又は可変長混在文字列型の場合は外部サーバで実行しません。

4：引数が文字列型以外（数，日付，時刻，日間隔，及び時間隔）の場合は外部サーバで実行しません。

5：引数が BLOB 型の場合は外部サーバで実行しません。

6：引数が混在文字列型，可変長混在文字列型，又は文字列型以外（数，日付，時刻，日間隔，及び時間隔）の場合は外部サーバで実行しません。

7：引数が文字列型以外（数，日付，時刻，日間隔，及び時間隔），又は BLOB 型の場合は外部サーバで実行しません。

8：引数が混在文字列型，可変長混在文字列型，又は BLOB 型の場合は外部サーバで実行しません。

《注意事項》

外部サーバでの定義長と，外部表での定義長が異なる列を含む値式に対する LENGTH を外部サーバで実行すると，外部サーバでの実行結果と HiRDB がデータを受け取って LENGTH を実行したときの結果が異なります。このような場合はこのオペランドに 2 を指定してください。

13）　pd_hub_opt_lower_upper_type = 1～8　～<符号なし整数>《2》

スカラ関数 LOWER 及び UPPER を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：引数が固定長文字列型及び可変長文字列型の場合は外部サーバで実行しません。

4：引数が固定長各国文字列型及び可変長各国文字列型の場合は外部サーバで実行しません。

5：引数が固定長混在文字列型及び可変長混在文字列型の場合は外部サーバで実行しません。

6：引数が固定長文字列型，可変長文字列型，固定長各国文字列型，及び可変長各国文字列型の場合は外部サーバで実行しません。

7：引数が固定長各国文字列型，可変長各国文字列型，固定長混在文字列型，及び可変長混在文字列型の場合は外部サーバで実行しません。

8：引数が固定長文字列型，可変長文字列型，固定長混在文字列型，及び可変長混在文字列型の場合は外部サーバで実行しません。

14）　pd_hub_opt_mod_div_type = 1～3　～<符号なし整数>《2》

スカラ関数 MOD()，集合関数 AVG，及び割り算を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：次に示す場合は外部サーバで実行しません。

- 引数が DECIMAL 型の MOD()の場合
- 引数が DECIMAL 型の集合関数 AVG の場合
- 演算結果が DECIMAL 型になる割り算の場合

15）　pd_hub_opt_substr = 1～3　～<符号なし整数>《2》

スカラ関数 SUBSTR()を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：引数が混在文字列型及び可変長混在文字列型の場合，外部サーバで実行しません。

16）　pd_hub_opt_num = 1～2　～<符号なし整数>《2》

数データ型へ型変換をするスカラ関数 DECIMAL，FLOAT，及び INTEGER を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

17）　pd_hub_opt_datetime = 1～2　～<符号なし整数>《2》

日付又は時刻を抽出するスカラ関数を外部サーバで実行するかどうかを指定します。対象は次に示す関数です。ただし，引数が時刻印型の場合は外部サーバで実行しません。

- DAY
- HOUR
- MINUTE
- MONTH
- SECOND
- YEAR

1：外部サーバで実行します。

2：外部サーバで実行しません。

18）　pd_hub_opt_datetime_op = 1～5　～<符号なし整数>《2》

日付演算及び時刻演算を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

3：ラベル付き間隔の値式にスカラ関数 ABS 又は MOD を含む場合は，外部サーバで実行しません。

4：3 の条件に加えて次に示す演算を外部サーバで実行しません。

- 日間隔±日間隔
- 日間隔±10 進定数
- 日間隔±DECIMAL
- 日間隔＊整数演算
- 日間隔/整数演算
- 時間隔±時間隔
- 時間隔±10 進定数
- 時間隔±DECIMAL
- 時間隔＊整数演算
- 時間隔/整数演算
- 日付データ±日付データ
- 時刻データ±時刻データ

5：日付データ±値式｛YEARS|MONTHS|DAYS｝だけを外部サーバで実行します。時刻演算は外部サーバで実行しません。

19）　pd_hub_opt_trailing_spc = 1～2　～<符号なし整数>《2》

後続空白を含む，又はその可能性がある可変長文字列型（比較述語，BETWEEN 述語，IN 述語，集合関数の MIN・MAX，SELECT DISTINCT 指定，DISTINCT 指定集合関数，GROUP BY，及び ORDER BY 句）を外部サーバで実行するかどうかを指定します。

ただし，列オプション COLLATING_SEQUENCE に'DIFFERENT'の指定がある列指定を含む文字列型の大小比較（比較述語<，<=，>，>=），BETWEEN 述語，集合関数の MIN・MAX，ORDER BY 句），文字列型定数同士の比較処理，文字列型定数と変数の比較処理，及び変数同士の比較処理は常に外部サーバで実行しません。

1：外部サーバで実行します。

2：後続空白を含む，又はその可能性がある可変長文字列型の比較処理を外部サーバで実行しません。次に示す値式が相当します。変数を用いる場合は，変数に設定する値に後続空白を含まないようにしてください。

- 値式は後続空白を含む文字列定数である場合
- 値式が連結演算ではなく，かつ列オプション TRAILING_SPACE に'YES'が指定された可変長文字列型の列を含む場合
- 値式が戻り値の型が可変長文字列型のスカラ関数 SUBSTR である場合
- 値式が戻り値の型が可変長文字列型の連結演算であり，かつ連結演算の最も右側の一次子が 1～3 の条件を満たす場合
- 値式が戻り値の型が可変長文字列型の連結演算であり，かつ連結演算の最も右側の一次子が固定長文字列型である場合（ただし，最も右側の一次子がスカラ関数 CHAR 及び DIGITS の場合を除く）

20）　pd_hub_opt_in_value_num = 1～30000　～<符号なし整数>《255》

外部サーバで実行する場合，IN 述語の右辺に指定する行値構成子数の最大値を指定します。IN 述語の右辺に指定する行値構成子数が指定値を超える場合は HiRDB 上で処理を実行します。

21）　pd_hub_opt_nullable = 1～2　～<符号なし整数>《2》

列オプションの NULLABLE に YES の指定がある場合，又は列オプションの NULLABLE を指定した場合，この列を含む連結演算を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

22）　pd_hub_opt_use_zero_string = 1～2　～<符号なし整数>《2》

0 バイト文字列を含むか，又はその可能性のある値指定を外部サーバで評価するかどうかを指定します。0 バイト文字列定数，データ型として文字列型を仮定する埋込み変数，SQL 変数，SQL パラメタ，及び？パラメタが対象となります。

1：外部サーバで実行します。

2：0 バイト文字列を含むか，又はその可能性のある値指定を次に示すどれかに含む場合，該当値指定の評価を外部サーバで実行しません。

- 述語
- GROUP BY 句
- ORDER BY 句
- 選択式（かつ GROUP BY 句，ORDER BY 句，DISTINCT，集合関数，HAVING 句のどれかを指定している）

外部サーバが 0 バイト文字列をナル値として扱う場合は 2 を指定してください。

23）　pd_hub_opt_nchar = 1～2　～<符号なし整数>《2》

外部サーバの各国文字型の文字コードセットが 2 バイト固定幅かどうか，及び埋め字が 2 バイト文字の空白かどうかを指定します。

1：2 バイト固定幅であり，かつ埋め字が 2 バイト文字の空白である場合に指定します。

2：2 バイト固定幅でない，又は埋め字が 2 バイト文字の空白でない場合に指定します。

2 を指定した場合，各国文字型，可変長各国文字型を含む述語，値式を外部サーバで実行しません。外部サーバの各国文字型の文字コードセットが 2 バイト固定幅ではない場合は 2 を指定してください。

24）　pd_hub_opt_nest_scalar = 0～65535　～<符号なし整数>《0》

外部サーバで実行するスカラ演算又は集合関数によるネスト数の最大値を指定します。0 を指定した場合はネスト数に関係なく外部サーバで実行します。また，ネスト数の数え方は HiRDB の数え方とは異なり，スカラ演算，集合関数のネスト数は演算の種類に関係なくすべて 1 とします。

25）　pd_hub_opt_float = 1～2　～<符号なし整数>《1》

浮動小数点型を含む値式，述語，GROUP BY 句，ORDER BY 句，選択式（かつ，GROUP BY 句，ORDER BY 句，DISTINCT，集合関数，HAVING 句のどれかの指定がある）を外部サーバで実行するかどうかを指定します。ただし，値指定の処理は外部サーバでは実行されません。

1：外部サーバで実行します。

2：外部サーバで実行しません。

外部サーバのマシン又は DBMS と自 HiRDB の浮動小数点型の精度が異なる場合は 2 を指定してください。

26）　pd_hub_opt_table_num = 0～65535　～<符号なし整数>《0》

1SQL 文中に指定した表数がこのオペランドの値を超える場合は HiRDB 上で処理を実行します。このオペランドを省略するか，又は 0 を指定した場合は，表数に関係なく外部サーバで処理を実行します。

27）　pd_hub_opt_time_24hour = 1～2　～<符号なし整数>《1》

TIME 型を外部サーバで実行するかどうかを指定します。

1：外部サーバで実行します。

2：外部サーバで実行しません。

外部 DBMS の TIME 型の列に 24 時 00 分 00 秒が格納されている場合は 2 を指定してください。

13.3　注意事項

- 互換性のない機能を外部サーバで実行すると指定した場合，外部サーバで処理を実行するときにエラーになります。また，構文は同じであるが，仕様が異なる機能を外部サーバで実行すると指定した場合，その機能は外部サーバの仕様に従い処理されます。
- 次に示すものは外部サーバで実行されません。
  - 関数呼出し
  - システム定義スカラ関数
  - IS_USER_CONTAINED_IN_HDS_GROUP 関数
  - HEX 関数
  - TIMESTAMP 関数
  - TIMESTAMP_FORMAT 関数
  - BIT_AND_TEST 関数
  - VARCHAR_FORMAT 関数
  - 日時書式を指定した DATE 関数
  - 日時書式を指定した TIME 関数
  - POSITION 関数
  - 行値構成子要素が二つ以上ある行値構成子
  - IN 述語のうち，右辺に値指定以外の行値構成子を含むもの
  - WRITE 指定
  - CAST 指定
  - GET_JAVA_STORED_ROUTINE_SOURCE 指定
  - LIMIT 句
  - 文字データと数データ間のデータ変換を必要とする比較述語
  - 文字データと数データ間のデータ変換を必要とする連結演算
  - 文字データと数データ間のデータ変換を必要とするスカラ関数
  - 文字データと数データ間のデータ変換を必要とする四則演算
  - ウィンドウ関数
  - SIMILAR 述語
  - COUNT_FLOAT
  - 列指定，コンポネント指定以外を指定した NULL 述語
  - 順序数生成子
- 選択式に SQL パラメタ，SQL 変数，USER，CURRENT_DATE，CURRENT_TIME，又は CURRENT_TIMESTAMP を指定した場合，その選択式は外部サーバで実行されません。
- 四則演算，日付演算，連結演算，スカラ関数，及び CASE 式中に，埋込み変数，?パラメタ，SQL パラメタ，SQL 変数，USER，CURRENT_DATE，CURRENT_TIME，CURRENT_TIMESTAMP を記述した場合，四則演算，日付演算，連結演算，スカラ関数，及び CASE 式の処理は外部サーバで実行されません。

- 次に示すデータ型を含む値式，述語，GROUP BY 句，ORDER BY 句，選択式（かつ GROUP BY 句，ORDER BY 句，DISTINCT，集合関数，HAVING 句のどれかの指定がある）は外部サーバで実行されません。
  - 時刻印型
  - BINARY 型
  - 明示的な文字集合指定のある文字列型
  - 精度が 29 を超えた DECIMAL 型
- ビュー表，WITH 句の問合せ名，又は導出表を FROM 句に指定した問合せ指定があり，このビュー定義，WITH 句中の導出問合せ式，又は導出表の FROM 句に，さらに内部導出表となるビュー表，WITH 句又は導出表を指定している場合，前者の問合せ指定は外部サーバで実行されません。
- ビュー定義，又は WITH 句中の導出問合せ式の選択式に集合関数を指定し，このビュー表，又は WITH 句の問合せ名を FROM 句に指定した問合せ指定の選択式に，集合関数から導出した列指定を含まない場合，その SQL は外部サーバで実行されません。
- 集合関数，及び GROUP BY 句の指定がなく，HAVING を指定した問合せ指定の選択式に列指定を含まない場合，その SQL は外部サーバで実行されません。
- ビュー定義，又は WITH 句中の導出問合せ式の複数の選択式に同じ列を指定し，かつその列をこのビュー表，又は WITH 句の問合せ名を FROM 句に指定した問合せ指定に GROUP BY 句を指定した場合，その SQL は外部サーバで実行されません。
- INSERT 文，UPDATE 文，及び DELETE 文中に，外部サーバで実行されない構文が含まれているとエラーになります。INSERT 文，UPDATE 文，及び DELETE 文以外の場合，外部サーバで実行されない構文の処理は HiRDB が実行します。
- 列指定を含まない比較述語のオペランドに，SQL パラメタ，SQL 変数，USER，CURRENT_DATE，CURRENT_TIME，又は CURRENT_TIMESTAMP を指定した場合，その SQL は外部サーバで実行されません。

13.4　各外部サーバの推奨値

各オペランドの外部サーバごとの推奨値は次のようになります。

Hub 最適化情報定義サンプルファイルの値はこれらの推奨値と同じです。したがって，Hub 最適化情報定義サンプルファイルをそのまま利用してください。Hub 最適化情報定義サンプルファイルの格納場所については，マニュアル「HiRDB External Data Access Version 8」を参照してください。

オペランド名	HiRDB Version 08-04	HiRDB Version 6～08-03	HiRDB Version 5.0	XDM/RD E2	ORACLE
pd_hub_opt_on_cnd	1	1	1	1	2
pd_hub_opt_joined_table	1	1	3	1	2
pd_hub_opt_set_func	1	1	3	3	1
pd_hub_opt_case	1	1	1	1	2
pd_hub_opt_like※1	1	1	1	4	4
pd_hub_opt_grouping	1	1	1	4	3
pd_hub_opt_data_len	255	255	255	255	255
pd_hub_opt_abs	1	1	1	1	3
pd_hub_opt_date	1	1	1	1	3
pd_hub_opt_time	1	1	1	1	2
pd_hub_opt_digits	1	1	1	1	2
pd_hub_opt_length	1	1	1	3	7
pd_hub_opt_lower_upper_type	1	1	1	1	6
pd_hub_opt_mod_div_type	1	1	1	2	1
pd_hub_opt_substr	1	1	1	3	2
pd_hub_opt_num	1	1	1	1	2
pd_hub_opt_datetime	1	1	1	1	2
pd_hub_opt_datetime_op	1	1	1	1	5
pd_hub_opt_trailing_spc	1	1	1	1	1
pd_hub_opt_in_value_num	30000	255	255	255	255
pd_hub_opt_nullable	1	1	1	1	2
pd_hub_opt_use_zero_string	1	1	1	1	2
pd_hub_opt_nchar※2	1	1	1	2	2
pd_hub_opt_nest_scalar	0	0	0	0	0
pd_hub_opt_float	2	2	2	2	2

オペランド名	HiRDB Version 08-04	HiRDB Version 6～08-03	HiRDB Version 5.0	XDM/RD E2	ORACLE
pd_hub_opt_table_num	64	64	64	64	64
pd_hub_opt_time_24hour	1	1	1	1	1

注※ 1

文字列型，可変長文字列型，混在文字列型，可変長混在文字列型のデータに対する LIKE 述語，NOT LIKE 述語，XLIKE 述語，NOT XLIKE 述語に全角の特殊文字を含む場合，又は各国文字列型，可変長各国文字列型のデータに対する LIKE 述語，NOT LIKE 述語，XLIKE 述語，NOT XLIKE 述語に半角の特殊文字を含む場合，これらの述語を外部サーバで実行しないようにしてください。

注※ 2

外部サーバの各国文字型の文字コードセットが 2 バイト固定幅ではない場合は 2 を指定してください。

付録

付録 A　オペランドで指定する内容

HiRDB システム定義で定義する内容，及びそれに対応するオペランド名を示します。なお，次に示す定義のオペランドはここには記載していません。

- 外部サーバ情報定義（HiRDB External Data Access 機能）
- Hub 最適化情報定義（HiRDB External Data Access 機能）

(1)　システム構成に関するオペランド

オペランドの内容	オペランド名
HiRDB 識別子を指定します。	pd_system_id
HiRDB のポート番号を指定します。	pd_name_port
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名を指定します。	pd_master_file_name
ユニットをホストに割り当てる指定をします。	pdunit
サーバをホストに割り当てる指定をします。	pdstart
ユニット識別子を指定します。	pd_unit_id
標準ホスト名を指定します。	pd_hostname

(2)　同時実行最大数に関するオペランド

オペランドの内容	オペランド名
最大同時接続数を指定します。	pd_max_users
1 ユニット内で同時に起動するサーバプロセス数の最大数を指定します。	pd_max_server_process
1 トランザクションで同時にアクセスできる表数の最大数を指定します。	pd_max_access_tables
ユティリティの同時実行最大数を変更します。	pd_utl_exec_mode
pdreclaim コマンド（-p オプション）の最大同時実行数を指定します。	pd_max_commit_write_reclaim_no

(3)　プロセスに関するオペランド

オペランドの内容	オペランド名
バックエンドサーバ当たりの最大起動プロセス数を指定します。	pd_max_bes_process
ディクショナリサーバ当たりの最大起動プロセス数を指定します。	pd_max_dic_process
あらかじめサーバ開始時から起動しておく常駐プロセス数を指定します。	pd_process_count
HiRDB の非常駐サーバプロセスの停止処理を行う間隔を分単位で指定します。この機能は，稼働中のサーバプロセス数が常駐プロセス数（pd_process_count オペランドの値）より多くなった場合に適用されます。	pd_server_cleanup_interval

オペランドの内容	オペランド名
非同期 READ 機能を使用する場合に，非同期 READ 処理に必要なプロセス数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（バックエンドサーバ又はディクショナリサーバ）当たりのプロセス数を指定します。	pd_max_ard_process
デファードライト処理の並列 WRITE 機能を使用する場合，並列処理するプロセス数を指定します。	pd_dfw_awt_process

(4)　作業表に関するオペランド

オペランドの内容	オペランド名
SQL 実行時に，一時的に情報を格納する作業表を作成する場合のバッファの確保方式を指定します。	pd_work_buff_mode
SQL 実行時に使用する作業表用バッファ長を指定します。	pd_work_buff_size
作業表用バッファが不足したとき，増分する作業表用バッファの上限値を指定します。	pd_work_buff_expand_limit
作業表用ファイル用の HiRDB ファイルシステム領域の名称を指定します。	pdwork

(5)　HiRDB の開始方法に関するオペランド

オペランドの内容	オペランド名
HiRDB の開始方法を指定します。	pd_mode_conf
pdstart コマンドの完了待ち時間を指定します。	pd_system_complete_wait_time
HiRDB の開始準備処理の最大待ち時間を指定します。	pd_start_time_out

(6)　縮退起動に関するオペランド

オペランドの内容	オペランド名
縮退起動をするかどうかを指定します。	pd_start_level
縮退起動開始の連絡待ち時間を設定します。	pd_reduced_check_time
HiRDB の開始時，障害などで開始しないユニットを指定します。	pd_start_skip_unit

(7)　HiRDB の処理方式に関するオペランド

オペランドの内容	オペランド名
データベースの更新内容をファイルに反映するタイミングを指定します。	pd_dbsync_point
次に示す RD エリアの更新内容をファイルに反映するタイミングを指定します。 • マスタディレクトリ用 RD エリア • データディレクトリ用 RD エリア • データディクショナリ用 RD エリア • データディクショナリ LOB 用 RD エリア • レジストリ用 RD エリア	pd_system_dbsync_point

オペランドの内容	オペランド名
• レジストリ LOB 用 RD エリア	
シンクポイント取得処理中に更新バッファに対して参照要求が発生した場合，その更新バッファの内容をデータベースに書き込む処理は，参照要求をしたトランザクションを実行するサーバプロセスが代行します。この代行処理をスキップするかどうかを指定します。	pd_dbsync_altwrite_skip
演算中のエラー抑止をするかどうかを指定します。	pd_overflow_suppress
HiRDB のプロセスが異常終了したときに後処理をする後処理プロセスを HiRDB の開始時に起動しておくかどうかを指定します。	pd_process_terminator
HiRDB の開始時に起動する後処理プロセスの数を指定します。	pd_process_terminator_max
HiRDB の 1 プロセス当たりのデスクトップヒープ消費量を指定します。	pd_process_desktopheap_size
空白変換機能を使用する場合の空白変換レベルを指定します。	pd_space_level
DECIMAL 型の符号正規化機能を使用するかどうかを指定します。	pd_dec_sign_normalize
演算結果が 29 けた以下の DECIMAL 型となるとき，演算結果の精度の最大値を指定します。	pd_sql_dec_op_maxprec
UAP の多重実行時に HiRDB サーバプロセスで処理が競合すると，処理要求を一時的にキューイングすることがあります。このときの HiRDB の処理方式を指定します。	pd_server_entry_queue
スレッド間のロック獲得処理などで使用するスリープ（一定時間処理を待つこと）関数の種類を指定します。	pd_thdlock_sleep_func
スレッド間ロックの解放通知方式を指定します。	pd_thdlock_wakeup_lock
スレッド間ロックの解放調査を行う間隔をマイクロ秒単位で指定します。	pd_thdlock_pipe_retry_interval
スレッド間ロックのスリープ時間をマイクロ秒単位で指定します。	pd_thdlock_retry_time
スレッド間スピンロックのスピン回数を指定します。	pd_thdspnlk_spn_count
データベース検索時のページアクセス方式を指定します。	pd_pageaccess_mode
RD エリアの閉塞状態のチェック方式を指定します。	pd_cmdhold_precheck
RD エリア（マスタディレクトリ用 RD エリアを除く）の入出力エラーが発生したときの HiRDB の処理を指定します。	pd_db_io_error_action
CONNECT 失敗時にエラーの要因を隠すかどうかを指定します。	pd_connect_errmsg_hide
受信用ポートの生成時，ループバックアドレスで bind()するかどうかを指定します。	pd_rpc_bind_loopback_address
サーバプロセスの強制終了が発生した場合に，出力されるエラーメッセージを変更するかどうかを指定します。	pd_cancel_down_msgchange

(8)　全面回復処理に関するオペランド

オペランドの内容	オペランド名
全面回復処理をするときの回復処理プロセス（REDO プロセス）数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（ディクショナリサーバ又はバックエンドサーバ）当たりの回復処理プロセス（REDO プロセス）数になります。	pd_max_recover_process

オペランドの内容	オペランド名
全面回復処理で，シンクポイント以降に更新されたページをすべてデータベースに書き出すかを指定します。	pd_redo_allpage_put

(9)　トランザクション決着処理に関するオペランド

オペランドの内容	オペランド名
トランザクションが COMMIT 一相目未完了状態でユニットが異常終了した場合，ユニットの再開始時にこのトランザクションからブランチした未決着状態のトランザクションを自動決着するかどうかを指定します。	pd_trn_rerun_branch_auto_decide
ブランチしたトランザクションに自動決着指示を送信した場合，送信が何らかの原因で失敗したときに決着指示を再送するまでの時間を秒単位で指定します。	pd_trn_send_decision_intval_sec
ブランチしたトランザクションに自動決着指示を送信した場合，送信が何らかの原因で失敗したときに決着指示を再送するまでの時間を分単位で指定します。	pd_trn_send_decision_interval
ブランチしたトランザクションに自動決着指示を送信してから，決着完了の連絡が戻ってくるまでの待ち時間を指定します。	pd_trn_send_decision_retry_time
HiRDB サーバプロセス上で実行されるトランザクションの同期点処理で，トランザクションブランチ間で行う通信（プリペア，コミット指示，又は応答など）の受信待ち時間の最大値を指定します。	pd_trn_watch_time
HiRDB/パラレルサーバのコミットメント制御で一相コミットを使用するかどうかを指定します。	pd_trn_commit_optimize

(10)　SQL の最適化に関するオペランド

オペランドの内容	オペランド名
SQL 最適化オプションを指定します。	pd_optimize_level
SQL 拡張最適化オプションを指定します。	pd_additional_optimize_level
SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシング方式を指定します。	pd_hashjoin_hashing_mode
SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシュ表サイズを指定します。	pd_hash_table_size
作業表を使用する SQL を実行する場合，次に示す HiRDB の処理方式を指定します。 • AND の複数インデクス利用時の排他取得方式 • 作業表用バッファ長の自動増分時のメッセージ出力	pd_work_table_option
バックエンドサーバの中からフロータブルサーバとして使用するサーバを指定ます。	pd_floatable_bes
バックエンドサーバをフロータブルサーバとして使用しないことを指定します。	pd_non_floatable_bes

(11)　拡張 SQL エラー情報出力機能に関するオペランド

オペランドの内容	オペランド名
拡張 SQL エラー情報出力機能を使用するかどうかを指定します。	pd_uap_exerror_log_use

オペランドの内容	オペランド名
SQL エラーレポートファイルの格納ディレクトリを指定します。	pd_uap_exerror_log_dir
SQL エラーレポートファイルの最大サイズを指定します。	pd_uap_exerror_log_size
エラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。	pd_uap_exerror_log_param_size

(12)　SQL 予約語削除機能に関するオペランド

オペランドの内容	オペランド名
SQL 予約語削除機能を使用する場合，SQL 予約語削除ファイルの名称を指定します。	pd_delete_reserved_word_file

(13)　SQL からのコマンド実行に関するオペランド

オペランドの内容	オペランド名
SQL の CALL COMMAND 文でコマンド及びユティリティを実行できる認可識別子を指定します。	pd_sql_command_exec_users

(14)　SQLSTATE の細分化に関するオペランド

オペランドの内容	オペランド名
SQLSTATE の値を詳細に出力するかどうかを指定します。	pd_standard_sqlstate

(15)　絞込み検索に関するオペランド

オペランドの内容	オペランド名
同時にリストを所有できるユーザの最大数を指定します。	pd_max_list_users
1 ユーザ当たりのリスト作成数を指定します。	pd_max_list_count
リストの初期化（削除）時期を指定します。通常，HiRDB を開始（再開始を含む）するときにリストが初期化されます。このオペランドでリストの初期化時期を変更できます。	pd_list_initialize_timing
インデクスを用いた検索で，IN 述語及び限定述語（= ANY（表副問合せ），= SOME（表副問合せ））を指定する場合，サーチ条件 ATS 又は RANGES を適用する絞り込み値の組み合わせ個数の上限を指定します。	pd_apply_search_ats_num

(16)　システム監視に関するオペランド

オペランドの内容	オペランド名
運用コマンド・ユティリティ実行時に，プロセスを割り当てるまでの監視時間を指定します。	pd_cmd_process_conwaittime
次に示すユティリティの実行時間を監視する場合，その監視時間を分単位で指定します。 • データベース作成ユティリティ（pdload コマンド）	pd_utl_exec_time

オペランドの内容	オペランド名
• データベース再編成ユティリティ（pdrorg コマンド） このオペランドで指定した監視時間を超えてもユティリティが終了しない場合，実行中のユティリティを強制終了して，無応答原因を調査するための障害情報を出力します。	
HiRDB サーバプロセス上で実行される SQL の最大実行時間を指定します。このオペランドで指定した時間を過ぎても SQL の実行が完了しない場合は，その SQL の実行を中断することがあります。	pd_watch_time
メッセージキュー中のメッセージが，このオペランドに指定した時間を過ぎても取り出されない場合，警告メッセージ（KFPS00888-W 又は KFPS00889-E）を出力します（メッセージキュー監視機能）。	pd_queue_watch_time
pd_queue_watch_time オペランドの値以上の時間，メッセージキューからメッセージが取り出されない場合の HiRDB の処理を指定します。	pd_queue_watch_timeover_action
サーバプロセスの異常終了回数がこのオペランドに指定した値を超えた場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を異常終了させます（プロセスの異常終了回数監視機能）。	pd_down_watch_proc
ほかのホストの稼働状況を監視するために問い合わせをする時間間隔を指定します。	pd_host_watch_interval
Windows 対応の HiRDB クライアントからの要求に対する応答をサーバが返してから，次に HiRDB クライアントからの要求があるまでの，サーバ側の最大待ち時間を指定します。	pd_watch_pc_client_time
シンクポイントダンプの有効化処理が連続してスキップされるとき，そのスキップ回数（1 トランザクション中のスキップ回数）の上限値を指定します。	pd_spd_syncpoint_skip_limit
デファードライト処理によってシンクポイントダンプの取得が遅れ，シンクポイントダンプの有効化処理がスキップされるとき，そのスキップ回数（1 トランザクション中のスキップ回数）の上限値を指定します。	pd_dfw_syncpoint_skip_limit
リソースの使用率に関する警告メッセージを出力するかどうかを指定します。	pd_watch_resource
pd_max_users オペランドで指定した最大同時接続数に対する警告メッセージを出力するかどうかを指定します。	pd_max_users_wrn_pnt
pd_max_access_tables オペランドで指定した同時アクセス可能実表数に対する警告メッセージを出力するかどうかを指定します。	pd_max_access_tables_wrn_pnt
pd_max_rdarea_no オペランドで指定した RD エリアの最大数に対する警告メッセージを出力するかどうかを指定します。	pd_max_rdarea_no_wrn_pnt
pd_max_file_no オペランドで指定した RD エリアを構成する HiRDB ファイルの最大数に対する警告メッセージを出力するかどうかを指定します。	pd_max_file_no_wrn_pnt
pdwork オペランドで指定した作業表用ファイル用の HiRDB ファイルシステム領域の使用率に対する警告メッセージを出力するかどうかを指定します。	pdwork_wrn_pnt
pd_max_list_users オペランドで指定したリスト作成ユーザ数に対する警告メッセージを出力するかどうかを指定します。	pd_max_list_users_wrn_pnt
pd_max_list_count オペランドで指定した 1 ユーザ当たりのリスト作成数に対する警告メッセージを出力するかどうかを指定します。	pd_max_list_count_wrn_pnt
サーバ内のリスト作成数に対する警告メッセージを出力するかどうかを指定します。	pd_rdarea_list_no_wrn_pnt

(17)　排他制御に関するオペランド

オペランドの内容	オペランド名
デッドロック情報を出力するかどうかを指定します。	pd_lck_deadlock_info
排他要求が待ち状態になってから解除されるまでの待ち時間を指定します。	pd_lck_wait_timeout
HiRDB が排他の解除を検知する方法を指定します。	pd_lck_release_detect
HiRDB が排他管理用領域を参照するとき，そのインターバル時間を指定します。	pd_lck_release_detect_interval
WITHOUT LOCK NOWAIT 指定の検索を実行したときの処理方式を指定します。	pd_nowait_scan_option
排他待ちユーザ数警告メッセージの出力契機を指定します。	pd_lck_queue_limit
デッドロックの優先順位を制御するかどうかを指定します。	pd_deadlock_priority_use
コマンドのデッドロックプライオリティ値を指定します。	pd_command_deadlock_priority
インデクスキー値の排他資源の作成方法を指定します。	pd_key_resource_type
サーバ当たりの排他制御用プールの大きさを指定します。	pd_lck_pool_size
フロントエンドサーバの排他制御用プールの大きさを指定します。	pd_fes_lck_pool_size
排他制御の分散をする場合，サーバ当たりの排他制御用プールの分割数を指定します。	pd_lck_pool_partition
排他制御の分散をする場合，フロントエンドサーバの排他制御用プールの分割数を指定します。	pd_fes_lck_pool_partition
トランザクションを越えて保持する表，及び RD エリアの排他資源数を指定します。	pd_lck_until_disconnect_cnt
UNTIL DISCONNECT 指定の LOCK 文を実行しない表に対してホールダブルカーソルを使用する場合に，そのホールダブルカーソルのトランザクション当たりの最大同時オープン数を指定します。	pd_max_open_holdable_cursors
排他制御用プールで使用するハッシュテーブルのエントリ数を指定します。	pd_lck_hash_entry
B-tree インデクスの排他制御方式を指定します（インデクスキー値無排他を実施するかどうかを指定します）。	pd_indexlock_mode
コミットしていない削除データの排他待ち制御方式を指定します。	pd_lock_uncommited_delete_data
デッドロックの発生を監視するかどうかを指定します。	pd_lck_deadlock_check
デッドロックの発生を監視する間隔を指定します。	pd_lck_deadlock_check_interval
シンクポイント時に発生するグローバルバッファへの排他を解除する間隔を指定します。	pd_dbsync_lck_release_count

(18)　バッファに関するオペランド

オペランドの内容	オペランド名
SQL オブジェクトを置くバッファの大きさを指定します。	pd_sql_object_cache_size
表定義情報用バッファ，ビュー解析情報用バッファ，ユーザ定義型情報用バッファ，及びルーチン定義情報用バッファの管理領域を HiRDB 開始時に一括して確保するか，又はトランザクション開始時に確保するかを指定します。	pd_def_buf_control_area_assign

オペランドの内容	オペランド名
1 スレッドが使用するスタックサイズの最大値を指定します。	pd_thread_max_stack_size
表定義情報を置くバッファの大きさを指定します。	pd_table_def_cache_size
ユーザ権限情報を置くバッファの大きさを指定します。	pd_auth_cache_size
ビュー解析情報を置くバッファの大きさを指定します。	pd_view_def_cache_size
ユーザ定義型情報を置くバッファの大きさを指定します。	pd_type_def_cache_size
ルーチン定義情報を置くバッファの大きさを指定します。	pd_routine_def_cache_size
レジストリ情報を置くバッファの大きさを指定します。	pd_registry_cache_size

(19)　共用メモリに関するオペランド

オペランドの内容	オペランド名
HiRDB が使用する共用メモリを固定するかどうかを指定します。	pd_shmpool_attribute
グローバルバッファが使用する共用メモリを実メモリ上に固定するかどうかを指定します。	pd_dbbuff_attribute
シングルサーバが使用する共用メモリの大きさを指定します。	pd_sds_shmpool_size
ディクショナリサーバが使用する共用メモリの大きさを指定します。	pd_dic_shmpool_size
バックエンドサーバが使用する共用メモリの大きさを指定します。	pd_bes_shmpool_size
グローバルバッファ用の共用メモリセグメントの大きさの上限値を指定します。	SHMMAX

(20)　RD エリアに関するオペランド

オペランドの内容	オペランド名
RD エリアの最大数を指定します。	pd_max_rdarea_no
RD エリアを構成する HiRDB ファイルの最大数を指定します。	pd_max_file_no
RD エリアのセグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I），又は増分する HiRDB ファイルの領域使用率通知メッセージ（KFPH22037-W）の出力契機を指定します。	pd_rdarea_warning_point
RD エリアのセグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I）を出力するかどうかを指定します。	pd_rdarea_warning_point_msgout
RD エリアの自動増分機能を適用している場合，増分領域を初期化するか否かを指定します。	pd_rdarea_expand_format
RD エリアの自動増分機能を適用している場合，自動的に RD エリアの容量を拡張する自動増分契機を指定します。	pd_rdarea_extension_timing
RD エリアのオープン属性の DEFER 属性，又は SCHEDULE 属性を使用するかどうかを指定します。	pd_rdarea_open_attribute_use
RD エリアのオープン契機の標準値を指定します。	pd_rdarea_open_attribute
共用 RD エリアを使用するかどうかを指定します。	pd_shared_rdarea_use

オペランドの内容	オペランド名
RD エリアへのアクセス時にファイルアクセスエラーが発生した場合の動作を指定します。	pd_db_access_error_action

(21)　グローバルバッファに関するオペランド

オペランドの内容	オペランド名
グローバルバッファの LRU 管理方式を指定します。	pd_dbbuff_lru_option
UAP がアクセスするバイナリデータに対してグローバルバッファの LRU 管理を行うかどうかを指定します。	pd_dbbuff_binary_data_lru
グローバルバッファを動的変更するかどうかを指定します。	pd_dbbuff_modify
グローバルバッファの排他解除の検知方式を指定します。	pd_dbbuff_lock_release_detect
グローバルバッファの排他獲得待ち処理中のスピン回数を指定します。	pd_dbbuff_lock_spn_count
グローバルバッファの排他獲得待ち処理中のインターバル時間を指定します。	pd_dbbuff_lock_interval
グローバルバッファの占有状態の調査する間隔を指定します。	pd_dbbuff_wait_interval
グローバルバッファの占有状態調査のスピンループ回数上限値を指定します。	pd_dbbuff_wait_spn_count
デファードライトトリガのトリガ契機を更新バッファ比率で指定します。	pd_dbbuff_rate_updpage
DB バッファ制御情報トレースの取得レベルを指定します	pd_dbbuff_trace_level
グローバルバッファに割り当てる RD エリアを定義します。	pdbuffer
動的追加用グローバルバッファ数の上限値を指定します。	pd_max_add_dbbuff_no
動的追加用共用メモリセグメント数の上限値を指定します。	pd_max_add_dbbuff_shm_no

(22)　インメモリデータ処理に関するオペランド

オペランドの内容	オペランド名
インメモリ RD エリアの最大数を指定します。	pd_max_resident_rdarea_no
インメモリデータバッファが使用する共用メモリセグメントの最大数を指定します。	pd_max_resident_rdarea_shm_no

(23)　表又はインデクスの予約数に関するオペランド

オペランドの内容	オペランド名
表予約数の最低保証値を指定します。	pd_assurance_table_no
インデクス予約数の最低保証値を指定します。	pd_assurance_index_no

(24)　参照制約及び検査制約に関するオペランド

オペランドの内容	オペランド名
参照制約及び検査制約で，制約名の定義を制約定義の前後どちらにするかを指定します。	pd_constraint_name
参照制約及び検査制約使用時に指定します。	pd_check_pending

