

Cosminexus V11 アプリケーションサーバ リファ レンス 定義編(サーバ定義)

文法書

3021-3-J16-80

前書き

■ 対象製品

マニュアル「アプリケーションサーバ & BPM/ESB 基盤 概説」の前書きの対象製品の説明を参照してください。

■ 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

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

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

Intel は、Intel Corporation またはその子会社の商標です。

Linux は、Linus Torvalds 氏の米国およびその他の国における登録商標です。

Microsoft, Internet Explorer, Microsoft Edge, Windows, Windows Server は、マイクロソフト企業グループの商標です。

Oracle(R), Java , MySQL 及び NetSuite は、Oracle, その子会社及び関連会社の米国及びその他の国における登録商標です。

Red Hat, and Red Hat Enterprise Linux are registered trademarks of Red Hat, Inc. in the United States and other countries. Linux(R) is the registered trademark of Linus Torvalds in the U.S. and other countries.

Red Hat, および Red Hat Enterprise Linux は、米国およびその他の国における Red Hat, Inc.の登録商標です。Linux(R)は、米国およびその他の国における Linus Torvalds 氏の登録商標です。

UNIX は、The Open Group の登録商標です。

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。

Eclipse は、開発ツールプロバイダのオープンコミュニティである Eclipse Foundation, Inc.により構築された開発ツール統合のためのオープンプラットフォームです。

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

■ 発行

2026 年 1 月 3021-3-J16-80



■ 著作権

All Rights Reserved. Copyright (C) 2020, 2026, Hitachi, Ltd.

変更内容

変更内容(3021-3-J16-80) uCosminexus Application Server 11-70, uCosminexus Client 11-70, uCosminexus Developer 11-70, uCosminexus Service Architect 11-70, uCosminexus Service Platform 11-70

追加・変更内容	変更箇所
JDK21 サポートに関する記述を追加した。	2.2.2(4) , 2.2.2(7) , 4.11.2(38) , 12.2.1(7)
次のファイルについて、Windows 環境でのマルチバイト文字の使用に関する記述を追加した。 • Smart Composer で使用するファイル • Manager で使用するファイル • ログの運用で使用するファイル	4.2 , 8.2 , 10.2
次のオプションの「出力形式 (G1GC を使用している場合)」に、JDK21 サポートに関する記述を追加した。 • -XX:[+ -]HitachiVerboseGC • -XX:[+ -]HitachiCommaVerboseGC	14.2
次のオプションの「GC 要因一覧」を示す表に、JDK21 サポートに関する記述を追加した。 • -XX:[+ -]HitachiVerboseGCPrintCause	14.2
次のオプションを追加した。 • -XX:[+ -]AlwaysPreTouch • -XX:SoftMaxHeapSize =<size>	14.5
マニュアル訂正の内容を反映した。	-

単なる誤字・脱字などはお断りなく訂正しました。



はじめに

このマニュアルをお読みになる際の前提情報については、マニュアル「アプリケーションサーバ & BPM/ESB 基盤 概説」のはじめにの説明を参照してください。

目次

前書き	2
変更内容	4
はじめに	5

1 ファイルの概要 14

1.1	ファイルの種類	15
1.2	ファイルの説明の記述形式	18
1.2.1	記述形式	18
1.2.2	使用する記号	18
1.3	ファイル編集時の注意事項	20

2 J2EE サーバで使用するファイル 21

2.1	J2EE サーバで使用するファイルの一覧	22
2.2	J2EE サーバで使用するファイルの詳細	23
2.2.1	hitachi_web.properties (J2EE サーバ用 Web アプリケーションプロパティファイル)	23
2.2.2	usrconf.cfg (J2EE サーバ用オプション定義ファイル)	24
2.2.3	usrconf.properties (J2EE サーバ用ユーザプロパティファイル)	34
2.2.4	server.policy (J2EE サーバ用セキュリティポリシーファイル)	133
2.2.5	criticalList.cfg (保護区リストファイル)	138

3 バッチサーバで使用するファイル 140

3.1	バッチサーバで使用するファイルの一覧	141
3.2	バッチサーバで使用するファイルの詳細	142
3.2.1	usrconf.cfg (バッチサーバ用オプション定義ファイル)	142
3.2.2	usrconf.properties (バッチサーバ用ユーザプロパティファイル)	146
3.2.3	server.policy (バッチサーバ用セキュリティポリシーファイル)	159
3.2.4	criticalList.cfg (保護区リストファイル)	163
3.2.5	usrconf.cfg (バッチアプリケーション用オプション定義ファイル)	165
3.2.6	usrconf.properties (バッチアプリケーション用ユーザプロパティファイル)	170

4 Smart Composer 機能で使用するファイル 172

4.1	Smart Composer 機能で使用するファイルの一覧	173
4.1.1	J2EE アプリケーションを実行するシステムで使用するファイル	173
4.1.2	バッチアプリケーションを実行するシステムで使用するファイル	174
4.2	Smart Composer 機能のコマンドの動作環境設定ファイル	175

4.2.1	cmxserver.properties (サーバ設定プロパティファイル)	175
4.2.2	.cmxrc (クライアント設定プロパティファイル)	176
4.2.3	cmxclient.properties (クライアント共通設定プロパティファイル)	179
4.2.4	lb.properties (負荷分散機定義プロパティファイル)	183
4.3	簡易構築定義ファイル	186
4.3.1	概要	186
4.3.2	簡易構築定義ファイルの指定内容	187
4.3.3	Web システムの定義	196
4.3.4	負荷分散機の定義	197
4.3.5	cookie スイッチングの定義	200
4.3.6	仮想サーバの定義	202
4.3.7	物理ティアの定義	205
4.3.8	サービスユニットの定義	210
4.3.9	論理サーバの定義	213
4.3.10	論理サーバのコンフィグレーション定義	217
4.3.11	ホストの定義	219
4.4	構成変更定義ファイル	222
4.4.1	構成変更定義ファイルの詳細	222
4.5	論理サーバ参照定義ファイル	227
4.5.1	論理サーバ参照定義ファイルの詳細	227
4.6	スケールアウト用ホスト定義ファイル	230
4.6.1	スケールアウト用ホスト定義ファイルの詳細	230
4.7	簡易構築定義ファイルおよび構成変更定義ファイルに指定できるタグ	233
4.7.1	J2EE アプリケーションを実行するシステムで指定できるタグ	233
4.7.2	バッチアプリケーションを実行するシステムで指定できるタグ	245
4.8	システムの構成パターンと定義する論理サーバ	253
4.8.1	J2EE アプリケーションを実行するシステムを構築する場合	253
4.8.2	バッチアプリケーションを実行するシステムを構築する場合	254
4.9	各論理サーバ共通で指定できるパラメタ	256
4.9.1	論理サーバ共通で指定できるパラメタ	256
4.10	論理 Web サーバで指定できるパラメタ	259
4.10.1	Web サーバの J2EE サーバとの連携方法を設定するパラメタ	259
4.10.2	Web サーバ用定義を設定するパラメタ	261
4.10.3	HTTP Server のログの出力方法を設定するパラメタ	267
4.10.4	リバースプロキシを設定するパラメタ	274
4.11	論理 J2EE サーバで指定できるパラメタ	277
4.11.1	J2EE サーバの互換モードを設定するパラメタ	277
4.11.2	J2EE サーバ用ユーザプロパティを設定するパラメタ	278
4.11.3	J2EE サーバ用オプション定義を設定するパラメタ	311

4.11.4	J2EE サーバの JavaVM のシステムプロパティに設定できるパラメタ	313
4.11.5	J2EE サーバの拡張パラメタ	317
4.11.6	バッチサーバ用ユーザプロパティを設定するパラメタ	318
4.11.7	バッチサーバ用オプション定義を設定するパラメタ	327
4.11.8	バッチサーバの JavaVM のシステムプロパティに設定できるパラメタ	329
4.11.9	バッチサーバの拡張パラメタ	330
4.11.10	運用監視エージェントプロパティを設定するパラメタ	332
4.11.11	Management イベント発行用プロパティを設定するパラメタ	332
4.11.12	JP1 連携を設定するパラメタ	333
4.11.13	SecurityManager の使用を設定するパラメタ	337
4.11.14	統合ユーザ管理の使用を設定するパラメタ	338
4.11.15	JavaVM の起動パラメタで使用するファイルを設定するパラメタ	339
4.12	論理パフォーマンストレーサで指定できるパラメタ	341
4.13	論理 CTM ドメインマネージャで指定できるパラメタ	343
4.14	論理 CTM で指定できるパラメタ	345
4.15	論理スマートエージェントで指定できるパラメタ	350
4.16	論理ユーザサーバで指定できるパラメタ	351
4.17	論理ネーミングサービスで指定できるパラメタ	352
5	サーバ管理コマンドで使用するファイル	353
5.1	サーバ管理コマンドで使用するファイルの一覧	354
5.2	サーバ管理コマンドで使用するファイルの詳細	355
5.2.1	usrconf (サーバ管理コマンド用オプション定義ファイル)	355
5.2.2	usrconf.bat (サーバ管理コマンド用オプション定義ファイル)	357
5.2.3	usrconf.properties (サーバ管理コマンド用システムプロパティファイル)	360
6	CJMS プロバイダで使用するファイル	367
6.1	CJMS プロバイダで使用するファイルの一覧	368
6.2	CJMS プロバイダで使用するファイルの詳細	369
6.2.1	admin.properties (管理コマンドプロパティファイル)	369
6.2.2	commonconfig.properties (CJMSP ブローカー共通プロパティファイル)	372
6.2.3	config.properties (CJMSP ブローカー個別プロパティファイル)	373
7	CTM で使用するファイル	380
7.1	CTM で使用するファイルの一覧	381
7.2	CTM で使用するファイルの詳細	382
7.2.1	CTM ユーザ環境変数定義ファイル	382
7.2.2	CTM コマンドオプションファイル	384

8 Manager で使用するファイル 386

- 8.1 Manager で使用するファイルの一覧 387
- 8.2 Manager で使用するファイルの詳細 389
 - 8.2.1 adminagent.properties (運用管理エージェントプロパティファイル) 389
 - 8.2.2 AdminAgentrc (運用管理エージェント自動起動用設定ファイル) 405
 - 8.2.3 adminagentuser.cfg (運用管理エージェント用オプション定義ファイル) 406
 - 8.2.4 adminagent.xml (運用管理エージェント設定ファイル) 408
 - 8.2.5 mngagent.<実サーバ名>.properties (運用監視エージェントプロパティファイル) 411
 - 8.2.6 mserver.properties (Management Server 環境設定ファイル) 413
 - 8.2.7 mserver.cfg (Management Server 用オプション定義ファイル) 424
 - 8.2.8 mserverenv.cfg (Management Server 用環境変数定義ファイル) 427
 - 8.2.9 manager.cfg (Manager 設定ファイル) 429
 - 8.2.10 maction.properties (Management アクション実行用プロパティファイル) 433
 - 8.2.11 Management イベント発行用プロパティファイル 437
 - 8.2.12 Management イベント発行用メッセージ ID リストファイル 439
 - 8.2.13 Management Server 管理ファイル用退避対象定義ファイル 442
 - 8.2.14 .mngsvrutilrc (mngsvrutil コマンドのクライアント側定義ファイル) 443
 - 8.2.15 mngsvrutil.properties (mngsvrutil コマンドのサーバ側定義ファイル) 447
 - 8.2.16 mngsvrutilcl.properties (mngsvrutil コマンドのクライアント側共通定義ファイル) 449
 - 8.2.17 .mngsvrmonitorrc (JP1/IM 連携用モニタ起動コマンドの設定ファイル) 451
 - 8.2.18 setup.cfg (セットアップウィザード用設定ファイル) 452
 - 8.2.19 論理ユーザサーバ定義ファイル 454
- 8.3 JP1/IM 連携用システムログメッセージマッピングファイル 462
 - 8.3.1 mserver.jp1event.system.mapping.properties (Management Server 用メッセージマッピングファイル) 462
 - 8.3.2 manager.jp1event.system.mapping.properties (J2EE サーバ共通用メッセージマッピングファイル) 463
 - 8.3.3 manager.<論理サーバ名>.jp1event.system.mapping.properties (J2EE サーバ個別用メッセージマッピングファイル) 464
 - 8.3.4 JP1 イベントへの変換 465

9 仮想化システムで使用するファイル 468

- 9.1 仮想化システムで使用するファイルの一覧 469
- 9.2 仮想化システムで使用するファイルの詳細 470
 - 9.2.1 <LB 接続情報の識別名>.properties (仮想サーバマネージャ側の負荷分散機接続設定プロパティファイル) 470
 - 9.2.2 rasetup.cfg (rasetup コマンド用設定ファイル) 471
 - 9.2.3 sinaviagent.cfg (サーバ通信エージェント用オプション定義ファイル) 473
 - 9.2.4 sinaviagent.properties (サーバ通信エージェント用プロパティファイル) 479
 - 9.2.5 tier.properties (ティア別プロパティファイル) 480

9.2.6	tierlb.properties (ティア側の負荷分散機接続設定プロパティファイル)	483
9.2.7	version.properties (インタフェースバージョン定義ファイル)	485
9.2.8	vmi.properties (仮想サーバマネージャプロパティファイル)	486
9.2.9	vmiclient.properties (仮想サーバマネージャのクライアント共通設定プロパティファイル)	488
9.2.10	.vmirc (仮想サーバマネージャのクライアント設定プロパティファイル)	489
9.2.11	vmsetup.properties (管理者設定用プロパティファイル)	490
9.2.12	仮想サーバグループ情報ファイル	492
10	ログの運用で使用するファイル	495
10.1	ログの運用で使用するファイルの一覧	496
10.2	ログの運用で使用するファイルの詳細	497
10.2.1	snapshot ログ収集対象定義ファイル	497
11	監査ログの設定で使用するファイル	501
11.1	監査ログの設定で使用するファイルの一覧	502
11.2	監査ログの設定で使用するファイルの詳細	503
11.2.1	監査ログ定義ファイル	503
12	Java アプリケーションで使用するファイル	509
12.1	Java アプリケーションで使用するファイルの一覧	510
12.2	Java アプリケーションで使用するファイルの詳細	512
12.2.1	usrconf.cfg (Java アプリケーション用オプション定義ファイル)	512
12.2.2	usrconf.properties (Java アプリケーション用ユーザプロパティファイル)	519
12.2.3	Java アプリケーションに指定するシステムプロパティ	537
12.2.4	Java アプリケーションのユーザログ用プロパティ設定ファイル	545
13	Web アプリケーションで使用するファイル	548
13.1	Web アプリケーションで使用するファイルの一覧	549
13.2	Web アプリケーションで使用するファイルの詳細	550
13.2.1	コンパイル対象外リストファイル	550
13.2.2	実行結果リストファイル	551
14	JavaVM 起動オプション	553
14.1	JavaVM 拡張オプションの一覧	554
14.2	JavaVM 拡張オプションの詳細	563
	-XX:+Hitachi (一覧表示オプション)	564
	-XX:[+ -]HitachiThreadDump (拡張スレッドダンプ情報出力オプション)	564
	-XX:[+ -]HitachiThreadDumpToStdout (拡張スレッドダンプ標準出力抑止オプション)	577
	-XX:[+ -]HitachiThreadDumpWithHashCode (拡張スレッドダンプハッシュコード出力オプション)	577
	-XX:[+ -]HitachiThreadDumpWithCpuTime (拡張スレッドダンプ CPU 利用時間出力オプション)	578