(25)　HiRDB ファイルシステム領域に関するオペランド

オペランドの内容	オペランド名
2048 メガバイト以上の HiRDB ファイルシステム領域を使用するかどうかを指定します。	pd_large_file_use
HiRDB ファイルシステム領域へのアクセス時にキャッシュ未使用アクセスを使用するかどうかを指定します。	pd_ntfs_cache_disable

(26)　システムログファイルに関するオペランド

オペランドの内容	オペランド名
システムログファイルを二重化するかどうかを指定します。	pd_log_dual
システムログファイルの空き率が警告値未満になったときの HiRDB の処理方式を指定します（システムログファイルの空き容量監視機能）。	pd_log_remain_space_check
システムログの自動ログアンロード機能を使用する場合に，アンロードログファイルの出力先ディレクトリ名又は出力先 HiRDB ファイルシステム領域名を絶対パス名で指定します。	pd_log_auto_unload_path
システムログファイルの片系運転をするかどうかを指定します。	pd_log_singleoperation
オープンされたファイルがすべてスワップできない状態の場合，予約ファイルを使用するかどうかを指定します。	pd_log_rerun_reserved_file_open
再開始時にシステムログファイルをスワップするかどうかを指定します。	pd_log_rerun_swap
システムログファイルのスワップが完了するまでの待ち時間を指定します。	pd_log_swap_timeout
システムログファイルをスワップ先にできる状態にする条件から，アンロード済み状態であることを除くかどうかを指定します。アンロード済み状態であることを除くと，データベースの回復にシステムログを使用しないときの運用となります。	pd_log_unload_check
システムログの入出力に使用するバッファの大きさを指定します。	pd_log_max_data_size
システムログの出力に使用するバッファの面数を指定します。	pd_log_write_buff_count
システムログファイルのレコード長を指定します。	pd_log_rec_leng
ロールバック用ログ入力バッファ面数を指定します。	pd_log_rollback_buff_count
一つのシステムログファイルが拡張契機 1 回当たりに拡張するサイズ，及び拡張できるファイルサイズの上限を指定します。	pd_log_auto_expand_size
システムログファイルを構成するファイルグループの名称を指定します。	pdlogadfg -d sys
ファイルグループに割り当てるシステムログファイルの名称を指定します。	pdlogadpf -d sys

(27)　シンクポイントダンプファイルに関するオペランド

オペランドの内容	オペランド名
シンクポイントダンプファイルを二重化するかどうかを指定します。	pd_spd_dual
シンクポイントダンプが有効化されたときにメッセージを出力するかどうかを指定します。	pd_spd_assurance_msg
有効保証世代数を指定します。	pd_spd_assurance_count
縮退運転機能を使用するかどうかを指定します。	pd_spd_reduced_mode
予約ファイルを自動的にオープンするかどうかを指定します。	pd_spd_reserved_file_auto_open
シンクポイントダンプの入出力に使用するバッファの大きさを指定します。	pd_spd_max_data_size
シンクポイントダンプを取得する間隔を指定します。	pd_log_sdinterval
シンクポイントダンプファイルを構成するグループ名を指定します。	pdlogadfg -d spd
ファイルグループに割り当てるシンクポイントダンプファイルを指定します。	pdlogadpf -d spd

(28)　ステータスファイルに関するオペランド

オペランドの内容	オペランド名
ユニット用ステータスファイル名を指定します。	pd_syssts_file_name_1～7
ユニット開始時にステータスファイルに障害がある場合の処置を指定します。	pd_syssts_initial_error
ユニット用ステータスファイルの片系運転をするかどうかを指定します。	pd_syssts_singleoperation
現用として使用するユニット用ステータスファイル名を指定します。	pd_syssts_last_active_file
再開始時に，前回の稼働時に正常だったユニット用ステータスファイルの系を指定します。	pd_syssts_last_active_side
サーバ用ステータスファイル名を指定します。	pd_sts_file_name_1～7
サーバ開始時にステータスファイルの障害がある場合の処置を指定します。	pd_sts_initial_error
サーバ用ステータスファイルの片系運転をするかどうかを指定します。	pd_sts_singleoperation
現用として使用するサーバ用ステータスファイル名を指定します。	pd_sts_last_active_file
再開始時に，前回の稼働時の正常だったサーバ用ステータスファイルの系を指定します。	pd_sts_last_active_side

(29)　メッセージログファイルに関するオペランド

オペランドの内容	オペランド名
メッセージログの出力先ユニットを指定します。	pd_mlg_msg_log_unit
メッセージログファイルの最大容量を指定します。	pd_mlg_file_size

(30)　統計情報に関するオペランド

オペランドの内容	オペランド名
HiRDB の開始時から統計情報を取得するかどうかを指定します。	pd_statistics
統計ログファイルの最大容量を指定します。	pd_stj_file_size
統計ログバッファの大きさを指定します。	pd_stj_buff_size
HiRDB の開始時から統計情報を取得する場合に指定します。	pdstbegin
HiRDB の開始時から CONNECT 及び DISCONNECT に関する履歴情報を取得する場合に指定します。	pdhibegin

(31)　RPC トレース情報に関するオペランド

オペランドの内容	オペランド名
RPC トレース情報を取得するかどうかを指定します。	pd_rpc_trace
RPC トレース情報を取得するファイル名を指定します。	pd_rpc_trace_name
RPC トレース情報を取得するファイルの容量を指定します。	pd_rpc_trace_size

(32)　トラブルシュート情報に関するオペランド

オペランドの内容	オペランド名
トラブルシュート情報を出力するかどうかを指定します。	pd_cancel_dump
トランザクションの実行中にクライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）を超えた場合にトラブルシュート情報を出力するかどうかを指定します。	pd_client_waittime_over_abort
トランザクションの実行中にクライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）を超えた場合に出力される共用メモリダンプの出力対象ユニットを制限するかどうかを指定します。	pd_clt_waittime_over_dump_level
トラブルシュート情報の再出力を抑止する時間を指定します。	pd_dump_suppress_watch_time
HiRDB プロセスの異常終了，又は HiRDB（ユニット）が異常終了したときに取得するトラブルシュート情報中に，ネットワーク情報を取得するかどうかを指定します。	pd_debug_info_netstat
HiRDB が出力したトラブルシュート情報ファイル（%PDDIR%¥spool 下のファイル），及び作業用一時ファイル（%PDDIR%¥tmp 下のファイル）の削除間隔を指定します。	pd_spool_cleanup_interval
pd_spool_cleanup_interval オペランドの削除対象となるファイルを指定します。	pd_spool_cleanup_interval_level
HiRDB が出力したトラブルシュート情報ファイル（%PDDIR%¥spool 下のファイル）を HiRDB 開始時に削除するかどうかを指定します。	pd_spool_cleanup
pd_spool_cleanup オペランドの削除対象となるトラブルシュート情報ファイルを指定します。	pd_spool_cleanup_level
モジュールトレース格納最大数を指定します。	pd_module_trace_max
モジュールトレース出力時刻取得方法を指定します。	pd_module_trace_timer_level

オペランドの内容	オペランド名
通信トレース格納最大数を指定します。	pd_pth_trace_max

(33)　バックエンドサーバ接続保持機能に関するオペランド

オペランドの内容	オペランド名
バックエンドサーバ接続保持機能を使用するかどうかを指定します。	pd_bes_connection_hold
バックエンドサーバ接続保持時間を指定します。	pd_bes_conn_hold_trn_interval

(34)　再編成時期予測機能に関するオペランド

オペランドの内容	オペランド名
再編成時期予測機能を使用するかどうかを指定します。	pd_rorg_predict

(35)　セキュリティに関するオペランド

オペランドの内容	オペランド名
HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時から監査証跡を取得するかどうかを指定します。	pd_audit
監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。	pd_aud_file_name
1 監査証跡ファイルの最大容量をメガバイト単位で指定します。	pd_aud_max_generation_size
監査証跡ファイル用の HiRDB ファイルシステム領域内に作成する監査証跡ファイルの最大数（世代数）を指定します。	pd_aud_max_generation_num
スワップ先にできる監査証跡ファイルがない場合の HiRDB の処理方式を指定します。	pd_aud_no_standby_file_opr
監査証跡を非同期に出力する場合に使用するバッファ長をバイト単位で指定します。	pd_aud_async_buff_size
監査証跡を非同期に出力する場合に使用するバッファの面数を指定します。	pd_aud_async_buff_count
監査証跡を非同期に出力する場合に使用するバッファがすべて使用されているとき，未使用のバッファが確保できるまでバッファを監視する処理のリトライ間隔を指定します。	pd_aud_async_buff_retry_intvl
セキュリティ監査機能を使用する場合に監査証跡に出力する SQL 文のサイズをバイト単位で指定します。	pd_aud_sql_source_size
セキュリティ監査機能を使用する場合に監査証跡に出力する SQL データのサイズをバイト単位で指定します。	pd_aud_sql_data_size
スワップ先にできない監査証跡ファイル数が警告値以上になったとき，警告メッセージを出力します。このオペランドには警告値を，pd_aud_max_generation_num オペランドに指定した監査証跡ファイルの最大数に対する比率で指定します。	pd_aud_file_wrn_pnt
監査証跡表の自動データロード機能を使用するかどうかを指定します。	pd_aud_auto_loading
監査証跡表の自動データロード機能で動作させるデータベース作成ユティリティ（pdload）の環境情報を定義します。	pdaudload

オペランドの内容	オペランド名
セキュリティ監査機能の情報を置くバッファの大きさを指定します。	pd_audit_def_buffer_size
HiRDB サーバを構成するネットワークで使用している全ホスト名を指定します。	pd_security_host_group

(36)　系切り替え機能に関するオペランド

オペランドの内容	オペランド名
系切り替え機能を使用するかどうかの指定をします。	pd_ha
系切り替え機能を使用する場合に IP アドレスを引き継ぐかどうかを指定します。	pd_ha_ipaddr_inherit
特定のユニットだけ系切り替えしない場合に指定します。	pd_ha_unit
系切り替え機能の運用をモニタモードで行うか，又はサーバモードで行うかを指定します。	pd_ha_acttype
ユーザサーバホットスタンバイを使用するかどうかを指定します。	pd_ha_server_process_standby
高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用するかどうかを指定します。	pd_ha_agent
影響分散スタンバイレス型系切り替え機能を使用する場合，ユニット内で同時に実行系として稼働できるゲスト BES 数の最大値を指定します。	pd_ha_max_act_guest_servers
影響分散スタンバイレス型系切り替え機能を使用する場合，ユニット内の最大起動ユーザサーバプロセス数を指定します。	pd_ha_max_server_process
影響分散スタンバイレス型系切り替え機能を使用する場合，ゲスト BES 受け入れ後のユニット内のホスト BES とゲスト BES の常駐プロセス数の合計値を指定します。	pd_ha_process_count
影響分散スタンバイレス型系切り替え機能を使用する場合，ユニットを開始するときに，実行系サーバのリソースが活性化されるまでの最大待ち時間を指定します。	pd_ha_resource_act_wait_time
トランザクションキューイング機能を使用するかどうかを指定します。また，系の切り替え中に最大同時接続数（pd_max_users オペランドの値）を超えた場合の処理を指定します。	pd_ha_transaction
トランザクションキューイング機能を使用する場合のトランザクションのキューイング待ち時間を指定します。	pd_ha_trn_queuing_wait_time
系切り替えが発生してから待機系ユニットが再開始するまでの間で，トランザクションの開始要求のリトライ処理時間の上限を指定します。	pd_ha_trn_restart_retry_time
系切り替え時の HiRDB（HiRDB/パラレルサーバの場合はユニット）の内部停止処理がサーバ障害監視時間を超えた場合に，HiRDB の内部停止処理を待たないで系を切り替えるかどうかを指定します。	pd_ha_switch_timeout
系切り替えをサーバモードで運用している場合，実行系での強制停止処理時にユニットが異常終了したときに系切り替えを抑止するかどうかを指定します。	pd_deter_restart_on_stop_fail
システムマネジャユニットの系が切り替わる時に，ほかのユニットの開始を待ち合わせるかどうか指定します。	pd_ha_mgr_rerun
HA グループを定義します。	pdhagroup
ユニットの識別子を指定します。	pdunit

オペランドの内容	オペランド名
HiRDB を構成するサーバをどのサーバマシンで実行するか，各サーバマシンをどのように使用するかを指定します。	pdstart
現用系の標準ホスト名を指定します。	pd_hostname

(37)　HiRDB Datareplicator に関するオペランド

オペランドの内容	オペランド名
HiRDB Datareplicator 連携機能を HiRDB 開始時から使用するかどうかを指定します。	pd_rpl_init_start
HiRDB Datareplicator 連携機能を使用する場合に，トランザクションを反映する単位を指定します。	pd_rpl_reflect_mode
HiRDB Datareplicator 連携機能を使用しているときに，HiRDB Datareplicator でのシステムログの抽出が完了していないため，すべてのシステムログファイルがスワップ先にできない状態でスワップ要求が発生した場合の処理方法を指定します。	pd_log_rpl_no_standby_file_opr
HiRDB Datareplicator 連携機能を使用する場合，抽出側 HiRDB Datareplicator 運用ディレクトリ名を指定します。	pd_rpl_hdepath
HiRDB Datareplicator 連携機能を使用する場合の，抽出側 HiRDB で実行する機能を指定します。	pd_rpl_func_control

(38)　JP1 との連携に関するオペランド

オペランドの内容	オペランド名
HiRDB のイベントを JP1 に登録するかどうかを指定します。	pd_jp1_use
JP1 に通知する HiRDB のイベントの種類を指定します。	pd_jp1_event_level
JP1 にメッセージ出力イベントを通知するかどうかを指定します。	pd_jp1_event_msg_out

(39)　ディレクトリサーバ連携機能に関するオペランド

オペランドの内容	オペランド名
ディレクトリサーバ連携機能を使用する場合に指定します。	pd_directory_server

(40)　HiRDB External Data Access 機能に関するオペランド

オペランドの内容	オペランド名又は定義名
接続する外部サーバの最大数を指定します。	pd_max_foreign_server
外部サーバのクライアントライブラリのパス名を指定します。	pd_foreign_server_libpath
Hub 最適化情報定義を適用する外部サーバを指定します。	pdhubopt
HiRDB External Data Access 機能の実行環境を定義します。	外部サーバ情報定義
入力された問合せの各構文を外部サーバで実行するかどうかを定義します。また，コスト式計算に必要なパラメタの値を定義します。	Hub 最適化情報定義

(41)　OLTP に関するオペランド

オペランドの内容	オペランド名
OLTP 環境下の UAP で，ホールダブルカーソル機能を使用するかどうかを指定します。	pd_oltp_holdcr

(42)　日付・時刻に関するオペランド

オペランドの内容	オペランド名
うるう秒を使用するかどうかを指定します。	pd_leap_second
タイムゾーンを指定します。	TZ

(43)　メッセージの出力抑止機能に関するオペランド

オペランドの内容	オペランド名
HiRDB がイベントログに出力するメッセージに対して，出力又は出力抑止の制御をします。	pdmlgput

(44)　クライアントグループに関するオペランド

オペランドの内容	オペランド名
クライアントグループの接続枠保証機能を使用する場合に，接続保証ユーザ数を指定します。	pdcltgrp

(45)　プラグインに関するオペランド

オペランドの内容	オペランド名
使用するプラグインの名称を指定します。	pdplugin
プラグインが使用する共用メモリの大きさを指定します。	pdplgprm
プラグインインデクスの遅延一括作成をするときに，インデクス情報ファイルを作成するディレクトリを指定します。	pd_plugin_ixmk_dir

(46)　バージョンアップに関するオペランド

オペランドの内容	オペランド名
HiRDB をバージョンアップするときに実行する pdvrup コマンドを自動起動するかどうかを指定します。	pd_auto_vrup
HiRDB システム定義のオペランドの省略値を HiRDB Version 6 以前のままにするかどうかを指定します。	pd_sysdef_default_option

(47)　通信処理に関するオペランド

オペランドの内容	オペランド名
スケジューラプロセスのポート番号を指定します。	pd_service_port
ユニット間のシステムサーバと通信する場合のアドレス解決方法を指定します。	pd_name_fixed_port_lookup
pd_name_fixed_port_lookup オペランドに Y を指定した場合に，スケジューラプロセスのポート番号を指定します。	pd_scd_port
pd_name_fixed_port_lookup オペランドに Y を指定した場合に，トランザクションサーバプロセスのポート番号を指定します。	pd_trn_port
pd_name_fixed_port_lookup オペランドに Y を指定した場合に，メッセージログサーバプロセスのポート番号を指定します。	pd_mlg_port
pd_name_fixed_port_lookup オペランドに Y を指定した場合に，ユニット監視プロセスのポート番号を指定します。	pd_alv_port
HiRDB サーバから HiRDB クライアントへの通信時に使用するネットワークを指定します。	pd_change_clt_ipaddr
HiRDB が通信処理で使用するポート番号の範囲を指定します。	pd_registered_port
pd_registered_port オペランドで範囲指定したポート番号と%windir%¥system32¥drivers¥etc¥services に登録されたポート番号との重複チェックをするかどうかを指定します。	pd_registered_port_check
HiRDB 予約ポート機能の対象範囲を指定します。	pd_registered_port_level
プロセス間通信の送信リトライ処理の連続繰り返し回数を指定します。	pd_ipc_send_retrycount
プロセス間通信の送信リトライ処理のスリープ時間を指定します。	pd_ipc_send_retrysleeptime
サーバ間の送信処理が終了するまでの送信処理のリトライ回数を指定します。	pd_ipc_send_count
サーバ間の受信処理が終了するまでの受信処理のリトライ回数を指定します。	pd_ipc_recv_count
HiRDB サーバ間通信（ユニット間の通信）で，コネクション確立をノンブロックモードで行うかどうかを指定します。	pd_ipc_conn_nblock
HiRDB のサーバ間通信（ユニット間の通信）におけるノンブロックモードでのコネクション確立処理のリトライ時間と，コネクションの確立監視時間を指定します。	pd_ipc_conn_nblock_time
データ送信時に行うコネクション確立処理のリトライ間隔を指定します。	pd_ipc_conn_interval
データ送信時に行うコネクション確立処理のリトライ回数を指定します。	pd_ipc_conn_count
サーバのユニット間通信（TCP INET ドメイン）で使用する送受信バッファの最大値を指定します。	pd_ipc_inet_bufsize
HiRDB サーバがあるホスト以外のホスト上から接続した HiRDB クライアントとの通信（TCP INET ドメイン）で使用する送受信バッファの最大値を指定します。	pd_tcp_inet_bufsize
ユティリティのプロセスが使用する通信用バッファ 1 面のサイズを指定します。	pd_utl_buff_size
ユティリティがファイルに 1 回当たりにアクセスするときのデータサイズを指定します。	pd_utl_file_buff_size
HiRDB サーバ間で検索結果のデータ転送をする場合の通信用バッファ 1 面のサイズを指定します。	pd_sql_send_buff_size

(48)　Java に関するオペランド

オペランドの内容	オペランド名
Java 仮想マシンの起動オプションを指定します。	pd_java_option
外部 Java ルーチンが使用するスタック領域長を指定します。	pd_java_routine_stack_size
Java ストアドプロシジャ又は Java ストアドファンクションで使用する JAR ファイルを格納するディレクトリ名を指定します。	pd_java_archive_directory
Java 仮想マシンが使用するクラスパスを指定します。	pd_java_classpath
Java Runtime Environment のルートディレクトリを絶対パス名で指定します。	pd_java_runtimepath
Java 仮想マシンのライブラリが格納されているディレクトリを，Java Runtime Environment のルートディレクトリへの相対パス名で指定します。	pd_java_libpath
Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合，そのファイル名を指定します。	pd_java_stdout_file

(49)　外部 C ストアドルーチンに関するオペランド

オペランドの内容	オペランド名
外部 C ストアドルーチンで使用する C ライブラリファイルを格納するディレクトリを指定します。	pd_c_library_directory

(50)　SQL 実行時間警告出力機能に関するオペランド

オペランドの内容	オペランド名
SQL 実行時間警告情報を出力する条件を指定します。	pd_cwaittime_wrn_pnt
SQL 実行時間警告情報ファイルの出力先ディレクトリを指定します。	pd_cwaittime_report_dir
SQL 実行時間警告情報ファイルの最大容量を指定します。	pd_cwaittime_report_size

(51)　ローカルバッファに関するオペランド

オペランドの内容	オペランド名
ローカルバッファを使用してアクセスする RD エリア又はインデクスがほかのユーザに使用されている場合の UAP の動作を指定します。	pd_uap_wait
UAP が使用するローカルバッファを定義します。	pdlbuffer

(52)　文字コードに関するオペランド

オペランドの内容	オペランド名
1 文字を表現する最大バイト数を指定します。	pd_substr_length

(53)　ワークファイル出力先ディレクトリの変更に関するオペランド

オペランドの内容	オペランド名
HiRDB が出力するワークファイルの出力先ディレクトリ名を指定します。	pd_tmp_directory

付録 B　定義例

付録 B.1　HiRDB/シングルサーバの場合

HiRDB/シングルサーバのシステム定義を作成します。システム構成は次のとおりです。

●ユニット構成

注　（　）内はサーバ名を表しています。

●システムファイルの構成

(1)　システム共通定義

```
set pd_system_id = PDB1                                          1
set pd_name_port = 20001                                         2
set pd_master_file_name = "C:¥rdarea¥mast¥mast01"                3
set pd_max_users = 100                                           4
set pd_max_server_process = 220                                  5
set pd_max_access_tables = 50                                    6
set pd_sql_object_cache_size = 3000                              7
set pd_max_rdarea_no = 200                                       8
set pd_max_file_no = 600                                         9
set pd_optimize_level = "PRIOR_NEST_JOIN", ¥                     10
    "PRIOR_OR_INDEXES","DETER_AND_INDEXES", ¥
    "RAPID_GROUPING","DETER_WORK_TABLE_FOR_UPDATE", ¥
    "APPLY_ENHANCED_KEY_COND"
set pd_additional_optimize_level = "COST_BASE_2"                 11
pdunit -x host1 -u UNT1 -d "C:¥HiRDB"                            12
pdstart -t SDS -s sds1 -u UNT1                                   13
pdbuffer -a gbuf01 -r RDMAST,RDDIC,RDDIR -n 1000                 14
```

```
pdbuffer -a gbuf02 -r RDAREA1,RDAREA2,RDAREA3 -n 1000         15
pdbuffer -a gbuf03 -r RDAREA4,RDAREA5,RDAREA6 -n 1000         16
pdbuffer -a gbuf04 -o -n 1000                                 17
putenv SHMMAX 16                                              18
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。
4. 最大同時接続数を指定します。
5. 最大同時起動サーバプロセス数を指定します。
6. 同時アクセス可能実表数を指定します。
7. SQL オブジェクト用バッファ長を指定します。
8. RD エリアの最大数を指定します。
9. RD エリアを構成する HiRDB ファイルの最大数を指定します。
10. SQL 最適化オプションを指定します。
11. SQL 拡張最適化オプションを指定します。
12. ユニットの構成を指定します。
    -x：ホスト名を指定します。
    -u：ユニット識別子を指定します。
    -d：HiRDB 運用ディレクトリ名を指定します。
13. サーバの構成を指定します。
    -t：サーバの種別（シングルサーバ：SDS）を指定します。
    -s：サーバ名を指定します。
    -u：ユニット識別子を指定します。
14. マスタディレクトリ用 RD エリア，データディクショナリ用 RD エリア，及びデータディレクトリ用 RD エリアにグローバルバッファを割り当てます。
15. ユーザ用 RD エリア（RDAREA1～RDAREA3）にグローバルバッファを割り当てます。
16. ユーザ用 RD エリア（RDAREA4～RDAREA6）にグローバルバッファを割り当てます。
17. そのほかの RD エリアにグローバルバッファを割り当てます。
18. 共用メモリセグメントサイズの上限値を指定します。

(2)　ユニット制御情報定義

```
set pd_unit_id = UNT1                                                1
set pd_syssts_file_name_1 = ”untsts1”,”D:¥sysfile_a¥usts1a”,¥        2
                                      ”E:¥sysfile_b¥usts1b”
set pd_syssts_file_name_2 = ”untsts2”,”D:¥sysfile_a¥usts2a”,¥
                                      ”E:¥sysfile_b¥usts2b”
```

〔説明〕

1. ユニット識別子を指定します。
2. ユニット用ステータスファイルの構成を指定します。

(3)　シングルサーバ定義

```
set pd_sds_shmpool_size = 10000                                          1
set pd_log_dual = Y                                                      2
set pd_sts_file_name_1 = ”svrsts1”,”D:¥sysfile_a¥ssts1a”,¥               3
                                 ”E:¥sysfile_b¥ssts1b”
set pd_sts_file_name_2 = ”svrsts2”,”D:¥sysfile_a¥ssts2a”,¥
                                 ”E:¥sysfile_b¥ssts2b”
pdwork -v ”C:¥work01”,”C:¥work02”                                        4
pdlogadfg -d sys -g log1 ONL                                             5
pdlogadfg -d sys -g log2 ONL
pdlogadfg -d sys -g log3 ONL
pdlogadfg -d sys -g log4 ONL
pdlogadpf -d sys -g log1 -a ”D:¥sysfile_a¥log1a” -b ”E:¥sysfile_b¥log1b”
pdlogadpf -d sys -g log2 -a ”D:¥sysfile_a¥log2a” -b ”E:¥sysfile_b¥log2b”
pdlogadpf -d sys -g log3 -a ”D:¥sysfile_a¥log3a” -b ”E:¥sysfile_b¥log3b”
pdlogadpf -d sys -g log4 -a ”D:¥sysfile_a¥log4a” -b ”E:¥sysfile_b¥log4b”
pdlogadfg -d spd -g sync1 ONL                                            6
pdlogadfg -d spd -g sync2 ONL
pdlogadfg -d spd -g sync3 ONL
pdlogadfg -d spd -g sync4 ONL
pdlogadpf -d spd -g sync1 -a ”D:¥sysfile_a¥sync1”
pdlogadpf -d spd -g sync2 -a ”E:¥sysfile_b¥sync2”
pdlogadpf -d spd -g sync3 -a ”D:¥sysfile_a¥sync3”
pdlogadpf -d spd -g sync4 -a ”E:¥sysfile_b¥sync4”
```

〔説明〕

1. シングルサーバが使用する共用メモリサイズを指定します。
2. システムログファイルの二重化を指定します。
3. サーバ用ステータスファイルの構成を指定します。
4. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
5. システムログファイルの構成を指定します。
6. シンクポイントダンプファイルの構成を指定します。

(4)　UAP 環境定義

```
set pd_uap_wait = Y                                        1
pdlbuffer -a localbuf1 -r RDAREA10 -n 1000 -p 16           2
pdlbuffer -a localbuf2 -r RDAREA11,RDAREA12 -n 1000        3
pdlbuffer -a localbuf3 -i USER01.INDX01 -n 1000            4
```

〔説明〕

1. ローカルバッファを使用してアクセスする RD エリア又はインデクスがほかのユーザに使用されている場合の UAP の動作を指定します。
2. ユーザ用 RD エリア（RDAREA10）にローカルバッファを割り当てます。
3. ユーザ用 RD エリア（RDAREA11，RDAREA12）にローカルバッファを割り当てます。
4. インデクス（INDX01）にローカルバッファを割り当てます。

付録 B.2　HiRDB/シングルサーバの場合：系切り替え機能使用時

HiRDB/シングルサーバのシステム定義を作成します。システム構成は次のとおりです。

- HiRDB/シングルサーバ 1 と HiRDB/シングルサーバ 2 の相互系切り替え構成です。
- IP アドレスを引き継がない系切り替え機能とします。

なお, 定義例はシステム共通定義及びユニット制御情報定義の関連のあるオペランドについてだけ説明しています。

注　網掛けが現用系のHiRDBです。

(1)　HiRDB/シングルサーバ 1 のシステム共通定義

```
set pd_system_id = SDS1                                  1
set pd_name_port = 20003                                 2
         :
         :
set pd_ha = use                                          3
set pd_ha_ipaddr_inherit = N                             4
         :
         :
pdunit -x host1 -u UNT1 -d "C:¥HiRDB1" -c host2          5
pdstart -t SDS -s sds01 -u UNT1                          6
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. 系切り替え機能を使用することを指定します。
4. IP アドレスを引き継がないことを指定します。
5. HiRDB/シングルサーバ 1 のユニット構成を指定します。
   -x：ホスト名を指定します。
   -u：ユニット識別子を指定します。
   -d：HiRDB 運用ディレクトリ名を指定します。
   -c：予備系のホスト名を指定します。
6. HiRDB/シングルサーバ 1 のサーバ構成を指定します。
   -t：サーバの種別を指定します。
   -s：サーバ名を指定します。
   -u：ユニット識別子を指定します。

(2)　HiRDB/シングルサーバ 1 のユニット制御情報定義

```
set pd_unit_id = UNT1                                    1
         :
         :
set pd_hostname = host1                                  2
```

〔説明〕

1. ユニット識別子を指定します。
2. 現用系のホスト名を指定します。

(3)　HiRDB/シングルサーバ 2 のシステム共通定義

```
set pd_system_id = SDS2                                  1
set pd_name_port = 20004                                 2
          :
          :
set pd_ha = use                                          3
set pd_ha_ipaddr_inherit = N                             4
          :
          :
pdunit -x host2 -u UNT2 -d ”C:¥HiRDB2” -c host1          5
pdstart -t SDS -s sds02 -u UNT2                          6
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. 系切り替え機能を使用することを指定します。
4. IP アドレスを引き継がないことを指定します。
5. HiRDB/シングルサーバ 2 のユニット構成を指定します。
   -x：ホスト名を指定します。
   -u：ユニット識別子を指定します。
   -d：HiRDB 運用ディレクトリ名を指定します。
   -c：予備系のホスト名を指定します。
6. HiRDB/シングルサーバ 2 のサーバ構成を指定します。
   -t：サーバの種別を指定します。
   -s：サーバ名を指定します。
   -u：ユニット識別子を指定します。

(4)　HiRDB/シングルサーバ 2 のユニット制御情報定義

```
set pd_unit_id = UNT2                                    1
          :
          :
set pd_hostname = host2                                  2
```

〔説明〕

1. ユニット識別子を指定します。
2. 現用系のホスト名を指定します。

付録 B.3　HiRDB/パラレルサーバの場合

HiRDB/パラレルサーバのシステム定義を作成します。システム構成は次のとおりです。

●ユニット構成

注　（　）内はサーバ名を表しています。

●UNT1 のシステムファイルの構成

●UNT2 のシステムファイルの構成

●UNT3 のシステムファイルの構成

(1)　システム共通定義

```
set pd_system_id = PDB1                                          1
set pd_name_port = 20001                                         2
set pd_master_file_name = "C:¥rdarea¥mast¥mast01"                3
set pd_max_users = 100                                           4
set pd_max_server_process = 520                                  5
set pd_max_access_tables = 50                                    6
set pd_sql_object_cache_size = 3000                              7
set pd_max_rdarea_no = 200                                       8
set pd_max_file_no = 600                                         9
set pd_optimize_level = "PRIOR_NEST_JOIN","PRIOR_OR_INDEXES", ¥  10
    "SORT_DATA_BES","DETER_AND_INDEXES","RAPID_GROUPING", ¥
    "DETER_WORK_TABLE_FOR_UPDATE", ¥
    "APPLY_ENHANCED_KEY_COND"
set pd_additional_optimize_level = "COST_BASE_2"                 11
pdunit -x host1 -u UNT1 -d "C:¥HiRDB"                            12
pdunit -x host2 -u UNT2 -d "C:¥HiRDB"
pdunit -x host3 -u UNT3 -d "C:¥HiRDB"
pdstart -t MGR -u UNT1                                           13
pdstart -t FES -s fes -u UNT1
pdstart -t DIC -s dic -u UNT2
pdstart -t BES -s bes1 -u UNT1
pdstart -t BES -s bes2 -u UNT2
pdstart -t BES -s bes3 -u UNT3
pdbuffer -a gbuf01 -r RDMAST,RDDIC,RDDIR -n 1000                 14
pdbuffer -a gbuf02 -r RDAREA1,RDAREA2,RDAREA3 -n 1000            15
pdbuffer -a gbuf03 -r RDAREA4,RDAREA5,RDAREA6 -n 1000            16
pdbuffer -a gbuf04 -o -n 1000                                    17
putenv SHMMAX 16                                                 18
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。
4. 最大同時接続数を指定します。
5. 最大同時起動サーバプロセス数を指定します。
6. 同時アクセス可能実表数を指定します。
7. SQL オブジェクト用バッファ長を指定します。
8. RD エリアの最大数を指定します。
9. RD エリアを構成する HiRDB ファイルの最大数を指定します。
10. SQL 最適化オプションを指定します。
11. SQL 拡張最適化オプションを指定します。
12. HiRDB/パラレルサーバの全ユニットの構成を指定します。
    -x：ホスト名を指定します。
    -u：ユニット識別子を指定します。
    -d：HiRDB 運用ディレクトリ名を指定します。
13. HiRDB/パラレルサーバの全サーバの構成を指定します。
    -t：サーバの種別を指定します。
    -s：サーバ名を指定します。
    -u：ユニット識別子を指定します。
14. マスタディレクトリ用 RD エリア，データディクショナリ用 RD エリア，及びデータディレクトリ用 RD エリアにグローバルバッファを割り当てます。
15. ユーザ用 RD エリア（RDAREA1～RDAREA3）にグローバルバッファを割り当てます。

16. ユーザ用 RD エリア（RDAREA4～RDAREA6）にグローバルバッファを割り当てます。

17. そのほかの RD エリアにグローバルバッファを割り当てます。

18. 共用メモリセグメントサイズの上限値を指定します。

(2)　UNT1 のユニット制御情報定義

```
set pd_unit_id = UNT1                                                    1
set pd_syssts_file_name_1 = "u1sts1","D:¥unt1¥sysfile_a¥u1sts1a",¥       2
                                     "E:¥unt1¥sysfile_b¥u1sts1b"
set pd_syssts_file_name_2 = "u1sts2","D:¥unt1¥sysfile_a¥u1sts2a",¥
                                     "E:¥unt1¥sysfile_b¥u1sts2b"
```

〔説明〕

1. ユニット識別子を指定します。

2. ユニット用ステータスファイルの構成を指定します。

(3)　UNT2 のユニット制御情報定義

```
set pd_unit_id = UNT2                                                    1
set pd_syssts_file_name_1 = "u2sts1","D:¥unt2¥sysfile_a¥u2sts1a",¥       2
                                     "E:¥unt2¥sysfile_b¥u2sts1b"
set pd_syssts_file_name_2 = "u2sts2","D:¥unt2¥sysfile_a¥u2sts2a",¥
                                     "E:¥unt2¥sysfile_b¥u2sts2b"
```

〔説明〕

1. ユニット識別子を指定します。

2. ユニット用ステータスファイルの構成を指定します。

(4)　UNT3 のユニット制御情報定義

```
set pd_unit_id = UNT3                                                    1
set pd_syssts_file_name_1 = "u3sts1","D:¥unt3¥sysfile_a¥u3sts1a",¥       2
                                     "E:¥unt3¥sysfile_b¥u3sts1b"
set pd_syssts_file_name_2 = "u3sts2","D:¥unt3¥sysfile_a¥u3sts2a",¥
                                     "E:¥unt3¥sysfile_b¥u3sts2b"
```

〔説明〕

1. ユニット識別子を指定します。

2. ユニット用ステータスファイルの構成を指定します。

(5)　フロントエンドサーバ定義

```
set pd_log_dual = N                                                      1
set pd_sts_file_name_1 = "fsts1","D:¥unt1¥sysfile_a¥fsts1a",¥            2
                                 "E:¥unt1¥sysfile_b¥fsts1b"
set pd_sts_file_name_2 = "fsts2","D:¥unt1¥sysfile_a¥fsts2a",¥
                                 "E:¥unt1¥sysfile_b¥fsts2b"
pdlogadfg -d sys -g flog1 ONL                                            3
pdlogadfg -d sys -g flog2 ONL
pdlogadfg -d sys -g flog3 ONL
pdlogadfg -d sys -g flog4 ONL
pdlogadpf -d sys -g flog1 -a "D:¥unt1¥sysfile_a¥flog1a"
pdlogadpf -d sys -g flog2 -a "D:¥unt1¥sysfile_a¥flog2a"
pdlogadpf -d sys -g flog3 -a "D:¥unt1¥sysfile_a¥flog3a"
pdlogadpf -d sys -g flog4 -a "D:¥unt1¥sysfile_a¥flog4a"
pdlogadfg -d spd -g fsync1 ONL                                           4
pdlogadfg -d spd -g fsync2 ONL
pdlogadfg -d spd -g fsync3 ONL
pdlogadfg -d spd -g fsync4 ONL
pdlogadpf -d spd -g fsync1 -a "D:¥unt1¥sysfile_a¥fsync1"
pdlogadpf -d spd -g fsync2 -a "E:¥unt1¥sysfile_b¥fsync2"
```

```
pdlogadpf -d spd -g fsync3 -a "D:¥unt1¥sysfile_a¥fsync3"
pdlogadpf -d spd -g fsync4 -a "E:¥unt1¥sysfile_b¥fsync4"
```

〔説明〕

1. システムログファイルを二重化しないことを指定します。
2. サーバ用ステータスファイルの構成を指定します。
3. システムログファイルの構成を指定します。
4. シンクポイントダンプファイルの構成を指定します。