-XX:[+|-]HitachiThreadDumpWithBlockCount (拡張スレッドダンプブロック回数出力オプション) 579

-XX:HitachiJavaLog (ログファイル名のプリフィックス指定オプション) 579

-XX:HitachiJavaLogFileSize (最大ログファイルサイズ指定オプション) 580

-XX:[+|-]HitachiJavaLogNoMoreOutput (ログファイル入出力エラー発生時指定オプション) 581

-XX:HitachiJavaLogNumberOfFile (最大ログファイル数指定オプション) 582

-XX:[+|-]JavaLogAsynchronous 583

-XX:[+|-]HitachiOutputMilliTime (詳細時間出力オプション) 583

-XX:[+|-]HitachiVerboseGC (拡張 verbosegc 情報出力オプション) 584

-XX:[+|-]HitachiCommaVerboseGC (CSV 出力オプション) 599

-XX:HitachiVerboseGCIntervalTime (拡張 verbosegc 情報出力間隔指定オプション) 613

-XX:[+|-]HitachiVerboseGCPrintCause (GC 要因内容出力オプション) 614

-XX:[+|-]HitachiVerboseGCPrintDate (拡張 verbosegc 情報日付出力オプション) 617

-XX:[+|-]HitachiVerboseGCCpuTime (拡張 verbosegc 情報 CPU 利用時間出力オプション) 617

-XX:[+|-]HitachiVerboseGCPrintTenuringDistribution (Survivor 領域の年齢分布出力オプション) 618

-XX:[+|-]HitachiVerboseGCPrintJVMInternalMemory (C ヒープ情報出力オプション) 619

-XX:[+|-]HitachiVerboseGCPrintThreadCount (スレッド数の出力オプション) 621

-XX:[+|-]HitachiVerboseGCPrintDeleteOnExit (java.io.File.deleteOnExit()が使用するヒープサイズの出力オプション) 622

-XX:[+|-]PrintCodeCacheInfo (コードキャッシュ領域情報出力オプション) 624

-XX:CodeCacheInfoPrintRatio (コードキャッシュ領域使用率指定オプション) 627

-XX:[+|-]PrintCodeCacheFullMessage (コードキャッシュ領域枯渇メッセージ出力オプション) 627

-XX:[+|-]HitachiOutOfMemoryCause (例外発生要因種別出力オプション) 629

-XX:[+|-]HitachiOutOfMemoryStackTrace (スタックトレース出力オプション) 630

-XX:HitachiOutOfMemoryStackTraceLineSize (スタックトレース行サイズ指定オプション) 631

-XX:[+|-]HitachiOutOfMemorySize (メモリサイズ出力オプション) 632

-XX:[+|-]HitachiOutOfMemoryAbort (強制終了オプション) 633

-XX:[+|-]HitachiOutOfMemoryAbortThreadDump (スレッドダンプ出力オプション) 634

-XX:[+|-]HitachiOutOfMemoryAbortThreadDumpWithJHeapProf (クラス別統計情報出力オプション) 635

-XX:[+|-]HitachiOutOfMemoryHandling (OutOfMemory ハンドリングオプション) 636

-XX:HitachiOutOfMemoryHandlingMaxThrowCount (最大発生回数の設定オプション) 639

-XX:[+|-]HitachiJavaClassLibTrace (クラスライブラリのスタックトレース出力オプション) 641

-XX:HitachiJavaClassLibTraceLineSize (クラスライブラリのスタックトレース行サイズ指定オプション) 642

-XX:[+|-]HitachiLocalsInThrowable (例外発生時のローカル変数情報収集オプション) 643

-XX:[+|-]HitachiLocalsInStackTrace (スレッドダンプ出力時のローカル変数出力オプション) 646

-XX:[+|-]HitachiLocalsSimpleFormat (ローカル変数情報の出力フォーマット変更オプション) 648

-XX:[+|-]HitachiTrueTypeInLocals (ローカル変数情報の実型名出力オプション) 649

-XX:HitachiCallToString (ローカル変数情報出力オプション) 650

-XX:[+|-]HitachiFullCore (システムリソース解除オプション) 652

-XX:[+|-]HitachiUseExplicitMemory (明示管理ヒープ機能オプション) 652

-XX:HitachiExplicitHeapMaxSize (Explicit メモリブロックの最大サイズ指定オプション) 653

-XX:HitachiExplicitMemoryLogLevel (明示管理ヒープ機能のログ出力レベル指定オプション) 654

-XX:HitachiExplicitMemoryJavaLog (明示管理ヒープ機能のログファイル出力先指定オプション) 655

-XX:HitachiExplicitMemoryJavaLogFileSize (明示管理ヒープ機能の最大ログファイルサイズ指定オプション) 656

-XX:HitachiExplicitMemoryJavaLogNumberOfFile (明示管理ヒープ機能の最大ログファイル数指定オプション) 657

-XX:[+|-]HitachiExplicitMemoryMoveToTenuredFirst (明示管理ヒープ機能の明示解放機能オプション) 657

-XX:[+|-]HitachiExplicitMemoryAutoReclaim (明示管理ヒープ機能の自動解放機能オプション) 658

-XX:[+|-]HitachiExplicitMemoryCompatibleToV8 (明示管理ヒープ機能のバージョン互換設定オプション) 659

-XX:[+|-]HitachiAutoExplicitMemory (明示管理ヒープ機能の自動配置機能オプション) 660

-XX:HitachiAutoExplicitMemoryFile (明示管理ヒープ機能の自動配置機能ファイルパス指定オプション) 661

-XX:ExplicitMemoryFullGCPolicy (明示管理ヒープ機能の Explicit メモリブロックへのオブジェクト移動制御オプション) 661

-XX:[+|-]ExplicitMemoryUseExcludeClass (明示管理ヒープ機能適用除外クラス指定機能オプション) 662

-XX:ExplicitMemoryExcludeClassListFile (明示管理ヒープ機能適用除外設定ファイルパス指定オプション) 663

-XX:ExplicitMemoryNotExcludeClassListFile (明示管理ヒープ機能適用除外無効設定ファイルパス指定オプション) 665

-XX:[+|-]HitachiExplicitMemoryAutoRefReclaim (HTTP セッションで利用する Explicit ヒープの省メモリ化オプション) 666

-XX:HitachiJITCompileMaxMemorySize (JIT コンパイル時の確保メモリ上限値指定オプション) 667

-XX:[+|-]JITCompilerContinuation (JIT コンパイラ稼働継続機能オプション) 669

-XX:[+|-]UseCompressedOops (圧縮オブジェクトポインター機能で使用する Java オプション) 671

-XX:HitachiThreadLimit (スレッド数の上限値指定オプション) 671

-XX:[+|-]HitachiVerboseGCPrintDirectBuffer (ダイレクトバッファの使用状況出力オプション) 673

14.3 JavaVM で使用するプロパティ 676

jvm.userprf.Enable 677

jvm.userprf.ExtendedSetting 678

jvm.userprf.File 679

jvm.userprf.Limit 680

jvm.userprf.Trace 681

jvm.userprf.LineNumber 682

jvm.userprf.ThrowableName 682

jvm.userprf.ThrowableNameEditMethod 683

jvm.userprf.LogLevel 685

JP.co.Hitachi.soft.jvm.autofinalizer 686

	sun.nio.cs.map	686
14.4	Application Server で指定できる Java HotSpot VM のオプションのデフォルト値	688
14.5	Application Server で指定できる Java HotSpot VM のオプション	690
14.6	JavaVM で使用する環境変数の一覧	701
14.7	JavaVM で使用する環境変数の詳細	702
	JAVACOREDIR	702

付録 704

付録 A	拡張 MIB オブジェクト定義ファイル	705
付録 A.1	拡張 MIB オブジェクト定義ファイルのテーブル	705
付録 B	Web アプリケーション用 DD (web.xml)	781
付録 B.1	拡張子と MIME タイプの対応づけ	781

索引 786

1

ファイルの概要

この章では、アプリケーションサーバで使用するファイルの種類、説明の記述形式、編集時の注意事項について説明します。

なお、この章ではアプリケーションサーバで使用するファイルのうち、サーバに関する定義情報を記載するファイルについて説明します。

J2EE アプリケーションの設定で使用する属性ファイルについては、マニュアル「アプリケーションサーバ リファレンス 定義編(アプリケーション／リソース定義)」の「3. J2EE アプリケーションの設定で使用する属性ファイル」を参照してください。リソースの属性ファイルの定義については、マニュアル「アプリケーションサーバ リファレンス 定義編(アプリケーション／リソース定義)」の「4. リソースの設定で使用する属性ファイル」を参照してください。

1.1 ファイルの種類

アプリケーションサーバでは、実行するアプリケーションの種類によって使用できるファイルが異なります。このマニュアルでは、ファイルを次の5種類に分類しています。

- J2EE アプリケーションを実行するシステムで使用するファイル
- バッチアプリケーションを実行するシステムで使用するファイル
- Java アプリケーションを実行するシステムで使用するファイル
- Web アプリケーションを実行するシステムで使用するファイル
- Web サービスを実行するシステムで使用するファイル

それぞれのファイルについて次の表に示します。

表 1-1 J2EE アプリケーションを実行するシステムで使用するファイル

ファイルの種類	ファイルの説明	参照先マニュアル	参照先
J2EE サーバで使用するファイル	J2EE サーバが使用する JavaVM のプロパティや J2EE サーバの起動オプションなどを設定するファイルです。 例：usrconf.properties, usrconf.cfg※	このマニュアル	2 章
Smart Composer 機能で使用するファイル（サーバ設定プロパティファイル）	Smart Composer 機能のコマンドで構築するシステムについて設定するファイルです。 例：cmxserver.properties, .cmxrc		4 章
サーバ管理コマンドで使用するファイル	サーバ管理コマンド実行時の JavaVM のプロパティやサーバ管理コマンドの動作を設定するファイルです。 例：usrconf, usrconf.bat		5 章
CJMS プロバイダで使用するファイル	CJMS プロバイダ機能を使用する場合に、コネクションサービス、メッセージルーティングサービスなどについて設定するファイルです。 例：commonconfig.properties, config.properties		6 章
CTM で使用するファイル	CTM を使用してリクエストのスケジューリングや負荷分散をする場合に、CTM で使用する環境変数やコマンドのオプションなどを設定するファイルです。 例：CTM ユーザ環境変数定義ファイル なお、CTM は、構成ソフトウェアに Component Transaction Monitor を含む製品だけで利用できます。利用できる製品については、マニュアル「アプリケーションサーバ & BPM/ESB 基盤 概説」の「2.2.1 製品と構成ソフトウェアの対応」を参照してください。		7 章
Manager で使用するファイル	運用管理ドメインを管理するためのプロパティや、環境変数やログの収集先などを設定するためファイルです。 例：adminagent.properties, mserver.properties		8 章

ファイルの種類	ファイルの説明	参照先マニュアル	参照先
仮想化システムで使用するファイル	仮想サーバ上にアプリケーションサーバを構築する場合に、負荷分散機への接続情報や、仮想サーバマネージャのクライアント共通設定プロパティなどを設定するファイルです。 例：tierlb.properties, vmclient.properties		9 章
ログの運用で使用するファイル	アプリケーションサーバのログを収集する場合に、snapshot ログの収集先などを設定するファイルです。 例：snapshotlog.conf		10 章
監査ログの設定で使用するファイル	監査ログのセットアップ時に、監査ログ出力時のログファイルなどを設定するファイルです。 例：auditlog.properties		11 章
統合ユーザ管理で使用するファイル	統合ユーザ管理機能を使用する場合に、JAAS 対応ユーザ管理やシングルサインオンの情報などを設定するファイルです。 例：jaas.conf, ua.conf	アプリケーションサーバ 機能解説 セキュリティ管理機能編	12 章

注※

J2EE サーバのオプション定義ファイル (usrconf.cfg) に設定する製品の JavaVM (以降、JavaVM と呼びます) 拡張オプションについては、「[14. JavaVM 起動オプション](#)」を参照してください。

表 1-2 バッチアプリケーションを実行するシステムで使用するファイル

ファイルの種類	ファイルの説明	参照先
バッチサーバで使用するファイル	バッチサーバが使用する JavaVM のプロパティやバッチサーバの起動オプションなどを設定するファイルです。 例：usrconf.properties, usrconf.cfg※	3 章
Smart Composer 機能で使用するファイル (サーバ設定プロパティファイル)	Smart Composer 機能のコマンドの実行環境を設定したり、共通引数のデフォルト値を設定したりするファイルです。 例：cmxserver.properties, .cmxrc	4 章
サーバ管理コマンドで使用するファイル	サーバ管理コマンド実行時の JavaVM のプロパティやサーバ管理コマンドの動作を設定するファイルです。 例：usrconf, usrconf.bat	5 章
Manager で使用するファイル	運用管理ドメインを管理するためのプロパティや、環境変数やログの収集先などを設定するためファイルです。 例：adminagent.properties, mserver.properties	8 章
ログの運用で使用するファイル	アプリケーションサーバのログを収集する場合に、snapshot ログの収集先などを設定するファイルです。 例：snapshotlog.conf	10 章
監査ログの設定で使用するファイル	監査ログのセットアップ時に、監査ログ出力時のログファイルなどを設定するファイルです。 例：auditlog.properties	11 章

注※

バッチサーバのオプション定義ファイル（usrconf.cfg）に設定する JavaVM 拡張オプションについては、「[14. JavaVM 起動オプション](#)」を参照してください。

表 1-3 Java アプリケーションを実行するシステムで使用するファイル

ファイルの種類	ファイルの説明	参照先
Java アプリケーションで使用するファイル	Java アプリケーションを使用する場合に、JavaVM のプロパティや起動オプションなどを設定するファイルです。 例：usrconf.properties, usrconf.cfg※	12 章

注※

Java アプリケーションのオプション定義ファイル（usrconf.cfg）に設定する JavaVM 拡張オプションについては、「[14. JavaVM 起動オプション](#)」を参照してください。

表 1-4 Web アプリケーションを実行するシステムで使用するファイル

ファイルの種類	ファイルの説明	参照先
Web アプリケーションで使用するファイル	アプリケーション起動時に、アプリケーションに含まれる JSP ファイルをコンパイルする場合に、コンパイル対象外にするファイルや実行結果を出力するファイルを設定するファイルです。	13 章

表 1-5 Web サービスを実行するシステムで使用するファイル

ファイルの種類	ファイルの説明	参照先マニュアル	参照先
Web サービスで使用するファイル	システム共通の動作、またはプロセス固有の動作を設定する場合に設定するファイルです。 例：cjwconf.properties, cjrconf.properties	アプリケーションサーバ Web サービス開発ガイド	10.1, 13.1

1.2 ファイルの説明の記述形式

ファイルの説明の記述形式と使用する記号について説明します。

1.2.1 記述形式

2 章以降、ファイルの説明を次の形式で記述します。

形式

ファイルの記述形式を示します。

ファイルの格納先

ファイルの格納先を示します。

機能

ファイルの機能について説明します。

指定できるキー

ファイルの中で指定できるキーについて説明します。

指定できるパラメタ

ファイルの中で指定できるパラメタについて説明します。

記述例

ファイルの内容の記述例を示します。

注意事項

ファイルを設定または使用する際の注意事項を説明します。

参考

すべてのファイルで上記の項目をすべて説明しているわけではありません。また、上記以外に、各ファイルの固有情報を記載している場合があります。

1.2.2 使用する記号

ファイルの説明は次の表に示す記号および構文要素を使用して記述します。

表 1-6 ファイルの説明で使用している記号

記号	意 味
	横に並べられた複数の項目に対する項目間の区切りを示し、「または」を意味します。 (例) A B

記号	意 味
	A または B を指定することを示します。
{ }	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することを示します。項目が横に並べられ、記号 で区切られている場合は、そのうちの一つを選択します。 (例) {A B C} A, B または C のどれかを指定することを示します。
[]	この記号で囲まれている項目は省略してもよいことを示します。複数の項目が横に並べて記述されている場合には、すべてを省略するか、記号 { } と同じくどれか一つを選択します。 (例 1) [A] 「何も指定しない」か「A を指定する」ことを示します。 (例 2) [B C] 「何も指定しない」か「B または C を指定する」ことを示します。
...	記述が省略されていることを示します。 (例) ABC... ABC の後ろに記述があり、その記述が省略されていることを示します。
< >	この記号で囲まれている項目は、該当する要素やファイルなどを指定することを示します。 (例 1) <プロパティ> プロパティを記述します。またはプロパティが表示されます。 (例 2) <ファイル名> ファイル名を指定します。
...	この記号の直前に示す記号を繰り返し、複数個指定できることを示します。 (例) <プロパティ>... プロパティは複数個、繰り返して指定できます。

注 すべての半角文字を使用してください。

1.3 ファイル編集時の注意事項

ファイルを編集する場合の注意事項を次に示します。なお、システムドライブを C ドライブとして説明します。

定義ファイルを更新する場合

アプリケーションサーバが提供する定義ファイルは、管理者特権で更新する必要があります。管理者特権のないユーザが定義ファイルを更新しても、C:\Program Files 以下のディレクトリにある定義ファイルは更新されません。管理者特権のないユーザが更新したファイルは、次に示すディレクトリ以下に保存されます。

C:\Users<ユーザ名>\AppData\Local\VirtualStore

なお、アプリケーションサーバは管理者特権で起動されるため、管理者特権のないユーザが更新した定義ファイルの内容は無視されます。

Unicode の補助文字を使用する場合

アプリケーションサーバで使用する定義ファイルに、Unicode の補助文字は使用できません。

Unicode の補助文字を使用できない定義の例を示します。

- EAR, WAR, JAR, EJB-JAR, サブレット, JSP, クラス, メソッド, 引数, および変数の名称
- DD 内の各種定義
- そのほか, 各種定義ファイルの設定値

また、実行時の文字エンコーディングに変換できない文字をログファイルへ出力した場合、正しく出力されません。

2

J2EE サーバで使用するファイル

この章では、J2EE サーバで使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

2.1 J2EE サーバで使用するファイルの一覧

J2EE サーバで使用するファイルの一覧を、次の表に示します。

表 2-1 J2EE サーバで使用するファイルの一覧

ファイル名	分類	概要	参照先
hitachi_web.properties	J2EE サーバ用 Web アプリケーションプロパティファイル	Web アプリケーション個別のプロパティを指定します。	2.2.1
usrconf.cfg	J2EE サーバ用オプション定義ファイル	J2EE サーバを実行する JavaVM の起動オプションを指定します。	2.2.2
usrconf.properties	J2EE サーバ用ユーザプロパティファイル	J2EE サーバを実行する JavaVM のシステムプロパティを指定します。	2.2.3
server.policy	J2EE サーバ用セキュリティポリシーファイル	J2EE サーバを実行する JavaVM のセキュリティポリシーを指定します。	2.2.4
criticalList.cfg	保護区リストファイル	保護区としてメソッドキャンセルを禁止するクラスを設定します。	2.2.5

2.2 J2EE サーバで使用するファイルの詳細

2.2.1 hitachi_web.properties (J2EE サーバ用 Web アプリケーションプロパティファイル)

(1) 形式

次のようにキーを指定します。

`<キー名称> = <値>`

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値にスペースを含むパスを指定する場合、パスをダブルクォーテーション (") で囲む必要はありません。

(2) ファイルの格納先

- Windows の場合
 <Web アプリケーションの WEB-INF ディレクトリ>¥
- UNIX の場合
 <Web アプリケーションの WEB-INF ディレクトリ>/

(3) 機能

Web アプリケーション個別のプロパティを指定します。なお、J2EE サーバ全体でのシステムプロパティ設定は、usrconf.properties で指定します。usrconf.properties と WEB-INF/hitachi_web.properties で同じキーが指定されている場合は、WEB-INF/hitachi_web.properties の値が優先されます。

(4) 指定できるキー

指定できるキーとデフォルト値を次に示します。このキーに不正な値を指定した場合、動作は保証されません。

キー名称	内容	デフォルト値
webserver.jsp.keepgenerated	JSP ファイルをコンパイルした結果得られた Java ファイルを保持するかどうかを指定します。	false

キー名称	内容	デフォルト値
	true を指定した場合： Java ファイルを保持します。 false を指定した場合： Java ファイルを保持しません。	
webserver.xml.validate	タグライブラリ・ディスクリプタ（TLD ファイル）が DTD で記述されたスキーマに沿って書かれているかチェックするかどうかを指定します。 true を指定した場合： チェックします。 false を指定した場合： チェックをしません。	true

(5) 記述例

```
webserver.jsp.keepgenerated=false
webserver.xml.validate=true
```

2.2.2 usrconf.cfg（J2EE サーバ用オプション定義ファイル）

(1) 形式

次のようにキーを指定します。

```
<キー名称> = <値>
```

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) <キー名称> = <値> #<コメント>

- 実行時の文字エンコーディングと異なる文字エンコーディングでは記述できません。

(2) ファイルの格納先

- Windows の場合

<Application Server のインストールディレクトリ>%CC%server%usrconf%ejb%<サーバ名称>%

- UNIX の場合
/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/

(3) 機能

J2EE サーバを実行する JavaVM の起動オプションを指定します。

J2EE サーバの稼働中に、このファイルの内容を変更した場合、変更した内容は次に J2EE サーバを起動したときに反映されます。

(4) 指定できるキー

指定できるキーとデフォルト値を次の表に示します。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 2-2 usrconf.cfg に指定できるキーとデフォルト値 (J2EE サーバ)

キー名称	内容	デフォルト	VR
add.jvm.arg	指定されたオプションを使って JavaVM を起動します。 なお、システムプロパティを指定したい場合は、"-D"を利用しないで、usrconf.properties ファイルに指定します。 複数のオプションを指定する場合、同じキー名称を使用して、次のように複数回指定してください。 (指定例) <pre>add.jvm.arg= -Xms256m add.jvm.arg= -Xmx512m</pre> なお、次のように複数のオプションを指定できません。指定をした場合、JavaVM の初期化に失敗します。 (指定例) <pre>add.jvm.arg=-Xms256m -Xmx512m</pre> add.jvm.arg で指定できる JavaVM のオプションについては、「14.1 JavaVM 拡張オプションの一覧」および「14.5 Application Server で指定できる Java HotSpot VM のオプション」を参照してください。	デフォルト値については「2.2.2(5) J2EE サーバでの JavaVM オプションのデフォルト値」を参照してください。	—
add.class.path	コンテナ拡張ライブラリ用のクラスパスを指定します。 なお、スペースを含むパスを指定する場合、パスをダブルクォーテーション (") で囲む必要はありません。 また、ダブルクォーテーションでの指定は有効となりません。	なし	—

キー名称	内容	デフォルト	VR
add.library.path ^{※1}	JNI 用の共有ライブラリを指定します。コンテナ拡張ライブラリが JNI を利用する場合だけ指定します。	なし	—
cpp.library.version	プロセス内で使用する libstdc++ライブラリのバージョンを指定します。なお、このキーは、Linux 用です。 指定できる文字列を次に示します。 6: libstdc++.so.6 ライブラリを使用します。 09-00 以降では、設定できるのは「6」だけです。このキーは Red Hat Enterprise Linux 以外のプラットフォームで設定しても無効です。	6	—
ejb.public.directory	指定されたディレクトリを J2EE サーバの作業ディレクトリとして J2EE サーバを開始します。J2EE サーバのカレントディレクトリを次に示します。 - Windows の場合 <製品の作業ディレクトリ>¥ejb¥<サーバ名称> - UNIX の場合 <製品の作業ディレクトリ>/ejb/<サーバ名称> ディレクトリ名称は絶対パスで指定してください。指定できる文字は、半角英数字、アンダースコア「_」、またはハイフン「-」です。パス長は、アプリケーションサーバの作業ディレクトリの見積もり式で算出した範囲内で指定できます。J2EE サーバの作業ディレクトリについては、マニュアル「アプリケーションサーバ システム構築・運用ガイド」の「付録 C.1 J2EE サーバの作業ディレクトリ」を参照してください。	- Windows の場合 <Application Server のインストールディレクトリ>¥CC¥server¥public - UNIX の場合 /opt/Cosminexus/CC/server/public	—
ejb.server.corefilenum (UNIX 用)	J2EE サーバ再起動時に残す core ファイルの数を 0～16 の整数で指定します。なお、このキーは、UNIX 用です。 次の対象 core ファイルのうち、作成日時が新しいものから指定した数のファイルを残し、それ以外のファイルは削除されます。通常は J2EE サーバダウン時に出力する core ファイル数の 1 を指定します。 対象 core ファイル <製品の作業ディレクトリ>/ejb/<サーバ名称>/core* 再起動時にすべての core ファイルを削除する場合は、0 を指定します。	1	—

キー名称	内容	デフォルト	VR
	- 手動による運用または Management Server を使用する場合は、1 以上を指定します。 対象 core ファイルのうち、上書き、または削除される可能性のある core ファイルについては再起動時にリネームされます。 Linux（プロセス ID 付きの設定）の場合 core.<core 生成プロセス ID> → core.<core 生成プロセス ID>.<core ファイル作成日時> 上記以外の場合 core → core.<core ファイル作成日時> <core ファイル作成日時>の形式は、 yymmddHHMMSS です。 なお、core ファイルのリネーム、または削除に失敗した場合はメッセージ KDJE40047-E を出力し、J2EE サーバ起動処理を中止して異常終了します。 また、次のことに注意してください。 - 対象 core ファイルと同じ名称のファイルを作成した場合はユーザが作成したファイルが削除されます。 - core ファイルの削除は J2EE サーバ再起動時に処理されるため、再起動するまでは定義数以上の core ファイルが出力されている場合があります。 - 定義数+n 分の core ファイルが残るためディスク容量に注意してください (n:再起動されるまでに J2EE サーバが出力する core ファイル数)。		
ejb.server.log.directory	指定されているディレクトリをログファイル※2の出力先として使用します。パス長は、1～200 バイトで指定します。Windows の場合、UNC 名を含むパスは指定できません。UNIX の場合、nfs マウントされたディスク上へのパスは指定できません。 このキーを指定する場合、サーバ開始前にログ出力先ディレクトリを作成してください。存在しないディレクトリが指定された場合は、J2EE サーバ開始時に KDJE40024-E のメッセージが出力され、異常終了します。 ほかの J2EE サーバや Web コンテナサーバ（互換機能）、バッチサーバ、Manager とログ出力先が、同じディレクトリにならないように注意してください。同じディレクトリを指定した場合、動作は保証されません。 なお、このキーを指定して、作業ディレクトリ以外にログを出力する場合、変更先ディレクトリ下	- Windows の場合 <製品の作業ディレクトリ>¥ejb¥<サーバ名称>¥logs - UNIX の場合 <製品の作業ディレクトリ>/ejb/<サーバ名称>/logs	—

キー名称	内容	デフォルト	VR
	のログファイルはサーバをアンセットアップする時に削除されません。ログファイルを削除したい場合には、手動で削除してください。 Windows 環境の JDK21 以降で使用する場合、パスに日本語を含めないでください。指定できる文字は、半角英数字、アンダースコア (_), またはハイフン (-) です。		
ejb.server.log.stdout.filesize	cjstdout.log ファイルのサイズの上限値を 4096～2147483647 の範囲（単位：バイト）の整数で指定します。	1048576	—
ejb.server.log.stderr.filesize	cjstderr.log ファイルのサイズの上限値を 4096～2147483647 の範囲（単位：バイト）の整数で指定します。	1048576	—
jvm.type	使用する JavaVM のタイプを指定します。指定できる値は次のどちらかです。 • server Java HotSpot Server VM が使用されます。 • client Java HotSpot Client VM が使用されます。 J2EE サーバ、または J2EE アプリケーションの開始および停止の頻度が高い開発環境などでは、"client"を指定することで性能が向上する場合があります。指定値に誤りがあった場合は、"server"→"client"の順で JavaVM を検索します。なお、このとき KDJE40020-W メッセージが出力されます。	server	—

(凡例)

— : 08-00 より前のバージョンを示します。

注※1

cjstartsv コマンドまたは cjstartweb コマンドの実行時に add.library.path と OS の環境変数 (LIBPATH など) が両方設定されていた場合、add.library.path の設定が優先されます。

注※2

次に示すログファイルが変更の対象になります。

ログの種類	内容	ログ取得ディレクトリおよびログファイル名
メッセージログ	稼働ログ	• Windows の場合 <ejb.server.log.directory キーの値>*cjmessage?.log • UNIX の場合 <ejb.server.log.directory キーの値>/cjmessage?.log
	ログ稼働ログ	• Windows の場合 <ejb.server.log.directory キーの値>*cjlogger.log • UNIX の場合

2. J2EE サーバで使用するファイル

ログの種類	内容	ログ取得ディレクトリおよびログファイル名
		<ejb.server.log.directory キーの値>/cjlogger.log
ユーザログ	Web サブレットログ	- Windows の場合 <ejb.server.log.directory キーの値>*web_servlet?.log - UNIX の場合 <ejb.server.log.directory キーの値>/web_servlet?.log
	ユーザ出力ログ	- Windows の場合 <ejb.server.log.directory キーの値>*user_out?.log - UNIX の場合 <ejb.server.log.directory キーの値>/user_out?.log
	ユーザエラーログ	- Windows の場合 <ejb.server.log.directory キーの値>*user_err?.log - UNIX の場合 <ejb.server.log.directory キーの値>/user_err?.log
	JavaVM の保守情報および GC のログ	- Windows の場合 <ejb.server.log.directory キーの値>*javalog??.log - UNIX の場合 <ejb.server.log.directory キーの値>/javalog??.log
イベントログ	JavaVM の明示管理ヒープ機能イベントログ	- Windows の場合 <ejb.server.log.directory キーの値>*ehjavalog??.log - UNIX の場合 <ejb.server.log.directory キーの値>/ehjavalog??.log
例外ログ	障害発生時の例外情報	- Windows の場合 <ejb.server.log.directory キーの値>*cjexception?.log - UNIX の場合 <ejb.server.log.directory キーの値>/cjexception?.log
リソースアダプタの保守用ログ	—	- Windows の場合 <ejb.server.log.directory キーの値>*connectors*リソースアダプタの表示名?.log - UNIX の場合 <ejb.server.log.directory キーの値>/connectors/リソースアダプタの表示名?.log
TPBroker のトレース	TPBroker のトレース情報	- Windows の場合 <ejb.server.log.directory キーの値>*TPB 配下 - UNIX の場合 <ejb.server.log.directory キーの値>/TPB 配下

(凡例)