(6)　ディクショナリサーバ定義

```
set pd_dic_shmpool_size = 5000                                          1
set pd_log_dual = Y                                                     2
set pd_sts_file_name_1 = "dsts1","D:¥unt2¥sysfile_a¥dsts1a",¥           3
                                 "E:¥unt2¥sysfile_b¥dsts1b"
set pd_sts_file_name_2 = "dsts2","D:¥unt2¥sysfile_a¥dsts2a",¥
                                 "E:¥unt2¥sysfile_b¥dsts2b"
pdwork -v "C:¥unt2¥work01","C:¥unt2¥work02"                             4
pdlogadfg -d sys -g dlog1 ONL                                           5
pdlogadfg -d sys -g dlog2 ONL
pdlogadfg -d sys -g dlog3 ONL
pdlogadfg -d sys -g dlog4 ONL
pdlogadpf -d sys -g dlog1 -a "D:¥unt2¥sysfile_a¥dlog1a"¥
                          -b "E:¥unt2¥sysfile_b¥dlog1b"
pdlogadpf -d sys -g dlog2 -a "D:¥unt2¥sysfile_a¥dlog2a"¥
                          -b "E:¥unt2¥sysfile_b¥dlog2b"
pdlogadpf -d sys -g dlog3 -a "D:¥unt2¥sysfile_a¥dlog3a"¥
                          -b "E:¥unt2¥sysfile_b¥dlog3b"
pdlogadpf -d sys -g dlog4 -a "D:¥unt2¥sysfile_a¥dlog4a"¥
                          -b "E:¥unt2¥sysfile_b¥dlog4b"
pdlogadfg -d spd -g dsync1 ONL                                          6
pdlogadfg -d spd -g dsync2 ONL
pdlogadfg -d spd -g dsync3 ONL
pdlogadfg -d spd -g dsync4 ONL
pdlogadpf -d spd -g dsync1 -a "D:¥unt2¥sysfile_a¥dsync1"
pdlogadpf -d spd -g dsync2 -a "E:¥unt2¥sysfile_b¥dsync2"
pdlogadpf -d spd -g dsync3 -a "D:¥unt2¥sysfile_a¥dsync3"
pdlogadpf -d spd -g dsync4 -a "E:¥unt2¥sysfile_b¥dsync4"
```

〔説明〕

1. ディクショナリサーバが使用する共用メモリサイズを指定します。
2. システムログファイルの二重化を指定します。
3. サーバ用ステータスファイルの構成を指定します。
4. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
5. システムログファイルの構成を指定します。
6. シンクポイントダンプファイルの構成を指定します。

(7)　bes1 のバックエンドサーバ定義

```
set pd_bes_shmpool_size = 5000                                          1
set pd_log_dual = Y                                                     2
set pd_sts_file_name_1 = "b1sts1","D:¥unt1¥sysfile_a¥b1sts1a",¥         3
                                  "E:¥unt1¥sysfile_b¥b1sts1b"
set pd_sts_file_name_2 = "b1sts2","D:¥unt1¥sysfile_a¥b1sts2a",¥
                                  "E:¥unt1¥sysfile_b¥b1sts2b"
pdwork -v "C:¥unt1¥work01","C:¥unt1¥work02"                             4
pdlogadfg -d sys -g b1log1 ONL                                          5
pdlogadfg -d sys -g b1log2 ONL
pdlogadfg -d sys -g b1log3 ONL
pdlogadfg -d sys -g b1log4 ONL
pdlogadpf -d sys -g b1log1 -a "D:¥unt1¥sysfile_a¥b1log1a"¥
                           -b "E:¥unt1¥sysfile_b¥b1log1b"
pdlogadpf -d sys -g b1log2 -a "D:¥unt1¥sysfile_a¥b1log2a"¥
                           -b "E:¥unt1¥sysfile_b¥b1log2b"
```

```
pdlogadpf -d sys -g b1log3 -a "D:¥unt1¥sysfile_a¥b1log3a"¥
                           -b "E:¥unt1¥sysfile_b¥b1log3b"
pdlogadpf -d sys -g b1log4 -a "D:¥unt1¥sysfile_a¥b1log4a"¥
                           -b "E:¥unt1¥sysfile_b¥b1log4b"
pdlogadfg -d spd -g b1sync1 ONL                                    6
pdlogadfg -d spd -g b1sync2 ONL
pdlogadfg -d spd -g b1sync3 ONL
pdlogadfg -d spd -g b1sync4 ONL
pdlogadpf -d spd -g b1sync1 -a "D:¥unt1¥sysfile_a¥b1sync1"
pdlogadpf -d spd -g b1sync2 -a "E:¥unt1¥sysfile_b¥b1sync2"
pdlogadpf -d spd -g b1sync3 -a "D:¥unt1¥sysfile_a¥b1sync3"
pdlogadpf -d spd -g b1sync4 -a "E:¥unt1¥sysfile_b¥b1sync4"
```

〔説明〕

1. バックエンドサーバ（bes1）が使用する共用メモリサイズを指定します。
2. システムログファイルの二重化を指定します。
3. サーバ用ステータスファイルの構成を指定します。
4. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
5. システムログファイルの構成を指定します。
6. シンクポイントダンプファイルの構成を指定します。

(8)　bes2 のバックエンドサーバ定義

```
set pd_bes_shmpool_size = 5000                                     1
set pd_log_dual = Y                                                2
set pd_sts_file_name_1 = "b2sts1","D:¥unt2¥sysfile_a¥b2sts1a",¥    3
                                  "E:¥unt2¥sysfile_b¥b2sts1b"
set pd_sts_file_name_2 = "b2sts2","D:¥unt2¥sysfile_a¥b2sts2a",¥
                                  "E:¥unt2¥sysfile_b¥b2sts2b"
pdwork -v "C:¥unt2¥work03","C:¥unt2¥work04"                        4
pdlogadfg -d sys -g b2log1 ONL                                     5
pdlogadfg -d sys -g b2log2 ONL
pdlogadfg -d sys -g b2log3 ONL
pdlogadfg -d sys -g b2log4 ONL
pdlogadpf -d sys -g b2log1 -a "D:¥unt2¥sysfile_a¥b2log1a"¥
                           -b "E:¥unt2¥sysfile_b¥b2log1b"
pdlogadpf -d sys -g b2log2 -a "D:¥unt2¥sysfile_a¥b2log2a"¥
                           -b "E:¥unt2¥sysfile_b¥b2log2b"
pdlogadpf -d sys -g b2log3 -a "D:¥unt2¥sysfile_a¥b2log3a"¥
                           -b "E:¥unt2¥sysfile_b¥b2log3b"
pdlogadpf -d sys -g b2log4 -a "D:¥unt2¥sysfile_a¥b2log4a"¥
                           -b "E:¥unt2¥sysfile_b¥b2log4b"
pdlogadfg -d spd -g b2sync1 ONL                                    6
pdlogadfg -d spd -g b2sync2 ONL
pdlogadfg -d spd -g b2sync3 ONL
pdlogadfg -d spd -g b2sync4 ONL
pdlogadpf -d spd -g b2sync1 -a "D:¥unt2¥sysfile_a¥b2sync1"
pdlogadpf -d spd -g b2sync2 -a "E:¥unt2¥sysfile_b¥b2sync2"
pdlogadpf -d spd -g b2sync3 -a "D:¥unt2¥sysfile_a¥b2sync3"
pdlogadpf -d spd -g b2sync4 -a "E:¥unt2¥sysfile_b¥b2sync4"
```

〔説明〕

1. バックエンドサーバ（bes2）が使用する共用メモリサイズを指定します。
2. システムログファイルの二重化を指定します。
3. サーバ用ステータスファイルの構成を指定します。
4. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
5. システムログファイルの構成を指定します。
6. シンクポイントダンプファイルの構成を指定します。

(9)　bes3 のバックエンドサーバ定義

```
set pd_bes_shmpool_size = 5000                                          1
set pd_log_dual = Y                                                     2
set pd_sts_file_name_1 = ”b3sts1”,”D:¥unt3¥sysfile_a¥b3sts1a”,¥         3
                                 ”E:¥unt3¥sysfile_b¥b3sts1b”
set pd_sts_file_name_2 = ”b3sts2”,”D:¥unt3¥sysfile_a¥b3sts2a”,¥
                                 ”E:¥unt3¥sysfile_b¥b3sts2b”
pdwork -v ”C:¥unt3¥work01”,”C:¥unt3¥work02”                             4
pdlogadfg -d sys -g b3log1 ONL                                          5
pdlogadfg -d sys -g b3log2 ONL
pdlogadfg -d sys -g b3log3 ONL
pdlogadfg -d sys -g b3log4 ONL
pdlogadpf -d sys -g b3log1 -a ”D:¥unt3¥sysfile_a¥b3log1a”¥
                           -b ”E:¥unt3¥sysfile_b¥b3log1b”
pdlogadpf -d sys -g b3log2 -a ”D:¥unt3¥sysfile_a¥b3log2a”¥
                           -b ”E:¥unt3¥sysfile_b¥b3log2b”
pdlogadpf -d sys -g b3log3 -a ”D:¥unt3¥sysfile_a¥b3log3a”¥
                           -b ”E:¥unt3¥sysfile_b¥b3log3b”
pdlogadpf -d sys -g b3log4 -a ”D:¥unt3¥sysfile_a¥b3log4a”¥
                           -b ”E:¥unt3¥sysfile_b¥b3log4b”
pdlogadfg -d spd -g b3sync1 ONL                                         6
pdlogadfg -d spd -g b3sync2 ONL
pdlogadfg -d spd -g b3sync3 ONL
pdlogadfg -d spd -g b3sync4 ONL
pdlogadpf -d spd -g b3sync1 -a ”D:¥unt3¥sysfile_a¥b3sync1”
pdlogadpf -d spd -g b3sync2 -a ”E:¥unt3¥sysfile_b¥b3sync2”
pdlogadpf -d spd -g b3sync3 -a ”D:¥unt3¥sysfile_a¥b3sync3”
pdlogadpf -d spd -g b3sync4 -a ”E:¥unt3¥sysfile_b¥b3sync4”
```

〔説明〕

1. バックエンドサーバ（bes3）が使用する共用メモリサイズを指定します。
2. システムログファイルの二重化を指定します。
3. サーバ用ステータスファイルの構成を指定します。
4. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
5. システムログファイルの構成を指定します。
6. シンクポイントダンプファイルの構成を指定します。

(10)　UAP 環境定義

```
set pd_uap_wait = Y                                       1
pdlbuffer -a localbuf1 -r RDAREA10 -n 1000 -p 16          2
pdlbuffer -a localbuf2 -r RDAREA11,RDAREA12 -n 1000       3
pdlbuffer -a localbuf3 -i USER01.INDX01 -n 1000           4
```

〔説明〕

1. ローカルバッファを使用してアクセスする RD エリア又はインデクスがほかのユーザに使用されている場合の UAP の動作を指定します。
2. ユーザ用 RD エリア（RDAREA10）にローカルバッファを割り当てます。
3. ユーザ用 RD エリア（RDAREA11，RDAREA12）にローカルバッファを割り当てます。
4. インデクス（INDX01）にローカルバッファを割り当てます。

付録 B.4　HiRDB/パラレルサーバの場合：スタンバイ型系切り替え機能使用時

HiRDB/パラレルサーバのシステム定義を作成します。システム構成は次のとおりです。

- 相互系切り替え構成です。

- IP アドレスを引き継がない構成です。
- 全ユニットに高速系切り替え機能を適用します。

なお,定義例はシステム共通定義及びユニット制御情報定義の関連のあるオペランドについてだけ説明しています。

注　網掛けが現用系のユニットです。

ポイント

- IP アドレスを引き継がない場合はサーバマシンからの独立性を図るために，pdunit オペランドの-x 及び-c オプションに標準ホスト名ではなく，エイリアス IP アドレスのホスト名を指定してください。
- IP アドレスを引き継がない相互系切り替え構成の場合は，pdunit オペランドの-x 及び-c オプションに指定するホスト名は重複できません。(1)の pdunit オペランドの指定例を参考にしてください。

(1)　システム共通定義

```
        :
set pd_name_port = 5001                                  1
        :
set pd_ha = use                                          2
set pd_ha_ipaddr_inherit = N
set pd_ha_switch_timeout = Y
set pd_ha_transaction = queuing
set pd_ha_trn_queuing_wait_time = 240
```

```
set pd_ha_trn_restart_retry_time = 90
        :
pdunit -x host1A -u UNT1 -d "C:¥HiRDB_A" -c host2A -p 5001       3
pdunit -x host2B -u UNT2 -d "C:¥HiRDB_B" -c host1B -p 5002
pdunit -x host3A -u UNT3 -d "C:¥HiRDB_C" -c host4A -p 5003
pdunit -x host4B -u UNT4 -d "C:¥HiRDB_D" -c host3B -p 5004
pdstart -t MGR -u UNT1                                           4
pdstart -t FES -s fes1 -u UNT1
pdstart -t DIC -s dic  -u UNT2
pdstart -t BES -s bes1 -u UNT3
pdstart -t BES -s bes2 -u UNT4
        :
```

〔説明〕

1. HiRDB のポート番号を指定します。
2. 系切り替え機能に関するオペランドの指定です。
3. HiRDB/パラレルサーバのユニット構成を指定します。
   -x：ホスト名を指定します。
   -u：ユニット識別子を指定します。
   -d：HiRDB 運用ディレクトリ名を指定します。
   -c：予備系のホスト名を指定します。
   -p：ユニットのポート番号を指定します。
4. HiRDB/パラレルサーバのサーバ構成を指定します。
   -t：サーバの種別を指定します。
   -s：サーバ名を指定します。
   -u：ユニット識別子を指定します。

(2)　UNT1～UNT4 のユニット制御情報定義

```
        :
set pd_ha_acttype = server                        1
set pd_ha_agent = standbyunit                     2
        :
```

〔説明〕

1. 系切り替え機能をサーバモードで運用することを指定します。
2. 高速系切り替え機能を使用することを指定します。

付録 B.5　HiRDB/パラレルサーバの場合：1：1 スタンバイレス型系切り替え機能使用時

HiRDB/パラレルサーバのシステム定義を作成します。システム構成は次のとおりです。

- スタンバイ型とスタンバイレス型を混合した構成です。
- スタンバイ型系切り替え機能を適用するユニットは，相互系切り替え構成でクライアント接続用の IP アドレスだけを引き継ぐ構成です。また，ユーザサーバホットスタンバイを適用します。
- 1：1 スタンバイレス型系切り替え機能を適用するユニットは相互代替構成です。

なお，定義例はシステム共通定義及びユニット制御情報定義の関連のあるオペランドについてだけ説明しています。

注

- 網掛けが現用系のユニットです。
- (　) 内にはサーバ名を表記しています。
- バックエンドサーバ (bes1) は正規 BES であると同時に，バックエンドサーバ (bes2) の代替 BES です。
- バックエンドサーバ (bes2) は正規 BES であると同時に，バックエンドサーバ (bes1) の代替 BES です。

(1)　システム共通定義

```
        :
set pd_name_port = 5001                                          1
        :
set pd_ha = use                                                  2
set pd_ha_ipaddr_inherit = N
set pd_ha_switch_timeout = Y
set pd_ha_transaction = queuing
set pd_ha_trn_queuing_wait_time = 240
set pd_ha_trn_restart_retry_time = 90
        :
pdunit -x host1 -u UNT1 -d "C:¥HiRDB_A" -c host1A -p 5001        3
pdunit -x host2 -u UNT2 -d "C:¥HiRDB_B" -c host2A -p 5002
pdunit -x host3 -u UNT3 -d "C:¥HiRDB_A" -p 5003
pdunit -x host4 -u UNT4 -d "C:¥HiRDB_A" -p 5003
pdstart -t MGR -u UNT1                                           4
pdstart -t FES -s fes1 -u UNT1
pdstart -t DIC -s dic  -u UNT2
pdstart -t BES -s bes1 -u UNT3 -c bes2                           5
pdstart -t BES -s bes2 -u UNT4 -c bes1
        :
```

［説明］

1. HiRDB のポート番号を指定します。
2. 系切り替え機能に関するオペランドの指定です。

3. HiRDB/パラレルサーバのユニット構成を指定します。
   -x：ホスト名を指定します。
   -u：ユニット識別子を指定します。
   -d：HiRDB 運用ディレクトリ名を指定します。
   -c：予備系のホスト名を指定します。
   -p：ユニットのポート番号を指定します。
4. HiRDB/パラレルサーバのサーバ構成を指定します。
   -t：サーバの種別を指定します。
   -s：サーバ名を指定します。
   -u：ユニット識別子を指定します。
5. HiRDB/パラレルサーバのサーバ構成を指定します。
   -t：サーバの種別を指定します。
   -s：正規 BES 名を指定します。
   -u：ユニット識別子を指定します。
   -c：代替 BES 名を指定します。

(2)　UNT1～UNT2 のユニット制御情報定義

```
          :
set pd_ha_acttype = server                                    1
set pd_ha_server_process_standby = Y                          2
          :
```

〔説明〕

1. 系切り替え機能をサーバモードで運用することを指定します。
2. ユーザサーバホットスタンバイを使用することを指定します。

(3)　UNT3～UNT4 のユニット制御情報定義

```
          :
set pd_ha_acttype = server                                    1
set pd_ha_agent = server                                      2
          :
```

〔説明〕

1. 系切り替え機能をサーバモードで運用することを指定します。
2. 1：1 スタンバイレス型系切り替え機能を使用することを指定します。

付録 B.6　HiRDB External Data Access 機能使用時

HiRDB External Data Access 機能使用時の HiRDB/パラレルサーバのシステム定義を作成します。なお，定義例は HiRDB External Data Access 機能に関連のあるオペランドについてだけ説明しています。

(1)　システム共通定義

```
        :
set pd_max_foreign_server = 2
        :
```

〔説明〕

外部サーバ数（最大値）を指定します。

(2)　フロントエンドサーバ定義（FES1）

```
        :
pdhubopt -s hirdb_v8_01 -f hubopt1
pdhubopt -s xdmrd_01 -f hubopt2
        :
```

〔説明〕

外部サーバに HiRDB Version 8（サーバ名：hirdb_v8_01），及び XDM/RD E2（サーバ名：xdmrd_01）を使用することを定義します。-f オプションに Hub 最適化情報定義ファイル名を指定します。

(3)　フロントエンドサーバ定義（FES2）

```
        :
pdhubopt -s hirdb_v8_01 -f hubopt1
pdhubopt -s xdmrd_01 -f hubopt2
        :
```

〔説明〕

外部サーバに HiRDB Version 8（サーバ名：hirdb_v8_01），及び XDM/RD E2（サーバ名：xdmrd_01）を使用することを定義します。-f オプションに Hub 最適化情報定義ファイル名を指定します。

(4)　バックエンドサーバ定義（BES1）

```
        :
set pd_foreign_server_libpath = "C:¥hirdb¥client¥lib"
        :
```

〔説明〕

外部サーバ（HiRDB Version 8）のクライアントライブラリの絶対パス名を指定します。

(5)　バックエンドサーバ定義（BES2）

```
        :
set pd_foreign_server_libpath = "C:¥hirdb¥client¥lib"
        :
```

〔説明〕

外部サーバ（XDM/RD E2）のクライアントライブラリの絶対パス名を指定します。

(6)　外部サーバ情報定義（HiRDB Version 8 用）

```
putenv PDHOST host01                                    1
putenv PDNAMEPORT 20000                                 2
```

〔説明〕

1. 外部サーバ（HiRDB Version 8）のホスト名を指定します。
2. 外部サーバ（HiRDB Version 8）のポート番号を指定します。

(7)　外部サーバ情報定義（XDM/RD E2 用）

```
putenv PDHOST host02                                    1
putenv PDNAMEPORT 20100                                 2
putenv PDCLTRDNODE rd_node1                             3
```

〔説明〕

1. DB コネクションサーバのホスト名を指定します。
2. DB コネクションサーバのポート番号を指定します。
3. 外部サーバ（XDM/RD E2）の RD ノード名を指定します。

(8)　Hub 最適化情報定義（HiRDB Version 8 用）

Hub 最適化情報定義ファイル名：hubopt1

```
set pd_hub_opt_on_cnd =1
set pd_hub_opt_joined_table =1
set pd_hub_opt_set_func = 1
set pd_hub_opt_case = 1
set pd_hub_opt_like = 1
set pd_hub_opt_grouping = 1
set pd_hub_opt_data_len = 255
set pd_hub_opt_abs = 1
set pd_hub_opt_date = 1
set pd_hub_opt_time = 1
set pd_hub_opt_digits = 1
set pd_hub_opt_length = 1
set pd_hub_opt_lower_upper_type = 1
set pd_hub_opt_mod_div_type = 1
set pd_hub_opt_substr = 1
set pd_hub_opt_num = 1
set pd_hub_opt_datetime = 1
set pd_hub_opt_datetime_op = 1
set pd_hub_opt_trailing_spc = 1
set pd_hub_opt_in_value_num = 255※
set pd_hub_opt_nullable = 1
set pd_hub_opt_use_zero_string = 1
set pd_hub_opt_nchar = 1
set pd_hub_opt_nest_scalar = 0
set pd_hub_opt_float = 2
set pd_hub_opt_table_num = 64
set pd_hub_opt_time_24hour = 1
```

注※

HiRDB のバージョンが 08-04 以降の場合は，30000 が推奨値となります。

(9)　Hub 最適化情報定義（XDM/RD E2 用）

Hub 最適化情報定義ファイル名：hubopt2

```
set pd_hub_opt_on_cnd =1
set pd_hub_opt_joined_table =1
set pd_hub_opt_set_func = 3
set pd_hub_opt_case = 1
set pd_hub_opt_like = 4
set pd_hub_opt_grouping = 4
set pd_hub_opt_col_len = 255
set pd_hub_opt_abs = 1
set pd_hub_opt_date = 1
set pd_hub_opt_time = 1
set pd_hub_opt_digits = 1
set pd_hub_opt_length = 3
set pd_hub_opt_lower_upper_type = 1
set pd_hub_opt_mod_div_type = 2
set pd_hub_opt_substr = 3
set pd_hub_opt_num = 1
set pd_hub_opt_datetime = 1
set pd_hub_opt_datetime_op = 1
set pd_hub_opt_trailing_spc = 1
set pd_hub_opt_in_value_num = 255
set pd_hub_opt_nullable = 1
set pd_hub_opt_use_zero_string = 1
set pd_hub_opt_nchar = 2
set pd_hub_opt_nest_scalar = 0
set pd_hub_opt_float = 2
set pd_hub_opt_table_num = 64
set pd_hub_opt_time_24hour = 1
```

付録 C　オペランド指定値の見積もり式

ここでは，システム定義の各オペランドの指定値の見積もり式について説明します。説明する項目は次のとおりです。

1. 統計ログファイル（pd_stj_file_size）の見積もり式
2. SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式
3. ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式
4. 表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式
5. 1 サーバ当たりの UNTIL DISCONNECT 指定排他の表と RD エリアの総和（pd_lck_until_disconnect_cnt）の見積もり式
6. ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式

付録 C.1　統計ログファイル（pd_stj_file_size）の見積もり式

出力される統計ログ量は統計情報ごとに異なります。統計情報ごとの統計ログ量を(1)以降に示します。pd_stj_file_size オペランドには，(1)以降で求めた統計ログ量の合計値以上の値を指定してください。

注意事項

計算式の単位はバイトのため，合計後，キロバイトに変換してください。

(1)　システムの稼働に関する統計情報（sys）

pdstbegin オペランド又は pdstbegin コマンドに指定したオプションによって異なります。

-a 及び-s オプションを指定しなかった場合

統計ログ量＝ 2412×（↓ a ÷ b ↓＋統計情報出力時間中の pdstjsync コマンド実行回数）（単位：バイト）

-a オプションを指定した場合

統計ログ量＝（1836× c ＋ 2412）×（↓ a ÷ b ↓＋統計情報出力時間中の pdstjsync コマンド実行回数）（単位：バイト）

-s オプションを指定した場合

統計ログ量＝（1836× d ＋ 2412）×（↓ a ÷ b ↓＋統計情報出力時間中の pdstjsync コマンド実行回数）（単位：バイト）

a：統計情報出力時間　（単位：分）

b：統計情報出力間隔　（単位：分）

pdstbegin オペランド又は pdstbegin コマンドの-m オプションで指定した時間間隔です。

c：ユニット内のサーバ数

システムマネジャは数に入れません。

d：pdstbegin オペランド又は pdstbegin コマンドの-s オプションで指定したサーバ数

(2)　UAP に関する統計情報（uap）

〈計算式〉

統計ログ量＝ 1040×統計情報の取得中に実行する UAP の数　（単位：バイト）

実際に実行する UAP の延べ数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する UAP 数が不明確な場合，UAP 数は次の式から求められます。

- UAP 数＝単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間

例えば，30 分に 10 個の UAP を実行し，統計情報を 60 分ほど収集した場合は，10×60÷30=20 となります。

(3)　SQL に関する統計情報（sql）

〈計算式〉

統計ログ量＝ 728×統計情報の取得中に実行する SQL の数※　（単位：バイト）

実際に実行する SQL の延べ数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する SQL 数が不明確な場合，SQL 数は次の式から求められます。

SQL 数※＝ UAP が実行する SQL 文の平均的な数×
（単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間）

注※

ストアドプロシジャ又はストアドファンクションを実行する場合，手続き又は関数中に記述する SQL 文数も数えてください。

(4)　グローバルバッファに関する統計情報（buf）

〈計算式〉

統計ログ量＝ 380× a × b　（単位：バイト）

a：pdbuffer オペランドの指定数

b：統計情報取得中のシンクポイントダンプ取得回数

統計情報収集時間内に取得するシンクポイントダンプ取得回数は，次の計算式から算出してください。

↑統計情報収集時間内に出力するシステムログ量
÷(各サーバの定義で指定した pd_log_max_data_size オペランドの指定値の合計値÷3)
×各サーバの定義で指定した pd_log_sdinterval オペランドの指定値の合計値↑

システムログ量については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

(5)　データベース操作に関する HiRDB ファイルの統計情報（fil）

〈計算式〉

統計ログ量＝ 428× a × b　（単位：バイト）

a：データベース初期設定ユティリティの制御文で指定した HiRDB ファイル数

b：統計情報取得中のシンクポイントダンプ取得回数

統計情報収集時間内に取得するシンクポイントダンプ取得回数は，次の計算式から算出してください。

↑統計情報収集時間内に出力するシステムログ量
÷(各サーバの定義で指定した pd_log_max_data_size オペランドの指定値の合計値÷3)
×各サーバの定義で指定した pd_log_sdinterval オペランドの指定値の合計値↑

システムログ量については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

(6)　デファードライト処理に関する統計情報（dfw）

〈計算式〉

統計ログ量＝384×ａ×2＋（384×ｂ×ｃ）÷（ｄ×ｅ×ｆ）（単位：バイト）

ａ：統計情報取得中のシンクポイントダンプ取得回数

統計情報収集時間内に取得するシンクポイントダンプ取得回数は，次の計算式から算出してください。

↑統計情報収集時間内に出力するシステムログ量

÷(各サーバの定義で指定した pd_log_max_data_size オペランドの指定値の合計値÷3)

×各サーバの定義で指定した pd_log_sdinterval オペランドの指定値の合計値↑

システムログ量については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

ｂ：1 トランザクション当たりの平均更新ページ数

ｃ：統計情報時間内に実行するトランザクション数

ｄ：pdbuffer オペランドの－ｎオプションで指定したバッファ面数の総数

ｅ：全トランザクションに占めるデータベースを更新するトランザクションの比率

ｆ：pdbuffer オペランドで指定したデファードライトトリガ時の更新ページ出力比率

(7)　インデクスに関する統計情報（idx）

〈計算式〉

統計ログ量＝3768×↑ａ÷128↑×ｂ　（単位：バイト）

ａ：分割されたインデクスの総数

分割されたインデクスの総数は, 定義系 SQL の各 CREATE INDEX の IN RD エリア名で指定する RD エリア名の数の合計値です。

ｂ：統計情報取得中のシンクポイントダンプ取得回数

統計情報収集時間内に取得するシンクポイントダンプ取得回数は，次の計算式から算出してください。

↑統計情報収集時間内に出力するシステムログ量

÷(各サーバの定義で指定した pd_log_max_data_size オペランドの指定値の合計値÷3)

×各サーバの定義で指定した pd_log_sdinterval オペランドの指定値の合計値↑

システムログ量については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」を参照してください。

(8)　SQL 静的最適化に関する統計情報（sop）

〈計算式〉

統計ログ量＝92×ａ　（単位：バイト）

ａ：統計情報の取得中に実行する SQL 文で SQL オブジェクト生成処理を行う数

ここでいう SQL 文とは，PREPARE 文，EXECUTE IMMEDIATE 文，及び静的 SQL 文のことです。

実際に SQL オブジェクト生成処理数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する SQL オブジェクト生成処理数が不明確な場合, SQL オブジェクト生成処理数は次の式から求められます。

SQL オブジェクト生成処理数＝

UAP が実行する SQL 文の平均的な数×SQL オブジェクトキャッシュのミスヒット率×（単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間）

SQL オブジェクトキャッシュのミスヒット率はシステムに関する統計情報で調べてください。

(9)　SQL 動的最適化に関する統計情報（dop）

〈計算式〉

統計ログ量＝

$$\sum_{i=1}^{a}\left\{\begin{array}{l}28+16\times\text{SQL中の表数}\times 2\\ +12\times\left(1+\sum_{j=1}^{b}\text{表cを定義しているRDエリア数}\times 2\right.\end{array}\right\}$$

（単位：バイト）

a：実行する SQL 文数

b：c 番目の SQL 文中の表数

(10)　SQL オブジェクト実行に関する統計情報（pcd）

●HiRDB/シングルサーバの場合

〈計算式〉

統計ログ量＝ 388×｛(a ＋ b ＋ c ＋ d ＋ e ＋ f) ＋ g ＋ h｝（単位：バイト）

●HiRDB/パラレルサーバの場合

〈計算式〉

統計ログ量＝ 388×｛(a ＋ b ＋ c ＋ d ＋ e ＋ f) ×i ＋ g ＋ h｝（単位：バイト）

a：OPEN 文の実行数

b：CLOSE 文の実行数

c：INSERT 文の実行数

d：DELETE 文の実行数

e：ASSIGN LIST 文の実行数

f：UPDATE 文の実行数

g：FETCH 文の実行数

h：DESCRIBE 文の実行数

i：次に示す計算式の値

$$\sum_{k=1}^{n}\left\{1+\sum_{m=1}^{Tm}\text{表mを定義しているRDエリア数}\times 2\right\}$$

n：実行する SQL 文数

Tm：m 番目の SQL 文中の表数

(11)　SQL 文の履歴に関する統計情報（sqh）

〈計算式〉

統計ログ量＝（728 ＋ a）× b　（単位：バイト）

a：SQL 文の平均 SQL 長

b：統計情報の取得中に実行する SQL の数

実際に実行する SQL の延べ数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する SQL 数が不明確な場合，SQL 数は次の式から求められます。

SQL 数※＝ UAP が実行する SQL 文の平均的な数×
（単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間）

注※

ストアドプロシジャ又はストアドファンクションを実行する場合，手続き又は関数中に記述する SQL 文数も数えてください。

(12)　SQL オブジェクト転送に関する統計情報（obj）

〈計算式〉

統計ログ量＝ 300×統計情報の取得中に実行する SQL の数　（単位：バイト）

実際に実行する SQL の延べ数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する SQL 数が不明確な場合，SQL 数は次の式から求められます。

SQL 数※＝ UAP が実行する SQL 文の平均的な数×
（単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間）

注※

ストアドプロシジャ又はストアドファンクションを実行する場合，手続き又は関数中に記述する SQL 文数も数えてください。

(13)　外部サーバの稼働に関する統計情報（fsv）

〈計算式〉

統計ログ量＝ 188×a×b　（単位：バイト）

a：外部サーバ情報定義をした外部サーバ数

b：統計情報取得中のトランザクション数

(14)　外部サーバの利用状況に関する統計情報（hba）

〈計算式〉

統計ログ量＝ 404×a　（単位：バイト）

a：統計情報取得中に実行する外部サーバに対する SQL 数

SQL 数＝ UAP が実行する外部サーバに対する SQL 文の平均的な数×（単位時間当たりの UAP 実行数×統計情報取得時間÷単位時間）

付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式

計算式で使用する変数については(3)を参照してください。

(1)　1SQL 文の SQL オブジェクト長の計算式

1SQL 文の SQL オブジェクト長は次に示す計算式から求めます。

〈計算式〉

1SQL 文の SQL オブジェクト長（単位：キロバイト）＝
↑｛
1840 ＋ 46×RCN ＋ 298×Si ＋ 20×Pi ＋ 1138×Ti ＋ 76×Ti×Di ＋ 80×Ci ＋ 40×Ii ＋ 534×Wi ＋ 20×Ki ＋ Li
＋ 8×TCi ＋ 656×Di ＋ 48×nFF ＋ 100×nFP ＋ 148×nFC ＋ 696×nPFF ＋ 16×（nAT ＋ nPAT）＋ 20×nCAT
＋ 28×（nAF ＋ nCAF）＋ 20×（nAA ＋ nPAA ＋ nCAA）＋ 1057×nSPA ＋ 120×nSPP ＋ 287×nSFF ＋ 8×nSFP
＋ 813×nJFC ＋ 20×nJFP
〔＋ 60×nFT ＋ nFTS ＋ 32×nFTQ〕※1
〔＋ 1057×nTR ＋ 120×（nTSN ＋ nTSO）＋ 20×（nTCN ＋ nTCO)〕※2
〔＋ 760 ＋ 376×RCC ＋ 1880×RCT〕※3
〔＋ 32×Si ＋ 16〕※4
〔＋↑（42×SiT）＋｛52 ＋ 152×（SiTA ＋ SiSA ＋ SiNA）×（SiT ＋ SiS ＋ SiN)｝↑〕※5
｝÷1024↑

注※1　外部表を指定する場合に加算する計算式です。

注※2　トリガを使用する場合に加算する計算式です。

注※3　参照制約を使用する場合に加算する計算式です。

注※4　列名記述領域長の計算式です。動的 SQL の場合に加算します。

注※5　型名記述領域長の計算式です。動的 SQL の場合に加算します。

注意事項

- ストアドプロシジャ又はストアドファンクションを使用する場合は，手続き又は関数中に記述する SQL 文も計算対象にしてください。
- トリガを使用する場合はトリガ中に記述するトリガ SQL 文も計算対象にしてください。
- ストアドプロシジャ，ストアドファンクション，又はトリガを使用する場合は，ストアドプロシジャ，ストアドファンクション，又はトリガごとにルーチン制御用オブジェクト長を計算して SQL オブジェクト用バッファ長に加算してください。1 ルーチンのルーチン制御用オブジェクト長の計算式を(2)に示します。

(2)　1 ルーチンのルーチン制御用オブジェクト長の計算式

(a)　ユーザが定義する場合

ストアドプロシジャ，ストアドファンクション，又はトリガを定義した場合の，1 ルーチンのルーチン制御用オブジェクト長は次に示す計算式から求めます。

〈計算式〉

1 ルーチンのルーチン制御用オブジェクト長（単位：キロバイト）＝
↑｛
600 ＋ 28×sRi ＋ 32×（sRUi ＋ sDi）＋ 56×sSXi ＋ sCUi ＋ sSi ＋ sPi ＋ sLA ＋ sKi ＋ sL ＋ 80×sWi ＋ 24×sCM
＋ 32×sCCR ＋ 2×sDCR ＋ 60×sCHD ＋ 72×sDHD ＋ 64×sHCN ＋ 8×sCHD×sHCN ＋ 48×nRFF ＋ 100×

nRFP + 148×nRFC + 200×nPRFF + 8×nPRFP + 196×nPA + 64×nPP + 36×nPPI + 20×nPPO + 200×nPPA + 8×nPPP + 20×nAR + 48×nARA + 16×nRPAT + 20×nCAT + 28×（nRPAF + nRCAF）+ 20×（nRPAA + nRCAA）+ 287×nRSFF + 8×nRSFP + 813×nPJA + 20×nPJP + 813×nRJFC + 20×nRJFP
〔+ 28×（nTSN×2 + nTSO)〕※
｝÷1024↑

注※　トリガを使用する場合に加算する計算式です。

(b)　HiRDB が自動的に作成する場合

表定義時，参照動作に CASCADE を指定した場合，HiRDB が制約制御のためにトリガを作成したときの 1 ルーチンのルーチン制御用オブジェクト長は次に示す計算式から求めます。

〈計算式〉

1 ルーチンのルーチン制御用オブジェクト長（単位：キロバイト）＝
↑｛
240 + 608×RCC +（5120 + 100×RDi + 256×RIi）×RCP×RCT
｝÷1024↑

(3)　計算式で使用する変数

変数名	説明
RCN	SQL オブジェクトで使用する表，及びインデクスの合計数
Si	SQL 文中にある検索項目数（SQL 文で指定する列がインデクス列の場合はその列数）
Pi	SQL 文中にある埋込み変数又はパラメタ数
Ti	SQL 文中にある表名数
Ci	SQL 文中にある列名数 表定義時に次に示す定義をした列数も加算してください（SQL 文中にない列であっても加算します）。 • 列定義に DEFAULT を指定している列 • 非ナル値制約指定に WITH DEFAULT を指定している列
TCi	SQL 文中にある表の構成列数
Wi	SQL 文中にある論理演算子数※1
Ki	SQL 文中にある定数の数※1
Li	SQL 文中にある定数の合計長※1（単位：バイト）
Ii	SQL 文実行時に使用するインデクス数（SQL 文で指定する表のうち，検索条件に指定するインデクス数）
Di	SQL 文中の表に定義された格納条件の総数（マトリクス分割表は 2 倍する）
SiT	SQL 文中にある選択式の抽象データ型数
SiS	SQL 文中にある選択式の抽象データ型のスーパタイプ数
SiN	SQL 文中にある選択式のサブタイプである抽象データ型のスーパタイプの総数
SiTA	SQL 文中にある選択式の抽象データ型の属性数