— : 該当しない。

(5) J2EE サーバでの JavaVM オプションのデフォルト値

J2EE サーバでの Java VM オプションのデフォルト値を示します。

(a) V9 互換モードの場合

JDK11 を使用する場合

- -Xms256m
- -Xmx512m
- -XX:MetaspaceSize=128m
- -XX:MaxMetaspaceSize=128m
- -XX:SurvivorRatio=8
- -XX:HitachiJavaLog:<ejb.server.log.directory>
- -XX:HitachiJavaLogFileSize=4m
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:-HitachiThreadDumpToStdout
- -XX:+HitachiOutOfMemoryAbort
- -XX:+HitachiJavaClassLibTrace
- -XX:HitachiJavaClassLibTraceLineSize=1024
- -XX:+HitachiLocalsSimpleFormat
- -XX:+HitachiTrueTypeInLocals
- -XX:+HitachiLocalsInStackTrace
- -XX:+HitachiVerboseGC
- -XX:+HitachiVerboseGCPrintCause
- -XX:+HitachiOutputMilliTime
- -XX:+HitachiUseExplicitMemory
- -XX:HitachiExplicitHeapMaxSize=64m
- -XX:HitachiExplicitMemoryLogLevel:normal
- -XX:HitachiExplicitMemoryJavaLog:<ejb.server.log.directory>
- -XX:HitachiExplicitMemoryJavaLogFileSize=4m
- -XX:-HitachiAutoExplicitMemory
- -XX:+HitachiExplicitMemoryAutoReclaim
- -XX:-HitachiExplicitMemoryCompatibleToV8
- -XX:+HitachiOutOfMemoryHandling

(b) 推奨モードの場合

JDK11 を使用する場合

- -Xms256m
- -Xmx512m
- -XX:MetaspaceSize=128m
- -XX:MaxMetaspaceSize=128m
- -XX:SurvivorRatio=8
- -XX:HitachiJavaLog:<ejb.server.log.directory>
- -XX:HitachiJavaLogFileSize=4m
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:-HitachiThreadDumpToStdout
- -XX:+HitachiOutOfMemoryAbort
- -XX:+HitachiJavaClassLibTrace
- -XX:HitachiJavaClassLibTraceLineSize=1024
- -XX:+HitachiLocalsSimpleFormat
- -XX:+HitachiTrueTypeInLocals
- -XX:+HitachiLocalsInStackTrace
- -XX:+HitachiVerboseGC
- -XX:+HitachiVerboseGCPrintCause
- -XX:+HitachiOutputMilliTime
- -XX:+HitachiUseExplicitMemory
- -XX:HitachiExplicitHeapMaxSize=64m
- -XX:HitachiExplicitMemoryLogLevel:normal
- -XX:HitachiExplicitMemoryJavaLog:<ejb.server.log.directory>
- -XX:HitachiExplicitMemoryJavaLogFileSize=4m
- -XX:-HitachiAutoExplicitMemory
- -XX:+HitachiExplicitMemoryAutoReclaim
- -XX:-HitachiExplicitMemoryCompatibleToV8
- -XX:+HitachiOutOfMemoryHandling
- -XX:ReservedCodeCacheSize=240M

JDK17 以降を使用する場合

- -Xms1g
- -Xmx1g
- -XX:MetaspaceSize=128m
- -XX:MaxMetaspaceSize=128m
- -XX:SurvivorRatio=8
- -XX:HitachiJavaLog:<ejb.server.log.directory>
- -XX:HitachiJavaLogFileSize=4m
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:-HitachiThreadDumpToStdout
- -XX:+HitachiOutOfMemoryAbort
- -XX:+HitachiJavaClassLibTrace
- -XX:HitachiJavaClassLibTraceLineSize=1024
- -XX:+HitachiLocalsSimpleFormat
- -XX:+HitachiTrueTypeInLocals
- -XX:+HitachiLocalsInStackTrace
- -XX:+HitachiVerboseGC
- -XX:+HitachiVerboseGCPrintCause
- -XX:+HitachiOutputMilliTime
- -XX:+HitachiOutOfMemoryHandling
- -XX:ReservedCodeCacheSize=240M

(6) 記述例

- Windows の場合

```
# java vm options
add.jvm.arg=-Xms256m
add.jvm.arg=-Xmx512m
add.jvm.arg=-XX:MetaspaceSize=128m
add.jvm.arg=-XX:MaxMetaspaceSize=128m
add.jvm.arg=-XX:+HitachiVerboseGC
add.jvm.arg=-XX:HitachiVerboseGCIntervalTime=600

# ejb public directory
ejb.public.directory=c:¥workdir

# ejb.server.log.directory
ejb.server.log.directory=c:¥logdir¥servername
```

- UNIX の場合


```
# java vm options
add.jvm.arg=-Xms256m
add.jvm.arg=-Xmx512m
add.jvm.arg=-XX:MetaspaceSize=128m
add.jvm.arg=-XX:MaxMetaspaceSize=128m
add.jvm.arg=-XX:+HitachiVerboseGC
add.jvm.arg=-XX:HitachiVerboseGCIntervalTime=600

# ejb public directory
ejb.public.directory=/opt/workdir

# ejb.server.log.directory
ejb.server.log.directory=/CClogs/server/servername

# ejb.server.corefilenum
ejb.server.corefilenum=3
```

(7) 注意事項

- コマンドプロンプトで設定されている CLASSPATH 環境変数は、J2EE サーバには渡りませんが、その他の環境変数（PATH など）は渡ります。
- OS のロケールと異なる文字エンコーディングで記述しないでください。また、OS のロケールに UTF-8 を使用する場合、BOM 付き UTF-8 では記述しないでください。
- usrconf.properties と usrconf.cfg の add.jvm.arg の -D に同じキーが指定されている場合は、usrconf.properties の値が優先されます。
- usrconf.cfg で設定されていない JavaVM のオプションは JavaVM のデフォルト値になります。
- 次に示すプロパティは "-D" で指定できません。
 - java.endorsed.dirs
 - java.security.policy
 - java.class.path
 - java.library.path
 - java.locale.providers
- add.class.path キーに指定されている値の中に、<cosminexus.home>タグを使用してコンテナ拡張ライブラリ用の JAR を指定しているものがあります。これは Application Server のインストール先を意味しているため、add.class.path キーの記述内容を変更する必要はありません。なお、add.class.path キーにコンテナ拡張ライブラリを追加する場合は、インストールおよびアンインストール時の動作を保証できなくなるおそれがあるので、<cosminexus.home>タグを使用しないでください。
- add.jvm.arg キーで --add-modules、および --module-path オプションを指定しないでください。指定した場合は、動作を保証しません。
- -XX:SoftRefLRUPolicyMSPerMB オプションは、開発環境で Metaspace 領域が圧迫された場合にだけ指定してください。このオプションに 0 を指定すると、すべてのソフト参照が無効になります。ソフ

ト参照は、性能向上のためのキャッシュとして使用されることが多いため、このオプションの指定によってアプリケーションの性能が劣化するおそれがあります。

- Windows で使用する場合、JDK21 より前のバージョンでは、Java のデフォルトの文字コードは Shift_JIS でしたが、JDK21 以降のバージョンからは UTF-8 に変更されました。ただし、usrconf.cfg ファイルについては、JDK のバージョンに関係なく Shift_JIS で記述してください。

2.2.3 usrconf.properties (J2EE サーバ用ユーザプロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

次のようにキーを指定します。

<キー名称> = <値>

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) <キー名称> = <値> #<コメント>

- 記載する文字は Java の仕様に従って、ISO 8859-1 文字エンコーディングを使用してください。
- Java の仕様に従わない形式の場合、J2EE サーバの起動に失敗することがあります。

(例) 不正な Unicode エスケープシーケンスを含む場合 ("¥u"に続く文字列が適切な Unicode 16 進数の値ではない)。

(2) ファイルの格納先

- Windows の場合
<Application Server のインストールディレクトリ>¥CC¥server¥usrconf¥ejb¥<サーバ名称>¥
- UNIX の場合
/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/

(3) 機能

J2EE サーバを実行する JavaVM のシステムプロパティを指定します。

usrconf.properties と usrconf.cfg の add.jvm.arg の-D に同じキーが指定されている場合は、usrconf.properties の値が優先されます。

J2EE サーバの稼働中に、このファイルの内容を変更した場合、変更した内容は次に J2EE サーバを起動したときに反映されます。

(4) J2EE サーバの予約済みキー

J2EE サーバでは、次に示す接頭子で始まるキーを内部的に利用します。このため、アプリケーションでこれらの接頭子で始まるキーを利用してはいけません。

- ejbserver.*
- webserver.*
- appclient.*
- j2eeserver.*
- manager.j2ee.*

(5) J2EE サーバのカスタマイズ用キー

J2EE サーバでは、次に示すシステムプロパティのキーに値を設定することで、J2EE サーバの動作をカスタマイズできます。

ここでは、キーを次の分類に分けて説明します。

- ejbserver.application から始まるキー
- ejbserver.bv から始まるキー
- ejbserver.client から始まるキー
- ejbserver.commonj から始まるキー
- ejbserver.compiler から始まるキー
- ejbserver.connectionpool から始まるキー
- ejbserver.connector から始まるキー
- ejbserver.container から始まるキー
- ejbserver.ctm から始まるキー
- ejbserver.deploy から始まるキー
- ejbserver.distributedtx から始まるキー
- ejbserver.DynamicStubLoading から始まるキー
- ejbserver.ejb から始まるキー
- ejbserver.ext から始まるキー

- ejbserver.http から始まるキー
- ejbserver.instrumentation から始まるキー
- ejbserver.javaee から始まるキー
- ejbserver.jca から始まるキー
- ejbserver.jndi から始まるキー
- ejbserver.jpa から始まるキー
- ejbserver.jta から始まるキー
- ejbserver.logger から始まるキー
- ejbserver.management から始まるキー
- ejbserver.manager から始まるキー
- ejbserver.naming から始まるキー
- ejbserver.rmi から始まるキー
- ejbserver.security から始まるキー
- ejbserver.server から始まるキー
- ejbserver.stateful から始まるキー
- ejbserver.stdoutlog から始まるキー
- ejbserver.watch から始まるキー
- ejbserver.webj2ee から始まるキー
- java から始まるキー
- vbj から始まるキー
- vbroker から始まるキー
- webserver.application から始まるキー
- webserver.connector から始まるキー
- webserver.container から始まるキー
- webserver.context から始まるキー
- webserver.dbsfo から始まるキー
- webserver.errorpage から始まるキー
- webserver.http から始まるキー
- webserver.http2 から始まるキー
- webserver.jsp から始まるキー
- webserver.prf から始まるキー
- webserver.servlet から始まるキー

- webserver.ServletContainerInitializer_jar から始まるキー
- webserver.session から始まるキー
- webserver.sessionManager から始まるキー
- webserver.static から始まるキー
- webserver.work から始まるキー
- webserver.xml から始まるキー
- mail.mime から始まるキー

(a) ejbserver.application から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.application.InitTermProcessClasses	コンテナ拡張ライブラリを使用する場合に、サーバ起動・停止フックのクラス名を指定します。複数のクラスを指定する場合は、","で区切って指定してください。なお、","とクラス名の間には、空白を入れないでください。J2EE サーバ起動時には、指定した順で、サーバ起動・停止フック機能のサーバ起動フックメソッドが呼び出されます。J2EE サーバ終了時には、指定した逆順にサーバ起動・停止フック機能のサーバ停止フックメソッドが呼び出されます。	なし	—
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>* 1.appname	AppName のフィールドに出力するデフォルトのアプリケーション名を 0~16 バイトの半角文字セットで指定します。プログラム上で、AppName のフィールドの値が指定されていない場合、このプロパティの値が AppName のフィールドに出力されます。 使用できる文字セットの定義を満たしているが、値の長さが制限を超えている場合、警告メッセージが出力されて、制限の長さより大きい部分のデータが切り捨てられ、制限の長さまでのデータが使用されます。 "null"という文字列が指定された場合は、空文字列""の指定として扱われ、appname フィールドには何も表示されません。また、このプロパティから appname に"null"という文字列の出力指定はできません。	user_app	—
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>*1.count	ログファイルの面数を次の範囲の整数で指定します。 CJMessageFileHandler を使用した場合： 2~16 ここで指定した面数に到達すると、次の番号は初めの"1"に戻ります。同名の旧ログファイル、または同名のユーザファイルがあった場合は上書きされます。	2	—
ejbserver.application.userlog.CJLogHandler.<	出力する文字列のエンコーディングを 0~1024 バイトで指定します。省略した場合は、デフォルトのエンコーディングが使用されます。	null	—

キー名称	内容	省略値	VR
ハンドラ名称>※ 1.encoding	指定された値が範囲を超えた場合、および"null"という文字列が指定された場合は、encoding の指定なし（encoding にはデフォルト値を使用する）と扱われます。null という名称のエンコードは使用できません。 JavaVM 上で java.nio.charset.Charset.isSupported(String charsetName) が true となる文字セットで指定してください。エンコーディングに指定できる文字セットと指定する文字列については、Java 標準 API の java.nio.charset.Charset などの仕様に従ってください。次に指定値の例を示します。 • US-ASCII : 7 ビット ASCII (ISO646-US/Unicode charset の Basic Latin ブロック) • ISO-8859-1 : ISO Latin Alphabet No. 1 (ISO-LATIN-1) • UTF-8 : 8 ビット UCS 変換形式 • SJIS : Shift-JIS, 日本語 • EUC_JP : JIS X 0201, 0208, 0212, EUC エンコーディング, 日本語 • MS932 : Windows 日本語 (Windows 用の指定値) • JIS0201 : JIS X 0201, 日本語 • JIS0208 : JIS X 0208, 日本語 • JIS0212 : JIS X 0212, 日本語 なお、性能劣化の原因となるため、不要なエンコーディング処理をしないでください。		
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>※ 1.filter	使用するフィルタを、パッケージ名を含むフルネームで指定します。値は 0~4096 バイトのクラス文字セットで指定してください。 指定されたハンドラ名称のハンドラは、ここで指定されたフィルタを Reflection 機能で作成して利用します。 指定された値が範囲を超えた場合、フィルタの指定はなしとみなされます。また、"null"という文字列が指定された場合は、フィルタなし（フィルタは使わない）として扱われます。 "null"という名前のフィルタを作成しないでください。	null	—
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>※ 1.formatter	使用するフォーマッタを、パッケージ名を含むフルネームで指定します。値は 0~4096 バイトのクラス文字セットで指定してください。 指定できる値を次に示します。 CJMessageFileHandler クラスに接続する場合： com.hitachi.software.ejb.application.userlog.CJSimpleFormatter か、またはユーザ作成の Formatter クラスを指定します。 Java2-1.4 標準の SimpleFormatter や XMLFormatter を CJMessageFileHandler に指定しないでください（セパレータが改行であるため、メッセージレコードが 1 行で表示されません）。また、ユーザ作成の Formatter クラスを指定する場合は、改行などの制御コードを含まないメッセージを作成するフォーマッタを作成する必要があります。 Java2-1.4 の Handler クラスに接続する場合： 特に決まりはありません。	null	—