変数名	説明
SiSA	SQL 文中にある選択式の抽象データ型のスーパタイプの属性数
SiNA	SQL 文中にある選択式のサブタイプである抽象データ型のコンポネント指定総数
nSPA	SQL 文中にある手続き文の呼び出し数
nSPP	SQL 文中にある手続き文の引数の総数
nFF	SQL 文中にある関数の呼び出し数※1
nFP	SQL 文中にある関数の引数の総数※1
nFC	SQL 文中にある関数の総関数定義候補数（関数呼出し数 nFF に，引数が抽象データ型の各関数呼出しに対して，サブタイプを引数とする関数定義数を加算する）
nPFF	SQL オブジェクトが使用するプラグイン関数呼出し数（SQL 文中にあるプラグイン関数呼出し数＋SELECT の場合は 1，INSERT，UPDATE，DELETE の場合は 6）
nSFF	SQL 文中にあるシステム定義スカラ関数の呼び出し数※1
nSFP	SQL 文中にあるシステム定義スカラ関数の引数の総数※1
nJFC	SQL 文中にある外部 Java 関数の呼び出し数
nJFP	SQL 文中にある外部 Java 関数の引数の総数
nAT	SQL 文中にあるコンポネント指定で使用する抽象データ型数（スーパタイプ，抽象データ型属性によって現れる抽象データ型を除く）
nAA	SQL 文中にあるコンポネント指定で使用する抽象データ型数（スーパタイプ，抽象データ型属性によって現れる抽象データ型を含む）
nAF	SQL 文中にあるコンポネント指定で使用する属性総数
nPAT	SQL オブジェクトが使用するプラグイン関数の引数で使用する抽象データ型数（スーパタイプ，抽象データ型属性によって現れる抽象データ型を除く）
nPAA	SQL オブジェクトが使用するプラグイン関数の引数で使用する抽象データ型数（スーパタイプ，サブタイプを含む）
nCAT	SQL 文中にあるコンストラクタ関数の呼び出し数
nCAA	SQL 文中にあるコンストラクタ関数の抽象データ型数（スーパタイプを含む）
nCAF	SQL 文中にあるコンストラクタ関数の抽象データ型の属性総数
nFT	SQL 文中にある外部表数
nFTS	外部表を検索する SQL 文の長さ※2
nFTQ	外部表を検索する SQL 文中の埋込み変数又は？パラメタの数※2
nTR	SQL 文の実行によって起動されるトリガ数
nTSN	SQL 文の実行によって起動される各トリガのトリガ SQL 文中の新値相関名によって修飾された列の総数
nTSO	SQL 文の実行によって起動される各トリガのトリガ SQL 文中の旧値相関名によって修飾された列の総数
nTCN	SQL 文の実行によって起動される各トリガのトリガ動作条件中の新値相関名によって修飾された列の総数

変数名	説明
nTCO	SQL 文の実行によって起動される各トリガのトリガ動作条件中の旧値相関名によって修飾された列の総数
RCC	SQL 文中で，更新対象の表を参照する表の外部キーの構成列数と，主キーの構成列数の総数
RCT	SQL 文中で，更新対象の表を参照する表，及び更新対象の表が参照する表の総数
RCP	参照表定義時に，参照動作に指定した CASCADE の総数
RIi	参照表定義時，参照指定をする被参照表に定義されたインデクスの総数
RDi	参照表定義時，参照指定をする被参照表に定義された分割格納条件の総数（マトリクス分割表の場合は，2 倍する）
sRi	手続き及び関数中の SQL パラメタ数（INOUT 指定の SQL パラメタ数は 2 倍する）
sRUi	手続き及び関数中の SQL パラメタの総数（又は，トリガ定義中のトリガ SQL 文中の新旧値相関名によって修飾された列の総数）
sDi	手続き，関数，及びトリガ SQL 文中の SQL 変数（declare）の総数
sSXi	手続き，関数，及びトリガ SQL 文中の SQLCODE，SQLCOUNT 変数の総数
sCUi	手続き，関数，及びトリガ SQL 文中の CURRENT_TIME，CURRENT_DATE 定数の総数
sSi	手続き及びトリガ SQL 文中の操作系 SQL の数（カーソル宣言を除く SQL 文：OPEN，FETCH，CLOSE，UPDATE，DELETE，INSERT 文など）
sPi	手続き，関数，及びトリガ SQL 文中のルーチン制御 SQL 数（BEGIN，SET，IF，ELSEIF，WHILE など）
sLA	手続き，関数，及びトリガ SQL 文中のラベル数
sKi	手続き，関数，及びトリガ SQL 文中の定数の数（手続き及びトリガ SQL 文中に記述された操作系 SQL 文の定数を除く）
sL	手続き，関数，及びトリガ SQL 文中の定数の長さの合計（手続き及びトリガ SQL 文中に記述された操作系 SQL 文の定数を除く）
sWi	手続き，関数，及びトリガ SQL 文中の条件述語数
sCM	手続き，関数，及びトリガ SQL 文中の複合文の数
sCCR	手続き及びトリガ SQL 文のカーソル宣言を記述した複合文の数
sDCR	手続き及びトリガ SQL 文のカーソル宣言の数
sCHD	手続き，関数，及びトリガ SQL 文のハンドラ宣言を記述した複合文の数
sDHD	手続き，関数，及びトリガ SQL 文のハンドラ宣言の数
sHCN	手続き，関数，及びトリガ SQL 文のハンドラ宣言中に記述された条件値の数
nRFF	ルーチン中にある関数呼出し数
nRFP	ルーチン中にある関数の引数の総数
nRFC	ルーチン中にある関数の関数定義候補の総数（関数呼出し数 nFF に，引数が抽象データ型の各関数呼出しに対して，サブタイプを引数とする関数定義数を加算する）
nPRFF	ルーチンの SQL オブジェクトが使用するプラグイン関数呼出し数

変数名	説明
nPRFP	ルーチンの SQL オブジェクトが使用するプラグイン関数のプラグインパラメタ総数
nPA	ルーチン中にある手続き呼び出し数
nPP	ルーチン中にある手続きのパラメタ総数
nPPI	ルーチン中にある手続きの入力パラメタ総数（入出力パラメタを含む）
nPPO	ルーチン中にある手続きの出力パラメタ総数（入出力パラメタを含む）
nPPA	ルーチンの SQL オブジェクトが使用するプラグイン手続き呼び出し数
nPPP	ルーチンの SQL オブジェクトが使用するプラグイン手続きのプラグインパラメタ総数
nRSFF	ルーチン中にあるシステム定義スカラ関数の呼び出し数
nRSFP	ルーチン中にあるシステム定義スカラ関数の引数の総数
nPJA	ルーチン中にある外部 Java 手続きの呼び出し数
nPJP	ルーチン中にある外部 Java 手続きの引数の総数
nRJFC	ルーチン中にある外部 Java 関数の呼び出し数
nRJFP	ルーチン中にある外部 Java 関数の引数の総数
nAR	ルーチン中にあるコンポネント指定で使用する抽象データ型数（スーパタイプ，抽象データ型属性によって現れる抽象データ型を除く）
nARA	ルーチン中にあるコンポネント指定で使用する属性総数
nRPAT	ルーチンの SQL オブジェクトが使用するプラグインルーチンのパラメタで使用する抽象データ型数（ただし，スーパタイプ，抽象データ型属性の抽象データ型を除く）
nRPAA	ルーチンの SQL オブジェクトが使用するプラグインルーチンのパラメタで使用する抽象データ型数（スーパタイプを含む）
nRPAF	ルーチンの SQL オブジェクトが使用するプラグインルーチンのパラメタで使用する抽象データ型の属性総数
nRCAT	ルーチン中にあるコンストラクタ関数の呼び出し数
nRCAA	ルーチン中にあるコンストラクタ関数の抽象データ型数（スーパタイプを含む）
nRCAF	ルーチン中にあるコンストラクタ関数の抽象データ型の属性総数

注※ 1

トリガを使用する場合は，SQL 文の実行によって起動される各トリガのトリガ動作条件についても数える必要があります。

注※ 2

外部表を検索する SQL 文はアクセスパス表示ユティリティ（pdvwopt コマンド）で確認できます。

付録 C.3　ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式

(1)　一つのビュー表当たりのビュー解析情報用バッファ長の計算式

一つのビュー表当たりのビュー解析情報用バッファ長は，次に示す計算式から求めます。計算式で使用する変数については，(2)及び(3)を参照してください。

〈32 ビットモードの HiRDB の計算式〉

---

1 ビュー表当たりのビュー解析情報用バッファ長（単位：キロバイト）＝
　↑1 ビュー表当たりのビュー解析情報長÷1024↑

1 ビュー表当たりのビュー解析情報長（単位：バイト）＝
　↑（1024 ＋ LCNST ＋ 15）÷16↑×16
　＋↑（LPTREE ＋ 15）÷16↑×16
　＋↑（28×NINCC ＋ 15）÷16↑×16
　＋↑（28×NINCP ＋ 15）÷16↑×16
　＋↑（24 ＋ 512×NTBL ＋ 40×（NTBL ＋ NDTBL）
　＋ 128×NCLM ＋ 15）÷16↑×16
　＋↑（16×MAX（↑NADTL÷50↑，1）
　＋ 200×NADTL）÷16↑×16
　＋↑（16×MAX（↑NATTL÷50↑，1）
　＋ 144×NATTL）÷16↑×16
　＋ 16 ＋ 16×（3 ＋ NINCC ＋ NINCP）

〈64 ビットモードの HiRDB の計算式〉

---

1 ビュー表当たりのビュー解析情報用バッファ長（単位：キロバイト）＝
　↑1 ビュー表当たりのビュー解析情報長÷1024↑

1 ビュー表当たりのビュー解析情報長（単位：バイト）＝
　↑（1600 ＋ LCNST ＋ 15）÷16↑×16
　＋↑（LPTREE ＋ 15）÷16↑×16
　＋↑（40×NINCC ＋ 15）÷16↑×16
　＋↑（40×NINCP ＋ 15）÷16↑×16
　＋↑（32 ＋ 768×NTBL ＋ 48×（NTBL ＋ NDTBL）
　＋ 184×NCLM ＋ 15）÷16↑×16
　＋↑（20×MAX（↑NADTL÷50↑，1）
　＋ 224×NADTL）÷16↑×16
　＋↑（20×MAX（↑NATTL÷50↑，1）
　＋ 160×NATTL）÷16↑×16
　＋ 16 ＋ 24×（3 ＋ NINCC ＋ NINCP）

注意事項

ビュー定義に，内部導出表を生成するビュー表を含む場合，そのビュー表のビュー解析情報用バッファ長の計算式も加算してください。

内部導出表を生成しないビュー表の場合は，そのビュー表を介さずに直接指定した SQL 文で計算した値を加算してください。

(2)　LPTREE の計算式

LPTREE は，次に示す計算式から求めます。計算式で使用する変数については，(3)を参照してください。

〈32 ビットモードの HiRDB の計算式〉

```
LPTREE（単位：バイト）＝
    276×NQRY
    ＋12×（2×NQRY＋NSBQ＋NSTOP）
    ＋12×（2×NQRY＋NSLST＋NSLAS）
    ＋12×（4×NTBL＋4×NJTBL＋5×NDTBL＋NDCLM）
    ＋12×（NWHRC＋NGPHV）
    ＋12×（NVLCM＋NEXCM＋NNLCM
    ＋3×NRANG＋3×NLKCM＋3×NSMCM＋NETCM）
    ＋12×NRVCL
    ＋12×NTVCL
    ＋12×NLGEX
    ＋12×（NCNST＋NSREG＋NPRCS）
    ＋12×4×NCSCV
    ＋12×（NARTH＋NCNCT）
    ＋12×NSFNC
    ＋12×NLBLD
    ＋12×（3×NCLM）
    ＋12×（NCASE＋NWHEN）
    ＋12×2×（NSCLF＋2×NCSSP）
    ＋12×（2×NPOS＋3×NDTV）
    ＋12×（NEXTR＋2×NSBST＋3×NVALU＋NBTEX＋NCAST）
    ＋12×（4×NFCSP＋NFPRM）
    ＋12×（3×NATTL＋NATNM）
    ＋28×NOPTL＋20×NOPTJ＋32×NOPTT＋20×NOPTIX
    ＋8×NSLAS
    ＋120×NJTBL
    ＋480×NCSET
    ＋84×NSPDT
    ＋72×（NSCLF＋NCASE＋NCSSP×2）
    ＋（144＋20＋32）×NFCSP＋（300＋136）×NRTNL
    ＋260×（NEXRTN＋NCLASS＋NJAR）＋32×NPVOW
    ＋140×NPLGL＋172×NPPRL
    ＋（260＋257＋257）×NPLGL
    ＋32×NCUD
    ＋512×NDTBL
    ＋40×NDTBL
    ＋128×NDCLM
    ＋182
    ＋376
    ＋1384
    ＋356×NVCLM
    ＋24×NVCSC
```

〈64 ビットモードの HiRDB の計算式〉

```
LPTREE（単位：バイト）＝
    496×NQRY
    ＋24×（2×NQRY＋NSBQ＋NSTOP）
    ＋24×（2×NQRY＋NSLST＋NSLAS）
    ＋24×（4×NTBL＋4×NJTBL＋5×NDTBL＋NDCLM）
    ＋24×（NWHRC＋NGPHV）
    ＋24×（NVLCM＋NEXCM＋NNLCM
    ＋3×NRANG＋3×NLKCM＋3×NSMCM＋NETCM）
    ＋24×NRVCL
    ＋24×NTVCL
    ＋24×NLGEX
    ＋24×（NCNST＋NSREG＋NPRCS）
    ＋24×4×NCSCV
    ＋24×（NARTH＋NCNCT）
    ＋24×NSFNC
    ＋24×NLBLD
    ＋24×（3×NCLM）
    ＋24×（NCASE＋NWHEN）
    ＋24×2×（NSCLF＋2×NCSSP）
    ＋24×（2×NPOS＋3×NDTV）
    ＋24×（NEXTR＋2×NSBST＋3×NVALU＋NBTEX＋NCAST）
    ＋24×（4×NFCSP＋NFPRM）
    ＋24×（3×NATTL＋NATNM）
    ＋56×NOPTL＋40×NOPTJ＋64×NOPTT＋40×NOPTIX
    ＋16×NSLAS
    ＋208×NJTBL
    ＋480×NCSET
    ＋96×NSPDT
    ＋96×（NSCLF＋NCASE＋NCSSP×2）
    ＋（176＋24＋40）×NFCSP＋（328＋144）×NRTNL
    ＋260×（NEXRTN＋NCLASS＋NJAR）＋32×NPVOW
    ＋168×NPLGL＋184×NPPRL
    ＋（260＋264＋257）×NPLGL
    ＋40×NCUD
    ＋768×NDTBL
    ＋48×NDTBL
    ＋184×NDCLM
    ＋226
    ＋568
    ＋1496
    ＋480×NVCLM
    ＋32×NVCSC
```

(3)　計算式で使用する変数

変数名	説明
LCNST	次に示す式の演算結果の定数の合計長※1，※2，※3 • ビュー定義中にある定数の合計長＋ビュー定義中にある文字集合変換 • 対象となる定数の変換後の合計長＋ビュー定義中にある数値文字変換 • 対象となる定数の変換後の合計長＋ビュー定義中にあるスカラ演算
NINCC	• LCNST ＞ 11516 バイトの場合 32 ビットモード：↑（LCNST-11516）÷4020↑ 64 ビットモード：↑（LCNST-11516）÷4024↑ • LCNST≦11516 バイトの場合 0
NINCP	• LTREE ＞ 11516 バイトの場合 32 ビットモード：↑（LTREE－11516）÷4020↑ 64 ビットモード：↑（LTREE－11516）÷4024↑ • LCNST≦11516 バイトの場合 0
NQRY	ビュー定義中にある問合せ指定の数
NSBQ	ビュー定義中にある副問合せの数
NSTOP	ビュー定義中にある集合演算の数
NSLST	ビュー定義中にある全問合せ指定の選択項目数の合計※4
NWHRC	ビュー定義中にある WHERE 句の数
NGPHV	ビュー定義中にある GROUP BY 句の数＋ビュー定義中にある HAVING 句の数
NVLCM	ビュー定義中にある比較述語の数＋ビュー定義中にある限定述語の数
NEXCM	ビュー定義中にある IN 述語の数
NNLCM	ビュー定義中にある NULL 述語の数
NRANG	ビュー定義中にある BETWEEN 述語の数
NLKCM	ビュー定義中にある LIKE 述語と XLIKE 述語の数
NSMCM	ビュー定義中にある SIMILAR 述語の数
NETCM	ビュー定義中にある EXISTS 述語の数
NRVCL	ビュー定義中にある行値構成子の要素の数
NTVCL	ビュー定義中にある IN 述語右側の行値構成子の要素の数
NLGEX	ビュー定義中にある論理演算子の数
NCNST	ビュー定義中にある定数の延べ数＋ビュー定義中にあるパターン文字の延べ数
NSREG	ビュー定義中にある USER，CURRENT DATE，CURRENT TIME，又は CURRENT TIMESTAMP の総数

変数名	説明
NPRCS	ビュー定義中にある CURRENT TIMESTAMP の小数秒精度指定の数
NCSCV	ビュー定義中にある文字集合変換対象となる定数の数※2
NARTH	ビュー定義中にある四則演算，日付演算，及び時刻演算の総数
NCNCT	ビュー定義中にある連結演算の数
NLBLD	ビュー定義中にあるラベル付き間隔の数
NSFNC	ビュー定義中にある集合関数の数
NSCLF	ビュー定義中にあるシステム組込みスカラ関数の数
NFCSP	ビュー定義中にある関数呼出し，又はスカラ関数の数
NCLM	ビュー定義中にある列の延べ数
NCASE	ビュー定義中にある CASE 式の数＋ビュー定義中にある CASE 略式 NULLIF の数
NWHEN	ビュー定義中にある CASE 式の WHEN の数＋ 1
NCSSP	ビュー定義中にある CAST 指定の数
NPOS	ビュー定義中にあるシステム組込みスカラ関数 POSITION の数
NDTV	ビュー定義中にあるシステム組込みスカラ関数（DATE，TIME，TIMESTAMP，又は VERCHAR_FORMAT）の数
NEXTR	ビュー定義中にあるシステム組込みスカラ関数（YEAR，MONTH，DAY，HOUR，MINUTE，又は SECOND）の数
NSBST	ビュー定義中にあるシステム組込みスカラ関数 SUBSTR の数
NVALU	ビュー定義中にあるシステム組込みスカラ関数 VALUE の数＋ビュー定義中にある CASE 略式 COALESCE の数
NBTEX	ビュー定義中にあるシステム組込みスカラ関数 BIT_AND_TEST の数
NCAST	ビュー定義中にあるシステム組込みスカラ関数（INTEGER，又は DECIMAL）の数
NOPTL	ビュー定義中にある SQL 最適化指定の数
NOPTJ	ビュー定義中にある結合方式の SQL 最適化指定の数
NOPTT	ビュー定義中にある表の SQL 最適化指定の数
NOPTIX	ビュー定義中にあるインデクスの SQL 最適化指定の数
NSLAS	ビュー定義中にある選択式の AS 句の数
NJTBL	ビュー定義中にある結合表の数
NCSET	HiRDB で指定できる文字集合の種別の数＋ 1※5
NSPDT	ビュー定義中にある「AS データ型」指定の数
NFPRM	ビュー定義中にある関数呼出しの引数とシステム定義スカラ関数の引数の総数
NRTNL	ビュー定義中にある関数呼出しの呼び出し候補となる関数と，システム定義スカラ関数の呼び出し候補となる関数の総数

変数名	説明
NEXRTN	ビュー定義中にある呼び出し候補となる関数のうち外部ルーチンの総数
NCLASS	ビュー定義中にある JAVA 関数のクラスの総数
NJAR	ビュー定義中にある JAVA 関数の JAVA アーカイブの総数
NPVOW	ビュー定義中にあるパブリックビューの所有者名の総数
NDTBL	ビュー定義中にある FROM 句の導出表の数＋ビュー定義中にある内部導出表を生成するビュー表の数
NTBL	• ビュー定義中にある表の延べ数が 15 個以下で，かつビュー定義中にある相関名の数と NDTBL の合計が 15 個以下の場合 15 • ビュー定義中にある表の延べ数が 16 個以上か，又はビュー定義中にある相関名の数と NDTBL の合計が 16 個以上の場合 Max（ビュー定義中にある表の延べ数，ビュー定義中にある相関名の数＋ NDTBL の値）
NDCLM	ビュー定義中にある FROM 句の導出表の合計列数＋ビュー定義中にある内部導出表を生成するビュー表の合計列数
NVCLM	ビュー定義によって導出される列数
NVCSC	ビュー定義によって導出される列で，既定文字集合以外の文字集合を指定した列
NADTL	ビュー定義中にあるすべての抽象データ型の継承関係※6を含めた数
NATTL	ビュー定義中にあるコンポネント指定の数
NATNM	ビュー定義中にあるすべてのコンポネント指定の属性名数
NPLGL	ビュー定義中にあるプラグイン提供関数の数
NPPRL	ビュー定義中にあるプラグイン提供関数のパラメタ数
NCUD	ビュー定義中にある抽象データ型の列数

注※1

定数の種別によって次のように長さを計算します。

定数の種別	長さ（単位：バイト）	境界調整によって必要となる長さの最大値（単位：バイト）
整数定数	4	7
10 進数定数	↑（10 進数定数の精度＋ 1）÷2↑	↑（10 進数定数の精度＋ 1）÷2↑＋ 7
浮動小数点数定数	8	15
文字列定数	2 ＋文字列定数の文字数	2 ＋文字列定数の文字数＋ 3
16 進文字列定数	4 ＋（16 進文字列定数の文字列長）÷2	4 ＋（16 進文字列定数の文字列長）÷2 ＋ 3
各国文字列定数	2 ＋各国文字列定数の文字数×2	2 ＋各国文字列定数の文字数×2 ＋ 3
混在文字列定数	2 ＋混在文字列定数の半角文字の数＋混在文字列定数の全角文字の文字数×2	2 ＋混在文字列定数の半角文字の数＋混在文字列定数の全角文字の文字数×2 ＋ 3

注※2

文字集合変換対象となる定数とは，次の定数です。

- 比較対象が文字集合の項目である文字列定数
- CAST 指定で変換後のデータ型に文字集合指定をした文字列定数

注※3

数値文字変換対象となる定数とは，次の定数です。

- 比較対象が数値属性の項目である文字列定数
- 連結演算のオペランドに指定された数値定数
- 四則演算のオペランドに指定された文字列定数

注※4

選択項目に＊を指定した場合は，FROM 句に指定したすべての表の列数の合計を加算します。また，選択式に表名.＊を指定した場合は，その表の列数を加算します。

注※5

HiRDB で指定できる文字集合は，EBCDIK だけです。

注※6

プラグイン提供の抽象データ型の継承関係は，各プラグインのマニュアルを参照してください。

付録 C.4　表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式

(1)　計算式で使用する変数

a：ディクショナリ表（SQL_TABLES 表）の DEFINITION_CACHE_SIZE の値（単位：バイト）

DEFINITION_CACHE_SIZE の値が分からない場合は，「(3)表定義キャッシュサイズの求め方」を参照してください。

b：表の列数

c：表のインデクス数

d：表分割条件数

e：表格納 RD エリア数

f：インデクス格納 RD エリア数

g：BLOB 列数

h：BLOB 属性を含む抽象データ型数

i：抽象データ型中の BLOB 属性の総数

j：抽象データ型数

k：プラグインオプション数

n：インデクス除外キー値数

p：ディクショナリ表（SQL_TABLES 表）の STATISTICS_CACHE_SIZE の値（単位：バイト）

pdgetcst コマンドで表の最適化情報を取得した場合に加算します。単位がバイトなのでキロバイトに変換して代入してください。STATISTICS_CACHE_SIZE の値が分からない場合は次に示す計算式で求めます。

$(2.6\times q_1 + 3.0\times q_2 + 0.04\times c + 0.02)\times 1024$（単位：バイト）

q_1：列のデータ型が DECIMAL 型以外の列最適化情報数，及び精度 31 けたまでの DECIMAL 型の列最適化情報数の合計

q_2：列のデータが精度 32 けた以上の DECIMAL 型の列最適化情報数

r：CREATE TABLE の DEFAULT オペランドの指定列数

s：CREATE TABLE の DEFAULT オペランドに指定する既定値長の合計値（単位：バイト）

既定値を指定するすべての列数分加算します。既定値長を長く変更することがある場合は，変更後の長さを加算してください。

t：パブリックビュー表の表名と同じ表識別子を所有する認可識別子数

u：表に定義しているトリガ数

v：新旧値相関名で修飾して使用している列数

w：トリガ動作条件数

x：トリガ動作条件の総解析ツリー長（単位：バイト）

トリガ動作条件の解析ツリー長の見積もり式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「通常のデータディクショナリ用 RD エリアの容量の見積もり」を参照してください。

y：UPDATE トリガのトリガ契機列数

z：表に定義している外部キー数

aa：表の主キーを参照している外部キー数

ab：表に定義している検査制約数

ac：検査制約用解析ツリーの合計長（単位：バイト）

検査制約用解析ツリー長の見積もり式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「通常のデータディクショナリ用 RD エリアの容量の見積もり」を参照してください。

ad：インデクス構成部分構造パス数

ae：該当するインデクスの部分構造パスの合計長

af：該当するインデクスの部分構造パス用解析ツリーの合計長（単位：バイト）

部分構造パス用解析ツリー長の見積もり式については，マニュアル「HiRDB Version 8 システム導入・設計ガイド」の「通常のデータディクショナリ用 RD エリアの容量の見積もり」を参照してください。

(2)　1 表当たりの表定義情報バッファサイズ（単位：キロバイト）

1 表当たりの表定義情報バッファサイズは，次に示す概算式で求めてください。なお，ビュー表の場合は，ビュー表の基となる実表又は外部表についても計算してください。

表の種類	1 表当たりの表定義情報バッファサイズの計算式（単位：キロバイト）
1 実表又は 1 外部表の表定義情報バッファサイズ	↓（4 ＋ ↑ a ÷1024↑ ＋ 0.01 × b ＋ ↑ p ÷1024↑ ＋ 7）÷8↓ ×8
1 ビュー表の表定義情報バッファサイズ	↓（4 ＋ ↑ a ÷1024↑ ＋ 0.01 × b ＋ 7）÷8↓ ×8

(3)　表定義キャッシュサイズの求め方

(a)　DEFINITION_CACHE_SIZE の値が分からない場合

次のどれかに該当する場合は，ディクショナリ表を検索しても DEFINITION_CACHE_SIZE の値が分かりません。

- まだ表を定義していない場合
- HiRDB Version 4.0 からバージョンアップした場合※ 1
- 32 ビットモードの HiRDB から 64 ビットモードの HiRDB に移行した場合※ 1 ※ 2

注※ 1

この場合，DEFINITION_CACHE_SIZE に正しい値が入っていません。

注※ 2

32 ビットモードから 64 ビットモードに移行した表であるかどうかは，pdvrup コマンドの実行時間と表の作成時間を比較してください。表の作成時間の方が古い表が，32 ビットモードから 64 ビットモードに移行した表です。pdvrup コマンドの実行時間は，ディクショナリ表（SQL_TABLES 表）の MASTER.SQL_TABLES の CHANGE_TIME で分かります。表の作成時間は，ディクショナリ表（SQL_TABLES 表）の作成した表の CREATE_TIME で分かります。これらの列は，次に示す SQL で検索できます。

- pdvrup コマンド実行時間

```
select CHANGE_TIME from MASTER.SQL_TABLES where
    TABLE_SCHEMA='MASTER' and TABLE_NAME='SQL_TABLES'
```

- 表の作成時間

```
select CREATE_TIME from MASTER.SQL_TABLES where
    TABLE_SCHEMA='認可識別子' and TABLE_NAME='表名'
```

(b)　表定義キャッシュサイズの計算式

表定義キャッシュサイズは次に示す計算式で求めてください。

- 実表又は外部表の場合：（計算式 1 ＋計算式 2）×1024（単位：バイト）
- ビュー表の場合：（計算式 1 ＋計算式 3）×1024（単位：バイト）

計算式の種類	計算式
計算式 1	●HiRDB が 32 ビットモードの場合 0.6 ＋ 0.13× b ＋ 0.15× c ＋ 0.18×（d ＋ 1）＋ 0.01×（e ＋ f） ●HiRDB が 64 ビットモードの場合 0.9 ＋ 0.19× b ＋ 0.18× c ＋ 0.18×（d ＋ 1）＋ 0.01×（e ＋ f）
計算式 2	●LOB 列を定義した表の場合に加算します。

計算式の種類	計算式
	（0.02 ＋ 0.01× e ）× g ●BLOB 属性を含む抽象データ型の列を定義した表の場合に加算します。 0.02× h ＋（0.02 ＋ 0.01× e ）× i ●抽象データ型の列を定義した表の場合に加算します。 0.3× j ＋ 0.3× k ●インデクス除外キー値を定義した表の場合に加算します。 0.3× n ●SEGMENT REUSE を指定（SEGMENT REUSE NO を除く）した表の場合に加算します。 0.01× e ●DEFAULT オペランドに既定値を指定した表の場合に加算します。 0.01× r ＋↑ s ÷1024↑ ●トリガを定義した表の場合に加算します。 0.3× u ＋ 0.2× v ＋ 0.1× w ＋↑ x ÷1024↑＋ 0.2× y ●外部キーを定義した表の場合に加算します。 0.3× z ●被参照表の場合に加算します。 0.1 ＋ 0.1× a a ●検査制約を定義した表の場合に加算します。 0.1× a b ＋↑ a c ÷1024↑ ●部分構造インデクスを定義した表の場合に，部分構造インデクスごとに加算します。 0.1× a d ＋↑ a e ÷1024↑＋↑ a f ÷1024↑
計算式 3	●パブリックビュー表の場合に加算します。 0.1 ＋（0.1× t ） ●ビュー表の構成列が抽象データ型となる場合に加算します。 0.3× j

付録 C.5　1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式

1 サーバ当たりの UNTIL DISCONNECT 指定排他の表と RD エリアの総和（pd_lck_until_disconnect_cnt）の見積もり式を次に示します。なお，計算式はサーバで異なります。

〈計算式〉

サーバの種類	計算式
シングルサーバ	（a＋b＋o＋p＋q）×2＋｛2×（f＋g＋i＋1）＋（g×h）｝×e
ディクショナリサーバ	（a＋b）×2＋｛2×（f＋g＋i＋1）＋（g×h）｝×e
バックエンドサーバ	（a＋b＋o＋p＋q＋r＋s）×2＋｛2×（f＋g＋i＋1）＋（g×h）｝×e

a：同時実行する UNTIL DISCONNECT 指定の LOCK TABLE 文で指定する表の数

b：同時実行する UNTIL DISCONNECT 指定の LOCK TABLE 文で指定する表を格納する RD エリア数

e：同時実行するユティリティの数

f：同時実行するユティリティの処理対象表を格納する LOB 用 RD エリアの数

g：同時実行するユティリティの処理対象インデクスを格納する RD エリアの数

h：同時実行するユティリティの処理対象インデクスの数

i：同時実行するユティリティの処理対象表を格納する RD エリアの数

o：同時に使用するデータ用ローカルバッファが割り当てられた表の格納 RD エリアの総数
データ用ローカルバッファを使用する場合に加算します。

p：同時に使用するインデクス用ローカルバッファが割り当てられたインデクスの格納 RD エリアの総数
インデクス用ローカルバッファを使用する場合に加算します。

q：同時に使用するインデクス用ローカルバッファが割り当てられたインデクスの対象の表の総数
インデクス用ローカルバッファを使用する場合に加算します。

r：同時実行する IN EXCLUSIVE MODE 指定の LOCK 文で指定する共用表格納 RD エリアの総数
共用 RD エリアを使用する場合に加算します。

s：同時実行する IN EXCLUSIVE MODE 指定の LOCK 文で指定する共用表に定義されているすべての共用インデクス格納 RD エリアの総数
共用 RD エリアを使用する場合に加算します。

注　ここでいうユティリティとは次に示すユティリティのことです。

- データベース作成ユティリティ
- データベース再編成ユティリティ
- 空きページ解放ユティリティ
- グローバルバッファ常駐化ユティリティ
- リバランスユティリティ（占有モード）

付録 C.6　ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式

ルーチン定義情報用バッファ長は次に示す計算式から求めます。

使用頻度が高いルーチンの定義情報長の合計値
＋使用するプラグインのプラグイン関数の定義情報長の合計値
＋使用頻度が高いシステム定義スカラ関数の定義情報長の合計値

---

(1)　1 ルーチン当たりのルーチン定義情報長の求め方

1 ルーチン当たりのルーチン定義情報長は，次に示す概算式で求めてください。

〈計算式〉

↑（1.3 ＋ 0.2× a）↑× b　　（単位：キロバイト）

a：使用頻度が高いルーチンの総パラメタ数

b：使用頻度が高いルーチンの定義数

(2)　プラグイン関数の定義情報長の求め方

プラグイン関数の定義情報長は，次に示す概算式で求めてください。

〈計算式〉

0.6 ＋ c ＋ 0.2× d　　（単位：キロバイト）

c：一つのプラグインが DML で利用するプラグイン関数の合計数※

d：一つのプラグインが DML で利用するプラグイン関数のパラメタの合計数※

注

上記の計算式は 1 プラグイン当たりの計算式です。複数のプラグインをインストールしている場合は，インストールしたプラグインの数だけ計算して加算してください。

注※

DML で利用するプラグイン関数，及び DML で利用するプラグイン関数のパラメタの合計数は次に示す SQL で求めます。

```
SELECT COUNT(*),SUM(N_PARAM) FROM MASTER.SQL_PLUGIN_ROUTINES
  WHERE PLUGIN_NAME ='プラグイン名称'
  AND (TIMING_DESCRIPTOR = 'ADT_FUNCTION'
    OR TIMING_DESCRIPTOR IS NULL
    OR TIMING_DESCRIPTOR = 'BEFORE_INSERT'
    OR TIMING_DESCRIPTOR = 'AFTER_INSERT'
    OR TIMING_DESCRIPTOR = 'BEFORE_UPDATE'
    OR TIMING_DESCRIPTOR = 'AFTER_UPDATE'
    OR TIMING_DESCRIPTOR = 'BEFORE_DELETE'
    OR TIMING_DESCRIPTOR = 'AFTER_DELETE'
    OR TIMING_DESCRIPTOR = 'BEFORE_PURGE_TABLE'
    OR TIMING_DESCRIPTOR = 'AFTER_PURGE_TABLE'
    OR TIMING_DESCRIPTOR = 'INDEX_SEARCH'
    OR TIMING_DESCRIPTOR = 'INDEX_COUNT'
    OR TIMING_DESCRIPTOR = 'INDEX_INSERT'
    OR TIMING_DESCRIPTOR = 'INDEX_BEFORE_UPDATE'
    OR TIMING_DESCRIPTOR = 'INDEX_AFTER_UPDATE'
    OR TIMING_DESCRIPTOR = 'INDEX_DELETE'
    OR TIMING_DESCRIPTOR = 'PURGE_INDEX'
    OR TIMING_DESCRIPTOR = 'INDEX_MAINTENANCE_DEFERRED'
    OR TIMING_DESCRIPTOR = 'BEFORE_INSERT_DC'
    OR TIMING_DESCRIPTOR = 'BEFORE_UPDATE_DC'
    OR TIMING_DESCRIPTOR = 'BEFORE_DATA_CHECK'
    OR TIMING_DESCRIPTOR = 'AFTER_DATA_CHECK')
```

(3)　一つ当たりのシステム定義スカラ関数の定義情報長の求め方

一つ当たりのシステム定義スカラ関数の定義情報長を次の表に示します。

表 C-1　一つ当たりのシステム定義スカラ関数の定義情報長

関数名	定義情報長 （単位：キロバイト）
ACOS，ADD_INTERVAL，ASCII，ASIN，ATAN，ATAN2，CENTURY，COS，COSH，CHR，DATE_TIME，DAYNAME，DAYOFWEEK，DAYOFYEAR，DEGREES，EXP，INTERVAL_DATETIMES，LAST_DAY，LN，LOG10，MIDNIGHTSECONDS，MONTHNAME，MONTHS_BETWEEN，NEXT_DAY，PI，RADIANS，SIN，SINH，SQRT，TAN，TANH，WEEK，WEEKOFMONTH，YEARS_BETWEEN	2
POWER，IS_DBLBYTES，IS_SNGLBYTES，ISDIGITS，ROUNDMONTH，TRANSL_LONG	4

関数名	定義情報長 （単位：キロバイト）
CEIL，FLOOR，HALF，INSERTSTR，INSERTSTR_LONG，LEFTSTR，LTRIMSTR，NUMEDIT，QUARTER，REPLACE_LONG，REVERSESTR，RIGHTSTR，RTRIMSTR，SIGN，STRTONUM，TRUNCYEAR	6
LTRIM，REPLACE，RTRIM，TRANSL，TRUNC	12
POSSTR，ROUND	18
GREATEST，LEAST	32

付録 D　排他資源数の見積もり

SQL 文を実行するときに必要とする排他資源数の概算式を示します。これらの値から，最大の排他資源数を算出し余裕値を加算して，次に示すオペランドにプールサイズを設定してください。

- 各サーバのサーバ定義の pd_lck_pool_size オペランド
- フロントエンドサーバ定義の pd_fes_lck_pool_size オペランド
- 各サーバのサーバ定義の pd_lck_until_disconnect_cnt オペランド

注意事項

1. ここで示す排他資源数は，トランザクション内で有効となります。一つのトランザクションで複数の SQL を実行する場合，その排他資源の総和が必要となりますが，既に排他が掛かっている場合は必要としません。排他制御については，マニュアル「HiRDB Version 8 UAP 開発ガイド」を参照してください。
2. HiRDB/パラレルサーバの場合，各バックエンドサーバのリソース数は，対象バックエンドサーバで管理されるリソース数（RD エリア数，インデクス数，行数など）で算出する必要があります。