キー名称	内容	省略値	VR
	指定されたハンドラ名称のハンドラは、ここで指定されたフォーマットを Reflection 機能で作成して利用します。 指定された値が範囲を超えた場合、フォーマットの指定はなしとみなされます。また、"null"という文字列が指定された場合は、フォーマットなし（フォーマットは使わない）として扱われます。"null"という名前のフォーマットを作成しないでください。		
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>* ¹ .level	ログの取得レベルの上限を半角英字で指定します。指定は、Java の定数で定義されている値で指定します。大文字と小文字は区別されません。 ここで指定した値よりも低いレベルのメッセージだけがログへ出力されます。指定できるレベルを次に示します。 • OFF：ロギングをオフにするために使われる特殊なレベル • SEVERE：重大な障害を示すメッセージレベル • WARNING：潜在的な問題を示すメッセージレベル • INFO：メッセージを情報として提供するメッセージレベル • CONFIG：静的な構成メッセージのメッセージレベル • FINE：トレース情報を提供するメッセージレベル • FINER：トレースレベル 2（FINE より高い） • FINEST：トレースレベル 3（FINER より高い） • ALL：すべてのメッセージのログを取得 OFF と ALL は特殊なレベルです。例えば、ロガーとハンドラに対して OFF を指定すると、ログのレベルに関係なくログは一切取得されません。ALL を指定すると、すべてのレベルのログが出力されます。	SEVERE	—
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>* ¹ .limit	ログファイルの容量（単位：バイト）を次の範囲の整数で指定します。CJMessageFileHandler を使用した場合： 8192～2147483647 ここで指定した容量まで増加し、指定した容量に到達すると次のログファイルに切り替わります。 トレース共通ライブラリの場合、容量チェックがメッセージの書き込み前であるため、実際の最大容量が設定値より、最大で約 4200 バイトを超えることがあります（1 バイトでも指定した容量に満たなければ、メッセージが書き込まれてしまうため）。 例えば、このキーに 8192 を指定した場合で out1.log の現在使用中の領域が 8000 バイト、空き容量が 192 バイトのときに 4000 バイトのメッセージが出力されると、メッセージは out1.log に出力されます。このため、out1.log は上限値を超えたサイズになります。	1048576	—
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>* ¹ .msgid	MsgID のフィールドに出力するデフォルトのメッセージ ID を 0～21 バイトの半角文字セットで指定します。プログラム上で、MsgID の値が指定されていない場合、このプロパティの値が MsgID のフィールドに出力されます。 使用できる文字セットの定義を満たしているが、値の長さが制限を超えている場合、警告メッセージが出力されて、制限の長さより大きい部分のデータが切り捨てられ、制限の長さまでのデータが使用されません。	0001	—

キー名称	内容	省略値	VR
	"null"という文字列が指定された場合は、空文字列""の指定として扱われ、msgid フィールドには何も表示されません。また、このプロパティから msgid に"null"という文字列の出力指定はできません。		
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>* ¹ .path	ログファイル名のプリフィックスを 1～255 バイトの範囲で指定します。パスは、相対パスで指定します*²。なお、パス名の終端に、半角数値を指定しないでください。 ここで指定したプリフィックスに"xx (1～16 の整数) .log"が付加された値が、トレース情報ファイルの名称になります。 パスに指定された半角英文字の大文字と小文字は区別されません。また、パス値の長さは、次の A+B+C 値の合計が値の範囲に収まるよう指定してください。 • A (Windows の場合) : <ログ出力先ルート (ejb.server.log.directory の値)>¥user¥ (デフォルトは<製品の作業ディレクトリ>¥ejb¥<J2EE サーバ名>¥logs¥user¥) • A (UNIX の場合) : <ログ出力先ルート (ejb.server.log.directory の値)>/user/ (デフォルトは<製品の作業ディレクトリ>/ejb/<J2EE サーバ名>/logs/user/) • B : ユーザの指定したプリフィックスの長さ • C : "xx.log"の 6 文字 このほか、パスを指定するときには次の点に注意してください。 • Properties ファイルには native の 2 バイトコードなどは使用できないため、native2ascii などを利用して Unicode の文字列を取得して指定してください。 • 絶対パスで指定しないでください。また、相対パスの"..*" (UNIX の場合, "../") は使用しないでください。 • パスに日本語文字などを含む場合は、その文字を処理できるロケール設定がされている環境下で JavaVM が動作している必要があります。 • パスをほかのハンドラと同一に設定することはできません。そのような設定にした場合、そのハンドラは作成されません。	user_log	—
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>* ¹ .separator	CJSimpleFormatter を利用して、1 文でメッセージを出力するためのセパレータを 0～1024 バイトの半角文字セットで指定します。セパレータは文字列で指定できます。 使用できる文字セットの定義を満たしているが、値の長さが制限を超えている場合、警告メッセージが出力されて、制限の長さより大きい部分のデータが切り捨てられ、制限の長さまでのデータが使用されません。 トレース共通ライブラリ形式の出力の場合、メッセージ内に次の制御文字を指定しないでください。 • Windows の場合 : "¥r", "¥n" など • UNIX の場合 : "/r", "/n" など これらの文字を指定した場合、出力内容がずれたり、1 レコードが 1 行で出力されなくなったりします。"null"という文字列が指定された場合は、空文字列""の指定として扱われます。そのため、セパレータ	ストローク「 」	—

キー名称	内容	省略値	VR
	に"null"という文字列は使用できません。また、このプロパティからセパレータに"null"という文字列の出力指定はできません。		
ejbserver.application.userlog.loggers	使用するログの名称を宣言するプロパティです。0～65535 バイトで指定します。";"で区切ることで、複数のログ名称を指定できます。また、ログ名称のすぐ後ろに、";"で区切ってそのログで使用する地域対応化用リソースバンドル名称を指定できます。 なお、ログ名称には、java.net または javax.swing のように"."で区切られ、サブシステムのパッケージ名またはクラス名に基づいた名前が推奨されています。^{※3} また、ログの名称には","や";"を使用しないでください。 複数のクラス名を定義しており、指定された値の長さが範囲を超えた場合は、長さの範囲内に収まり、値の規則を満たしたクラス名だけが設定されます。 ここで宣言された名称のログは、CJLogManager クラスのログ構築対象として処理されます (J2EE サーバの起動時にログが構築されます)。 ここで指定しなかったログについては、自動生成されないため、コード上でログを作成してください。	なし	—
ejbserver.application.userlog.Logger.<ログ名称> ^{※3} .filter	ログで使用するフィルタを、パッケージ名を含むフルネームで指定するプロパティです。値は 0～4096 バイトのクラス文字セットで指定してください。 指定されたログ名称のログは、ここで指定されたフィルタを Reflection 機能で作成してメッセージの取捨選択に利用します。 指定された値が範囲を超えた場合、フィルタの指定はなしとみなされます。また、"null"という文字列が指定された場合は、フィルタなし (フィルタを使わない) として扱われます。"null"という名前のフィルタを作成しないでください。	null	—
ejbserver.application.userlog.Logger.<ログ名称> ^{※3} .handlers	指定のログ名称に対して、ハンドラクラスを接続 (addHandler(Handler)) するためのプロパティです。ハンドラクラスは、Reflection 機能で作成するため、0～65535 バイトのパッケージ名を含むフルネームで指定してください。";"で区切ることで複数のハンドラクラスを指定できます。 指定できるハンドラクラスを次に示します。 - com.hitachi.software.ejb.application.userlog.CJMessageFileHandler - java.util.logging.ConsoleHandler - java.util.logging.FileHandler - java.util.logging.SocketHandler - java.util.logging.StreamHandler - java.util.logging.MemoryHandler - ユーザが独自に作成した Handler クラスのパッケージ名を含むフルネーム CJLogHandler 系の場合は、ハンドラクラス名の後ろに";"で区切ってハンドラ名称を指定することで、指定のハンドラ名称の設定で初期化してハンドラクラスを作成し、ログに接続します。Java2-1.4 標	なし	—

キー名称	内容	省略値	VR
	準の Logging クラスについては, ";"で区切ってハンドラ名称を指定しても無効になります。Java2-1.4 標準の Logging クラスの設定方法については, 該当するクラスの仕様に従ってください。 ハンドラ名称は, 半角英数字で始まる 1~1024 バイトのクラス文字セットで指定してください。 複数のクラス名を定義しており, 指定された値の長さが範囲を超えた場合は, 長さの範囲内に収まり, 値の規則を満たしたクラス名だけが設定されます。 ejbserver.application.userlog.loggers でロガー名称が指定されていない場合, ロガーは作成されません。		
ejbserver.application.userlog.Logger.<ロガー名称>* ³ .level	ロガーのログ出力レベルまたは"null"を半角英字で指定します。大文字と小文字は区別されます。ログメッセージを出力するためには, ロガーのログ出力レベルと出力に利用するハンドラの出力レベルをクリアする必要があります。指定できる値を次に示します。 • ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.level に指定できる値 • "null" "null"という名称が指定された場合, ロガーのレベルを null で初期化します。ロガーのレベルが"null"で初期化された場合, そのロガーは親ロガーのレベルを継承します。	SEVERE	—
ejbserver.application.userlog.Logger.<ロガー名称>* ³ .useParentHandlers	ログレコードを受け取ったロガーから, 親ロガーの接続しているハンドラへログレコードを伝播させるかどうかを半角英字で指定します。伝播させる場合は true を, 伝播させない場合は false を指定してください。 このロガーを通過したログの出力レベルは, 親ロガーの接続しているハンドラのレベルにだけ影響されます (親ロガーのレベルが OFF, 親ロガーが接続しているハンドラのレベルが ALL の場合, すべてのメッセージは表示されます)。	true	—

(凡例)

— : 08-00 より前のバージョンを示します。

注

ユーザログ機能についてのプロパティ定義 (ejbserver.application.InitTermProcessClasses キー以外) で使用するそれぞれの文字セットで指定できる値は, 次のとおりです。

• クラス文字セット (クラス名の定義に使用する文字セット)

半角英数字 (a~z, A~Z, 0~9), ピリオド (.), ドル記号 (\$) およびアンダースコア (_)

• 半角文字セット

半角英数字 (a~z, A~Z, 0~9), ピリオド (.), ドル記号 (\$), アンダースコア (_), ハイフン (-), ストローク (|), コロン (:), アンパサンド (&), 単価記号 (@), シャープ (#) およびパーセント (%)

指定した値が, 使用できる文字セットの条件を満たさない, または指定できる値の範囲を満たさない不正な値の場合は, 警告メッセージが出力され, その時点で有効なデフォルトの値が採用されます。その

時点で有効なデフォルトの値とは、<ハンドラ名称>=default の場合のそのキーの値、または「デフォルト値」欄のデフォルト値を指します。

注※1

ハンドラ名称はプロパティの値を区別するために使用します。半角英数字で始まる 1~1024 バイトのクラス文字セットで指定してください。"default"を指定した場合は、すべての CJLogHandler が共通のデフォルト値として認識するプロパティを示します。

注※2

出力先の起点は、次の規則に従ってください。

`<ejb.server.log.directoryの値>%user, または<ejb.server.log.directoryの値>/user`

ejb.server.log.directory の値は、ejb.server.log.directory の仕様に従ってください。J2EE サーバの ejb.server.log.directory のデフォルトの値を次に示します。

- Windows の場合
 <製品の作業ディレクトリ>%ejb%<J2EE サーバ名>%user%<ユーザ指定値>
- UNIX の場合
 <製品の作業ディレクトリ>/ejb/<J2EE サーバ名>/user/<ユーザ指定値>

注※3

ロガー名称は、Logger.getLogger(<ロガー名称>)でロガーのインスタンスを取得するときに指定する"."付きの名称で、あらかじめ、ejbserver.application.userlog.loggers プロパティで宣言する必要があります。ロガー名称については、java.util.logging.Logger の仕様に従ってください。

半角英数字で始まる 1~1024 バイトのクラス文字セットで指定してください。"default"を指定した場合は、すべての CJLogHandler が共通のデフォルト値として認識するプロパティを示します。

なお、ロガー名称で識別されるプロパティは、ユーザログ機能のロガークラスの初期化で使用されます。ロガー名称の最後を.handlers で終了させた場合、ejbserver.application.userlog.Logger.<ロガー名称>.handlers のプロパティ指定時に紛らわしくなるため、ロガー名称は".handlers"で終了させないことを推奨します。

(b) ejbserver.bv から始まるキー

キー名称	内容	省略値	VR
ejbserver.bv.limit.max_validation_message	制約違反発生時のメッセージ補間処理で、文字列長のアラートしきい値を、1~2147483647 までの整数で指定します。指定値がシステムプロパティの値より長い場合は、KDJE60110-W のメッセージを出力し、処理を続行します。数値以外の文字列や範囲外の数値を指定した場合は、KDJE60111-W のメッセージを出力し、デフォルト値が設定されます。	10000	09-85

(c) ejbserver.client から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.client.ctm.RequestPriority	CTM 内のキューに滞留したリクエストを引き出す時のプライオリティ（優先順位）を 1～8 の整数で指定します。指定値が小さいほどプライオリティは高くなります。指定値が誤っている場合、または省略した場合は、4 が設定されます。	4	—

(凡例)

— : 08-00 より前のバージョンを示します。

(d) ejbserver.commonj から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.commonj.WorkManager.non_daemon_work_threads	WorkManager で短寿命 Work の実行を行うスレッドプールの最大プールサイズを指定します 1～65535 の整数で指定します。	10	08-50

(e) ejbserver.compiler から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.compiler.jvm.maxHeapSize	J2EE アプリケーション開始時に起動される javac コマンドのヒープサイズの最大値を指定します。 指定する値は、java コマンドの-Xmx, -Xms オプションが受け付ける形式で指定してください。それ以外の形式で指定された場合の動作は保証されません。 次のように指定してください。 2048～4294966272 の値を指定してください。ただし、ejbserver.compiler.jvm.minHeapSize に指定した以上の値にしてください。 - キロバイト単位で指定する場合は、文字「k」または「K」を付けて指定してください。 - メガバイト単位で指定する場合は、文字「m」または「M」を付けて指定してください。	256m	—

キー名称	内容	省略値	VR
	このプロパティはアプリケーションサーバ 09-00-01 以前で有効です。アプリケーションサーバ 09-00-02 以降は指定しても無視されません。		
ejbserver.compiler.jvm.minHeapSize	J2EE アプリケーション開始時に起動される javac コマンドのヒープサイズの初期値を指定します。 指定する値は、java コマンドの-Xmx, -Xms オプションが受け付ける形式で指定してください。それ以外の形式で指定された場合の動作は保証されません。 次のように指定してください。 1024~-Xmx 指定値の値を指定してください。 - キロバイト単位で指定するには、文字「k」または「K」を付けて指定してください。 - メガバイト単位で指定するには、文字「m」または「M」を付けて指定してください。 このプロパティはアプリケーションサーバ 09-00-01 以前で有効です。アプリケーションサーバ 09-00-02 以降は指定しても無視されません。	32m	—

(凡例)

— : 08-00 より前のバージョンを示します。

(f) ejbserver.connectionpool から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.connectionpool.applicationAuthentication.disabled	コンテナ管理のサインオンの最適化機能を有効にするかどうかを指定します。 true を指定した場合： コンテナ管理のサインオンが最適化された動作となります。ただし、これを指定した場合、アプリケーション管理によるサインオンは使用できません。 false を指定した場合： コンテナ管理のサインオンは最適化されません。コンテナ管理によるサインオン、およびアプリケーション管理によるサインオンを使用できます。	false	—	「機能解説 基本・開発編(コンテナ共通機能)」の「3.14.8 DB Connector のコンテナ管理でのサインオンの最適化」

キー名称	内容	省略値	VR	関連情報
	なお、DB Connector 以外のリソースアダプタを使用する場合には、このプロパティに true を指定することはできません。			
ejbserver.connectionpool.association.enabled	コネクションアソシエーション機能を有効にするかどうかを指定します。 true を指定した場合： コネクションアソシエーション機能を有効にします。 false を指定した場合： コネクションアソシエーション機能を無効にします。	false	—	「機能解説 基本・開発編(コンテナ共通機能)」の「3.14 パフォーマンスチューニングのための機能」
ejbserver.connectionpool.association.enabledDespiteUnshareableSetting	サーブレットや Enterprise Bean の標準 DD の<res-sharing-scope>に Unshareable を指定している場合に、コネクションアソシエーションを行うかどうかを指定します。 true を指定した場合： サーブレットや Enterprise Bean の標準 DD の<res-sharing-scope>に Unshareable を指定している場合でも、コネクションアソシエーションを行います。ただし、ejbserver.connectionpool.association.enabled=true が指定されている必要があります。 false を指定した場合： サーブレットや Enterprise Bean の標準 DD の<res-sharing-scope>に Unshareable を指定している場合、コネクションアソシエーションを行いません。 なお、アプリケーションを新規に開発する場合には、このキーは指定しないでください。	false	—	
ejbserver.connectionpool.sharingOutsideTransactionScope.enabled	アプリケーションサーバが管理するトランザクションの外で複数回コネクションの取得を行ったときのコネクションシェアリングの動作を指定します。なお、リソースアダプタのトランザクションサポートレベルに NoTransaction を指定した場合は、コネクションシェアリングは行われません。 true を指定した場合： コネクションシェアリングを行います。なお、true を指定した場合でも、サーブレットや Enterprise Bean の標準 DD の<res-sharing-scope>に Unshareable を指定している場合には、コネクションシェアリングは行いません。	false	—	

キー名称	内容	省略値	VR	関連情報
	false を指定した場合： アプリケーションサーバが管理するトランザクションの外ではコネクションシェアリングを行いません。			
ejbserver.connectionpool.validation.timeout	コネクション障害検知機能のタイムアウト時間およびコネクション数調節機能によるコネクション削除処理のタイムアウト時間（単位：秒）を 1～2147483647 の整数で指定します。	5	08-00	

(凡例)

－：08-00 より前のバージョンを示します。

空欄：関連情報はありませぬ。

(g) ejbserver.connector から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.connector.logwriter.filenum	リソースアダプタ用ログファイルの面数を 1～16 の整数で指定します。	4	－
ejbserver.connector.logwriter.filesize	リソースアダプタ用ログファイルのサイズ（単位：バイト）を 4096～2147483647 の整数で指定します。	2097152	－
ejbserver.connector.statementpool.clear.backcompat	ステートメントプーリング機能で、ステートメントを再利用する際にステートメントを初期化する内容を指定します。 true を指定した場合： 次の内容が初期化されます。 ・パラメタ ・SQL コマンドリスト ・Statement オブジェクトに関して報告されたすべての警告 false を指定した場合： 次の内容が初期化されます。 ・パラメタ ・SQL コマンドリスト ・Statement オブジェクトに関して報告されたすべての警告 ・データベースから取り出す必要がある行数 ・文字またはバイナリの値を格納する ResultSet 列に対する最大バイト数の制限値	true	08-70

キー名称	内容	省略値	VR
	・ ResultSet オブジェクトが含むことのできる最大行数の制限値 ・ クエリタイムアウト		

(凡例)

－：08-00 より前のバージョンを示します。

(h) ejbserver.container から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.container.audit_trail.enabled	データベース監査証跡連携機能を有効にするかどうかを指定します。 true を指定した場合： データベース監査証跡連携機能を有効にします。 false を指定した場合： データベース監査証跡連携機能を無効にします。	false	－	
ejbserver.container.bmp.backcompatible	Entity Bean(BMP)のロック制御の動作を指定します。 true を指定した場合： EJB コンテナでロックを制御します。 false を指定した場合： EJB コンテナでロックを制御しません。 なお、このプロパティの変更を有効にするには、開始中の J2EE アプリケーションをいったん削除し、再作成または再インポートしてください。	false	－	
ejbserver.container.ejbhome.sessionbean.reconnect.enabled	EJB ホームオブジェクトへの再接続機能を有効にするかどうかを指定します。 true を指定した場合： EJB ホームオブジェクトへの再接続機能を有効にします。 true を指定した場合は、必ず <code>vbroker.se.iiop_tp.scm.iioptp.listener.port</code> プロパティにポート番号を指定してください。	false	－	

キー名称	内容	省略値	VR	関連情報
	false を指定した場合： EJB ホームオブジェクトへの再接続機能を無効にします。			
ejbserver.container.passivate.sc an.interval	Enterprise Bean のタイムアウトやプールの状態を監視するスレッドの起動間隔を、0～2147483 の整数（単位：秒）で指定します。0 を指定した場合は監視しません。 Enterprise Bean のタイムアウトやプールの状態を監視するスレッドはここに指定した間隔で、次の処理を実行します。 - タイムアウト時間を経過した Stateful Session Bean や Entity Bean のインスタンスを削除します。 - Stateless Session Bean や Entity Bean のプールを対象に最小値を超えてプーリングされているインスタンスを徐々に削減します。	0	—	「機能解説 基本・開発編(EJB コンテナ)」の「2.11 EJB コンテナでのタイムアウトの設定」
ejbserver.container.rebindpolicy	EJB メソッドの呼び出し時に通信障害が発生した場合の、EJB クライアント側での接続の再接続動作とリクエストの再送動作を指定します。 VB_TRANSPARENT： 接続を再接続してリクエストを再送します。 NO_RECONNECT： 接続を再接続しないで、リクエストを再送しません。 この場合、通信障害によって接続が切断されると、再接続されないため、そのオブジェクトリファレンスは再利用できなくなります。	VB_TRANSPARENT	—	「機能解説 基本・開発編(EJB コンテナ)」の「2.13 EJB のリモートインタフェースの呼び出し」
ejbserver.container.security.disabled	Enterprise Bean のアクセス制御機能を有効にするかどうかを指定します。Enterprise Bean のアクセス制御機能の詳細については、マニュアル「アプリケーションサーバ 機能解説 基本・開発編(EJB コンテナ)」の「2.10 Enterprise Bean へのアクセス制御」を参照してください。 このプロパティは server.policy (J2EE サーバ用セキュリティポリシーファイル) で指定するセキュリティポリシーとは関係ありません。 true を指定した場合： Enterprise Bean のアクセス制御機能が無効になります。この場合、EJB レベルおよびアプリケーションレベルでのセキュリティ	false	—	

キー名称	内容	省略値	VR	関連情報
	定義が無効となり、EJB 実行時のメソッド実行権限のチェックが一切行われません。 false を指定した場合： Enterprise Bean のアクセス制御機能が有効になります。			
ejbserver.container.remove.sca n.interval	活性化状態の Stateful Session Bean のタイムアウトを監視するスレッドの起動間隔を、0～153722867280912 の整数（単位：分）で指定します。 0 を指定した場合または 0 未満を指定した場合、監視しません。 153722867280912 より大きい値を指定した場合、153722867280912 が設定されます。 数値以外を指定した場合、デフォルト値が設定されます。	5	—	「機能解説 基本・開発編(EJB コンテナ)」の「2.11.2 Stateful Session Bean のタイムアウト」

(凡例)

—：08-00 より前のバージョンを示します。

空欄：関連情報ははありません。

(i) ejbserver.ctm から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.ctm.ActivateTimeOut	CTM 機能を使用する場合に必要なプロパティです。CTM 機能を使用する J2EE アプリケーションのデプロイ時、J2EE サーバはスケジュールキューを活性化します。このときの待ち時間（単位：秒）を 0～2147483647 で指定します。0 を指定した場合、無限に待ち続け、省略した場合は、180（秒）が設定されます。	180	—	
ejbserver.ctm.CTMDomain	CTM 機能を使用する場合に必要なプロパティです。J2EE サーバが属する CTM ドメイン名称を 1～31 文字の英数字またはアンダースコア () で指定します。"CTM"または"ctm"で始まる CTM ドメイン名称は指定できません。省略した場合、"CTMDOMAIN"が設定されます。	CTMDO MAIN	—	「機能解説 拡張編」の「3.3.5 CTM ドメインと CTM ドメインマネージャ」

キー名称	内容	省略値	VR	関連情報
ejbserver.ctm.CTMID	CTM 機能を使用する場合に必要なプロパティです。J2EE サーバを制御する CTM デーモンの CTM 識別子を 1～31 文字の英数字、アンダースコア (_) またはピリオド (.) で指定します。ただし、IP アドレスで識別子を指定する場合にだけピリオドが使用できます。 "CTM"または"ctm"で始まる CTM 識別子は指定できません。また CTM ドメイン名称と同じ名称を指定できません。省略した場合、IP アドレスが設定されます。	<IP アドレス>	—	
ejbserver.ctm.CTMMYHost	CTM 機能を使用する場合に必要なプロパティです。マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを 1～64 文字の文字列で指定します。 省略した場合は、hostname コマンドで取得できるホスト名称が設定されます。 ejbserver.ctm.CTMID プロパティを省略してこのプロパティを指定した場合、デフォルトの CTM 識別子は、このプロパティで指定した IP アドレスになります。	<hostname コマンドで取得されるホスト名称>	—	
ejbserver.ctm.DeactivateTimeOut	CTM 機能を使用する J2EE アプリケーションのアンデプロイ時、J2EE サーバはスケジュールキューを非活性化します。このときの待ち時間（実行中のリクエストの完了待ち）を 0～2147483647（単位：秒）で指定します。0 を指定した場合、無限に待ち続け、省略した場合は、180（秒）が設定されます。 CTM 経由で仕掛けり中のリクエストがある状態でアプリケーションを停止しようとした場合に、強制停止できなくなるため、0 は指定しないでください。	180	—	
ejbserver.ctm.enabled	CTM 機能の使用の有無を指定します。 true を指定した場合： CTM 機能を使用します。CTM は、構成ソフトウェアに Component Transaction Monitor を含む製品だけで利用できます。利用できる製品については、マニュアル「アプリケーションサーバ & BPM/ESB 基盤 概説」の「2.2.1 製品と構成ソフトウェアの対応」を参照してください。 J2EE サーバ起動時 CTM との接続の確立および初期化をします。これに成功すると CTM 機能が使用できる状態で J2EE サーバが起動します。	09-00 以降のバージョン： false 09-00 より前のバージョンの場合： true	—	

キー名称	内容	省略値	VR	関連情報
	CTM との接続および初期化に失敗した場合、J2EE サーバの起動に失敗します。 false を指定した場合： CTM 機能を使用しません。 J2EE サーバ起動時に、CTM との接続の確立および初期化をしないため、J2EE サーバは CTM を使用できる状態で起動しません。この状態で CTM を使用するアプリケーションのデプロイはできません。CTM 機能が使用できない状態でも J2EE サーバを起動したい場合は、false を指定します。			
ejbserver.ctm.QueueLength	CTM 機能を使用する J2EE アプリケーションのデプロイ時、J2EE サーバは CTM キューを生成します。このときの CTM キューの長さを、1～32767 で指定します。省略した場合、CTM デーモンで保持している CTM キューの生成時の長さ（-CTMMaxRequestCount オプションで指定した長さ）が有効となります。CTM キューを共有する場合、CTM キューはすでに作成されているため、指定された値ではなく作成済みの長さが有効となります。	なし	—	
ejbserver.ctm.useGlobalJNDI	CTM 機能を使用する構成で、Enterprise Bean に別名を指定しなかった場合に使用する、デフォルトのルックアップ名称を切り替えます。 true の場合： java:global/～から始まる Portable Global JNDI 名を使用します。 false の場合： HITACHI_EJB/～から始まる Global JNDI 名を使用します。	false	09-00	

(凡例)

—：08-00 より前のバージョンを示します。

(j) ejbserver.deploy から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.deploy.annotations.load_libjars.enabled	J2EE アプリケーションのデプロイで、ライブラリ JAR のアノテーションの情報を読み込むかどうかを指定します。 true を指定した場合： ライブラリ JAR のアノテーションの情報を読み込みます。 false を指定した場合： ライブラリ JAR のアノテーションの情報を読み込みません。 このキーは、ライブラリ JAR のアノテーションの情報が不要であり、クラス数が膨大なライブラリ JAR を含むアプリケーションに対してインポート時のメモリ使用量の制限に有効です。 注意事項： - ライブラリ JAR のアノテーションの情報を読み込まない場合、ライブラリ JAR に含まれるクラスのアノテーションの情報は使用できません。例えば、Enterprise-Bean アプリケーション内のクラスで、ライブラリ JAR に含まれるクラスをインターセプタに指定した場合、呼び出し先クラスのアノテーションの情報 (@PostConstruct など) が無効になります。そのため、呼び出し先クラスはインターセプタとして動作しません。 - このプロパティの設定前に読み込んだアノテーションの情報はインポート時に保存されるため、変更後はアプリケーションの停止、削除および再インポートが必要です。	true	09-50	「機能解説 基本・開発編(コンテナ共通機能)」の「17.3 ロード対象のクラスとロード時に必要なクラスパス」
ejbserver.deploy.app.stopforcibly.disabled	J2EE アプリケーションの強制停止を無効にするかどうかを指定します。 true を指定した場合： J2EE アプリケーションの強制停止を無効にします。 false を指定した場合： J2EE アプリケーションの強制停止を有効にします。	false	—	
ejbserver.deploy.annotations.load_check.enabled	アノテーションの情報取得のためのクラスロードで例外が発生した場合に無視するかどうかを指定します。 true を指定した場合： エラーとなり処理を中断します。	false	08-00	「機能解説 基本・開発編(コンテナ共通機能)」の「17.3 ロード対象のクラスとロード時に必要なクラスパス」

キー名称	内容	省略値	VR	関連情報
	false を指定した場合： 例外ログを取得し処理を続行します。			
ejbserver.deploy.context.check_interval	アプリケーション構成ファイルの更新を検知する間隔を、0～2147483647 の整数（単位：秒）で指定します。 0 や 1～2147483647 以外の数字を指定した場合は、自動リロードしません。コマンドリロードだけ受け付けます。	0	—	「機能解説 基本・開発編(コンテナ共通機能)」の「18.8.6 J2EE アプリケーションの更新検知インターバル」
ejbserver.deploy.context.reload_scope	app： EJB アプリケーション、サーブレット、JSP でリロード機能を使用します。 jsp： JSP でリロード機能を使用します。 none： リロード機能を使用しません。 web： サーブレット、JSP でリロード機能を使用します。 上記以外： デフォルト値が設定されます。	app	—	「機能解説 基本・開発編(コンテナ共通機能)」の「18.8.2 リロードの適用範囲」
ejbserver.deploy.context.update.interval	アプリケーション構成ファイルの更新を待つ時間を、0～2147483647 の整数（単位：秒）で指定します。 0 や 1～2147483647 以外の数字を指定した場合は、待ち時間なしとなります。	0	—	「機能解説 基本・開発編(コンテナ共通機能)」の「18.8.7 J2EE アプリケーションの構成ファイル更新用インターバル」
ejbserver.deploy.exclusive.lockAliveInterval	サーバ処理終了後、コマンドから送信された次のレスポンスを受け取るまでの最大待ち時間を、1～2147483647 の整数（単位：秒）で指定します。 このプロパティの影響を受けるコマンドは、J2EE サーバで使用するコマンドのうち、更新、参照、特権に分類されているコマンドです。更新、参照、特権に分類されるコマンドは、J2EE サーバで使用するコマンドの一覧で確認できます。J2EE サーバで使用するコマンドについては、マニュアル「アプリケーションサーバリファレンス コマンド編」の「2. J2EE サーバで使用するコマンド」を参照してください。 不正な値を指定した場合は、警告メッセージを出力して、デフォルト値が設定されます。	60	—	

キー名称	内容	省略値	VR	関連情報
	指定した時間以上コマンド側からの通信がない場合、サーバはコマンドが不正終了したと見なし、コマンド排他を強制解除します。 待ち時間とは、コマンドがサーバ間を往復する時間とコマンドの処理時間の合計時間です。			
ejbserver.deploy.resourcefile.scramble.enabled	作業ディレクトリ以下に展開される DataSource, JavaMail, ResourceAdapter に関する DD ファイルをスクランブルするかどうかを設定します。なお、このプロパティの値を切り替えて J2EE サーバを起動してもすでにインポート・デプロイされているリソースの DD ファイルは定義情報の更新を行わないかぎり状態は変化しません。 true を指定した場合： スクランブル化します。 false を指定した場合： スクランブル化しません。	false	—	
ejbserver.deploy.session.work.directory	Web アプリケーションのセッション情報ファイルを設定した <ejbserver.deploy.session.work.directory プロパティ値>¥web¥<コンテキストルート名>配下ディレクトリに出力します。	- Windows の場合 <Application Server のインストールディレクトリ>¥CC¥server¥repository¥<サーバ名称> - UNIX の場合 <Application Server のインストールディレクトリ>/CC/server	—	