付録 D.1　定義系 SQL

(1)　ALTER INDEX

(a)　HiRDB/シングルサーバの場合

5＋（インデクス構成列数×2）＋オブジェクトが無効となる手続き，及びトリガ数
オブジェクトが無効となる手続きがある場合に加算します。
＋オブジェクトが無効となる手続き数×3
＋オブジェクトが無効となる手続きが使用する資源数×5
オブジェクトが無効となるトリガがある場合に加算します。
＋オブジェクトが無効となるトリガ数×5
＋UPDATEトリガのトリガ契機に指定した列数×3
＋インデクスが定義されている表を参照するトリガ動作手続き数×5
　新旧値相関名がある場合にさらに加算します。
　＋4
インデクスに関する統計情報を出力している場合に加算します。
＋2
分割キーインデクスの場合に加算します。
＋分割条件数×2
インデクス型を使用しているインデクスの場合に加算します。
＋1
＋インデクス型を定義している型が使用する抽象データ型数×2
＋インデクス型が使用する関数数×2
部分構造インデクスの場合に加算します。
＋構成部分構造パス数×2
除外値指定がある場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクトが無効となる手続き，及びトリガ数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

4＋（インデクス構成列数×2）
オブジェクトが無効となる手続きがある場合に加算します。
＋オブジェクトが無効となる手続き数×3
＋オブジェクトが無効となる手続きが使用する資源数×5
オブジェクトが無効となるトリガがある場合に加算します。
＋オブジェクトが無効となるトリガ数×5
＋UPDATEトリガのトリガ契機に指定した列数×3
＋インデクスが定義されている表を参照するトリガ動作手続き数×5
　新旧値相関名がある場合にさらに加算します。

　＋4
インデクスに関する統計情報を出力している場合に加算します。
＋2
分割キーインデクスの場合に加算します。
＋分割条件数×2
インデクス型を使用しているインデクスの場合に加算します。
＋1
＋インデクス型を定義している型が使用する抽象データ型数×2
＋インデクス型が使用する関数数×2
部分構造インデクスの場合に加算します。
＋構成部分構造パス数×2
除外値指定がある場合に加算します。
＋2

(2)　ALTER PROCEDURE

(a)　HiRDB/シングルサーバの場合

5
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数
実行者が所有者以外の場合に加算します。
＋2
手続き名称指定の場合に加算します。
＋2
AUTHORIZATION指定の場合に加算します。
＋対象者が所有する無効なルーチン数
＋対象者が定義した無効なPUBLICルーチン数＋2
　ALL指定の場合にさらに加算します。
　＋対象者が所有する全ルーチン数
　＋対象者が定義した全PUBLICルーチン数＋1
　INDEX USING指定の場合にさらに加算します。
　＋指定した表をリソースとして使用するルーチン数×2＋3
AUTHORIZATION指定なしの場合に加算します。
＋システム内の無効な全ルーチン数
　ALL指定の場合にさらに加算します。
　＋システム内の全ルーチン数
　INDEX USING指定の場合にさらに加算します。
　＋指定した表をリソースとして使用するルーチン数×2＋3
ルーチンに使用するリソースがある場合に加算します。
＋使用するルーチン数×2＋1
使用するリソースに表がある場合に加算します。
＋使用する表数＋1
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数＋1
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数＋1
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型＋1
パラメタを指定している場合に加算します。
＋パラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5
実行者が所有者以外の場合に加算します。
＋2
手続き名称指定の場合に加算します。
＋2
AUTHORIZATION指定の場合に加算します。
＋対象者が所有する無効なルーチン数

＋対象者が定義した無効なPUBLICルーチン数
ALL指定の場合にさらに加算します。
＋対象者が所有する全ルーチン数
＋対象者が定義した全PUBLICルーチン数
INDEX USING指定の場合にさらに加算します。
＋指定した表をリソースとして使用するルーチン数×2＋3
AUTHORIZATION指定なしの場合に加算します。
＋システム内の無効な全ルーチン数
ALL指定の場合にさらに加算します。
＋システム内の全ルーチン数
INDEX USING指定の場合にさらに加算します。
＋指定した表をリソースとして使用するルーチン数×2＋3
ルーチンに使用するリソースがある場合に加算します。
＋使用するルーチン数×2＋1
使用するリソースに表がある場合に加算します。
＋使用する表数＋1
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数＋1
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数＋1
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型＋1
パラメタを指定している場合に加算します。
＋パラメタ数＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1

(3)　ALTER ROUTINE

(a)　HiRDB/シングルサーバの場合

5
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数
トリガがある場合に加算します。
＋1
実行者が所有者以外の場合に加算します。
＋2
AUTHORIZATION指定の場合に加算します。
＋対象者が所有する無効なルーチン数
＋対象者が定義した無効なPUBLICルーチン数＋1
ALL指定の場合にさらに加算します。
＋対象者が所有する全ルーチン数
＋対象者が定義した全PUBLICルーチン数
対象者の所有するトリガがある場合にさらに加算します。
＋対象者が所有するトリガ数×4＋3
＋対象者が所有するトリガを定義している表数
＋トリガの契機列数×2＋1
＋トリガの動作条件中で使用している資源情報数×2＋1
＋トリガで使用するパラメタ数×2＋1
AUTHORIZATION指定なしの場合に加算します。
＋システム内の無効な全ルーチン数
ALL指定の場合にさらに加算します。
＋システム内の全ルーチン数
システム内にトリガがある場合にさらに加算します。
＋全トリガ数×4＋3
＋トリガを定義している全表数
＋トリガの契機列数×2＋1
＋トリガの動作条件中で使用している資源情報数×2＋1
＋トリガで使用するパラメタ数×2＋1
ルーチンに使用するリソースがある場合に加算します。
＋使用するルーチン数×2
使用するリソースに表がある場合に加算します。
＋使用する表数＋1
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数＋1
＋使用するビュー表で使用する利用資源数＋1

使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数＋1
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型＋1
パラメタを指定している場合に加算します。
＋パラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数
トリガがある場合に加算します。
＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5
実行者が所有者以外の場合に加算します。
＋2
AUTHORIZATION指定の場合に加算します。
＋対象者が所有する無効なルーチン数
＋対象者が定義した無効なPUBLICルーチン数
　ALL指定の場合にさらに加算します。
　＋対象者が所有する全ルーチン数
　＋対象者が定義した全PUBLICルーチン数
　対象者の所有するトリガがある場合にさらに加算します。
　＋対象者が所有するトリガ数×4＋3
　＋対象者が所有するトリガを定義している表数＋1
　＋トリガの契機列数×2＋1
　＋トリガの動作条件中で使用している資源情報数×2＋1
　＋トリガで使用するパラメタ数×2＋1
AUTHORIZATION指定なしの場合に加算します。
＋システム内の無効な全ルーチン数
　ALL指定の場合にさらに加算します。
　＋システム内の全ルーチン数
　システム内にトリガがある場合にさらに加算します。
　＋全トリガ数×4＋3
　＋トリガを定義している全表数＋1
　＋トリガの契機列数×2＋1
　＋トリガの動作条件中で使用している資源情報数×2＋1
　＋トリガで使用するパラメタ数×2＋1
ルーチンに使用するリソースがある場合に加算します。
＋使用するルーチン数×2
使用するリソースに表がある場合に加算します。
＋使用する表数＋1
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数＋1
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数＋1
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型＋1
パラメタを指定している場合に加算します。
＋パラメタ数＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1

(4)　ALTER TABLE

(a)　HiRDB/シングルサーバの場合

4＋表を基に定義してあるビュー表数
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数×3＋2

トリガが定義されている場合に加算します。
＋トリガ定義数×2＋2
＋定義トリガの契機列数＋1
＋定義トリガで使用するパラメタ数＋1
ADD列名を指定する場合に加算します。
＋5
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数
BLOB列を追加する場合にさらに加算します。
＋表に追加するRDエリア数＋1
＋表で使用しているRDエリア数＋1
＋表のBLOB列数
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
インデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表のBLOB列数×表の分割数
マトリクス分割表の場合にさらに加算します。
＋3
ユーザ定義型列を追加する場合にさらに加算します。
＋4
＋ユーザ定義型で使用する属性数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
BLOB属性が定義されているユーザ定義型の場合にさらに加算します。
＋表に追加するRDエリア数＋1
＋表で使用しているRDエリア数＋1
＋表のBLOB列数
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
インデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表のBLOB列数×表の分割数
マトリクス分割表の場合にさらに加算します。
＋3
プラグインの抽象データ型を使用している場合にさらに加算します。
＋使用するプラグイン数＋1
＋使用するプラグインのルーチン数＋1
＋使用するプラグインルーチンのパラメタ数＋1
表がFIX表の場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
非FIX表にNOT NULL列を追加する場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。

＋表の分割数×2＋2
DEFAULT句の指定がある列を追加する場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
ADD RDAREAの場合に加算します。
＋9
＋表の構成列数
＋2
＋表に定義してあるインデクス数
＋4
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数
FIXハッシュ表の場合に加算します。
＋表で使用している表用RDエリア数
＋表の分割数
表で既に使用しているRDエリアを追加する場合にさらに加算します。
＋既に使用しているRDエリア数
表で未使用のRDエリアを追加する場合にさらに加算します。
＋追加するRDエリア数＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクスの使用RDエリア数×2＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるインデクス数
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×RDエリア追加後の分割数＋1
プラグインインデクスが定義してある場合にさらに加算します。
＋2
＋プラグイン列数＋1
＋表で使用しているプラグインの属性数＋1
＋表で使用しているプラグインのルーチン数＋1
＋表で使用しているプラグインルーチンのパラメタ数＋1
＋表で使用しているプラグインのBLOB属性数×RDエリア追加後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
＋表に定義してあるBLOB列数×RDエリア追加後の分割数＋1
＋表に定義してあるBLOB列の使用RDエリア数×2＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるBLOB列数
表にユーザ定義型列が定義してある場合にさらに加算します。
＋表に定義してあるユーザ定義型列数＋1
＋表で使用しているユーザ定義型の属性数＋1
ユーザ定義型にBLOB属性が定義してある場合にさらに加算します。
＋表で使用しているユーザ定義型のBLOB属性数＋1
＋表で使用しているユーザ定義型のBLOB属性の使用RDエリア数×2＋1
＋表で使用しているユーザ定義型のBLOB属性数×RDエリア追加後の分割数＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるユーザ定義型列数
CHANGE 列名の場合に加算します。
＋11
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表にトリガが定義してある場合にさらに加算します。
＋列をリソースとして使用するトリガ数×2
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
＋列をパラメタとして使用するルーチン数×2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義されているインデクス数
列がインデクス構成列の場合にさらに加算します。
列のデータ型がVARCHAR, NVARCHAR, MVARCHARの場合にさらに加算します。
＋表の構成列数
＋表で使用しているインデクス用RDエリア数＋1

＋表に定義してあるインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
列がBLOB列の場合にさらに加算します。
＋1
列がユーザ定義型列の場合にさらに加算します。
＋2
＋列で使用しているユーザ定義型の属性数＋1
NO SPLIT指定の場合又はRECOVERY指定の場合にさらに加算します。
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数
データ長の変更の場合にさらに加算します。
表にインデクスが定義されている場合にさらに加算します。
＋1
列回復制約が定義してある場合にさらに加算します。
列がBLOB列の場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
列がBLOB属性を持つユーザ定義型列の場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
CHANGE CLUSTER KEYの場合に加算します。
＋10
＋クラスタキーインデクスの構成列数×2＋1
＋表で使用しているRDエリア数
オブジェクトが無効となるルーチンがある場合にさらに加算します。
＋オブジェクトが無効となるルーチン数×2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数
分割表の場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
分割表の場合にさらに加算します。
＋表の分割数＋1
ハッシュ混在マトリクス分割表の場合にさらに加算します。
＋2
FIX表の場合にさらに加算します。
＋表の構成列数
CHANGE LOCKの場合に加算します。
＋5
＋表で使用しているRDエリア数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
分割表の場合にさらに加算します。
＋表の分割数＋1
CHANGE HASHの場合に加算します。
＋5
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
FIXハッシュ分割表の場合にさらに加算します。
＋表の分割数＋1
＋表で使用しているRDエリア数＋1
CHANGE SEGMENT REUSEの場合に加算します。

＋4
＋2
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数
＋表の分割条件数×2
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
SEGMENT REUSE指定の有無を変更する場合にさらに加算します。
＋表で使用しているRDエリア数＋1
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
CHANGE INSERT ONLYの場合に加算します。
＋5
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数＋1
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数＋1
行削除期間指定ありの場合にさらに加算します。
＋1
DROP列名の場合に加算します。
＋18
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×4
＋表を基に定義されているビュー表の構成列数
＋表を基に定義されているビュー表の使用資源数
＋表を基に定義されているビュー表にアクセス権限を持つユーザ数
表を基に定義されているパブリックビュー表がある場合にさらに加算します。
＋4
表にトリガが定義してある場合にさらに加算します。
＋列を契機列として使用するトリガ数×2
＋表に定義してあるトリガ数×4
＋表に定義してあるトリガの使用するリソース数×2
＋表に定義してあるトリガの使用するルーチン数
＋表に定義してあるトリガの使用するパラメタ数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数×2
＋列をパラメタとして使用するルーチン数×2
列に関する統計情報を出力している場合にさらに加算します。
＋2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義されているインデクス数
列がインデクス構成列の場合にさらに加算します。
＋列を構成列として使用しているインデクス数×2＋2
＋表で使用しているインデクス用RDエリア数＋1
＋列を構成列として使用しているインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
インデクスに関する統計情報を出力している場合にさらに加算します。
＋列を構成列として使用しているインデクス数＋1
インデクスに除外値指定がある場合にさらに加算します。
＋列を構成列として使用しているインデクスの除外値数＋1
表にBLOB列が定義されている場合にさらに加算します。
＋表に定義されているBLOB列数
分割表の場合にさらに加算します。
＋表に定義されているBLOB列数×表の分割数
分割表の場合にさらに加算します。
＋表の分割数＋1
マトリクス分割表の場合にさらに加算します。
＋2
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったインデクス用RDエリア数×2＋2
＋表もインデクスも格納しなくなったRDエリア数×2
RENAME TABLEの場合に加算します。

＋7
＋表の構成列数
＋表へのアクセス権限を持つユーザ数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×2
表に関する統計情報を出力している場合にさらに加算します。
＋2
列に関する統計情報を出力している場合にさらに加算します。
＋統計情報が存在する列数＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
分割表の場合にさらに加算します。
＋表の分割数＋1
横分割表の場合にさらに加算します。
＋表の分割条件数×2＋1
マトリクス分割表の場合にさらに加算します。
＋表の分割条件数＋4
ハッシュ混在マトリクス分割表の場合にさらに加算します。
＋表の分割キー数＋1
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
部分構造インデクスが定義してある場合にさらに加算します。
＋部分構造インデクス数＋1
プラグインインデクスが定義してある場合にさらに加算します。
＋プラグインインデクスが使用する関数数＋1
インデクスに除外値指定がある場合にさらに加算します。
＋表に定義してあるインデクスの除外値数＋1
インデクスに関する統計情報を出力している場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
分割表の場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
表にユーザ定義型列が定義してある場合にさらに加算します。
＋表に定義してあるユーザ定義型列数＋1
表にインデクス型を使用するインデクスが定義してある場合にさらに加算します。
＋インデクスで使用するユーザ定義型数＋1
ユーザ定義型にBLOB属性を定義している場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数＋1
分割表の場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数×分割数
変更前の表名と同名称のパブリックビュー表が存在する場合にさらに加算します。
＋3
変更後の表名と同名称のパブリックビュー表が存在する場合にさらに加算します。
＋3
RENAME COLUMNの場合に加算します。
＋7
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×2
列に関する統計情報を出力している場合にさらに加算します。
＋2
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数×2＋1
＋列をパラメタとして使用するルーチン数×2
マトリクス分割表の場合にさらに加算します。
＋2
列がインデクス構成列の場合にさらに加算します。
＋列を構成列とするインデクス数＋1
列がプラグインインデクスの構成列の場合にさらに加算します。
＋プラグインインデクスが使用する関数数＋1
列がBLOB列の場合にさらに加算します。
＋2
分割表の場合にさらに加算します。
＋分割数
列がユーザ定義型列の場合にさらに加算します。
列がインデクス型を使用するインデクスの構成列の場合にさらに加算します。
＋インデクスで使用するユーザ定義型数＋1
ユーザ定義型にBLOB属性を定義している場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数＋1
分割表の場合にさらに加算します。

＋ユーザ定義型で使用するBLOB属性数×分割数
CHANGE RDAREA横分割表の場合に加算します。
＋8
＋表の構成列数
＋変更前の分割条件数×2＋変更後の分割条件数×2
＋変更前の分割数＋変更後の分割数
＋変更前の表で使用しているRDエリア数
＋変更後の表で使用しているRDエリア数×2
＋4
＋表に定義してあるインデクス数
＋削除対象インデクス数
＋（2
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表より削除するRDエリア数）
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクス数×変更前の分割数＋1
＋表に定義してあるインデクス数×変更後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×変更前の分割数＋1
＋表に定義してあるBLOB列数×変更後の分割数
表を被参照表とする参照制約が定義してある場合にさらに加算します。
＋表を被参照表とする参照制約数＋1
＋表を被参照表とする参照表数
表を参照表とする参照制約が定義してある場合にさらに加算します。
＋表の参照制約数＋1
表に検査制約が定義してある場合にさらに加算します。
＋表の検査制約数＋1
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったRDエリア数×2＋2
＋データを削除されるRDエリア数
＋表もインデクスも格納しなくなったRDエリア数×2
CHANGE RDAREAマトリクス分割表の場合に加算します。
＋11
＋表の構成列数
＋変更前の分割条件数＋変更後の分割条件数
＋変更前の分割数＋変更後の分割数
＋変更前の表で使用しているRDエリア数
＋変更後の表で使用しているRDエリア数×2
＋4
＋表に定義してあるインデクス数
＋削除対象インデクス数
＋（2
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表より削除するRDエリア数）
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクス数×変更前の分割数＋1
＋表に定義してあるインデクス数×変更後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×変更前の分割数＋1
＋表に定義してあるBLOB列数×変更後の分割数
表を被参照表とする参照制約が定義してある場合にさらに加算します。
＋表を被参照表とする参照制約数＋1
＋表を被参照表とする参照表数
表を参照表とする参照制約が定義してある場合にさらに加算します。
＋表の参照制約数＋1
表に検査制約が定義してある場合にさらに加算します。

＋表の検査制約数＋1
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったRDエリア数×2＋2
＋データを削除されるRDエリア数
＋表もインデクスも格納しなくなったRDエリア数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクトが無効となるルーチン数
＋表を基に定義してあるビュー表数
DROP COLUMNの場合に加算します。
＋表に定義してあるトリガ数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

1
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数×3＋2
トリガが定義されている場合に加算します。
＋トリガ定義数×2＋2
＋定義トリガの契機列数＋1
＋定義トリガで使用するパラメタ数＋1
DROP COLUMNの場合にさらに加算します。
＋表に定義してあるトリガ数
ADD列名の場合に加算します。
＋5
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数
BLOB列を追加する場合にさらに加算します。
＋表に追加するRDエリア数＋1
＋表で使用しているRDエリア数＋1
＋表のBLOB列数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
インデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表のBLOB列数×表の分割数
マトリクス分割表の場合にさらに加算します。
＋3
ユーザ定義型列を追加する場合にさらに加算します。
＋4
＋ユーザ定義型で使用する属性数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
BLOB属性が定義されているユーザ定義型の場合にさらに加算します。
＋表に追加するRDエリア数＋1
＋表で使用しているRDエリア数＋1
＋表のBLOB列数
インデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表のBLOB列数×表の分割数
マトリクス分割表の場合にさらに加算します。
＋3
プラグインの抽象データ型を使用している場合にさらに加算します。
＋使用するプラグイン数＋1
＋使用するプラグインのルーチン数＋1
＋使用するプラグインルーチンのパラメタ数＋1
表がFIX表の場合にさらに加算します。
＋表で使用しているRDエリア数

表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
非FIX表にNOT NULL列を追加する場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
DEFAULT句の指定がある列を追加する場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
ADD RDAREAの場合に加算します。
＋9
＋表の構成列数
＋2
＋表に定義してあるインデクス数
FIXハッシュ表の場合にさらに加算します。
＋表で使用している表用RDエリア数
＋表の分割数
表で既に使用しているRDエリアを追加する場合にさらに加算します。
＋既に使用しているRDエリア数
表で未使用のRDエリアを追加する場合にさらに加算します。
＋追加するRDエリア数＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクスの使用RDエリア数×2＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるインデクス数
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×RDエリア追加後の分割数＋1
プラグインインデクスが定義してある場合にさらに加算します。
＋2
＋プラグイン列数＋1
＋表で使用しているプラグインの属性数＋1
＋表で使用しているプラグインのルーチン数＋1
＋表で使用しているプラグインルーチンのパラメタ数＋1
＋表で使用しているプラグインのBLOB属性数×RDエリア追加後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
＋表に定義してあるBLOB列数×RDエリア追加後の分割数＋1
＋表に定義してあるBLOB列の使用RDエリア数×2＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるBLOB列数
表にユーザ定義型列が定義してある場合にさらに加算します。
＋表に定義してあるユーザ定義型列数＋1
＋表で使用しているユーザ定義型の属性数＋1
ユーザ定義型にBLOB属性が定義してある場合にさらに加算します。
＋表で使用しているユーザ定義型のBLOB属性数＋1
＋表で使用しているユーザ定義型のBLOB属性の使用RDエリア数×2＋1
＋表で使用しているユーザ定義型のBLOB属性数×RDエリア追加後の分割数＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるユーザ定義型列数
CHANGE列名の場合に加算します。
＋11
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表にトリガが定義してある場合にさらに加算します。
＋列をリソースとして使用するトリガ数×2

表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
＋列をパラメタとして使用するルーチン数×2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義されているインデクス数
列がインデクス構成列の場合にさらに加算します。
列のデータ型がVARCHAR, NVARCHAR, MVARCHARの場合にさらに加算します。
＋表の構成列数
＋表で使用しているインデクス用RDエリア数＋1
＋表に定義してあるインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
列がBLOB列の場合にさらに加算します。
＋1
列がユーザ定義型列の場合にさらに加算します。
＋2
＋列で使用しているユーザ定義型の属性数＋1
NO SPLIT指定の場合又はRECOVERY指定の場合にさらに加算します。
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数
データ長の変更の場合にさらに加算します。
表にインデクスが定義されている場合にさらに加算します。
＋1
CHANGE CLUSTER KEYの場合に加算します。
＋10
＋クラスタキーの構成列数×2＋1
＋表で使用しているRDエリア数
オブジェクトが無効となるルーチンがある場合にさらに加算します。
＋オブジェクトが無効となるルーチン数×2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数
分割表の場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
分割表の場合にさらに加算します。
＋表の分割数＋1
ハッシュ混在マトリクス分割表の場合にさらに加算します。
＋2
FIX表の場合にさらに加算します。
＋表の構成列数
CHANGE LOCKの場合に加算します。
＋5
＋表で使用しているRDエリア数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
分割表の場合にさらに加算します。
＋表の分割数＋1
CHANGE HASHの場合に加算します。
＋5
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
FIXハッシュ分割表の場合にさらに加算します。
＋表の分割数＋1
＋表で使用しているRDエリア数＋1
CHANGE SEGMENT REUSEの場合に加算します。
＋4
分割表の場合にさらに加算します。

＋表の分割数
＋表の分割条件数×2
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
SEGMENT REUSE指定の有無を変更する場合にさらに加算します。
＋表で使用しているRDエリア数＋1
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
CHANGE INSERT ONLYの場合に加算します。
＋5
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数＋1
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数＋1
行削除期間指定ありの場合にさらに加算します。
＋1
DROP列名の場合に加算します。
＋18
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×4
＋表を基に定義されているビュー表の構成列数
＋表を基に定義されているビュー表の使用資源数
＋表を基に定義されているビュー表にアクセス権限を持つユーザ数
表を基に定義されているパブリックビュー表がある場合にさらに加算します。
＋4
表にトリガが定義してある場合にさらに加算します。
＋列を契機列として使用するトリガ数×2
＋表に定義してあるトリガ数×4
＋表に定義してあるトリガの使用するリソース数×2
＋表に定義してあるトリガの使用するルーチン数
＋表に定義してあるトリガの使用するパラメタ数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数×2
＋列をパラメタとして使用するルーチン数×2
列に関する統計情報を出力している場合にさらに加算します。
＋2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義されているインデクス数
列がインデクス構成列の場合にさらに加算します。
＋列を構成列として使用しているインデクス数×2＋2
＋表で使用しているインデクス用RDエリア数＋1
＋列を構成列として使用しているインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
インデクスに関する統計情報を出力している場合にさらに加算します。
＋列を構成列として使用しているインデクス数＋1
インデクスに除外値指定がある場合にさらに加算します。
＋列を構成列として使用しているインデクスの除外値数＋1
表にBLOB列が定義されている場合にさらに加算します。
＋表に定義されているBLOB列数
分割表の場合にさらに加算します。
＋表に定義されているBLOB列数×表の分割数
分割表の場合にさらに加算します。
＋表の分割数＋1
マトリクス分割表の場合にさらに加算します。
＋2
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったインデクス用RDエリア数×2＋2
＋表もインデクスも格納しなくなったRDエリア数×2
RENAME TABLEの場合に加算します。
＋7
＋表の構成列数
＋表へのアクセス権限を持つユーザ数
表を基に定義されているビュー表がある場合にさらに加算します。

＋表を基に定義されているビュー表数×2
表に関する統計情報を出力している場合にさらに加算します。
＋2
列に関する統計情報を出力している場合にさらに加算します。
＋統計情報が存在する列数＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
分割表の場合にさらに加算します。
＋表の分割数＋1
横分割表の場合にさらに加算します。
＋表の分割条件数×2＋1
マトリクス分割表の場合にさらに加算します。
＋表の分割条件数＋4
ハッシュ混在マトリクス分割表の場合にさらに加算します。
＋表の分割キー数＋1
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
部分構造インデクスが定義してある場合にさらに加算します。
＋部分構造インデクス数＋1
プラグインインデクスが定義してある場合にさらに加算します。
＋プラグインインデクスが使用する関数数＋1
インデクスに除外値指定がある場合にさらに加算します。
＋表に定義してあるインデクスの除外値数＋1
インデクスに関する統計情報を出力している場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
分割表の場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
表にユーザ定義型列が定義してある場合にさらに加算します。
＋表に定義してあるユーザ定義型列数＋1
表にインデクス型を使用するインデクスが定義してある場合にさらに加算します。
＋インデクスで使用するユーザ定義型数＋1
ユーザ定義型にBLOB属性を定義している場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数＋1
分割表の場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数×分割数
変更前の表名と同名称のパブリックビュー表が存在する場合にさらに加算します。
＋3
変更後の表名と同名称のパブリックビュー表が存在する場合にさらに加算します。
＋3
RENAME COLUMNの場合に加算します。
＋7
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×2
列に関する統計情報を出力している場合にさらに加算します。
＋2
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数×2＋1
＋列をパラメタとして使用するルーチン数×2
マトリクス分割表の場合にさらに加算します。
＋2
列がインデクス構成列の場合にさらに加算します。
＋列を構成列とするインデクス数＋1
列がプラグインインデクスの構成列の場合にさらに加算します。
＋プラグインインデクスが使用する関数数＋1
列がBLOB列の場合にさらに加算します。
＋2
分割表の場合にさらに加算します。
＋分割数
列がユーザ定義型列の場合にさらに加算します。
列がインデクス型を使用するインデクスの構成列の場合にさらに加算します。
＋インデクスで使用するユーザ定義型数＋1
ユーザ定義型にBLOB属性を定義している場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数＋1
分割表の場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数×分割数
CHANGE RDAREA 横分割表の場合に加算します。
＋8
＋表の構成列数

＋変更前の分割条件数×2＋変更後の分割条件数×2
＋変更前の分割数＋変更後の分割数
＋変更前の表で使用しているRDエリア数
＋変更後の表で使用しているRDエリア数×2
＋2
＋表に定義してあるインデクス数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクス数×変更前の分割数＋1
＋表に定義してあるインデクス数×変更後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×変更前の分割数＋1
＋表に定義してあるBLOB列数×変更後の分割数
表を被参照表とする参照制約が定義してある場合にさらに加算します。
＋表を被参照表とする参照制約数＋1
＋表を被参照表とする参照表数
表を参照表とする参照制約が定義してある場合にさらに加算します。
＋表の参照制約数＋1
表に検査制約が定義してある場合にさらに加算します。
＋表の検査制約数＋1
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったRDエリア数×2＋2
＋データを削除されるRDエリア数
＋表もインデクスも格納しなくなったRDエリア数×2
CHANGE RDAREA マトリクス分割表の場合に加算します。
＋11
＋表の構成列数
＋変更前の分割条件数＋変更後の分割条件数
＋変更前の分割数＋変更後の分割数
＋変更前の表で使用しているRDエリア数
＋変更後の表で使用しているRDエリア数×2
＋2
＋表に定義してあるインデクス数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクス数×変更前の分割数＋1
＋表に定義してあるインデクス数×変更後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×変更前の分割数＋1
＋表に定義してあるBLOB列数×変更後の分割数
表を被参照表とする参照制約が定義してある場合にさらに加算します。
＋表を被参照表とする参照制約数＋1
＋表を被参照表とする参照表数
表を参照表とする参照制約が定義してある場合にさらに加算します。
＋表の参照制約数＋1
表に検査制約が定義してある場合にさらに加算します。
＋表の検査制約数＋1
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったRDエリア数×2＋2
＋データを削除されるRDエリア数
＋表もインデクスも格納しなくなったRDエリア数×2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効となるルーチン数
DROP COLUMNの場合に加算します。
＋表に定義してあるトリガ数
ADD 列名の場合に加算します。
BLOB列を追加する場合にさらに加算します。
＋2

＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
BLOB属性を持つユーザ定義型列を追加する場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
ADD RDAREAの場合に加算します。
＋4
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数
CHANGE 列名の場合に加算します。
列回復制約が定義してある場合に加算します。
列がBLOB列の場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
列がBLOB属性を持つユーザ定義型列の場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
CHANGE SEGMENT REUSEの場合に加算します。
＋2
＋表で使用するRDエリア数
CHANGE RDAREAの場合に加算します。
＋2
＋削除対象インデクス数
＋（2
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表より削除するRDエリア数）

(5)　ALTER TRIGGER

(a)　HiRDB/シングルサーバの場合

19
実行者が所有者以外の場合に加算します。
＋2
トリガ契機列がある場合に加算します。
＋トリガの契機列数×2＋1
トリガ動作条件指定がある場合に加算します。
＋トリガの動作条件中で使用している資源情報数×2＋1
トリガが使用するリソースがある場合に加算します。
＋使用するリソース数×2＋1
使用するリソースに表がある場合に加算します。
＋使用する表数
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型数＋1
パラメタを指定している場合に加算します。
＋パラメタ数×2＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

2

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

17
実行者が所有者以外の場合に加算します。
＋2

トリガ契機列がある場合に加算します。
＋トリガの契機列数×2＋1
トリガ動作条件指定がある場合に加算します。
＋トリガの動作条件中で使用している資源情報数×2＋1
トリガが使用するリソースがある場合に加算します。
＋使用するリソース数×2＋1
使用するリソースに表がある場合に加算します。
＋使用する表数
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型数＋1
パラメタを指定している場合に加算します。
＋パラメタ数×2＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1

(6)　COMMENT

(a)　HiRDB/シングルサーバの場合

3
コメントが列の場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3
コメントが列の場合に加算します。
＋2

(7)　CREATE AUDIT

(a)　HiRDB/シングルサーバの場合

7
オブジェクト指定ありの場合に加算します。
＋該当するオブジェクトの定義数
オブジェクト指定なしの場合に加算します。
＋オブジェクト指定のない定義数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

7
オブジェクト指定ありの場合に加算します。
＋該当するオブジェクトの定義数
オブジェクト指定なしの場合に加算します。
＋オブジェクト指定のない定義数

(8)　CREATE CONNECTION SECURITY

(a)　HiRDB/シングルサーバの場合

7＋ディクショナリ表SQL_USERSに登録されているユーザ数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

6＋ディクショナリ表SQL_USERSに登録されているユーザ数

(9)　CREATE FOREIGN INDEX

(a)　フロントエンドサーバ

1

(b)　ディクショナリサーバ

20＋構成列数×5＋2

(10)　CREATE FOREIGN TABLE

(a)　ディクショナリサーバ

17＋列数×4＋2

(11)　CREATE 〔PUBLIC 〕FUNCTION

(a)　HiRDB/シングルサーバの場合

CREATE 〔PUBLIC 〕PROCEDUREの排他資源数＋オブジェクトが無効となるルーチン数
パラメタがある場合に加算します。
＋パラメタ数×2
パラメタにユーザ定義型がある場合に加算します。
＋ユーザ定義型数×2
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びバックエンドサーバ）

CREATE 〔PUBLIC 〕PROCEDUREの排他資源数＋オブジェクトが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

CREATE 〔PUBLIC 〕PROCEDUREの排他資源数＋オブジェクトが無効となるルーチン数
パラメタがある場合に加算します。
＋パラメタ数×2
パラメタにユーザ定義型がある場合に加算します。
＋ユーザ定義型数×2
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1

(12)　CREATE INDEX（プラグインインデクスでない場合）

(a)　HiRDB/シングルサーバの場合

25＋インデクスが無効となるルーチン数＋インデクス格納RDエリア×6＋構成列数×4
インデクスを定義する表を参照するルーチン数がある場合に加算します。
＋インデクスを定義する表を参照するルーチン数×4＋2
インデクスを定義した表を参照するトリガ動作手続きがある場合に加算します。
＋インデクスを定義した表を参照するトリガ動作手続き数×5
インデクス格納RDエリアが二つ以上の場合に加算します。
＋インデクス格納RDエリア数×3＋1
インデクスが部分構造インデクスの場合に加算します。
＋インデクス構成部分構造パス数×3

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋インデクスが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

19＋インデクス格納RDエリア×4＋構成列数×4
インデクスを定義する表を参照するルーチン数がある場合に加算します。
＋インデクスを定義する表を参照するルーチン数×4＋2
インデクスを定義した表を参照するトリガ動作手続きがある場合に加算します。
＋インデクスを定義した表を参照するトリガ動作手続き数×5
インデクス格納RDエリアが二つ以上の場合に加算します。
＋インデクス格納RDエリア数×3＋1＋インデクスが無効となるルーチン数
インデクスが部分構造インデクスの場合に加算します。
＋インデクス構成部分構造パス数×3

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋インデクス格納RDエリア数×2＋インデクスが無効となるルーチン数

(e)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ）

2

(13)　CREATE INDEX（プラグインインデクスの場合）

(a)　HiRDB/シングルサーバの場合

21＋インデクスが無効となるルーチン数＋インデクス格納RDエリア数×8＋構成列数×4＋適用関数数×5＋Σ適用関数パラメタ数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋インデクスが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

15＋インデクス格納RDエリア×6＋構成列数×4＋適用関数数×5＋Σ適用関数パラメタ数＋インデクスが無効となるルーチン数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋インデクス格納RDエリア数×2＋インデクスが無効となるルーチン数

(14)　CREATE 〔PUBLIC〕PROCEDURE

(a)　HiRDB/シングルサーバの場合

13＋SQL文前処理でアクセスする表数＋SQL文前処理でアクセスするビュー表数＋SQL文前処理でアクセスするビュー表の基となる実表数
使用するリソースがある場合に加算します。
＋使用するリソース数×7＋3
パラメタを指定している場合に加算します。
＋パラメタ数×3＋1
パラメタのデータ型がユーザ定義型の場合に加算します。
＋ユーザ定義型のパラメタ数×2＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋SQL文前処理でアクセスする表数＋SELECT文前処理でアクセスするビュー表数＋SELECT文前処理でアクセスするビュー表の基となる実表数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

15
使用するリソースがある場合に加算します。
＋使用するリソース数×7＋3

パラメタを指定している場合に加算します。
＋パラメタ数×3＋1
パラメタのデータ型がユーザ定義型の場合に加算します。
＋ユーザ定義型のパラメタ数×2＋1

(15)　CREATE SCHEMA

(a)　HiRDB/シングルサーバの場合

5

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5

(16)　CREATE SEQUENCE

(a)　HiRDB/シングルサーバの場合

9＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

9

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

2

(17)　CREATE SERVER

(a)　ディクショナリサーバ

9＋pd_max_foreign_serverオペランドの値

(18)　CREATE TABLE

(a)　HiRDB/シングルサーバの場合で，表を格納する RD エリアを省略し，かつ非横分割表の場合

29＋公用RDエリア数×3＋列数×3＋LOB列数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6＋LOB属性数×分割RDエリア数×4＋ユーザ定義型属性数×2＋5
クラスタキーを定義している場合に加算します。
＋5＋インデクス格納RDエリア数×2＋分割RDエリア数×6＋8＋構成列数×3
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
　無効となる関数がある場合にさらに加算します。
　＋オブジェクトが無効となる関数の数
　＋Σ（2＋プロシジャ内リソース数×5）
　＋被参照表を参照する関数の総数
検査制約を定義している場合に加算します。
＋検査制約数×9

(b)　HiRDB/シングルサーバの場合で，表を格納する RD エリアを指定し，かつ非横分割表の場合

28＋列数×4＋LOB列数×4＋表格納RDエリア数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6＋LOB属性数×分割RDエリア数×4＋ユーザ定義型属性数×2＋5
クラスタキーを定義している場合に加算します。
＋5＋インデクス格納RDエリア数×2＋分割RDエリア数×6＋8＋構成列数×3

参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
　無効となる関数がある場合にさらに加算します。
　＋オブジェクトが無効となる関数の数
　＋Σ（2＋プロシジャ内リソース数×5）
　＋被参照表を参照する関数の総数
検査制約を定義している場合に加算します。
＋検査制約数×9