キー名称	内容	省略値	VR	関連情報
		/ repository< サーバ 名称>		
ejbserver.deploy.stub.generation.scope	スタブなどの通信に必要なクラスを生成する対象範囲を指定します。新規インストール時は ejb, アップグレードインストール時は app を指定します。 ejb : EJB-JAR の DD に指定されているホームインタフェース, コンポーネントインタフェースからスタブなどの通信に必要なクラスを生成します。 app : EJB-JAR, WAR 内の java.rmi.Remote を継承したクラスからスタブなどの通信に必要なクラスを生成します。 上記以外 : サーバを起動時に KDJE42257-W のメッセージが出力され, デフォルト値が設定されます。	ejb	—	
ejbserver.deploy.LibraryConflictAvoid.<J2EE アプリケーション名>.enabled	ライブラリ競合回避機能を有効にするかどうかを指定します。 true : 有効にします。 false : 無効にします。 「<J2EE アプリケーション名>」に「*」を指定した場合は, 設定先 J2EE サーバ内のすべてのアプリケーションに適用されるデフォルト値となります。「<J2EE アプリケーション名>」に「*」を指定したサーバ単位でこのキーを設定したあと, アプリケーション単位に設定を変更できます。 対象パッケージが未設定の場合, この機能は無効となります。	false	11-10	
ejbserver.deploy.LibraryConflictAvoid.<J2EE アプリケーション名>.package	ライブラリ競合回避機能の対象パッケージを指定します。 ライブラリ競合回避機能が有効の場合, 次のパッケージをコンマ「,」で区切って指定します。 アプリケーションに含めたライブラリで, 優先して参照したいクラスのパッケージ	なし	11-10	

キー名称	内容	省略値	VR	関連情報
	この機能では、次のクラスは、アプリケーションに含めたライブラリを優先して参照します。 指定したパッケージの末尾に「.」を追加した値から始まるパッケージのクラス 対象パッケージのファイルを、リソースとして取得する場合も、この設定を適用します。 「java」および「java.」から始まるパッケージ、空白、および空文字だけの値は、未設定とします。また、値の前後の空白は無視します。 「<J2EE アプリケーション名>」に「*」を指定した場合は、設定先 J2EE サーバ内のすべてのアプリケーションに適用されるデフォルト値となります。「<J2EE アプリケーション名>」に「*」を指定したサーバ単位でこのキーを設定したあと、アプリケーション単位に設定を変更できます。			
ejbserver.deploy.LibraryConflictAvoid.<J2EE アプリケーション名>.resource	ライブラリ競合回避機能の対象リソースを指定します。 ライブラリ競合回避機能を有効にした場合、次のリソースを、コンマ「,」で区切って指定します。 - アプリケーションに含めたライブラリで、優先して参照したいリソース このプロパティは、J2EE 仕様のアプリケーションの構成で、クラスファイルを格納すると定められた場所以外にあるリソースを、この機能の対象としたい場合に設定します。 この機能では、次のリソースは、アプリケーションに含めたライブラリを優先して参照します。 - 指定したリソースに前方一致するリソース 空白、および空文字だけの値は、未設定とします。また、値の前後の空白は無視します。 「<J2EE アプリケーション名>」に「*」を指定した場合は、設定先 J2EE サーバ内のすべてのアプリケーションに適用されるデフォルト値となります。「<J2EE アプリケーション名>」に「*」を指定したサーバ単位でこのキーを設定したあと、アプリケーション単位に設定を変更できます。	なし	11-10	
ejbserver.deploy.multi_release_jar.enabled	アプリケーションの JAR をマルチリリース JAR として処理するかどうかを指定します。このプロパティは、Web アプリケーションクラスローダで読み込む JAR に対してだけ有効です。	true	11-50	

(凡例)

－：08-00 より前のバージョンを示します。

空欄：関連情報はありません。

(k) ejbserver.distributedtx から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.distributedtx.ots.recoverFailMessageCount	未決着トランザクションの走査をリトライする時に、何回リトライを実行したら KFCB40139-W メッセージを出力するかを指定します。0～2147483647 の整数で指定します。 J2EE サーバを開始する際のトランザクションリカバリ実行時、またはトランザクション決着処理中の通信障害発生時に、アプリケーションサーバはトランザクションに参加しているリソースマネージャに対して、未決着トランザクションの走査を実行します。走査が失敗した場合、アプリケーションサーバは成功するまで無限にリトライを続けますが、リソースアダプタ単位で実行される未決着トランザクションの走査が連続してこのプロパティで指定した回数分失敗したときには、KFCB40139-W メッセージを出力します。 なお、未決着トランザクションの走査は、5 秒ごとにリトライされます。 0 を指定した場合、未決着トランザクションの走査に何度失敗しても KFCB40139-W メッセージは出力されません。1 を指定した場合、走査が 1 回失敗したら KFCB40139-W メッセージが出力されます（2 回目以降の失敗時に KFCB40139-W メッセージは出力されません。ただし、一度走査に成功するか、または J2EE サーバが停止すると、回数はリセットされます。この場合は、次に 1 回失敗した時に KFCB40139-W メッセージが出力されます）。 このプロパティに 1 以上を指定すると、参加しているリソースマネージャに接続できない状態となったことを早期に知ることができます。しかし、アプリケーションサーバを起動したあとでリソースを起動する運用をしている場合、通常	0	08-70	

キー名称	内容	省略値	VR	関連情報
	の運用でも KFCB40139-W メッセージが出力されることがあります。			
ejbserver.distributedtx.ots.status.directory1※	インプロセストランザクションサービスのステータスファイルおよび保守資料としてのステータスファイルのバックアップを格納するディレクトリを指定します。パス区切り文字は” /” を使用してください。相対パスを指定した場合は、次のディレクトリからのパスとなります。 Windows の場合： <製品の作業ディレクトリ>¥ejb¥<サーバ名称> UNIX の場合： <製品の作業ディレクトリ>/ejb/<サーバ名称>	otsstatus	—	「機能解説 基本・開発編(コンテナ共通機能)」の「3.4 トランザクション管理」
ejbserver.distributedtx.ots.status.directory2※	インプロセストランザクションサービスのステータスファイルを二重化する場合、予備ステータスファイルおよび保守資料としての予備ステータスファイルのバックアップを格納するディレクトリを指定します。パス区切り文字は” /” を使用してください。 相対パスを指定した場合、J2EE アプリケーションの実行ディレクトリからのパスとなります。 同一マシン内で稼働するほかの J2EE サーバや J2EE アプリケーションのプロセスと重ならないように、別のディレクトリを指定する必要があります。 このプロパティを指定しない場合、予備ステータスファイルは作成されないため、二重化されません。	なし	—	「機能解説 基本・開発編(コンテナ共通機能)」の「3.4 トランザクション管理」
ejbserver.distributedtx.recovery.completionCheckOnStopping.timeout	グローバルトランザクションを使用する場合の J2EE サーバ停止時、および Transacted Delivery 機能を使用する際の J2EE アプリケーション停止時に行われるトランザクション仕掛り完了確認のタイムアウトを、-1～2147483647 の数値（単位：秒）で指定します。ただし、次の場合はタイムアウトしません。 • -1 を指定した場合 • 無効な文字列を指定した場合 • このキーを指定しない場合 この場合、J2EE サーバまたはリソースマネージャが認識する未決着トランザクションが存在しないことが確認できるまで、J2EE サーバまたは J2EE アプリケーションの停止は無限に保留されます。	-1（タイムアウトしません）	—	

キー名称	内容	省略値	VR	関連情報
	0 以上 2147483647 までの整数が指定された場合には、指定された時間（秒）が経過した時点でトランザクション仕掛かり完了確認をタイムアウトし、未決着トランザクションが存在しないことが確認できていなくても、J2EE サーバまたは J2EE アプリケーションを停止します。アプリケーションの運用時には、トランザクションの ACID を保証するために、タイムアウトしない設定にする必要があります。また、アプリケーションの開発時の、トランザクションの ACID よりも利便性を優先する場合には、タイムアウトする設定にできます。			
ejbserver.distributedtx.recovery.port	グローバルトランザクションを使用するとき、トランザクションリカバリで使用する固定ポート番号を 1～65535 の整数で指定します。無効な値を指定した場合や、指定したポートがすでに使用されていた場合には、起動が中断されます。なお、ライトトランザクションを有効として起動した場合は、このプロパティは無視されます。	20302	—	
ejbserver.distributedtx.rollbackClientTxOnSystemException	システム例外が発生した場合に、呼び出したクライアントトランザクションをロールバックにマークするかどうかを指定します。 true を指定した場合： クライアントトランザクションをロールバックにマークします。 false を指定した場合： クライアントトランザクションをロールバックにマークしません。 プロパティが指定されていない場合や、無効な値を指定した場合は、false が設定されます。クライアントのトランザクションを開始して J2EE サーバにあるリモートインタフェースを実装した EJB を、ローカル呼び出し最適化で呼び出し、呼ばれた EJB で新たなトランザクションを開始するかトランザクションを開始しない場合にだけ、このプロパティが有効になります。	false	—	
ejbserver.distributedtx.XATransaction.enabled	グローバルトランザクションを使用するかどうかを指定します。 true を指定した場合： ライトトランザクション機能が無効となり、グローバルトランザクションを利用できます。	false	—	「機能解説 基本・開発編(コンテナ共通機能)」の「3.4.2 ローカルトランザクションとグローバルトランザクション」

キー名称	内容	省略値	VR	関連情報
	false を指定した場合： ライトトランザクション機能が有効となり、ローカルトランザクションに最適化された環境となります。ただし、グローバルトランザクションは利用できません。			

(凡例)

－：08-00 より前のバージョンを示します。

空欄：関連情報ははありません。

注※

インプロセストランザクションサービスのステータスファイルには、J2EE サーバの識別情報としてホスト名または IP アドレスを取り込んでいます。このため、ホスト名または IP アドレスを変更する場合には、J2EE サーバの停止などの作業が必要です。ステータスファイルの格納ディレクトリに関するホスト名または IP アドレスの変更は、J2EE サーバの動作設定のカスタマイズ時に実行します。J2EE サーバの動作設定のカスタマイズについては、マニュアル「アプリケーションサーバ 機能解説 基本・開発編(コンテナ共通機能)」の「3.15.13 実行環境での設定」を参照してください。

(l) ejbserver.DynamicStubLoading から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.DynamicStubLoading.Enabled	ダイナミッククラスローディングを使用するかどうかを指定します。 false を指定した場合： ダイナミッククラスローディングは機能しません。EJB オブジェクトのスタブ、EJBHome オブジェクトのスタブ、またはビジネスインタフェースのリファレンスのスタブをダウンロードして、EJB クライアントのユーザクラスパスに指定してください。 true を指定した場合： ダイナミッククラスローディングが機能します。EJB オブジェクトのスタブ、EJBHome オブジェクトのスタブ、またはビジネスインタフェースのリファレンスのスタブを EJB クライアントのユーザクラスパスに指定する必要はありません。	false	－	「機能解説 基本・開発編(EJB コンテナ)」の「3.7.3 ダイナミッククラスローディング」

(凡例)

－：08-00 より前のバージョンを示します。

(m) ejbserver.ejb から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.ejb.cmp20.cmr.use.existing_table	デプロイ時に既存の CMR 表を使用するオプションです。通常はデプロイ時に CMR 表を生成するので既存の CMR 表を使用する必要はありませんが、障害から回復する場合に一時的に指定します。 false を指定した場合： デプロイ時にすでに CMR 表がある場合はエラーになります。 true を指定した場合： デプロイ時にすでにある CMR 表を使用します。	false	—
ejbserver.ejb.timerservice.maxCallbackThreads	J2EE サーバ全体で、TimerService でのタイムアウトメソッドをコールバックする最大スレッド数を、1～100 の整数で指定します。	1	—
ejbserver.ejb.timerservice.retryCount	TimerService での、タイムアウトメソッドのコールバックをリトライする最大回数を、0～2147483646 の整数で指定します。	1	—
ejbserver.ejb.timerservice.retryInterval	TimerService での、タイムアウトメソッドのコールバックをリトライする間隔を、1～604800 の整数（単位：秒）で指定します。	5	—

(凡例)

—：08-00 より前のバージョンを示します。

(n) ejbserver.ext から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.ext.method_observation.interval	このプロパティの設定には、次の二つの意味があります。 • J2EE アプリケーション実行時間監視機能を使用するかどうかを指定します。 • 動作中のリクエスト処理がタイムアウトしていないかを監視する時間間隔、タイムアウトしたリクエスト（メソッド）をキャンセルする時間間隔として、0～86400 の整数（単位：秒）を指定します。	0	—	「機能解説 運用／監視／連携編」の「5. J2EE アプリケーションの運用」

キー名称	内容	省略値	VR	関連情報
	0 を指定した場合： J2EE アプリケーション実行時間監視機能を使用しません。また、タイムアウトを監視しません 0 以外の有効値を指定した場合： J2EE アプリケーション実行時間監視機能を使用します。また、指定された時間間隔で、タイムアウトとメソッドキャンセルを異なるスレッドで実行します。 上記以外の無効な値を指定した場合： デフォルト値が設定されます。			

(凡例)

－：08-00 より前のバージョンを示します。

(o) ejbserver.http から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.http.port	管理用サーバのポート番号を 1～65535 の整数で指定します。 すでにほかのアプリケーションで使用または確保されているポート番号は指定できません。また、複数の J2EE サーバで Web サーバとの通信に使用するポートのポート番号に同じ値を設定しないでください。同一のポート番号を指定した J2EE サーバは cjstartsv コマンドで複数起動できません。	28008	11-00

(凡例)

－：08-00 より前のバージョンを示します。

(p) ejbserver.instrumentation から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.instrumentation.enabled	Management Server と連携するかどうかを指定します。 true を指定した場合： Management Server と連携できます。 false を指定した場合： Management Server と連携できません。	true	－

(凡例)