(c)　HiRDB/シングルサーバの場合で，横分割表の場合

29＋列数×4＋分割RDエリア数×21＋LOB列数×分割RDエリア数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6＋LOB属性数×分割RDエリア数×4＋ユーザ定義型属性数×2＋5
クラスタキーを定義している場合に加算します。
＋5＋インデクス格納RDエリア数×2＋分割RDエリア数×6＋8＋構成列数×3
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
　無効となる関数がある場合にさらに加算します。
　＋オブジェクトが無効となる関数の数
　＋Σ（2＋プロシジャ内リソース数×5）
　＋被参照表を参照する関数の総数
検査制約を定義している場合に加算します。
＋検査制約数×9

(d)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3
参照制約を定義している場合に加算します。
＋被参照表数
＋被参照表を参照する総ルーチン数

(e)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）で，表を格納する RD エリアを省略し，かつ非横分割表の場合

20＋公用RDエリア数×3＋列数×3＋LOB列数×分割RDエリア数　×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6＋LOB属性数×分割RDエリア数×4＋ユーザ定義型属性数×2＋5
クラスタキーを定義している場合に加算します。
＋分割RDエリア数×6＋8＋構成列数×3
参照制約を定義している場合に加算します。
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
　無効となる関数がある場合にさらに加算します。
　＋オブジェクトが無効となる関数の数
　＋Σ（2＋プロシジャ内リソース数×5）
検査制約を定義している場合に加算します。
＋検査制約数×9

(f)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）で，表を格納する RD エリアを指定し，かつ非横分割表の場合

25＋列数×4＋LOB列数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6＋LOB属性数×分割RDエリア数×4＋ユーザ定義型属性数×2＋5
クラスタキーを定義している場合に加算します。
＋分割RDエリア数×6＋8＋構成列数×3
参照制約を定義している場合に加算します。
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
　無効となる関数がある場合にさらに加算します。
　＋オブジェクトが無効となる関数の数
　＋Σ（2＋プロシジャ内リソース数×5）
検査制約を定義している場合に加算します。
＋検査制約数×9

(g)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）で，横分割表の場合

マトリクス分割表でない場合に加算します。
24＋列数×4＋分割RDエリア数×17＋LOB列数×分割RDエリア数×4
マトリクス分割表の場合に加算します。
21＋列数×4＋分割RDエリア数（重複排除しない数）×16＋LOB列数×分割RDエリア数×4＋分割キー数×2＋（格納条件数＋2）×2＋2
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6＋LOB属性数×分割RDエリア数×4＋ユーザ定義型属性数×2＋5
クラスタキーを定義している場合に加算します。
＋分割RDエリア数×6＋8＋構成列数×3
参照制約を定義している場合に加算します。
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
　無効となる関数がある場合にさらに加算します。
　＋オブジェクトが無効となる関数の数
　＋Σ（2＋プロシジャ内リソース数×5）
検査制約を定義している場合に加算します。
＋検査制約数×9

(h)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋表格納RDエリア数×4
クラスタキーを定義している場合に加算します。
＋5＋インデクス格納RDエリア数×2

(i)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ）

2
クラスタキーを定義している場合に加算します。
＋1

(19)　CREATE TRIGGER

(a)　HiRDB/シングルサーバの場合

21＋オブジェクトが無効になるルーチン数×5＋トリガ動作手続きで使用している資源数×5
トリガ動作条件指定がある場合に加算します。
＋トリガ動作条件指定中で使用している資源数×5＋3
トリガ動作手続き中に新旧値相関名で修飾した列名を使用している場合に加算します。
＋新値相関名で修飾した列種類数×3＋旧値相関名で修飾した列種類数×3＋3
トリガ契機列を指定する場合に加算します。
＋トリガ契機列数×3
無効になるルーチンがある場合に加算します。

＋無効になるルーチン数×5＋ルーチンが使用する資源数×5
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋オブジェクトが無効になるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

18＋オブジェクトが無効になるルーチン数＋トリガ動作手続きで使用している資源数×5
トリガ動作条件指定がある場合に加算します。
＋トリガ動作条件指定中で使用している資源数×5＋3
トリガ動作手続き中に新旧値相関名で修飾した列名を使用している場合に加算します。
＋新値相関名で修飾した列種類数×3＋旧値相関名で修飾した列種類数×3＋3
トリガ契機列を指定する場合に加算します。
＋トリガ契機列数×3
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×5＋ルーチンが使用する資源数×5
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効になるルーチン数

(20)　CREATE TYPE

(a)　HiRDB/シングルサーバの場合

22＋無効となるルーチン数＋属性数×3
属性にユーザ定義型がある場合に加算します。
＋（ユーザ定義型数×6＋2）
サブタイプ定義の場合に加算します。
＋11
無効となるルーチンがある場合に加算します。
＋（無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1）
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1
　＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1
関数又は手続き定義をする場合に加算します。
＋関数・手続き定義分排他資源数※

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋無効となるルーチン数
関数又は手続き定義をする場合に加算します。
＋関数・手続き定義数分排他資源数※

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

21＋属性数×3
属性にユーザ定義型がある場合に加算します。
＋（ユーザ定義型数×6＋2）
サブタイプ定義の場合に加算します。
＋11
無効となるルーチンがある場合に加算します。
＋（無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1）
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1
　＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1
関数又は手続き定義をする場合に加算します。
＋関数又は手続き定義数分排他資源数※

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

無効になるルーチン数

注※

関数定義又はプロシジャ定義の排他要求数を参照してください。

(21)　CREATE USER MAPPING

(a)　ディクショナリサーバ

11

(22)　CREATE 〔PUBLIC〕VIEW

(a)　HiRDB/シングルサーバの場合

31＋SELECT文前処理でアクセスする表数＋SELECT文前処理でアクセスするビュー表数＋SELECT文前処理でアクセスするビュー表の基となる実表数＋列数×4＋ビュー表の基表数×4＋SELECT文前処理の数
パブリックビュー表の場合に加算します。
＋2
＋パブリックビュー表と同じ名称の表数×2
＋パブリックビュー表と同じ名称のビュー表数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋SELECT文前処理でアクセスする表数＋SELECT文前処理でアクセスするビュー表数＋SELECT文前処理でアクセスするビュー表の基となる実表数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

28＋列数×4＋ビュー表の基表数×4＋SELECT文前処理の数
パブリックビュー表の場合に加算します。
＋2
＋パブリックビュー表と同じ名称の表数×2
＋パブリックビュー表と同じ名称のビュー表数×2

(23)　DROP AUDIT

(a)　HiRDB/シングルサーバの場合

7
オブジェクト指定ありの場合に加算します。
＋該当するオブジェクトの定義数
オブジェクト指定なしの場合に加算します。
＋オブジェクト指定のない定義数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

7
オブジェクト指定ありの場合に加算します。
＋該当するオブジェクトの定義数
オブジェクト指定なしの場合に加算します。
＋オブジェクト指定のない定義数

(24)　DROP CONNECTION SECURITY

(a)　HiRDB/シングルサーバの場合

7＋ディクショナリ表SQL_USERSに登録されているユーザ数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

6＋ディクショナリ表SQL_USERSに登録されているユーザ数

(25)　DROP DATA TYPE

(a)　HiRDB/シングルサーバの場合

26＋無効となるルーチン数＋属性数×3
サブタイプの場合に加算します。
＋6
ユーザ定義型の属性がある場合に加算します。
＋ユーザ定義型属性数×4
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1
　＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1
関数定義がある場合に加算します。
＋関数定義数×7＋1＋関数のパラメタ数×4＋1
＋関数で使用する手続き数×5＋1
手続き定義がある場合に加算します。
＋手続き定義数×5＋1＋手続きのパラメタ数×4＋1
＋手続きで使用するリソース数×5＋1
トリガ動作条件に指定している場合に加算します。
＋15
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

23＋属性数×3＋オブジェクトが無効となるルーチン数
サブタイプの場合に加算します。
＋6
ユーザ定義型の属性がある場合に加算します。
＋ユーザ定義型属性数×4
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1
　＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1
関数定義がある場合に加算します。
＋関数定義数×7＋1＋関数のパラメタ数×4＋1＋関数で使用する手続き数×5＋1
手続き定義がある場合に加算します。
＋手続き定義数×5＋1＋手続きのパラメタ数×4＋1＋手続きで使用するリソース数×5＋1
トリガ動作条件に指定している場合に加算します。
＋15
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効となるルーチン数

(26)　DROP FOREIGN INDEX

(a)　フロントエンドサーバ

1

(b)　ディクショナリサーバ

21＋構成列数×3
外部インデクスの最適化情報を作成する場合に加算します。
＋4

(27)　DROP FOREIGN TABLE

(a)　フロントエンドサーバ

1＋外部表を基に定義したビュー表の数＋削除する外部表又はビュー表を参照するルーチン数

(b)　ディクショナリサーバ

19＋列数×4＋削除する外部表への権限定義数×3＋オブジェクトが無効になるルーチン数
インデクスを定義している場合に加算します。
＋外部インデクス数×5＋Σ（構成列数×7＋4）
最適化情報を作成する場合に加算します。
＋2＋最適化情報を作成する列数×3＋最適化情報を作成する外部インデクス数×3
ビュー表の基表の場合に加算します。
＋削除対象のビュー表数
＋Σ｛20＋列数×4＋ビュー表の基表数＋ビュー表への権限定義数×3｝＋2
無効になるルーチンがある場合に加算します。
＋オブジェクトが無効になるルーチン数＋Σ（2＋プロシジャ内リソース数×5）＋1

(c)　バックエンドサーバ

オブジェクトが無効になるルーチン数

(28)　DROP 〔PUBLIC 〕FUNCTION

(a)　HiRDB/シングルサーバの場合

DROP 〔PUBLIC 〕PROCEDUREの排他資源数
パラメタがある場合に加算します。
＋パラメタ数×2
パラメタにユーザ定義型がある場合に加算します。
＋ユーザ定義型数×2
トリガ動作条件に指定している場合に加算します。
＋15
無効となるルーチンをビュー表で使用している場合に加算します。
＋ビュー表で使用している無効となるルーチン数＋1＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びバックエンドサーバ）

DROP 〔PUBLIC 〕PROCEDUREと同じ

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

DROP 〔PUBLIC 〕PROCEDUREの排他資源数
パラメタがある場合に加算します。
＋パラメタ数×2
パラメタにユーザ定義型がある場合に加算します。
＋ユーザ定義型数×2
トリガ動作条件に指定している場合に加算します。
＋15
無効となるルーチンをビュー表で使用している場合に加算します。
＋ビュー表で使用している無効となるルーチン数＋1＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1

(29)　DROP INDEX（プラグインインデクスでない場合）

(a)　HiRDB/シングルサーバの場合

22＋オブジェクト又はインデクスが無効となるルーチン数＋インデクス格納RDエリア数×7＋構成列数×3
＋インデクスセグメント数
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×7＋1
インデクスを定義した表を参照するトリガ動作手続きがある場合に加算します。
＋インデクスを定義した表を参照するトリガ動作手続き数×5
インデクスの最適化情報を取得している場合に加算します。
＋4
インデクスが部分構造インデクスの場合に加算します。
＋インデクス構成部分構造パス数×3

再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクト又はインデクスが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

16＋インデクス格納RDエリア数×3＋構成列数×3＋オブジェクト又はインデクスが無効となるルーチン数
無効となるルーチンがある場合に加算します。
無効となるルーチン数×7＋1
インデクスを定義した表を参照するトリガ動作手続きがある場合に加算します。
＋インデクスを定義した表を参照するトリガ動作手続き数×5
インデクスの最適化情報を取得している場合に加算します。
＋4
インデクスが部分構造インデクスの場合に加算します。
＋インデクス構成部分構造パス数×3
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋インデクス格納RDエリア数×4＋インデクスセグメント数＋オブジェクト又はインデクスが無効となるルーチン数

(e)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ）

2

(30)　DROP INDEX（プラグインインデクスの場合）

(a)　HiRDB/シングルサーバの場合

15＋オブジェクト又はインデクスが無効となるルーチン数＋インデクス格納RDエリア数×10＋構成列数×4＋適用関数数×3＋インデクスセグメント数
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクト又はインデクスが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

9＋インデクス格納RDエリア数×6＋構成列数×4＋適用関数数×3＋オブジェクト又はインデクスが無効となるルーチン数
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋インデクス格納RDエリア数×4＋インデクスセグメント数＋オブジェクト又はインデクスが無効となるルーチン数

(31)　DROP 〔PUBLIC 〕PROCEDURE

(a)　HiRDB/シングルサーバの場合

19＋パラメタ数×5＋リソース数×10＋オブジェクトが無効になるルーチン数
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×4＋2＋無効になるルーチンのリソース数×5＋1
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクトが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

18＋パラメタ数×3＋リソース数×10＋オブジェクトが無効となるルーチン数
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×4+2＋無効になるルーチンのリソース数×5+1
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効となるルーチン数

(32)　DROP SCHEMA

(a)　HiRDB/シングルサーバの場合

11＋スキーマ内の表数＋スキーマ内のビュー表数＋スキーマ内のルーチン数＋スキーマ内の表又はビュー表を基表にしたほかのスキーマのビュー表数＋削除する表，ビュー表，ルーチン，又はデータ型を参照するほかのスキーマのインデクス数
スキーマ内の表削除分を加算します（マトリクス分割表でない場合）。
＋Σ｛表格納RDエリア数×3＋分割条件指定数×9＋列数×4＋（LOB列数×4）×表格納RDエリア数＋1＋8｝＋2
スキーマ内の表削除分を加算します（マトリクス分割表の場合）。
＋Σ｛表格納RDエリア数×3＋表格納RDエリア指定数（重複排除しない数）×12＋列数×4＋（LOB列数×4）×表格納RDエリア数＋1＋8＋削除表の権限定義数＋分割キー数×2＋（格納条件指定数＋2）×2＋2｝＋3
スキーマ内のインデクス削除分を加算します。
＋3＋Σ（インデクス格納RDエリア数×7＋構成列数×3＋インデクス構成部分構造パス数×3※1）＋Σ（表格納RDエリア数×10＋10＋表データの使用中セグメント数）＋Σ（インデクス格納RDエリア数×8＋10＋インデクスの使用中セグメント数）＋8
最適化情報を収集している場合に加算します。
＋最適化情報収集表数×2＋最適化情報収集列数×3＋最適化情報収集インデクス数×3
表削除に伴い削除されるビュー表分を加算します。
＋2＋Σ（12＋ビュー表の基表数×4＋列数×4）
スキーマ内の表，ビュー表，ルーチン，又はデータ型を使用しているルーチン分を加算します。
＋3＋Σ（5＋プロシジャ内リソース数×5＋パラメタ定義数×3）
無効となるプロシジャがある場合に加算します。
＋無効となるプロシジャ数＋Σ（2＋リソース数×5）＋スキーマ内に定義されたデータ型数分※2
削除する表を参照するトリガ動作手続きがある場合に加算します。
＋削除する表を参照するトリガ動作手続き数×5
トリガを定義した表がある場合に加算します。
＋スキーマ内トリガ定義数×8＋スキーマ内UPDATEトリガ定義の総指定列数×3＋スキーマ内トリガ動作条件があるトリガ数×3＋スキーマ内トリガ動作条件中で参照する資源種類数×5
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
検査制約を定義している場合に加算します。
＋検査制約数×9
再編成時期予測機能を使用している場合に加算します。
＋Σ｛（表格納RDエリア数＋（LOB列数＋LOB属性数）×表格納用RDエリア数）×62｝
＋Σ｛（インデクス格納RDエリア数＋（LOB列数＋LOB属性数）×表格納用RDエリア数）×62｝＋1
スキーマ内に順序数生成子がある場合に加算します。
＋2＋スキーマ内の順序数生成子数×3
＋スキーマ内の順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋スキーマ内の表数＋スキーマ内の外部表数＋スキーマ内のビュー表数＋スキーマ内のルーチン数＋スキーマ内の順序数生成子数＋スキーマ内の表，外部表，又はビュー表を基表にしたほかのスキーマのビュー表数＋削除する表，外部表，ビュー表，ルーチン，又はデータ型を参照するほかのスキーマのルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

11＋ほかのスキーマのインデクス数
スキーマ内の表削除分を加算します（マトリクス分割表でない場合）。
＋Σ｛表格納RDエリア数×3＋分割条件指定数×9＋列数×4＋（LOB列数×4）×表格納RDエリア数＋削除する表への権限定義数×3＋6｝＋11
スキーマ内の表削除分を加算します（マトリクス分割表の場合）。
＋Σ｛表格納RDエリア数×3＋表格納RDエリア指定数（重複排除しない数）×12＋列数×4＋（LOB列数×4）×表格納RDエリア数＋削除表の権限定義数×3＋1＋6＋分割キー数×2＋（格納条件指定数＋2）×2＋

2}＋3
スキーマ内の表削除分を加算します（外部表の場合）。
＋Σ（列数×4＋削除する外部表への権限定義数×3＋4）＋8
スキーマ内のインデクス及び外部インデクス削除分を加算します。
＋4＋Σ（インデクス格納RDエリア数×7＋構成列数×7＋インデクス構成部分構造パス数×3※1）＋インデクス数×5＋外部インデクス数×5
最適化情報を収集している場合に加算します。
＋最適化情報収集表数×2＋表の最適化情報収集列数×3＋最適化情報作成外部表数×2＋外部表の最適化情報作成列数×3＋最適化情報収集インデクス数×3＋最適化情報作成外部インデクス数×3
表又は外部表削除に伴い削除されるビュー表分を加算します。
＋2＋Σ（18＋ビュー表の基表数×4＋列数×4＋ビュー表への権限定義数×3）
スキーマ内の表，外部表，ビュー表，ルーチン，データ型を使用しているルーチン分を加算します。
＋3＋Σ（5＋プロシジャ内リソース数×10＋パラメタ定義数×3）
無効となるプロシジャがある場合に加算します。
＋無効となるプロシジャ数＋Σ（5＋リソース数×5）＋スキーマ内に定義されたデータ型数分※2
削除する表を参照するトリガ動作手続きがある場合に加算します。
＋削除する表を参照するトリガ動作手続き数×5
トリガを定義した表がある場合に加算します。
＋スキーマ内トリガ定義数×8＋スキーマ内UPDATEトリガ定義の総指定列数×3＋スキーマ内トリガ動作条件があるトリガ数×3＋スキーマ内トリガ動作条件中で参照する資源種類数×5
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
検査制約を定義している場合に加算します。
＋検査制約数×9
再編成時期予測機能を使用している場合に加算します。
＋Σ（（表格納RDエリア数＋（LOB列数＋LOB属性数）×表格納用RDエリア数）×62）
＋Σ（（インデクス格納RDエリア数＋（LOB列数＋LOB属性数）×表格納用RDエリア数）×62）＋1
スキーマ内に順序数生成子がある場合に加算します。
＋2＋スキーマ内の順序数生成子数×2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

＋Σ（表格納RDエリア数×10＋10＋表データの使用中セグメント数）
＋Σ（インデクス格納RDエリア数×8＋10＋インデクスの使用中セグメント数）＋8＋オブジェクト又はインデクスオブジェクトが無効となるルーチン数
スキーマ内に順序数生成子が存在する場合に加算します。
＋スキーマ内の順序数生成子数×2

(e)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ））

表数＋インデクス格納RDエリア数

注※1　インデクスが部分構造インデクスの場合に繰り返し加算します。

注※2　ユーザ定義型削除時の排他要求数を参照してください。

(33)　DROP SEQUENCE

(a)　HiRDB/シングルサーバの場合

9＋2
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数×3＋2
トリガが定義されている場合に加算します。
＋トリガ定義数×2＋2＋定義トリガ契機列数＋1＋定義トリガで使用するパラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

8
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数×2＋2
トリガが定義されている場合に加算します。
＋トリガ定義数×2＋2＋定義トリガ契機列数＋1＋定義トリガで使用するパラメタ数＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

2

(34)　DROP SERVER

(a)　ディクショナリサーバ

9＋削除する外部サーバへの権限定義数

(35)　DROP TABLE

(a)　HiRDB/シングルサーバの場合

マトリクス分割表でない場合に加算します。
24＋表を基に定義したビュー表数＋削除する表又はビュー表を参照するルーチン数＋表格納RDエリア数×3＋分割条件指定数×9＋列数×4＋5＋表格納RDエリア数×4＋表データの使用中セグメント数
マトリクス分割表の場合に加算します。
22＋表を基に定義したビュー表数＋削除する表又はビュー表を参照するルーチン数＋表格納RDエリア数×3＋表格納RDエリア数（重複排除しない数）×8＋列数×4＋5＋表格納RDエリア数×4＋表データの使用中セグメント数＋分割キー数×2＋（格納条件指定数＋2）×2＋2
インデクスを定義している場合に加算します。
＋Σ（インデクス格納RDエリア数×2）＋インデクスの使用中セグメント数＋Σ（インデクス格納RDエリア数×7＋構成列数×3＋インデクス構成部分構造パス数×3※1）＋3
LOB列を定義している場合に加算します。
＋（LOB列数×4）×表格納RDエリア数＋1
最適化情報を収集している場合に加算します。
＋2＋最適化情報収集列数×3＋最適化情報収集インデクス数×3
ビュー表の基表の場合に加算します。
＋削除対象ビュー表数＋Σ｛（12＋ビュー表の基表数）＋列数×4｝＋2
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数＋Σ（2＋プロシジャ内リソース数×5）
削除する表を参照するトリガ動作手続きがある場合に加算します。
＋削除する表を参照するトリガ動作手続き数×5
ユーザ定義型の列が定義されている場合に加算します。
＋ユーザ定義型列数×4＋LOB属性数×分割RDエリア数×5＋抽象データ型属性数＋4
トリガが定義されている場合に加算します。
＋トリガ定義数×8＋UPDATEトリガ定義の総指定列数×3＋トリガ動作条件があるトリガ数×3＋トリガ動作条件中で参照する資源種類数×5
LOB列又はLOB属性を含むユーザ定義型を定義している場合に加算します。
＋Σ（↑RDエリア確保セグメント数÷64000↑＋1）※2
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
＋制約削除によって影響する被参照表の関数の総数
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
検査制約を定義している場合に加算します。
＋検査制約数×9
再編成時期予測機能を使用している場合に加算します。
＋｛表格納用RDエリア数＋インデクス格納RDエリア数
＋（LOB列数＋LOB属性数）×表格納用RDエリア数｝×62＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋表を基に定義したビュー表数＋削除する表又はビュー表を参照するルーチン数
参照制約を定義している場合に加算します。
＋被参照表数×4
＋制約削除によって影響する被参照表の関数の総数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

マトリクス分割表でない場合に加算します。
21＋表格納RDエリア数×3＋分割条件指定数×9＋列数×4＋オブジェクトが無効となるルーチン数
マトリクス分割表の場合に加算します。
19＋表格納RDエリア数×3＋分割条件指定数×9＋列数×4＋オブジェクトが無効となるルーチン数

インデクスを定義している場合に加算します。
＋Σ（インデクス格納RDエリア数×7＋構成列数×3＋インデクス構成部分構造パス数×3※1）＋3
LOB列を定義している場合に加算します。
＋（LOB列数×4）×表格納RDエリア数＋1
最適化情報を収集している場合に加算します。
＋2＋最適化情報収集列数×3＋最適化情報収集インデクス数×3
ビュー表の基表の場合に加算します。
＋削除対象ビュー表数＋Σ｛12＋ビュー表の基表数）＋列数×4｝＋2
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数＋Σ（2＋プロシジャ内リソース数×5）
削除する表を参照するトリガ動作手続きがある場合に加算します。
＋削除する表を参照するトリガ動作手続き数×5
ユーザ定義型の列が定義されている場合に加算します。
＋ユーザ定義型列数×4＋LOB属性数×分割RDエリア数×5＋抽象データ型属性数＋4
トリガが定義されている場合に加算します。
＋トリガ定義数×8＋UPDATEトリガ定義の総指定列数×3＋トリガ動作条件があるトリガ数×3＋トリガ動作条件中で参照する資源種類数×5
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
検査制約を定義している場合に加算します。
＋検査制約数×9
再編成時期予測機能を使用している場合に加算します。
＋｛表格納用RDエリア数＋インデクス格納RDエリア数
＋（LOB列数＋LOB属性数）×表格納用RDエリア数｝×62＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋表格納RDエリア数×4＋表データの使用中セグメント数＋オブジェクトが無効となるルーチン数
インデクスを定義している場合に加算します。
＋Σ（インデクス格納RDエリア数×2）＋インデクスの使用中セグメント数
LOB列又はLOB属性を含むユーザ定義型を定義している場合に加算します。
＋Σ（↑RDエリア確保セグメント数÷64000↑＋1）※2

(e)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ）

2
共用インデクスを定義している場合に加算します。
＋インデクス格納RDエリア数

注※1　インデクスが部分構造インデクスの場合に繰り返し加算します。

注※2　LOB 格納 RD エリアの数だけ繰り返し加算します。

(36)　DROP TRIGGER

(a)　HiRDB/シングルサーバの場合

19＋オブジェクトが無効になるルーチン数＋トリガ動作手続きで使用している資源数×5
トリガ動作条件指定がある場合に加算します。
＋トリガ動作条件指定中で使用している資源数×5＋3
トリガ動作手続き中に新旧値相関名で修飾した列名を使用している場合に加算します。
＋新値相関名で修飾した列種類数×3＋旧値相関名で修飾した列種類数×3＋3
トリガ契機列を指定する場合に加算します。
＋トリガ契機列数×3
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×5＋ルーチンが使用する資源数×5
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋オブジェクトが無効になるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

16＋オブジェクトが無効になるルーチン数＋トリガ動作手続きで使用している資源数×5
トリガ動作条件指定がある場合に加算します。
＋トリガ動作条件指定中で使用している資源数×5＋3
トリガ動作手続き中に新旧値相関名で修飾した列名を使用している場合に加算します。
＋新値相関名で修飾した列種類数×3＋旧値相関名で修飾した列種類数×3＋3
トリガ契機列を指定する場合に加算します。
＋トリガ契機列数×3
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×5＋ルーチンが使用する資源数×5
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効になるルーチン数

(37)　DROP USER MAPPING

(a)　ディクショナリサーバ

9

(38)　DROP 〔PUBLIC 〕VIEW

(a)　HiRDB/シングルサーバの場合

10＋削除されるビュー表数＋無効となるプロシジャ数＋Σ（15＋列数×4＋ビューの基表数×4）
無効となるプロシジャがある場合に加算します。
＋無効となるプロシジャ数＋Σ（2＋プロシジャ内リソース数×5）
削除するビュー表を参照するトリガ動作手続きがある場合に加算します。
＋削除するビュー表を参照するトリガ動作手続き数×5
パブリックビュー表の場合に加算します。
＋1
＋パブリックビュー表と同じ名称の表数
＋パブリックビュー表と同じ名称のビュー表数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

削除されるビュー表数＋無効となるプロシジャ数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

10＋Σ（15＋列数×4＋ビューの基表数×4）
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数＋Σ（2＋ルーチン内リソース数×5）
削除するビュー表を参照するトリガ動作手続きがある場合に加算します。
＋削除するビュー表を参照するトリガ動作手続き数×5
パブリックビュー表の場合に加算します。
＋1
＋パブリックビュー表と同じ名称の表数
＋パブリックビュー表と同じ名称のビュー表数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効となるルーチン数

(39)　GRANT AUDIT

(a)　HiRDB/シングルサーバの場合

3
パスワード文字制限機能を使用している場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3
パスワード文字制限機能を使用している場合に加算します。
＋2

(40)　GRANT CONNECT

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数
パスワード文字制限機能を使用している場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数
パスワード文字制限機能を使用している場合に加算します。
＋2

(41)　GRANT DBA

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数
パスワード文字制限機能を使用している場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数
パスワード文字制限機能を使用している場合に加算します。
＋2

(42)　GRANT RDAREA

(a)　HiRDB/シングルサーバの場合

5＋指定したユーザ数
PUBLIC指定の場合に加算します。
＋指定したRDエリア数
ユーザ指定の場合に加算します。
＋指定したRDエリア数×指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5＋指定したユーザ数
PUBLIC指定の場合に加算します。
＋指定したRDエリア数
ユーザ指定の場合に加算します。
＋指定したRDエリア数×指定したユーザ数

(43)　GRANT SCHEMA

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数

(44)　GRANT アクセス権限

(a)　HiRDB/シングルサーバの場合

4＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

4＋指定したユーザ数

(45)　REVOKE CONNECT

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数

(46)　REVOKE DBA

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数

(47)　REVOKE RDAREA

(a)　HiRDB/シングルサーバの場合

9
PUBLIC指定の場合に加算します。
＋指定したRDエリア数
ユーザ指定の場合に加算します。
＋指定したRDエリア数×指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

9
PUBLIC指定の場合に加算します。
＋指定したRDエリア数
ユーザ指定の場合に加算します。
＋指定したRDエリア数×指定したユーザ数

(48)　REVOKE SCHEMA

(a)　HiRDB/シングルサーバの場合

5＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5＋指定したユーザ数

(49)　REVOKE アクセス権限

(a)　HiRDB/シングルサーバの場合

7＋指定したユーザ数
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×2＋1

表を基に定義したビュー表がある場合に加算します。
＋DROP 〔PUBLIC 〕VIEWのシングルサーバの排他資源数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数
表を基に定義したビュー表がある場合に加算します。
＋DROP 〔PUBLIC 〕VIEWのフロントエンドサーバの排他資源数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

6＋指定したユーザ数
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×2＋1
表を基に定義したビュー表がある場合に加算します。
＋DROP 〔PUBLIC 〕VIEWのディクショナリサーバの排他資源数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

0
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数
表を基に定義したビュー表がある場合に加算します。
＋DROP 〔PUBLIC 〕VIEWのバックエンドサーバの排他資源数

付録 D.2　操作系 SQL

(1)　SELECT（LOCK TABLE なし及び WITHOUT LOCK 指定なしの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋検索対象表格納RDエリア数＋表数＋検索ヒット行数※3＋利用インデクス格納RDエリア数※2＋検索で利用したインデクスキー数※5
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※4
リストを介した表データ検索の場合，次の点に留意してください。

- 式中の前処理の排他要求数は，リストの基表に適用して求めてください。
- 式中の検索対象表格納 RD エリア数や表数は，リストの基表について求めてください。
- 次の値を加算してください。

  ＋ 1

ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※1
リストを介した表データ検索の場合，式中の前処理の排他要求数はリストの基表に適用して求めてください。
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

検索対象表格納RDエリア数＋表数＋検索ヒット行数※3＋利用インデクス格納RDエリア数※2＋検索で利用したインデクスキー数※5
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※4
リストを介した表データ検索の場合，次の点に留意してください。

- 式中の検索対象表格納 RD エリア数や表数は，リストの基表について求めてください。
- 次の値を加算してください。

  ＋ 1

ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

注※ 1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※ 2

検索条件式中の，AND,OR で指定した列にインデクスが定義してある場合は，基本的にそのインデクスは使用されます。

注※ 3

- 複数の表から検索する場合は，個々の表の検索ヒット行数を求めて加算してください。
- インデクス定義列に対する条件を二つ以上（FOR UPDATE がある場合は一つ以上）指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。
- ページ排他指定の場合は検索ヒットページ数となります。

注※ 4

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※ 5

インデクスキー値排他指定の場合は 1，インデクスキー値無排他指定の場合 0 になります。

(2)　SELECT（LOCK TABLE なし及び WITHOUT LOCK WAIT 指定ありの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋検索対象RDエリア数＋表数＋1＋利用インデクス格納RDエリア数※2＋検索で利用したインデクスキー数※4
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。

- 式中の前処理の排他要求数は，リストの基表に適用して求めてください。
- 式中の検索対象表格納 RD エリア数や表数は，リストの基表について求めてください。
- 次の値を加算してください。

  ＋ 1

ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※1
リストを介した表データ検索の場合，式中の前処理の排他要求数はリストの基表に適用して求めてください。
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

検索対象RDエリア数＋表数＋2＋利用インデクス格納RDエリア数※2＋検索で利用したインデクスキー数※4
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。

- 式中の検索対象表格納 RD エリア数や表数は，リストの基表について求めてください。
- 次の値を加算してください。

  ＋ 1

ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

注※ 1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※ 2

検索条件式中の AND，OR で指定した列にインデクスが定義してある場合は，基本的にそのインデクスは使用されます。

注※ 3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※ 4

インデクスキー値排他指定の場合は 1，インデクスキー値無排他指定の場合 0 になります。

(3)　SELECT（LOCK TABLE なし及び WITHOUT LOCK NOWAIT 指定ありの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋検索対象RDエリア数＋表数＋利用インデクス格納RDエリア数※2
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。

- 式中の前処理の排他要求数は，リストの基表に適用して求めてください。
- 式中の検索対象表格納 RD エリア数や表数は，リストの基表について求めてください。
- 次の値を加算してください。

  ＋ 1

pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※1
リストを介した表データ検索の場合，式中の前処理の排他要求数はリストの基表に適用して求めてください。
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

検索対象RDエリア数＋表数＋利用インデクス格納RDエリア数※2
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。

- 式中の検索対象表格納 RD エリア数や表数は，リストの基表について求めてください。
- 次の値を加算してください。

  ＋ 1

pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

注※ 1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※ 2

検索条件式中の AND，OR で指定した列にインデクスが定義してある場合は，基本的にそのインデクスは使用されます。

注※ 3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(4)　SELECT（LOCK TABLE ありの場合※1）

(a)　HiRDB/シングルサーバの場合

［前処理］※2＋検索対象RDエリア数＋利用インデクス格納RDエリア数
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。

- 式中の前処理の排他要求数は，リストの基表に適用して求めてください。
- 式中の検索対象表格納 RD エリア数や表数は，リストの基表について求めてください。
- 次の値を加算してください。

  ＋ 1

ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※2
リストを介した表データ検索の場合，式中の前処理の排他要求数はリストの基表に適用して求めてください。
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

検索対象RDエリア数＋利用インデクス格納RDエリア数
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。

＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。

- 式中の検索対象表格納 RD エリア数や表数は，リストの基表について求めてください。
- 次の値を加算してください。

  ＋ 1

ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

注※ 1

FOR UPDATE 指定の SELECT で LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

注※ 2

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※ 3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(5)　INSERT（INSERT～VALUES 句指定，LOCK TABLE なしの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋5＋挿入対象RDエリア数※3＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×分岐行数＋1ページに格納できないBINARY型の挿入列数×分岐行数＋インデクス数
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※2
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋挿入対象RDエリア数※3＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×分岐行数＋1ページに格納できないBINARY型の挿入列数×分岐行数＋インデクス数
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※2
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

注※ 1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※ 2

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※3

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(6)　INSERT（INSERT～VALUES 句指定及び LOCK TABLE ありの場合※1）

(a)　HiRDB/シングルサーバの場合

［前処理］※2＋1＋挿入対象RDエリア数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※2
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1＋挿入対象RDエリア数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

注※1

LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

注※2

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(7)　INSERT（INSERT～SELECT 節指定ありの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋挿入対象RDエリア数※5＋4＋挿入行数※2＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入行の列数×挿入行数※2×分岐行数＋1ページに格納できないBINARY型の挿入列数×挿入行数※2×分岐行数＋挿入行数×インデクス数＋1＋検索対象RDエリア数※5＋検索対象表数＋検索ヒット行数※3
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋挿入対象プラグインが使用する論理ファイル数※4＋検索に利用するプラグインが使用する論理ファイル数※4
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

4＋挿入対象RDエリア数※5＋挿入行数※2＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入行の列数×挿入行数×分岐行数＋1ページに格納できないBINARY型の挿入列数×挿入行数×分岐行数＋挿入行数×インデクス数＋1＋検索対象RDエリア数※5＋検索対象表数＋検索ヒット行数※3
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋挿入対象プラグインが使用する論理ファイル数※4＋検索に利用するプラグインが使用する論理ファイル数※4
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

ページ排他指定の場合は挿入ページ数となります。

注※3

- 複数の表から検索する場合は，個々の表の検索ヒット行数を求めて加算してください。
- インデクス定義列に対する条件を二つ以上指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。
- ページ排他指定の場合は検索ヒットページ数となります。

注※4

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※5

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(8)　INSERT（INSERT～VALUES 句指定，LOCK TABLE なし，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋5＋挿入対象RDエリア数※2＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×分岐行数＋1ページに格納できないBINARY型の挿入列数×分岐行数＋1
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋挿入対象RDエリア数※2＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×分岐行数＋1
ページに格納できないBINARY型の挿入列数×分岐行数＋1
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(9)　INSERT（INSERT～VALUES 句指定，LOCK TABLE あり，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋1＋挿入対象RDエリア数※2
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1＋挿入対象RDエリア数※2
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(10) INSERT（INSERT～SELECT 節指定，及びインデクスキー値無排他方式の場合）

(a) HiRDB/シングルサーバの場合

［前処理］※1＋挿入対象RDエリア数※2＋4＋挿入行数※3＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×挿入行数×分岐行数＋1ページに格納できないBINARY型の挿入列数×挿入行数×分岐行数＋インデクス数＋1＋検索対象RDエリア数※2＋検索対象表数＋検索ヒット行数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋挿入対象プラグインが使用する論理ファイル数※5＋検索に利用するプラグインが使用する論理ファイル数※5
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

(b) HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(c) HiRDB/パラレルサーバの場合（バックエンドサーバ）

挿入対象RDエリア数※2＋4＋挿入行数※3＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×挿入行数×分岐行数＋1ページに格納できないBINARY型の挿入列数×挿入行数×分岐行数＋インデクス数＋1＋検索対象RDエリア数※2＋検索対象表数＋検索ヒット行数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋挿入対象プラグインが使用する論理ファイル数※5＋検索に利用するプラグインが使用する論理ファイル数※5
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

対象となる表，インデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

注※3

ページ排他指定の場合は挿入ページ数となります。

注※4

- 複数の表から検索する場合は，個々の表の検索ヒット行数を求めて加算してください。
- インデクス定義列に対する条件を二つ以上指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。
- ページ排他指定の場合は検索ヒットページ数となります。

注※5

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(11)　UPDATE（LOCK TABLE なしの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋更新対象RDエリア数※5＋4＋更新行数※2＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新前列数×更新前行数×更新前分岐行数＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新後列数×更新後行数×更新後分岐行数＋1ページに格納できないBINARY型の更新前列数×更新前行数×更新前分岐行数＋1ページに格納できないBINARY型の更新後列数×更新後行数×更新後分岐行数＋更新対象インデクス数×2×更新行数＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

更新対象RDエリア数※5＋4＋更新行数※2＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新前列数×更新前行数×更新前分岐行数＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新後列数×更新後行数×更新後分岐行数＋1ページに格納できないBINARY型の更新前列数×更新前行数×更新前分岐行数＋1ページに格納できないBINARY型の更新後列数×更新後行数×更新後分岐行数＋更新対象インデクス数×2×更新行数＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

- 別の表からの検索が必要になる条件式がある場合は，その表の検索ヒット行数を求めて更新行数に加算してください。
- インデクス定義列に対する条件を指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。また，検索ヒット行数を更新行数にしてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。また，検索ヒット行数を更新行数にしてください。
- ページ排他指定の場合は更新ページ数となります。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

LOB 列及び LOB 属性の数だけ繰り返し加算します。

注※5

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(12)　UPDATE（LOCK TABLE ありの場合※1）

(a)　HiRDB/シングルサーバの場合

［前処理］※2＋更新対象RDエリア数※5＋1＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※2
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

更新対象RDエリア数※5＋1＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

注※1

LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

注※2

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

LOB 列及び LOB 属性の数だけ繰り返し加算します。

注※5

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(13)　UPDATE（LOCK TABLE なし，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋更新対象RDエリア数※2＋4＋更新行数※3＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新前列数×更新前行数×更新前分岐行数＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新後列数×更新後行数×更新後分岐行数＋1ページに格納できないBINARY型の更新前列数×更新前行数×更新前分岐行数＋1ページに格納できないBINARY型の更新後列数×更新後行数×更新後分岐行数＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

更新対象RDエリア数※2＋4＋更新行数※3＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新前列数×更新前行数×更新前分岐行数＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新後列数×更新後行数×更新後分岐行数＋1ページに格納できないBINARY型の更新前列数×更新前行数×更新前分岐行数＋1ページに格納できないBINARY型の更新後列数×更新後行数×更新後分岐行数＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

- 別の表からの検索が必要になる条件式がある場合は，その表の検索ヒット行数を求めて更新行数に加算してください。
- インデクス定義列に対する条件を指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。また，検索ヒット行数を更新行数にしてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。また，検索ヒット行数を更新行数にしてください。
- ページ排他指定の場合は更新ページ数となります。表にユニークインデクスが定義されている場合は，更新する行数分を更に加算してください。

注※4

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※5

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(14)　UPDATE（LOCK TABLE あり，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋更新対象RDエリア数※2＋1＋Σ（更新前LOBデータ格納セグメント数）※3
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※4
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

更新対象RDエリア数※2＋1＋Σ（更新前LOBデータ格納セグメント数）※3
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※4
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※4

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(15)　DELETE（LOCK TABLE なしの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋削除対象RDエリア数※5＋1＋削除行数※2＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の削除列数×分岐数×削除行数＋1ページに格納できないBINARY型の削除列数×分岐数×削除行数＋インデクス数×削除行数＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

削除対象RDエリア数※5＋1＋削除行数※2＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の削除列数×分岐数×削除行数＋1ページに格納できないBINARY型の削除列数×分岐数×削除行数＋インデクス数×削除行数＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

- 別の表からの検索が必要になる条件式がある場合は，その表の検索ヒット行数を求めて削除行数に加算してください。
- インデクス定義列に対する条件を指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。また，検索ヒット行数を削除行数にしてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。また，検索ヒット行数を削除行数にしてください。
- ページ排他指定の場合は削除ページ数となります。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※5

対象となる表，インデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

(16)　DELETE（LOCK TABLE ありの場合※1）

(a)　HiRDB/シングルサーバの場合

［前処理］※2＋削除対象RDエリア数※5＋Σ（削除したLOBデータ格納セグメント数）※4＋解放データページ数※6
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※2
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

削除対象RDエリア数※5＋Σ（削除したLOBデータ格納セグメント数※4＋解放データページ数※6
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

注※1

LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

注※2

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※5

対象となるインデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

注※6

ページ内の全データが削除されるページ数です。

(17)　DELETE（LOCK TABLE なし，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋削除対象RDエリア数※2＋1＋削除行数※3＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の削除列数×分岐数×削除行数＋1ページに格納できないBINARY型の削除列数×分岐数×削除行数＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※2
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

削除対象RDエリア数※2＋1＋削除行数※3＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の削除列数×分岐数×削除行数＋1ページに格納できないBINARY型の列数×分岐行数＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

- 別の表からの検索が必要になる条件式がある場合は，その表の検索ヒット行数を求めて削除行数に加算してください。
- インデクス定義列に対する条件を指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。また，検索ヒット行数を削除行数にしてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。また，検索ヒット行数を削除行数にしてください。
- ページ排他指定の場合は削除ページ数となります。表にユニークインデクスが定義されている場合は，削除する行数分を更に加算してください。

注※4

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※5

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(18)　DELETE（LOCK TABLE あり，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋削除対象RDエリア数※2＋Σ（削除したLOBデータ格納セグメント数）※3＋解放データページ数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

削除対象RDエリア数※2＋Σ（削除したLOBデータ格納セグメント数）※3＋解放データページ数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※4

ページ内の全データが削除されるページ数です。

注※5

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(19)　PURGE TABLE

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋対象RDエリア数※3＋3＋表の使用中セグメント数×2＋Σ（（インデクスの使用中セグメント数×2）＋2）※2＋Σ（↑HiRDBファイルにLOBデータを格納しているセグメント数÷64000↑）※4
pd_check_pendingオペランドにUSEを指定，又は省略した場合に加算します。
＋4
対象表が被参照表の場合にさらに加算します。
＋3＋対象表の表格納用RDエリア数＋対象表を参照する参照表数×2＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5＋Σ（参照表の対象RDエリア数※3＋2）※5
　対象表が分割表の場合の場合にさらに加算します。
　＋1＋対象表の表格納用RDエリア数
　対象表を参照する参照表が分割表の場合にさらに加算します。
　＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5
対象表が参照表，又は検査制約を定義した表の場合にさらに加算します。
＋1＋対象表に定義した参照制約の数
＋1＋対象表に定義した検査制約の数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1
pd_check_pendingオペランドにUSEを指定，又は省略した場合に加算します。
＋4
対象表が被参照表の場合にさらに加算します。
＋3＋対象表の表格納用RDエリア数＋対象表を参照する参照表数×2＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5
　対象表が分割表の場合の場合にさらに加算します。
　＋1＋対象表の表格納用RDエリア数
　対象表を参照する参照表が分割表の場合にさらに加算します。
　＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5
対象表が参照表，又は検査制約を定義した表の場合にさらに加算します。

＋1＋対象表に定義した参照制約の数
＋1＋対象表に定義した検査制約の数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

対象RDエリア数※3＋3＋表の使用中セグメント数×2＋Σ（（インデクスの使用中セグメント数×2）＋2）※2＋Σ（↑HiRDBファイルにLOBデータを格納しているセグメント数÷64000↑）※4
pd_check_pendingオペランドにUSEを指定，又は省略した場合，且つ対象表が被参照表の場合に加算します。
＋Σ（参照表の対象RDエリア数※3＋2）※5

注※1

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

注※2

インデクスの数だけ繰り返し加算します。

注※3

対象となる表，インデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

注※4

LOB 列と LOB 属性の LOB 用 RD エリアの HiRDB ファイル数だけ繰り返し加算します。

注※5

対象表を参照する参照表数だけ繰り返し加算します。

(20)　ASSIGN LIST 文（実表からのリスト作成）

(a)　HiRDB/シングルサーバの場合

基表SELECT排他要求数※1＋7＋2
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数＋3＋再作成前のリストデータの使用中セグメント数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

前処理※2

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

前処理※2＋7
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

基表SELECT排他要求数※1＋2
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋3＋再作成前のリストデータの使用中セグメント数

注※1

基表 SELECT 排他要求数については，(1)～(4)のうち基表の条件に合う SELECT 文を参照してください。

注※2

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

(21)　ASSIGN LIST 文（リストからのリスト作成）

(a)　HiRDB/シングルサーバの場合

前処理※＋7＋リストの基表格納RDエリア数＋4
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数＋3＋再作成前のリストデータの使用中セグメント数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

前処理※

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

前処理※＋7
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

リストの基表格納RDエリア数＋4
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋3＋再作成前のリストデータの使用中セグメント数

注※

前処理の排他要求数については，「(25)前処理の排他要求数」を参照してください。

(22)　ASSIGN LIST 文（リストの名前の変更）

(a)　HiRDB/シングルサーバの場合

7
変更後のリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

7
変更後のリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数

(23)　DROP LIST 文（リストの名前を指定した削除：DROP LIST～）

(a)　HiRDB/シングルサーバの場合

7＋リストの基表格納RDエリア数＋3＋リストデータの使用中セグメント数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

7＋リストの基表格納RDエリア数

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

3＋リストデータの使用中セグメント数

(24)　DROP LIST 文（自分の所有するすべてのリストの削除：DROP ALL LIST～）

(a)　HiRDB/シングルサーバの場合

1＋リスト用RDエリア数＋削除リスト数×2＋削除リストデータの使用中セグメント数の合計

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

リスト用RDエリア数＋削除リスト数×2＋削除リストデータの使用中セグメント数の合計

(25)　前処理の排他要求数

(a)　HiRDB/シングルサーバの場合

1＋使用する表数＋使用するビュー表数＋Σ※1（2＋列数×2＋インデクス数×2＋表分割数×4＋Σ※2（インデクス分割数×2）＋2）＋13
ユーザ定義型，又は関数を使用する場合に加算します。
＋1
最適化情報を収集している場合に加算します（１表当たり）。
＋1＋最適化情報作成列数×4＋1
ビュー表を使用する場合に加算します（１ビュー表当たり）。
＋Σ（列数×2＋4＋ビュー表を構成する列数＋2）
LOB列を定義している場合に加算します。
＋LOB列数×2＋表分割数
ユーザ定義型の列を定義している場合に加算します。
ユーザ定義型列数＋Σ（4＋属性数×2）
LOB属性がある場合に加算します。
＋LOB属性数×2＋表分割数
上位型がある場合に加算します。
＋2
関数を使用している場合に，使用関数分加算します。
＋Σ（2＋名称及びパラメタ構成の同じ関数数×2＋パラメタ数×2）
プラグインを使用する場合に，プラグイン関数数分加算します。
＋Σ（2＋パラメタ数×2）＋3＋コンテキスト数×4＋2
ルーチンを使用する場合に加算します。
＋使用するルーチン数＋1
順序数生成子を使用する場合に加算します。
＋使用する順序数生成子数＋1＋使用する順序数生成子数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋使用する表数＋使用するビュー表数
ユーザ定義型，又は関数を使用する場合に加算します。
＋1
ルーチンを使用する場合に加算します。
＋使用するルーチン数＋1
順序数生成子を使用する場合に加算します。
＋使用する順序数生成子数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

Σ※1（2＋列数×2＋インデクス数×2＋表分割数×4＋Σ※2（インデクス分割数×2）＋2）＋13
最適化情報を収集している場合に加算します（1表当たり）。
＋1＋最適化情報作成列数×4＋1
ビュー表を使用する場合に加算します（1ビュー表当たり）。
＋Σ（列数×2＋4＋ビュー表を構成する列数＋2）
LOB列を定義している場合に加算します。
＋LOB列数×2＋表分割数
ユーザ定義型の列を定義している場合に，ユーザ定義型列数分加算します。
＋Σ（4＋属性数×2）
LOB属性がある場合に加算します。
＋LOB属性数×2＋表分割数
上位型がある場合に加算します
＋2
関数を使用している場合に，使用関数分加算します。
＋Σ（2＋名称及びパラメタ構成の同じ関数数×2＋パラメタ数×2）
プラグインを使用する場合に，プラグイン関数数分加算します。
＋Σ（2＋パラメタ数×2）＋3＋コンテキスト数×4＋2
順序数生成子を使用する場合に加算します。
＋1＋使用する順序数生成子数

注※1　表数分加算します。

注※2　表内分割インデクス数分加算します。

付録 D.3　制御系 SQL

(1)　LOCK

(a)　HiRDB/シングルサーバの場合

［前処理］※＋表横分割数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

表横分割数＋1

注※

前処理時の排他資源数については，「付録 D.2(25)前処理の排他要求数」を参照してください。

付録 D.4　ユティリティ及びコマンド

(1)　データベース作成ユティリティ（pdload）

(a)　HiRDB/シングルサーバの場合

209＋α＋表格納用RDエリアの使用セグメント数＋インデクス格納RDエリアの使用セグメント数＋（↑LOB用RDエリアの使用セグメント数÷64000↑）＋β

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

206＋α

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

表格納用RDエリアの使用セグメント数＋インデクス格納RDエリアの使用セグメント数＋（↑LOB用RDエリアの使用セグメント数÷64000↑）＋β

α：MAX（表の列数，表のインデクス数，表の格納 RD エリア数）

β：見積もり値については，マニュアル「HiRDB Version 8 コマンドリファレンス」の「pdload 実行中に必要となるサーバ当たりの排他資源」で，トランザクションを越えた排他制御で使用する資源（pd_lck_until_disconnect_cnt オペランドで指定）を参照してください。

使用セグメント数はデータ格納予定セグメント数で求めます。なお，-d オプション指定時に，既にデータが格納されているセグメント数（使用中セグメント数）の方がデータ格納予定セグメント数より多い場合は，使用中セグメント数で計算してください。

(2)　データベース再編成ユティリティ（pdrorg）

(a)　HiRDB/シングルサーバの場合

209＋α＋表格納用RDエリアの使用中セグメント数＋インデクス格納RDエリアの使用中セグメント数＋（↑LOB用RDエリアの使用セグメント数÷64000↑）＋β

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

206＋α

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

表格納用RDエリアの使用中セグメント数＋インデクス格納RDエリアの使用中セグメント数＋（↑LOB用RDエリアの使用セグメント数÷64000↑）＋β

α：MAX（表の列数，表のインデクス数，表の格納 RD エリア数）

β：見積もり値については，マニュアル「HiRDB Version 8 コマンドリファレンス」の「pdrorg 実行中に必要となるサーバ当たりの排他資源」で，トランザクションを越えた排他制御で使用する資源（pd_lck_until_disconnect_cnt オペランドで指定）を参照してください。

(3)　リバランスユティリティ（pdrbal）

(a)　HiRDB/シングルサーバの場合で，共有モード（-k share）のとき

開始終了処理※1＋前処理※2＋リバランス処理※3＋2

(b)　HiRDB/シングルサーバの場合で，占有モード（-k exclusive）のとき

開始終了処理※1＋前処理※2＋リバランス処理※3

(c)　HiRDB/パラレルサーバの場合で，共有モード（-k share）のとき

- フロントエンドサーバ

開始終了処理※1＋前処理※2＋1

- ディクショナリサーバ

開始終了処理※1＋前処理※2

- バックエンドサーバ

開始終了処理※1＋リバランス処理※3

(d)　HiRDB/パラレルサーバの場合で，占有モード（-k exclusive）のとき

- フロントエンドサーバ

開始終了処理※1＋前処理※2

- ディクショナリサーバ

開始終了処理※1＋前処理※2

- バックエンドサーバ

開始終了処理※1＋リバランス処理※3

注※1

開始終了処理の排他要求数＝ ΣAi ＋ ΣBi

Ai：1 関数の排他要求数

関数の排他要求数については，「付録 D.1(11) CREATE 〔PUBLIC 〕FUNCTION」を参照してください。

Bi：1 手続きの排他要求数

手続きの排他要求数については，「付録 D.1(14) CREATE 〔PUBLIC 〕PROCEDURE」を参照してください。

注※ 2

前処理の排他要求数は次のとおりです。

- HiRDB/シングルサーバの場合：225
- HiRDB/パラレルサーバの場合（フロントエンドサーバ）：5
- HiRDB/パラレルサーバの場合（ディクショナリサーバ）：220

注※ 3

リバランス処理の排他要求数＝表格納用 RD エリアの使用中セグメント数＋インデクス格納用 RD エリアの使用中セグメント数＋（↑ユーザ LOB 用 RD エリアの使用セグメント数÷64000↑）

(4)　pdacunlck コマンド

(a)　HiRDB/シングルサーバの場合

```
0
認可識別子指定の場合に加算します。
＋128
ALL指定の場合に加算します。
＋ディクショナリ表SQL_USERSに登録されているユーザ数
```

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

```
0
認可識別子指定の場合に加算します。
＋128
ALL指定の場合に加算します。
＋ディクショナリ表SQL_USERSに登録されているユーザ数
```

付録 E　pdconfchk コマンドでチェックできるオペランド

pdconfchk コマンドでチェックできるオペランドを次の表に示します。オペランド名はアルファベット順に記載しています。なお，次に示す定義のオペランドについては pdconfchk コマンドでチェックしません。

- UAP 環境定義
- 外部サーバ情報定義（HiRDB External Data Access 機能）
- Hub 最適化情報定義（HiRDB External Data Access 機能）

表 E-1　pdconfchk コマンドでチェックできるオペランド

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_additional_optimize_level	○	×	×	×	×	○
pd_alv_port	○	×	×	×	×	○
pd_apply_search_ats_num	○	×	×	×	×	○
pd_assurance_index_no	○	×	×	×	×	○
pd_assurance_table_no	○	×	×	×	×	○
pd_audit	○	×	×	×	×	○
pd_aud_async_buff_count	○	×	×	×	×	○
pd_aud_async_buff_retry_intvl	○	×	×	×	×	×
pd_aud_async_buff_size	○	×	×	×	×	○
pd_aud_auto_loading	○	×	×	×	×	○
pd_aud_file_name	○	×	×	×	×	○
pd_aud_file_wrn_pnt	○	×	×	×	×	○
pd_aud_max_generation_num	○	×	×	×	×	○
pd_aud_max_generation_size	○	×	×	×	×	○
pd_aud_no_standby_file_opr	○	×	×	×	×	○
pd_aud_sql_data_size	○	×	×	×	×	×
pd_aud_sql_source_size	○	×	×	×	×	×
pd_audit_def_buffer_size	○	×	×	×	×	×
pd_auth_cache_size	○	×	×	×	×	○
pd_auto_vrup	○	×	×	×	×	○
pd_bes_connection_hold	○	×	×	×	×	×
pd_bes_conn_hold_trn_interval	○	×	×	×	×	×

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_bes_shmpool_size	○	×	×	×	×	○
pd_c_library_directory	○	×	×	×	×	○
pd_cancel_down_msgchange	○	×	×	×	×	○
pd_cancel_dump	○	×	×	×	×	○
pd_change_clt_ipaddr	○	×	×	×	×	○
pd_check_pending	○	×	×	×	×	×
pd_client_waittime_over_abort	○	×	×	×	×	○
pd_clt_waittime_over_dump_level	○	×	×	×	×	○
pd_cmdhold_precheck	○	×	×	×	×	○
pd_cmd_process_conwaittime	○	×	×	×	×	○
pd_command_deadlock_priority	○	×	×	×	×	○
pd_connect_errmsg_hide	○	×	×	×	×	○
pd_constraint_name	○	×	×	×	×	×
pd_cwaittime_report_dir	○	×	×	×	×	×
pd_cwaittime_report_size	○	×	×	×	×	×
pd_cwaittime_wrn_pnt	○	×	×	×	×	×
pd_dbbuff_attribute	○	×	×	×	×	○
pd_dbbuff_binary_data_lru	○	×	×	×	×	○
pd_dbbuff_lock_interval	○	×	×	×	×	○
pd_dbbuff_lock_release_detect	○	×	×	×	×	○
pd_dbbuff_lock_spn_count	○	×	×	×	×	○
pd_dbbuff_lru_option	○	×	×	×	×	○
pd_dbbuff_modify	○	×	×	×	×	○
pd_dbbuff_rate_updpage	○	×	×	×	×	○
pd_dbbuff_trace_level	○	×	×	×	×	○
pd_dbbuff_wait_interval	○	×	×	×	×	○
pd_dbbuff_wait_spn_count	○	×	×	×	×	○
pd_dbsync_altwrite_skip	○	×	×	×	×	○
pd_dbsync_lck_release_count	○	×	×	×	×	○
pd_dbsync_point	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_db_access_error_action	○	×	×	×	×	○
pd_db_io_error_action	○	×	×	×	×	×
pd_deadlock_priority_use	○	×	×	×	×	○
pd_debug_info_netstat	○	×	×	×	×	○
pd_dec_sign_normalize	○	×	×	×	×	○
pd_def_buf_control_area_assign	×	×	×	×	×	×
pd_delete_reserved_word_file	○	×	×	×	×	○
pd_deter_restart_on_stop_fail	○	×	×	×	×	○
pd_dfw_awt_process	○	×	×	×	×	○
pd_dfw_syncpoint_skip_limit	○	×	×	×	×	×
pd_dic_shmpool_size	○	×	×	×	×	○
pd_directory_server	○	×	×	×	×	○
pd_down_watch_proc	○	×	×	×	×	○
pd_dump_suppress_watch_time	○	×	×	×	×	○
pd_fes_lck_pool_partition	○	×	×	×	×	○
pd_fes_lck_pool_size	○	×	×	×	×	○
pd_floatable_bes	○	×	×	×	×	×
pd_foreign_server_libpath	○	×	×	×	×	×
pd_ha	○	×	×	×	×	○
pd_ha_acttype	○	×	×	×	×	×
pd_ha_agent	○	×	×	×	×	×
pd_ha_ipaddr_inherit	○	×	×	×	×	○
pd_ha_max_act_guest_servers	○	×	×	×	×	○
pd_ha_max_server_process	○	×	×	×	×	○
pd_ha_mgr_rerun	○	×	×	×	×	○
pd_ha_process_count	○	×	×	×	×	○
pd_ha_resource_act_wait_time	○	×	×	×	×	○
pd_ha_server_process_standby	○	×	×	×	×	×
pd_ha_switch_timeout	○	×	×	×	×	○
pd_ha_transaction	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_ha_trn_queuing_wait_time	○	×	×	×	×	○
pd_ha_trn_restart_retry_time	○	×	×	×	×	○
pd_ha_unit	○	×	×	×	×	×
pd_hash_table_size	○	×	×	×	×	○
pd_hashjoin_hashing_mode	○	×	×	×	×	○
pd_hostname	○	×	×	×	×	○
pd_host_watch_interval	○	×	×	×	×	○
pd_indexlock_mode	○	×	×	×	×	○
pd_ipc_conn_count	○	×	×	×	×	○
pd_ipc_conn_interval	○	×	×	×	×	○
pd_ipc_conn_nblock	○	×	×	×	×	○
pd_ipc_conn_nblock_time	○	×	×	×	×	○
pd_ipc_inet_bufsize	○	×	×	×	×	○
pd_ipc_recv_count	○	×	×	×	×	○
pd_ipc_send_count	○	×	×	×	×	○
pd_ipc_send_retrycount	○	×	×	×	×	○
pd_ipc_send_retrysleeptime	○	×	×	×	×	○
pd_java_archive_directory	○	×	×	×	×	○
pd_java_classpath	○	×	×	×	×	○
pd_java_libpath	○	×	×	×	×	○
pd_java_option	○	×	×	×	×	○
pd_java_routine_stack_size	○	×	×	×	×	○
pd_java_runtimepath	○	×	×	×	×	○
pd_java_stdout_file	○	×	×	×	×	○
pd_jp1_event_level	○	×	×	×	×	○
pd_jp1_event_msg_out	○	×	×	×	×	○
pd_jp1_use	○	×	×	×	×	○
pd_key_resource_type	○	×	×	×	×	○
pd_large_file_use	○	×	×	×	×	○
pd_lck_deadlock_check	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_lck_deadlock_check_interval	○	×	×	×	×	○
pd_lck_deadlock_info	○	×	×	×	×	○
pd_lck_hash_entry	○	×	×	×	×	○
pd_lck_pool_partition	○	×	×	×	×	○
pd_lck_pool_size	○	×	×	×	×	○
pd_lck_queue_limit	○	×	×	×	×	○
pd_lck_release_detect	○	×	×	×	×	○
pd_lck_release_detect_interval	○	×	×	×	×	○
pd_lck_until_disconnect_cnt	○	×	×	×	×	○
pd_lck_wait_timeout	○	×	×	×	×	○
pd_leap_second	○	×	×	×	×	○
pd_list_initialize_timing	○	×	×	×	×	×
pd_lock_uncommited_delete_data	○	×	×	×	×	○
pd_log_auto_expand_size	○	×	×	×	×	○
pd_log_auto_unload_path	○	×	×	×	×	○
pd_log_dual	○	×	×	×	×	○
pd_log_max_data_size	○	×	×	×	×	○
pd_log_rec_leng	○	×	×	×	×	○
pd_log_remain_space_check	○	×	×	×	×	×
pd_log_rerun_reserved_file_open	○	×	×	×	×	○
pd_log_rerun_swap	○	×	×	×	×	○
pd_log_rollback_buff_count	○	×	×	×	×	○
pd_log_rpl_no_standby_file_opr	○	×	×	×	×	○
pd_log_sdinterval	○	×	×	×	×	○
pd_log_singleoperation	○	×	×	×	×	○
pd_log_swap_timeout	○	×	×	×	×	○
pd_log_unload_check	○	×	×	×	×	○
pd_log_write_buff_count	○	×	×	×	×	○
pd_master_file_name	○	×	×	×	×	○
pd_max_access_tables	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_max_access_tables_wrn_pnt	○	×	×	×	×	○
pd_max_add_dbbuff_no	○	×	×	×	×	○
pd_max_add_dbbuff_shm_no	○	×	×	×	×	○
pd_max_ard_process	○	×	×	×	×	×
pd_max_bes_process	○	×	×	×	×	○
pd_max_commit_write_reclaim_no	○	×	×	×	×	○
pd_max_dic_process	○	×	×	×	×	○
pd_max_file_no	○	×	×	×	×	○
pd_max_file_no_wrn_pnt	○	×	×	×	×	○
pd_max_foreign_server	○	×	×	×	×	○
pd_max_list_users	○	×	×	×	×	○
pd_max_list_count	○	×	×	×	×	○
pd_max_list_users_wrn_pnt	○	×	×	×	×	○
pd_max_list_count_wrn_pnt	○	×	×	×	×	○
pd_max_open_holdable_cursors	○	×	×	×	×	○
pd_max_rdarea_no	○	×	×	×	×	○
pd_max_rdarea_no_wrn_pnt	○	×	×	×	×	○
pd_max_recover_process	○	×	×	×	×	○
pd_max_resident_rdarea_no	○	×	×	×	×	○
pd_max_resident_rdarea_shm_no	○	×	×	×	×	○
pd_max_server_process	○	×	×	×	×	○
pd_max_users	○	×	×	×	×	○
pd_max_users_wrn_pnt	○	×	×	×	×	○
pd_mlg_file_size	○	×	×	×	×	○
pd_mlg_msg_log_unit	○	×	×	×	×	○
pd_mlg_port	○	×	×	×	×	○
pd_mode_conf	○	×	×	×	×	○
pd_module_trace_max	○	×	×	×	×	○
pd_module_trace_timer_level	○	×	×	×	×	○
pd_name_fixed_port_lookup	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_name_port	○	×	×	×	×	○
pd_non_floatable_bes	○	×	×	×	×	×
pd_nowait_scan_option	○	×	×	×	×	○
pd_ntfs_cache_disable	○	×	×	×	×	○
pd_oltp_holdcr	○	×	×	×	×	○
pd_optimize_level	○	×	×	×	×	○
pd_overflow_suppress	○	×	×	×	×	○
pd_pageaccess_mode	○	×	×	×	×	○
pd_plugin_ixmk_dir	○	×	×	×	×	×
pd_process_count	○	×	×	×	×	○
pd_process_desktopheap_size	○	×	×	×	×	×
pd_process_terminator	○	×	×	×	×	○
pd_process_terminator_max	○	×	×	×	×	○
pd_pth_trace_max	○	×	×	×	×	○
pd_queue_watch_time	○	×	×	×	×	○
pd_queue_watch_timeover_action	○	×	×	×	×	○
pd_rdarea_extension_timing	○	×	×	×	×	○
pd_rdarea_list_no_wrn_pnt	○	×	×	×	×	○
pd_rdarea_open_attribute	○	×	×	×	×	○
pd_rdarea_open_attribute_use	○	×	×	×	×	○
pd_rdarea_warning_point	○	×	×	×	×	○
pd_rdarea_warning_point_msgout	○	×	×	×	×	○
pd_redo_allpage_put	○	×	×	×	×	○
pd_reduced_check_time	○	×	×	×	×	○
pd_registered_port	○	○	○	○	○	○
pd_registered_port_check	○	○	○	○	○	○
pd_registered_port_level	○	○	○	○	○	○
pd_registry_cache_size	○	×	×	×	×	○
pd_rorg_predict	○	×	×	×	×	○
pd_routine_def_cache_size	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_rpc_bind_loopback_address	○	×	×	×	×	○
pd_rpc_trace	○	×	×	×	×	○
pd_rpc_trace_name	○	×	×	×	×	○
pd_rpc_trace_size	○	×	×	×	×	○
pd_rpl_func_control	○	×	×	×	×	○
pd_rpl_hdepath	○	×	×	×	×	×
pd_rpl_init_start	○	×	×	×	×	○
pd_rpl_reflect_mode	○	×	×	×	×	○
pd_scd_port	○	×	×	×	×	○
pd_sds_shmpool_size	○	×	×	×	×	○
pd_security_host_group	○	×	×	×	○	○
pd_server_cleanup_interval	○	×	×	×	×	○
pd_server_entry_queue	○	×	×	×	×	○
pd_service_port	○	×	×	×	×	○
pd_shared_rdarea_use	○	×	×	×	×	○
pd_shmpool_attribute	○	×	×	×	×	○
pd_space_level	○	×	×	×	×	○
pd_spd_assurance_count	○	×	×	×	×	○
pd_spd_assurance_msg	○	×	×	×	×	○
pd_spd_dual	○	×	×	×	×	○
pd_spd_max_data_size	○	×	×	×	×	○
pd_spd_reduced_mode	○	×	×	×	×	○
pd_spd_reserved_file_auto_open	○	×	×	×	×	○
pd_spd_syncpoint_skip_limit	○	×	×	×	×	○
pd_spool_cleanup	○	×	×	×	×	×
pd_spool_cleanup_interval	○	×	×	×	×	×
pd_spool_cleanup_interval_level	○	×	×	×	×	×
pd_spool_cleanup_level	○	×	×	×	×	×
pd_sql_command_exec_users	○	×	×	×	×	○
pd_sql_dec_op_maxprec	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_sql_object_cache_size	○	×	×	×	×	○
pd_sql_send_buff_size	○	×	×	×	×	○
pd_start_level	○	×	×	×	×	○
pd_start_skip_unit	○	×	×	×	×	○
pd_start_time_out	○	×	×	×	×	○
pd_statistics	○	×	×	×	×	○
pd_stj_file_size	○	×	×	×	×	○
pd_stj_buff_size	○	×	×	×	×	○
pd_standard_sqlstate	○	×	×	×	×	○
pd_sts_file_name_1～7	○	○	○	○	×	×
pd_sts_initial_error	○	×	×	×	×	○
pd_sts_last_active_file	○	×	×	×	×	×
pd_sts_last_active_side	○	×	×	×	×	×
pd_sts_singleoperation	○	×	×	×	×	○
pd_substr_length	○	×	×	×	×	○
pd_sysdef_default_option	○	×	×	×	×	○
pd_syssts_file_name_1～7	○	×	×	×	×	○
pd_syssts_initial_error	○	×	×	×	×	○
pd_syssts_last_active_file	○	×	×	×	×	○
pd_syssts_last_active_side	○	×	×	×	×	○
pd_syssts_singleoperation	○	×	×	×	×	○
pd_system_complete_wait_time	○	×	×	×	×	○
pd_system_dbsync_point	○	×	×	×	×	○
pd_system_id	○	×	×	×	×	○
pd_table_def_cache_size	○	×	×	×	×	○
pd_tcp_inet_bufsize	○	×	×	×	×	○
pd_thdlock_pipe_retry_interval	○	×	×	×	×	○
pd_thdlock_retry_time	○	×	×	×	×	○
pd_thdlock_sleep_func	○	×	×	×	×	○
pd_thdlock_wakeup_lock	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_thdspnlk_spn_count	○	×	×	×	×	○
pd_thread_max_stack_size	○	×	×	×	×	○
pd_tmp_directory	○	○	○	×	×	×
pd_trn_commit_optimize	○	×	×	×	×	○
pd_trn_port	○	×	×	×	×	○
pd_trn_rerun_branch_auto_decide	○	×	×	×	×	○
pd_trn_send_decision_interval	○	×	×	×	×	○
pd_trn_send_decision_intval_sec	○	×	×	×	×	○
pd_trn_send_decision_retry_time	○	×	×	×	×	○
pd_trn_watch_time	○	×	×	×	×	○
pd_type_def_cache_size	○	×	×	×	×	○
pd_uap_exerror_log_dir	○	×	×	×	×	○
pd_uap_exerror_log_param_size	○	×	×	×	×	○
pd_uap_exerror_log_size	○	×	×	×	×	○
pd_uap_exerror_log_use	○	×	×	×	×	○
pd_unit_id	○	×	×	×	×	○
pd_utl_buff_size	○	×	×	×	×	○
pd_utl_exec_mode	○	×	×	×	×	○
pd_utl_exec_time	○	×	×	×	×	○
pd_utl_file_buff_size	○	×	×	×	×	×
pd_view_def_cache_size	○	×	×	×	×	○
pd_watch_pc_client_time	○	×	×	×	×	○
pd_watch_resource	○	×	×	×	×	○
pd_watch_time	○	×	×	×	×	○
pd_work_buff_expand_limit	○	×	×	×	×	○
pd_work_buff_mode	○	×	×	×	×	○
pd_work_buff_size	○	×	×	×	×	○
pd_work_table_option	○	×	×	×	×	○
pdaudload	○	×	×	×	×	○
pdbuffer	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pdcltgrp	○	×	×	×	×	○
pdhagroup	○	×	×	○	×	○
pdhibegin	○	×	×	×	×	○
pdhubopt	○	×	×	×	×	×
pdlogadfg -d spd	○	○	○	○	×	×
pdlogadfg -d sys	○	○	○	○	×	×
pdlogadpf -d spd	○	○	○	○	×	×
pdlogadpf -d sys	○	○	○	○	×	×
pdmlgput	○	×	×	×	×	○
pdplgprm	○	×	×	×	×	×
pdplugin	○	×	×	×	×	○
pdstart	○	×	×	×	○	○
pdstbegin	○	×	×	×	×	○
pdunit	○	×	×	×	○	○
pdwork	○	×	×	×	×	×
pdwork_wrn_pnt	○	×	×	×	×	○
SHMMAX	×	×	×	×	×	×
TZ	×	×	×	×	×	×

（凡例）

○：チェック対象です。

×：チェック対象外です。

文法のチェック：

オペランドの文法が正しいかチェックします。

ファイルのチェック：

システムログファイル，シンクポイントダンプファイル，及びステータスファイルの有無をチェックします。pdconfchk コマンドで-n オプションを指定した場合はファイルのチェックをしません。

アクセス権限のチェック：

HiRDB 管理者にファイルのアクセス権があるかを確認します。

pdconfchk コマンドで-n オプションを指定した場合はアクセス権限のチェックをしません。

pd_tmp_directory オペランドで指定したディレクトリにアクセス権があるかを確認します。
pd_tmp_directory オペランドと TMP 環境変数の指定を共に省略した場合は，アクセス権限のチェックをしません。

重複指定のチェック：

システムログファイル，シンクポイントダンプファイル，及びステータスファイルが重複していないかどうかをチェックします。

ホスト名のチェック：

ホスト名が hosts ファイルに記述されているかをチェックします。

サーバマシン間のチェック（HiRDB/パラレルサーバ限定）：

システムマネジャのサーバマシンを基準に，サーバマシン間の妥当性をチェックします。

付録 F　影響分散スタンバイレス型系切り替え機能使用時に指定できるオペランド一覧（ユニット制御情報定義）

影響分散スタンバイレス型系切り替え機能使用時, ユニット制御情報定義に指定できるオペランドを次に示します。ユニット制御情報定義にはこれら以外のオペランドを指定できません。指定した場合は HiRDB を開始できません（KFPS05618-E メッセージが出力されます）。

- pd_aud_file_name※
- pd_down_watch_proc
- pd_ha_acttype
- pd_ha_agent
- pd_ha_max_act_guest_servers
- pd_ha_max_server_process
- pd_ha_process_count
- pd_ha_resource_act_wait_time
- pd_ha_unit
- pd_hostname
- pd_ipc_conn_nblock_time
- pd_lck_deadlock_check
- pd_lck_deadlock_check_interval
- pd_process_desktopheap_size
- pd_registered_port※
- pd_rpc_trace_name※
- pd_rpl_hdepath
- pd_service_port
- pd_syssts_file_name_1～7
- pd_syssts_initial_error
- pd_syssts_last_active_file
- pd_syssts_last_active_side
- pd_syssts_singleoperation
- pd_tmp_directory
- pd_unit_id

注※

同じサーバマシンに複数の HiRDB ユニットを配置した場合は，ユニットごとに指定値を変更してください。

索引

数字

1：1 スタンバイレス型系切り替え機能　269
1 ビュー表当たりのビュー解析情報長　649
1 文字を表現する最大バイト数　184

A

AND の複数インデクス利用　82
AND の複数インデクス利用の抑止　78, 397

D

DB コネクションサーバ　572
DECIMAL 型の符号正規化機能　64
DESCRIBE 文に関するオペランド　583

F

FETCH に関するオペランド　550, 582

H

HA グループ　201, 209
HA グループに関するオペランド　209
HDEPATH　274
HiRDB Datareplicator〔HiRDB システム定義の変更〕　7, 15
HiRDB Datareplicator 運用ディレクトリ　274
HiRDB Datareplicator に関するオペランド　163, 274, 614
HiRDB External Data Access 機能（環境変数）に関するオペランド　535
HiRDB External Data Access 機能に関するオペランド　167, 435, 514, 614
HiRDB Non Recover FES　201
HiRDB Text Search Plug-in　312, 360, 411
HiRDB 運用ディレクトリ名　186
HiRDB 識別子　51
HiRDB システム定義の作成方法〔HiRDB/シングルサーバ〕　2
HiRDB システム定義の作成方法〔HiRDB/パラレルサーバ〕　9
HiRDB システム定義の種類〔HiRDB/シングルサーバ〕　2
HiRDB システム定義の種類〔HiRDB/パラレルサーバ〕　8
HiRDB システム定義の変更手順〔HiRDB/シングルサーバ〕　5
HiRDB システム定義の変更手順〔HiRDB/パラレルサーバ〕　13
HiRDB システム定義ファイル〔HiRDB/シングルサーバ〕　3
HiRDB システム定義ファイル〔HiRDB/パラレルサーバ〕　9
HiRDB システム定義ファイルの構成〔HiRDB/シングルサーバ〕　4
HiRDB システム定義ファイルの構成〔HiRDB/パラレルサーバ〕　10
HiRDB の開始方法に関するオペランド　56, 601
HiRDB の処理方式に関するオペランド 60, 228, 601
HiRDB のポート番号　51, 186
HiRDB ファイルシステム領域に関するオペランド　145, 609
HiRDB 予約ポート機能　171, 276
Hub 最適化情報定義　585
Hub 最適化情報定義サンプルファイル　596

I

IP アドレス　157, 274

J

JAR ファイル　182, 281
Java Runtime Environment のルートディレクトリ　183, 282
Java 仮想マシンの起動オプション　181
Java 仮想マシンのライブラリ　183, 282
Java クラスパス　182, 281
Java ストアドファンクション　181, 281, 319, 366, 415, 464, 512
Java ストアドプロシジャ　181, 281, 319, 366, 415, 464, 512
Java に関するオペランド　181, 281, 319, 366, 415, 464, 512, 617
JP1 との連携に関するオペランド　165, 614

K

KFPA20009-W　96, 350, 404
KFPS01861-E　58
KFPS05078-I　57
KFPS05623-I　163, 273

L

LOB 用グローバルバッファ　203
LRU 管理方式　135

O

OLTP に関するオペランド　167, 615
OR の複数インデクス利用の優先　78, 397

P

pd_additional_optimize_level　80, 399
pd_alv_port　171
pd_apply_search_ats_num　85
pd_assurance_index_no　143
pd_assurance_table_no　142
pd_aud_async_buff_count　151, 266
pd_aud_async_buff_retry_intvl　151, 266
pd_aud_async_buff_size　150, 265
pd_aud_auto_loading　153
pd_aud_file_name　148, 263
pd_aud_file_wrn_pnt　152
pd_aud_max_generation_num　149, 264
pd_aud_max_generation_size　148, 264
pd_aud_no_standby_file_opr　149
pd_aud_sql_data_size　152, 267
pd_aud_sql_source_size　151, 267
pd_audit　148, 263
pd_audit_def_buffer_size　319, 365, 414
pd_auth_cache_size　310, 357, 409
pd_auto_vrup　167
pd_bes_conn_hold_trn_interval　333, 532
pd_bes_connection_hold　333, 531
pd_bes_shmpool_size　314, 507
pd_c_library_directory　184, 283
pd_cancel_down_msgchange　72
pd_cancel_dump　113, 245
pd_change_clt_ipaddr　171, 275
pd_check_pending　144
pd_client_waittime_over_abort　113
pd_clt_waittime_over_dump_level　114
pd_cmd_process_conwaittime　85
pd_cmdhold_precheck　69
pd_command_deadlock_priority　103
pd_connect_errmsg_hide　71
pd_constraint_name　144
pd_cwaittime_report_dir　97, 235
pd_cwaittime_report_size　97, 235
pd_cwaittime_wrn_pnt　96, 350, 403
pd_db_access_error_action　132
pd_db_hold_action　133
pd_db_io_error_action　70, 231
pd_dbbuff_attribute　109, 242
pd_dbbuff_binary_data_lru　135
pd_dbbuff_lock_interval　138
pd_dbbuff_lock_release_detect　137
pd_dbbuff_lock_spn_count　138
pd_dbbuff_lru_option　135
pd_dbbuff_modify　136
pd_dbbuff_rate_updpage　140
pd_dbbuff_trace_level　140
pd_dbbuff_wait_interval　139, 254
pd_dbbuff_wait_spn_count　140, 255
pd_dbsync_altwrite_skip　61
pd_dbsync_lck_release_count　308, 355, 457, 505
pd_dbsync_point　60
pd_deadlock_priority_use　103
pd_debug_info_netstat　118
pd_dec_sign_normalize　64
pd_def_buf_control_area_assign　108
pd_delete_reserved_word_file　99
pd_deter_restart_on_stop_fail　163
pd_dfw_awt_process　296, 343, 448, 496
pd_dfw_syncpoint_skip_limit　300, 348, 452, 500
pd_dic_shmpool_size　313, 459
pd_directory_server　166
pd_down_watch_proc　89, 233
pd_dump_suppress_watch_time　115, 245
pd_fes_lck_pool_partition　305, 406
pd_fes_lck_pool_size　303, 405
pd_floatable_bes　401
pd_foreign_server_libpath　514
pd_ha　157
pd_ha_acttype　267
pd_ha_agent　269
pd_ha_ipaddr_inherit　157, 274
pd_ha_max_act_guest_servers　270
pd_ha_max_server_process　271
pd_ha_mgr_rerun　159
pd_ha_process_count　272
pd_ha_resource_act_wait_time　162, 273
pd_ha_server_process_standby　269
pd_ha_switch_timeout　158, 268
pd_ha_transaction　160
pd_ha_trn_queuing_wait_time　160
pd_ha_trn_restart_retry_time　161
pd_ha_unit　268
pd_hash_table_size　82
pd_hashjoin_hashing_mode　81

pd_hb_ary_fec_num 550, 582
pd_hb_db_con 548, 570, 580
pd_hb_db_dis_con 548, 570, 580
pd_hb_e_code 548, 570, 580
pd_hb_e_mode 548, 570, 580
pd_hb_e_size 548, 570, 580
pd_hb_get_lock 582
pd_hb_l_mode 549, 571, 581
pd_hb_l_path 549, 571, 581
pd_hb_l_prm 549, 571, 581
pd_hb_l_prm_size 549, 571, 581
pd_hb_l_size 549, 571, 581
pd_hb_sql_trace 582
pd_hb_use_describe 583
pd_host_watch_interval 90
pd_hostname 226
pd_hub_opt_abs 589
pd_hub_opt_case 588
pd_hub_opt_data_len 589
pd_hub_opt_date 589
pd_hub_opt_datetime 591
pd_hub_opt_datetime_op 591
pd_hub_opt_digits 589
pd_hub_opt_float 593
pd_hub_opt_grouping 588
pd_hub_opt_in_value_num 592
pd_hub_opt_joined_table 588
pd_hub_opt_length 589
pd_hub_opt_like 588
pd_hub_opt_lower_upper_type 590
pd_hub_opt_mod_div_type 590
pd_hub_opt_nchar 592
pd_hub_opt_nest_scalar 593
pd_hub_opt_nullable 592
pd_hub_opt_num 590
pd_hub_opt_on_cnd 588
pd_hub_opt_set_func 588
pd_hub_opt_substr 590
pd_hub_opt_table_num 593
pd_hub_opt_time 589
pd_hub_opt_time_24hour 593
pd_hub_opt_trailing_spc 591
pd_hub_opt_use_zero_string 592
pd_indexlock_mode 104
pd_ipc_conn_count 179
pd_ipc_conn_interval 177
pd_ipc_conn_nblock 175
pd_ipc_conn_nblock_time 176
pd_ipc_inet_bufsize 179, 280
pd_ipc_recv_count 175, 280
pd_ipc_send_count 175, 279
pd_ipc_send_retrycount 174, 279
pd_ipc_send_retrysleeptime 174, 279
pd_java_archive_directory 182, 281
pd_java_castoff 320, 367, 415, 464, 513
pd_java_classpath 182, 281
pd_java_libpath 183, 282
pd_java_option 181
pd_java_routine_stack_size 182
pd_java_runtimepath 183, 282
pd_java_stdout_file 183, 282, 319, 366, 415, 464, 512
pd_jp1_event_level 166
pd_jp1_event_msg_out 166
pd_jp1_use 165
pd_key_resource_type 103
pd_large_file_use 145
pd_lck_deadlock_check 105, 239
pd_lck_deadlock_check_interval 106, 240
pd_lck_deadlock_info 100, 239
pd_lck_hash_entry 306, 354, 407, 456, 505
pd_lck_pool_partition 304, 353, 455, 503
pd_lck_pool_size 302, 351, 454, 502
pd_lck_queue_limit 103
pd_lck_release_detect 101, 238
pd_lck_release_detect_interval 102, 239
pd_lck_until_disconnect_cnt 305, 353, 455, 503
pd_lck_wait_timeout 101, 237
pd_leap_second 185
pd_list_initialize_timing 84
pd_lock_uncommited_delete_data 104
pd_log_auto_expand_size 325, 372, 421, 470, 520
pd_log_auto_unload_path 368, 417, 466, 515
pd_log_dual 321, 367, 416, 465, 514
pd_log_max_data_size 323, 371, 420, 469, 518
pd_log_rec_leng 324, 372, 421, 470, 519
pd_log_remain_space_check 321, 368, 416, 465, 514
pd_log_rerun_reserved_file_open 322, 369, 418, 467, 516
pd_log_rerun_swap 322, 370, 418, 467, 516
pd_log_rollback_buff_count 324, 372, 421, 470, 519
pd_log_rpl_no_standby_file_opr 164
pd_log_sdinterval 327, 375, 424, 473, 523
pd_log_singleoperation 321, 369, 418, 467, 516
pd_log_swap_timeout 322, 370, 419, 468, 517
pd_log_unload_check 322, 370, 419, 468, 517

pd_log_write_buff_count 324, 371, 420, 469, 518
pd_master_file_name 51
pd_max_access_tables 54
pd_max_access_tables_wrn_pnt 92
pd_max_add_dbbuff_no 317, 363, 462, 510
pd_max_add_dbbuff_shm_no 318, 364, 463, 511
pd_max_ard_process 295, 343, 447, 495
pd_max_bes_process 292, 493
pd_max_commit_write_reclaim_no 55
pd_max_dic_process 292, 445
pd_max_file_no 125
pd_max_file_no_wrn_pnt 93
pd_max_foreign_server 167
pd_max_list_count 84
pd_max_list_count_wrn_pnt 94
pd_max_list_users 83
pd_max_list_users_wrn_pnt 94
pd_max_open_holdable_cursors 306, 354, 456, 504
pd_max_rdarea_no 124
pd_max_rdarea_no_wrn_pnt 93
pd_max_recover_process 73, 232
pd_max_resident_rdarea_no 141
pd_max_resident_rdarea_shm_no 142
pd_max_server_process 52, 226
pd_max_users 51
pd_max_users_wrn_pnt 92
pd_mlg_file_size 111
pd_mlg_msg_log_unit 110
pd_mlg_port 170
pd_mode_conf 56
pd_module_trace_max 123, 253, 316, 362, 413, 461, 509
pd_module_trace_timer_level 124, 253, 316, 363, 414, 461, 509
pd_name_fixed_port_lookup 168
pd_name_port 51
pd_non_floatable_bes 401
pd_nowait_scan_option 102
pd_ntfs_cache_disable 146
pd_oltp_holdcr 167
pd_optimize_level 77, 397
pd_overflow_suppress 61
pd_pageaccess_mode 69
pd_plugin_ixmk_dir 366, 512
pd_process_count 293, 341, 395, 445, 493
pd_process_desktopheap_size 63
pd_process_terminator 62
pd_process_terminator_max 62
pd_pth_trace_max 119, 248, 317, 363, 414, 462, 510
pd_queue_watch_time 88
pd_queue_watch_timeover_action 88
pd_rdarea_expand_format 127
pd_rdarea_extension_timing 129
pd_rdarea_list_no_wrn_pnt 95
pd_rdarea_open_attribute 130
pd_rdarea_open_attribute_use 129
pd_rdarea_warning_point 125
pd_rdarea_warning_point_msgout 127
pd_redo_allpage_put 73
pd_reduced_check_time 59
pd_registered_port 171, 276
pd_registered_port_check 173, 277
pd_registered_port_level 173, 277
pd_registry_cache_size 312, 359, 411
pd_rorg_predict 147
pd_routine_def_cache_size 311, 359, 410
pd_rpc_bind_loopback_address 71
pd_rpc_trace 112, 244, 315, 361, 412, 460, 508
pd_rpc_trace_name 112, 244, 315, 361, 412, 460, 508
pd_rpc_trace_size 112, 244, 315, 362, 413, 460, 508
pd_rpl_func_control 164
pd_rpl_hdepath 274
pd_rpl_init_start 163
pd_rpl_reflect_mode 163
pd_scd_port 170
pd_sds_shmpool_size 312, 360
pd_security_host_group 155
pd_server_cleanup_interval 294, 342, 396, 447, 495
pd_server_entry_queue 66, 228
pd_service_port 168, 275
pd_shared_rdarea_use 132
pd_shmpool_attribute 108, 241
pd_space_level 63
pd_spd_assurance_count 326, 374, 422, 471, 521
pd_spd_assurance_msg 326, 373, 422, 471, 521
pd_spd_dual 325, 373, 422, 471, 520
pd_spd_max_data_size 327, 375, 424, 473, 522
pd_spd_reduced_mode 326, 374, 423, 472, 521
pd_spd_reserved_file_auto_open 326, 374, 423, 472, 522
pd_spd_syncpoint_skip_limit 300, 347, 402, 451, 500
pd_spool_cleanup 122, 251

pd_spool_cleanup_interval 119, 249
pd_spool_cleanup_interval_level 120, 250
pd_spool_cleanup_level 122, 252
pd_sql_command_exec_users 100
pd_sql_dec_op_maxprec 65
pd_sql_object_cache_size 107, 308, 356, 408, 458, 506
pd_sql_send_buff_size 181
pd_standard_sqlstate 100
pd_start_level 58
pd_start_skip_unit 59
pd_start_time_out 58
pd_statistics 111
pd_stj_buff_size 112, 243
pd_stj_file_size 111, 243
pd_sts_file_name_1 376, 425, 474, 524
pd_sts_initial_error 328, 378, 427, 476, 525
pd_sts_last_active_file 383, 432, 481, 530
pd_sts_last_active_side 384, 433, 482, 531
pd_sts_singleoperation 332, 382, 431, 480, 529
pd_substr_length 184
pd_sysdef_default_option 167
pd_syssts_file_name_1 256
pd_syssts_initial_error 257
pd_syssts_last_active_file 262
pd_syssts_last_active_side 263
pd_syssts_singleoperation 261
pd_system_complete_wait_time 57
pd_system_dbsync_point 60
pd_system_id 51
pd_table_def_cache_size 309, 357, 408
pd_tcp_inet_bufsize 179, 280
pd_thdlock_pipe_retry_interval 67, 229
pd_thdlock_retry_time 68, 230
pd_thdlock_sleep_func 66
pd_thdlock_wakeup_lock 67, 229
pd_thdspnlk_spn_count 68, 230
pd_thread_max_stack_size 108
pd_tmp_directory 283
pd_trn_commit_optimize 77
pd_trn_port 170
pd_trn_rerun_branch_auto_decide 74
pd_trn_send_decision_interval 75
pd_trn_send_decision_intval_sec 74
pd_trn_send_decision_retry_time 76
pd_trn_watch_time 76
pd_type_def_cache_size 310, 358, 410
pd_uap_exerror_log_dir 98, 236
pd_uap_exerror_log_param_size 99, 237
pd_uap_exerror_log_size 98, 236
pd_uap_exerror_log_use 98, 351, 405
pd_uap_wait 539
pd_unit_id 226
pd_utl_buff_size 180
pd_utl_exec_mode 55
pd_utl_exec_time 86
pd_utl_file_buff_size 180
pd_view_def_cache_size 310, 358, 409
pd_watch_pc_client_time 299, 347, 402
pd_watch_resource 91
pd_watch_time 87, 232
pd_work_buff_expand_limit 298, 345, 450, 498
pd_work_buff_mode 296, 344, 448, 497
pd_work_buff_size 297, 344, 449, 497
pd_work_table_option 82
pdacunlck コマンド 719
PDADDITIONALOPTLVL 563
PDAGGR 563
pdaudload 153
PDAUTORECONNECT 553
PDBESCONHOLD 564
PDBESCONHTI 564
PDBLKF 574
pdbuffer 202
pdchgconf コマンド 6, 14
pdcltgrp 214
PDCLTGRP 554
PDCLTPATH 558, 573
PDCLTRCVADDR 553
PDCLTRCVPORT 552
PDCLTRDNODE 572
pdconfchk コマンド 4, 10
pdconfchk コマンドでチェックできるオペランド 720
PDCURSORLVL 550
PDCWAITTIME 556, 572
PDCWAITTIME オーバ 89, 233
PDDBACCS 562
PDDBLOG 554
PDDBORGUAP 562
PDFESHOST 551
pdhagroup 209
PDHASHTBLSIZE 563
PDHATRNQUEUING 565
pdhibegin 214
PDHOST 551, 572
pdhubopt 436
PDIPC 554

PDKALVL　555
PDKATIME　556
pdlbuffer　539
PDLOCKLIMIT　562
pdlogadfg -d spd　386, 434, 484, 534
pdlogadfg -d sys　385, 433, 483, 533
pdlogadpf -d spd　386, 435, 484, 534
pdlogadpf -d sys　385, 434, 483, 533
pdmlgput　217
PDNAMEPORT　552, 572
PDNETSERVICE　583
PDNODELAYACK　556
pdplgprm　387, 435, 485, 535
pdplugin　216
PDPRMTRC　559, 574
PDPRMTRCSIZE　559, 574
PDRCCOUNT　553
PDRCINTERVAL　553
pdreclaim コマンド（-p オプション）の最大同時実行数　55
PDRECVMEMSIZE　555
PDREPPATH　560
PDSENDMEMSIZE　555
PDSERVICEGRP　552
PDSERVICEPORT　552
PDSQLEXECTIME　559
PDSQLOPTLVL　562
PDSQLTEXTSIZE　559
PDSQLTRACE　559, 573
PDSQLTRCOPENMODE　561
PDSRVTYPE　552
pdstart　194
pdstart コマンドの完了待ち時間　57
pdstbegin　210
PDSWAITTIME　557
PDTCPCONOPT　555
pdtrcmgr　560
PDTRCMODE　560
PDTRCPATH　560
PDUAPERLOG　559, 573
PDUAPEXERLOGPRMSZ　564
PDUAPEXERLOGUSE　564
PDUAPREPLVL　560
pdunit　186
pdvrup コマンドの自動起動　167
PDVWOPTMODE　561
pdwork　384, 482, 532
pdwork_wrn_pnt　93
pipe ファイル　101, 238
putenv 形式　17

R

RD エリアに関するオペランド　124, 607
RD エリアのオープン契機　129
RD エリアの最大数　124
RD エリアの閉塞状態のチェック　69
RPC トレース情報に関するオペランド　112, 244, 315, 361, 412, 460, 508, 611

S

set 形式　17
SHMMAX　216, 284
SPsetup.bat　4
SQLSTATE の細分化に関するオペランド　100, 604
SQL オブジェクト長の計算式　644
SQL オブジェクト用バッファ長 107, 308, 356, 408, 458, 506
SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式　643
SQL 拡張最適化オプション　80, 399
SQL からのコマンド実行に関するオペランド　100, 604
SQL 最適化オプション　77, 397
SQL 実行時間警告出力機能に関するオペランド　96, 235, 350, 403, 617
SQL 実行時間警告情報ファイル　96, 350, 404
SQL トレースファイル　558, 573
SQL の最適化に関するオペランド 77, 397, 562, 603
SQL 予約語削除機能に関するオペランド　99, 604
SQL 予約語削除ファイル　99
Sun Java System Directory Server　166

T

TIME_WAIT 状態　555
TIMEOUT　58
TZ　217

U

UAP 環境定義　537
UAP 環境定義の追加又は変更手順〔HiRDB/シングルサーバ〕　7
UAP 環境定義の追加又は変更手順〔HiRDB/パラレルサーバ〕　15
UAP 統計レポート機能　560

W

WITHOUT LOCK NOWAIT 指定の検索　102

あ

空き領域の再利用機能　142
値式に対する結合条件適用機能　80, 400
後処理プロセス　62
アンロード状態のチェックを解除する運用　323, 370, 419, 468, 517
アンロードログファイルの出力先ディレクトリ　368, 417, 466, 515

い

一括入力最大ページ数　208
一相コミット　77
一相最適化　77
インデクスキー値の排他資源の作成方法　103
インデクスキー値無排他　104
インデクス情報ファイル　366, 512
インデクス用グローバルバッファ　204
インデクス用のローカルバッファ　540
インデクス予約数　143
インデクス利用の抑止　78, 397
インナレプリカ機能に関するオペランド　562
インメモリ RD エリア　141
インメモリデータ処理に関するオペランド　141, 608

う

受け入れ可能状態　270
受け入れユニット　272

え

影響分散スタンバイレス型系切り替え機能　269
エラーログファイル　558, 573
演算中のエラー抑止　61

お

オープン契機　129
オプション　17
オペランド一覧　19
オペランド指定値の見積もり式　639
オペランドで指定する内容　600
オペランドの指定形式　17

か

開始準備処理の最大待ち時間　58
開始処理完了の連絡待ち時間　59
開始方法　57
開始モード　57
外部 C ストアドルーチンに関するオペランド　184, 283
外部 HiRDB　551
回復不要 FES　201
外部サーバインタフェーストレース情報に関するオペランド　549, 571, 581
外部サーバエラー情報に関するオペランド　548, 570, 580
外部サーバ情報定義〔HiRDB〕　543
外部サーバ情報定義〔ORACLE〕　577
外部サーバ情報定義〔XDM/RD E2〕　567
外部サーバ数の最大値　167
外部サーバで必要な環境変数　536
外部サーバとの接続及び解除に関するオペランド　548, 570, 580
外部サーバとの接続に関するオペランド　551, 572, 583
拡張 SQL エラー情報出力機能に関するオペランド　98, 236, 351, 405, 564, 603
仮想ネットワーク名　188, 191
片系運転〔サーバ用ステータスファイル〕　332, 382, 431, 480, 529
片系運転〔システムログファイル〕　321, 369, 418, 467, 516
片系運転〔ユニット用ステータスファイル〕　261
簡易セットアップツール　3, 10
監査証跡に出力する SQL データのサイズ　152, 267
監査証跡に出力する SQL 文のサイズ　151, 267
監査証跡ファイルの最大数　149, 264
監査証跡ファイルの最大容量　148, 264
監査証跡ファイル用の HiRDB ファイルシステム領域名　148, 263
監視時間〔ユティリティ〕　86

き

キャッシュ未使用アクセス　146
行の継続　17
共用 RD エリア　132
共用メモリセグメントサイズの上限値　216, 284
共用メモリセグメントの最大数　142
共用メモリに関するオペランド　108, 216, 241, 284, 312, 360, 459, 507, 607

く

空白変換機能　63

空白変換レベル　63
クライアント環境定義との関係〔HiRDB/シングルサーバ〕　5
クライアント環境定義との関係〔HiRDB/パラレルサーバ〕　13
クライアントグループに関するオペランド 214, 554, 615
クライアントグループの接続枠保証機能　214, 554
クライアントの最大待ち時間　113
クラスパス　182, 281
グループ分け高速化処理　78, 397
グローバルバッファが使用する共用メモリ　109, 242
グローバルバッファに関するオペランド　135, 202, 254, 317, 363, 462, 510, 608
グローバルバッファの LRU 管理方式　135
グローバルバッファの動的変更　136

け

系切り替え機能に関するオペランド　157, 267, 565, 613
警告値　321, 368, 416, 465, 514
警告メッセージ　91, 103
ゲスト BES　272
検査制約　144
検査保留状態　144

こ

更新 SQL の作業表作成抑止　78, 397
更新バッファ量抑制機能　301, 348, 453, 501
高速系切り替え機能　269
コストベース最適化モード 2 の適用　80, 400
コマンド形式　17
コマンド引数　17
コミットメント制御　77
コメントの記述　17

さ

サーバ共通定義　287
サーバ構成に関するオペランド　194
サーバ障害監視時間　158, 268
サーバのプロセス数　53, 226
サーバプロセスの異常終了回数の上限値　89, 233
サーバプロセスの沈み込み　88, 89
サーバ名　194
サーバモード　267
サーバ用ステータスファイル（障害発生時）に関するオペランド　328, 378, 427, 476, 525
サーバ用ステータスファイルに関するオペランド 376, 425, 474, 524
最大応答待ち時間　87, 232
最大起動プロセス数　293, 341, 395, 446, 494
最大起動プロセス数〔ディクショナリサーバ〕　292, 445
最大起動プロセス数〔バックエンドサーバ〕 292, 493
最大同時起動サーバプロセス数　52, 226
最大同時接続数　51
最大待ち時間　299, 347, 402
再編成時期予測機能に関するオペランド　147
作業表に関するオペランド 296, 344, 448, 497, 601
作業表用バッファ長　297, 344, 449, 497
作業表用バッファの自動増分　298, 346, 450, 498
作業表用ファイルに関するオペランド 384, 482, 532
作業表用ファイル用の HiRDB ファイルシステム領域 93, 384, 482, 532
参照制約　144
参照制約及び検査制約に関するオペランド　144

し

識別子　51
システム監視に関するオペランド 85, 232, 299, 347, 402, 451, 500, 556, 572, 604
システム共通定義　37
システム構成に関するオペランド　51, 226, 600
システム構成変更コマンド　6, 14
システム定義スカラ関数の定義情報長の求め方　660
システムログファイルに関するオペランド 321, 367, 416, 465, 514, 609
システムログファイルの空き容量監視機能 321, 368, 416, 465, 514
システムログファイルのアンロード状態　322, 370, 419, 468, 517
システムログファイルの片系運転　321, 369, 418, 467, 516
システムログファイルの構成に関するオペランド 385, 433, 483, 533
システムログファイルの自動オープン 322, 369, 418, 467, 516
システムログファイルのスワップ　322, 370, 418, 467, 516
システムログファイルのスワップ完了待ち時間　322, 370, 419, 468, 517
システムログファイルの二重化 321, 367, 416, 465, 514
システムログファイルのファイルグループ 385, 433, 483, 533

システムログファイルの両系運転　321, 369, 418, 467
システムログファイルのレコード長　324, 372, 421, 470, 519
実体のないステータスファイル　256, 377, 426, 475, 524
指定不要になったオペランド　35
自動オープン〔システムログファイル〕　322, 369, 418, 467, 516
自動オープン〔シンクポイントダンプファイル〕　326, 374, 423, 472, 522
自動開始　56
自動再接続機能　553
自動増分〔作業表用バッファ〕　298, 346, 450, 498
自動ログアンロード機能　368, 417, 466, 515
自バックエンドサーバでのグループ化，ORDER BY，DISTINCT 集合関数処理　78, 397
絞込み検索に関するオペランド　83, 604
終了モード　57
縮退運転　326, 374, 423, 472, 522
縮退起動に関するオペランド　58, 601
手動開始　56
ジョインを含む SQL 文の外部サーバ実行の抑止　80, 400
常駐プロセス数　293
常駐プロセス数〔シングルサーバ〕　341
常駐プロセス数〔ディクショナリサーバ〕　445
常駐プロセス数〔バックエンドサーバ〕　493
常駐プロセス数〔フロントエンドサーバ〕　395
シンクポイントダンプの取得間隔　327, 375, 424, 473, 523
シンクポイントダンプの有効化　326, 373, 422, 471, 521
シンクポイントダンプファイルに関するオペランド　325, 373, 422, 471, 520, 610
シンクポイントダンプファイルの構成に関するオペランド　386, 434, 484, 534
シンクポイントダンプファイルの自動オープン　327, 375, 423, 472, 522
シンクポイントダンプファイルの縮退運転　326, 374, 423, 472, 521
シンクポイントダンプファイルの世代数　326, 374, 423, 471, 521
シンクポイントダンプファイルの二重化　325, 373, 422, 471, 520
シンクポイントダンプファイルのバッファ長　327, 375, 424, 473, 522
シンクポイントダンプファイルのファイルグループ　386, 434, 484, 534
シンクポイントダンプ有効化のスキップ回数監視機能　300, 348, 403, 452, 500
シングルサーバ定義　335
シングルサーバ用共用メモリサイズ　312, 360

す

スカラ演算を含むキー条件の適用　78, 398
スケジューラプロセスのポート番号　186
スタックサイズ　108
スタック領域長〔Java〕　182
スタンバイ型とスタンバイレス型を混合した構成〔HiRDB システム定義例〕　633
ステータスファイル　256, 376, 425, 474, 524
ステータスファイルに関するオペランド　610
ステータスファイルの片系運転〔サーバ用ステータスファイル〕　332, 382, 431, 480, 529
ステータスファイルの片系運転〔ユニット用ステータスファイル〕　261
スナップショット方式　69
スレッド間ロックの解放通知方式　67, 229

せ

セキュリティに関するオペランド　147, 263, 319, 365, 414, 612
セグメント使用率　125
接続枠保証機能　214
全面回復処理に関するオペランド　73, 232, 602
全面回復処理の並列実行プロセス数　73, 232

た

タイムアウト情報　100, 239
タイムゾーン　217
探索高速化条件の導出　78, 398

ち

直積を含む SQL 文の外部サーバ実行の強制　80, 400

つ

通信処理に関するオペランド　168, 275, 554, 616

て

定義の関係〔HiRDB/シングルサーバ〕　4
定義の関係〔HiRDB/パラレルサーバ〕　12
定義例　619
ディクショナリサーバ定義　439
ディクショナリサーバ用共用メモリサイズ　313, 459

ディレクトリサーバ連携機能に関するオペランド 166, 614
データ収集用サーバの分離機能 78, 397
データ転送に関するオペランド 574
データベース再編成ユティリティ〔排他資源数〕 717
データベース作成ユティリティ〔排他資源数〕 717
データベースの更新ログに関するオペランド 554
データ用のローカルバッファ 540
テーブルスキャン強制 78, 397
デッドロック情報 100, 239
デッドロックの優先順位 103
デファードライト処理の並列 WRITE 機能 296, 343, 448, 496
デファードライトトリガでの更新ページ出力比率 208

と

統計情報に関するオペランド 111, 210, 243, 611
統計ログ 111
統計ログバッファ長 112, 243
統計ログファイル(pd_stj_file_size)の見積もり式639
統計ログファイルの最大容量 112, 243
同時アクセス可能実表数 54, 92
同時実行最大数に関するオペランド 51, 226, 600
導出表の条件繰り込み機能 78, 398
動的 SQL トレースファイル 560
動的追加用共用メモリセグメント数の上限値 318, 364, 463, 511
動的追加用グローバルバッファ数の上限値 317, 363, 462, 510
トラブルシュート情報に関するオペランド 113, 245, 316, 362, 413, 461, 509, 558, 573, 582, 611
トラブルシュート情報の削除 122, 251
トラブルシュート情報の取得 113, 245
トラブルシュート情報の取得〔ネットワーク情報〕118
トランザクションキャンセル時のプロセスダウンメッセージ変更機能 72
トランザクションキューイング機能 160
トランザクション決着処理に関するオペランド 74, 603
トランザクションサーバプロセスのポート番号 186
トランザクションの同期点処理 76
トレース取得コマンド 560

に

二相コミット 77
入出力エラー 70, 231

ね

ネストループジョイン強制 77, 397
ネストループジョイン優先 77, 397

は

バージョンアップに関するオペランド 167, 615
バージョンによって省略値が異なるオペランド 33
排他解除検知インターバル時間 102, 239
排他管理用領域 101, 238
排他資源数 305, 353, 455, 504
排他資源数の見積もり 662
排他制御〔ORACLE〕 582
排他制御で使用する共用メモリ領域 302, 303, 352, 405, 454, 502
排他制御に関するオペランド 100, 237, 302, 351, 405, 454, 502, 562, 582, 606
排他制御用プール 302, 303, 352, 405, 454, 502
排他制御用プールサイズ 302, 303, 351, 502
排他の解除を検知する方法 101, 238
排他待ち限界経過時間 101, 237
排他待ち時間 101, 237
排他待ちユーザ数 103
バックエンドサーバ接続保持機能に関するオペランド 333, 531, 564, 612
バックエンドサーバ接続保持時間 333, 532
バックエンドサーバ定義 487
バックエンドサーバ用共用メモリサイズ 314, 507
ハッシュジョイン，副問合せのハッシュ実行 80, 400
ハッシュ表サイズ 82
バッチファイル 4
バッファサイズ 206
バッファに関するオペランド 107, 308, 356, 408, 458, 506, 606
バッファの確保方式 296, 344, 448, 497
バッファ面数〔グローバルバッファ〕 205
バッファ面数〔ローカルバッファ〕 541

ひ

日付・時刻に関するオペランド 185, 217, 615
非同期 READ 機能 295, 343, 447, 495
非同期 READ プロセス数 295
ビュー解析情報用バッファ長 310, 358, 409
表定義キャッシュサイズの求め方 657
表定義情報バッファサイズ 656
表定義情報用バッファ長 309, 357, 408
表定義情報用バッファ長 (pd_table_def_cache_size) の見積もり式 655

表又はインデクスの予約数に関するオペランド　142, 608
表予約数　142

ふ

ファイルグループ〔システムログファイル〕　385, 433, 483, 533
ファイルグループ〔シンクポイントダンプファイル〕　386, 434, 484, 534
複数インデクス利用の強制　78, 397
複数の SQL オブジェクト作成　77, 397
プラグイン　312, 360, 411
プラグインインデクスの遅延一括作成に関するオペランド　366, 512
プラグインが使用する共用メモリ　387, 435, 485, 535
プラグイン関数の定義情報長の求め方　660
プラグイン提供関数からの一括取得機能　78, 398
プラグインに関するオペランド　216, 387, 435, 485, 535, 615
フラグ引数　17
プリフェッチ機能　207, 541
フロータブルサーバ　401
フロータブルサーバ候補数の拡大　77, 397
フロータブルサーバ対象拡大　77, 397
フロータブルサーバ対象限定　78, 397
プロセス間メモリ通信機能　554
プロセス数の最大値　53, 226
プロセスに関するオペランド　292, 341, 395, 445, 493, 600
プロセスの異常終了回数監視機能　89, 233
フロントエンドサーバ定義　389
文法規則　17

へ

閉塞状態のチェック　69
ページアクセス方式　69

ほ

ポート番号　51
ポート番号の範囲を指定　171, 276
ホスト BES　272
ホスト間監視時間間隔　90
ホスト名　186, 194

ま

前処理の排他要求数　716
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称　51
マルチコネクションアドレス機能　195, 198

み

未決着状態のトランザクション　74

む

無応答障害　86
無条件に生成する，外部サーバで実行できる探索高速化条件の導出の抑止　80, 400

め

メッセージキュー監視機能　88
メッセージキュー監視時間　88
メッセージの出力抑止機能に関するオペランド　217, 615
メッセージログサーバプロセスのポート番号　186
メッセージログファイルに関するオペランド　110, 610
メッセージログファイルの最大容量　111

も

文字コードに関するオペランド　184, 617
モニタモード　267

ゆ

有効化処理　300, 347, 402, 451, 500
有効保証世代数　326, 374, 423, 472, 521
ユーザ権限情報用バッファ長　310, 357, 409
ユーザサーバホットスタンバイ　269
ユーザ定義型　310, 358, 410
ユーザ定義型情報用バッファ長　310, 358, 410
優先順位　32
ユティリティの最大同時実行数　55
ユティリティの実行監視時間　86
ユティリティの通信用バッファ長　180
ユティリティのファイル用バッファ長　180
ユニット監視プロセスのポート番号　186
ユニット構成に関するオペランド　186
ユニット識別子　186, 194, 226
ユニット制御情報定義　219
ユニット用ステータスファイル（障害発生時）に関するオペランド　257
ユニット用ステータスファイルに関するオペランド　256

り

リスト　84
リスト作成数　94, 95
リスト作成ユーザ数　94
リストの初期化時期　84
リソースの使用率　91
リバランスユティリティ〔排他要求数〕　718

る

ルーチン制御用オブジェクト長の計算式　644
ルーチン定義情報長の求め方　659
ルーチン定義情報用バッファ長　311, 359, 410
ルーチン定義情報用バッファ長
（pd_routine_def_cache_size）の見積もり式　659

れ

レジストリ情報用バッファ長　312, 359, 411

ろ

ローカルバッファ　539
ローカルバッファに関するオペランド　617
ログ出力バッファ面数　324, 371, 420, 469, 518
ログ入出力バッファ長　323, 371, 420, 469, 518
論理ファイル名　256, 376, 425, 474, 524

わ

ワークファイル出力先ディレクトリの変更　283
ワークファイル出力先ディレクトリの変更に関するオペランド　618