ー：08-00 より前のバージョンを示します。

(q) ejbserver.javaee から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.javaee.batch.<J2EE アプリケーション名>.appTag	Java Batch のアプリケーションタグを変更し ます。 クラスタ構成や HA 構成を組むケースなど、 複数の J2EE サーバでジョブリポジトリを共有 したい場合に、文字列で指定します。	<J2EE サーバ名 >:<J2EE アプリ ケーション名>	11-00
ejbserver.javaee.batch.<J2EE アプリケーション名 >.executorService.jndiName	Java Batch 実行用スレッドプールの JNDI 名 をアプリケーションごとに文字列で指定します。 <J2EE アプリケーション名>の部分に「*」を 指定した場合は、設定先 J2EE サーバ内のすべ てのアプリケーションに適用されるデフォルト 値となります。	concurrent/ __BatchExecutor Service	11-00
ejbserver.javaee.batch.executo rService.<JNDI 名 >.keepAliveTime	Java Batch 実行用スレッドプールで、最小ス レッド数を超えた使用後のスレッドを再利用す るまで待機する最大時間（単位：秒）を、0～ 2147483647 の整数で指定します。再利用さ れないまま指定時間を超えたスレッドは破棄さ れます。 <JNDI 名>には設定対象の Java Batch 実行 用スレッドプールの JNDI 名を指定します。	60	11-00
ejbserver.javaee.batch.executo rService.<JNDI 名 >.maxThreads	Java Batch 実行用スレッドプールの最大ス レッド数を 0～2147483647 の整数で指定し ます。 <JNDI 名>には設定対象の Java Batch 実行 用スレッドプールの JNDI 名を指定します。	2147483647	11-00
ejbserver.javaee.batch.executo rService.<JNDI 名 >.minThreads	Java Batch 実行用スレッドプールの最小ス レッド数を 0～2147483647 の整数で指定し ます。 <JNDI 名>には設定対象の Java Batch 実行 用スレッドプールの JNDI 名を指定します。	0	11-00
ejbserver.javaee.batch.executo rService.<JNDI 名>.queueSize	Java Batch 実行用スレッドプールのキューに 待機可能なタスクの最大数を 0～2147483647 の整数で指定します。 <JNDI 名>には設定対象の Java Batch 実行 用スレッドプールの JNDI 名を指定します。	2147483647	11-00
ejbserver.javaee.batch.jobRep ository.autoCreate.enabled	Java Batch のジョブリポジトリを自動生成に するかどうかを指定します。	false	11-00

キー名称	内容	省略値	VR
	true（大文字・小文字は区別しない）を指定した場合： Java Batch のジョブリポジトリを自動生成します。 上記以外の場合： Java Batch のジョブリポジトリを自動生成しません。		
ejbserver.javaee.batch.jobRepository.jndiName	Java Batch のジョブリポジトリの永続化先となるデータソースの JNDI 名を文字列で指定します。 DBConnector のユーザ指定名前空間機能を用いて JNDI 名前空間に別名を指定し、その別名をこのプロパティに指定します。	jdbc/____BatchPool	11-00
ejbserver.javaee.cdi.beanIdentifierIndexOptimization	効率的に CDI 管理 Bean の識別 ID にアクセスする機能を有効にするかどうかを指定します。 なお、CDI が有効なアプリケーションに対して、リロード機能を使用する場合、必ずこの機能を無効にしてリロード機能を使用してください。 true を指定した場合： 効率的に CDI 管理 Bean の識別 ID にアクセスする機能を有効にします。 上記以外の場合： 効率的に CDI 管理 Bean の識別 ID にアクセスする機能を無効にします。	false	11-30
ejbserver.javaee.cdi.beansXmlRequired	09-70 以前と同様に CDI 1.0 仕様に準じて「beans.xml」が存在するアーカイブだけを CDI の管理対象にするかどうかを指定します。 true（大文字・小文字は区別しない）を指定した場合： beans.xml が存在するアーカイブだけが CDI の管理対象になります。 上記以外の場合： beans.xml の有無に関わらず、すべてのアーカイブが CDI の管理対象になります。	false	11-00
ejbserver.javaee.cdi.webInfLibJarsEnabled	WAR ファイル内の WEB-INF/lib 以下の JAR ファイルに含まれるクラスを CDI の対象とするかどうかを指定します。 ただし CDI の対象となった JAR ファイルに含まれるクラスが CDI 管理クラスとなるかは、CDI 標準仕様に記載の条件に従います。また、展開ディレクトリ形式の場合、WAR ファイルは WAR ディレクトリを意味します。	true	11-20

キー名称	内容	省略値	VR
	true（大文字・小文字は区別しない）を指定した場合： WAR ファイル内の WEB-INF/lib 以下の JAR ファイルに含まれるクラスが CDI の管理対象になります。 上記以外の場合： WAR ファイル内の WEB-INF/lib 以下の JAR ファイルに含まれるクラスが CDI の管理対象になりません。		
ejbserver.javaee.concurrent.hungCheckIntervalSeconds	Concurrency Utilities で、無応答タスクの監視を行う間隔（単位：秒）を 1～2147483647 の整数で指定します。	60	11-00
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.awaitTerminationSeconds	Concurrency Utilities で、J2EE サーバ停止時に<JNDI 名>に指定した ManagedExecutorService で実行中のタスクが終了するまで待機する最大待機時間（単位：秒）を、0～2147483647 の整数で指定します。 0 を指定した場合はタスクの終了を待たずに強制終了します。	0	11-00
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.corePoolSize	Concurrency Utilities で、<JNDI 名>に指定した ManagedExecutorService の初期スレッド数を、0～2147483647 の整数で指定します。	16	11-00
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.hungAfterSeconds	Concurrency Utilities で、<JNDI 名>に指定した ManagedExecutorService の longRunningTasks が false の場合に、無応答と判断されるまでのスレッド実行時間（単位：秒）を、0～2147483647 の整数で指定します。0 以下の場合は無応答と判断されません。	0	11-00
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.keepAliveSeconds	Concurrency Utilities で、<JNDI 名>に指定した ManagedExecutorService で初期スレッド数を超えた使用後のスレッドを再利用するまで待機する最大時間（単位：秒）を、0～2147483647 の整数で指定します。指定時間内に実行要求がなかったスレッドは破棄されます。	60	11-00
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.longRunningTasks	Concurrency Utilities で、<JNDI 名>に指定した ManagedExecutorService が長時間実行のタスクかどうかを指定します。 false の場合： タスク実行開始から hungAfterSeconds 経過してもタスクが終了していない場合は無応答と判定され、メッセージログに KDJE60831-W が出力されます。	false	11-00

キー名称	内容	省略値	VR
	true の場合： 判定しません。		
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.maxPoolSize	Concurrency Utilities で、<JNDI 名>に指定した ManagedExecutorService の最大スレッド数を 0～2147483647 の整数で指定します。指定値を超える数のスレッドは生成されません。	16	11-00
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.taskQueueCapacity	Concurrency Utilities で、<JNDI 名>に指定した ManagedExecutorService のキューに待機可能なタスクの最大数を、0～2147483647 の整数で指定します。 同時実行要求タスク数が初期スレッド数を超えた場合、超えた分のタスクはキューで待機されます。キューが満杯になるとスレッド数を増やして処理します。キューが満杯で、かつスレッド数が最大スレッド数に達した場合、タスク実行要求時に java.util.concurrent.RejectedExecutionException がスローされ、実行要求が拒否されます。	2147483647	11-00
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.threadLifeTimeSeconds	Concurrency Utilities で、<JNDI 名>に指定した ManagedExecutorService の利用しなくなったスレッドを破棄するまでの時間（単位：秒）を、0～2147483647 の整数で指定します。0 の場合は破棄しないで、常に再利用します。	0	11-00
ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.threadPriority	Concurrency Utilities で、<JNDI 名>に指定した ManagedExecutorService が生成するスレッドの Priority を 1～10 の整数で指定します。	5	11-00
ejbserver.javaee.concurrent.managedExecutorService.jndiNames	Concurrency Utilities で、ユーザ定義の ManagedExecutorService の JNDI 名を、文字列で指定します。ルックアップに使用する JNDI 名を複数指定する場合は、文字列の間をコンマ (,) で区切ります。	なし	11-00
ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.awaitTerminationSeconds	Concurrency Utilities で、J2EE サーバ停止時に<JNDI 名>に指定した ManagedScheduledExecutorService で実行中のタスクが終了するまで待機する最大待機時間（単位：秒）を、0～2147483647 の整数で指定します。0 を指定した場合はタスクの終了を待たずに強制終了します。	0	11-00
ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.corePoolSize	Concurrency Utilities で、<JNDI 名>に指定した ManagedScheduledExecutorService の初期スレッド数を文字列で指定します。	なし	11-00

キー名称	内容	省略値	VR
ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.hungAfterSeconds	Concurrency Utilities で、<JNDI 名>に指定した ManagedScheduledExecutorService の longRunningTasks が false の場合に、無応答と判断されるまでのスレッド実行時間（単位：秒）を、0～2147483647 の整数で指定します。0 以下の場合は無応答と判断されません。	0	11-00
ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.keepAliveSeconds	Concurrency Utilities で、<JNDI 名>に指定した ManagedScheduledExecutorService で初期スレッド数を超えた使用後のスレッドを再利用するまで待機する最大時間（単位：秒）を、0～2147483647 の整数で指定します。指定時間内に実行要求がなかったスレッドは破棄されます。	60	11-00
ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.longRunningTasks	Concurrency Utilities で、<JNDI 名>に指定した ManagedScheduledExecutorService が長時間実行用のタスクかどうかを指定します。 false の場合： タスク実行開始から hungAfterSeconds 経過してもタスクが終了していない場合は無応答と判定され、メッセージログに KDJE60818-W が出力されます。 true： 判定しません。	false	11-00
ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.threadLifeTimeSeconds	Concurrency Utilities で、<JNDI 名>に指定した ManagedScheduledExecutorService の利用しなくなったスレッドを破棄するまでの時間（単位：秒）を、0～2147483647 の整数で指定します。0 の場合は破棄しないで、常に再利用します。	0	11-00
ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.threadPriority	Concurrency Utilities で、<JNDI 名>に指定した ManagedScheduledExecutorService が生成するスレッドの Priority を 1～10 の整数で指定します。	5	11-00
ejbserver.javaee.concurrent.managedScheduledExecutorService.jndiNames	Concurrency Utilities で、ユーザ定義の ManagedScheduledExecutorService の JNDI 名を、文字列で指定します。ルックアップに使用する JNDI 名を複数指定する場合は、文字列の間をコンマ (,) で区切ります。	なし	11-00
ejbserver.javaee.concurrent.managedThreadFactory.<JNDI 名>.threadPriority	Concurrency Utilities で、<JNDI 名>に指定した ManagedThreadFactory が生成するスレッドの Priority を 1～10 の整数で指定します。	5	11-00

キー名称	内容	省略値	VR
ejbserver.javaee.concurrent.managedThreadFactory.jndiNames	Concurrency Utilities で、ユーザ定義の ManagedThreadFactory の JNDI 名を、文字列で指定します。ルックアップに使用する JNDI 名を複数指定する場合は、文字列の間をコンマ (,) で区切ります。	なし	11-00
ejbserver.javaee.jaxrs.config.client.connectTimeout	JAX-RS のクライアント API での Client configuration property 「jersey.config.client.connectTimeout」のデフォルト値（単位：ミリ秒）を 0～2147483647 の整数で指定します。 0 を指定した場合、タイムアウトしません。	0	11-00
ejbserver.javaee.jaxrs.config.client.readTimeout	JAX-RS のクライアント API での Client configuration property 「jersey.config.client.readTimeout」のデフォルト値（単位：ミリ秒）を 0～2147483647 の整数で指定します。 0 を指定した場合、タイムアウトしません。	0	11-00
ejbserver.javaeeLogfilter.msgids	導入母体ログメッセージのメッセージ ID 単位のメッセージログおよび例外ログへの出力抑止設定で、抑止対象のメッセージ ID のリストを文字列で指定します。複数指定する場合は、文字列の間をコンマ (,) で区切ります。コンマとメッセージ ID の間には、スペース、タブ、または改行など、コンマとメッセージ ID 以外の文字を含むことはできません。 指定されたメッセージ ID と合致するログメッセージは、出力先がメッセージログ、例外ログから保守ログに変更されます。 なお、同一メッセージ ID を複数指定した場合、2 個目以降のメッセージ ID は無視されます。	なし	11-00

(r) ejbserver.jca から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.jca.adapter.tp1.bind_host	TP1 インバウンド連携機能の通信（受信および送信）で使用する IP アドレスまたはホスト名を指定します。ただし、ホスト名を指定した場合で、アドレスを解決できないときは、省略値を使用します。	システムによって自動選択された有効なローカルアドレス	08-50

(s) ejbserver.jndi から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.jndi.cache	ネーミングでのキャッシングをするかどうかを指定します。 キャッシングをする場合は on を、キャッシングをしない場合は off を指定します。	on	—	
ejbserver.jndi.cache.interval	ネーミングでのキャッシングをする場合、キャッシュをクリアする間隔（単位：秒）を指定します。 ただし、1～2147483647 以外の値を指定した場合、クリアしません。	0	—	
ejbserver.jndi.cache.interval.clear.option	インターバル経過後のネーミングでのキャッシュ領域に対する動作を決定します。 refresh を指定した場合： キャッシュの全領域をクリアします。 check を指定した場合： 無効なキャッシュをクリアします。CTM を使用している場合、キャッシュされたグローバル CORBA ネーミングサービスの EJB ホームオブジェクトのオブジェクトリファレンスは、アプリケーションが停止していても無効なキャッシュとしてクリアされません。詳細については、マニュアル「アプリケーションサーバ 機能解説 基本・開発編(コンテナ共通機能)」の「2.8.4 ネーミングでのキャッシングの注意事項」を参照してください。	refresh	—	
ejbserver.jndi.cache.reference	DataSource オブジェクトのキャッシング機能を使用するかどうかを指定します。 on を指定した場合： DataSource オブジェクトのキャッシング機能が有効になり、検索の要求に対して同じインスタンスが返ります。 off または無効な文字列を指定した場合： 通常どおりに動作し、検索ごとに異なるインスタンスが返ります。	off	—	「機能解説 基本・開発編(コンテナ共通機能)」の「3.14.7 DataSource オブジェクトのキャッシング」

キー名称	内容	省略値	VR	関連情報
ejbserver.jndi.global.enabled	アプリケーション開始時に、ネーミングサービスに対して Portable Global JNDI 名でオブジェクトを登録するかどうかを指定します。 true を指定した場合： Portable Global JNDI 名でオブジェクトを登録します。 false を指定した場合： Portable Global JNDI 名でオブジェクトを登録しません。	true	09-00	
ejbserver.jndi.namingservice.group.<Specify group name>.providerurls	各グループに属するネーミングサービスのルート位置をプロバイダ URL で指定します。 <Specify group name>には、ejbserver.jndi.namingservice.group.list 内に指定したグループ名を指定します。 プロバイダ URL は、URL スキーマ"corbaname"を使用して記述します。 指定形式： <プロバイダURL>;<プロバイダURL>*	なし	—	
ejbserver.jndi.namingservice.group.list	ラウンドロビン検索をする場合に、検索対象になる論理ネーミングサービスのグループを定義します。指定するグループ名は、英数字 (A～Z, a～z, 0～9) またはアンダースコア (_) で構成される、ejbserver.jndi.namingservice.group.list 内で一意に識別できる名前にします。 指定形式： <Specify group name>;<Specify group name>]* なお、*は繰り返しを示します。	なし	—	
ejbserver.jndi.request.timeout	ネーミングサービスとの通信タイムアウト時間 (単位：秒) を 0～86400 の整数で指定します。0 を指定した場合、またはこのプロパティを指定しない場合、タイムアウトしません。また、86400 を超える値を設定した場合、警告メッセージを出力して、タイムアウトしません。	0	—	「機能解説 基本・開発編(EJB コンテナ)」の「2.11.5 RMI-IIOP 通信のタイムアウト」

(凡例)

*：[]部分の繰り返しを示します。

—：08-00 より前のバージョンを示します。

(t) ejbserver.jpа から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.jpa.defaultJtaDsName	デフォルトの JTA データソースの参照を指定します。このプロパティは persistence.xml の jta-data-source を指定しなかった場合、または空白文字を指定した場合に使用されます。	なし	08-00
ejbserver.jpa.defaultNonJtaDsName	デフォルトの非 JTA データソースの参照を指定します。このプロパティは persistence.xml の non-jta-data-source を指定しなかった場合または空白文字を指定した場合に使用されます。	なし	08-00
ejbserver.jpa.defaultProviderClassName	デフォルトの JPA プロバイダクラス名を指定するプロパティです。このプロパティは、persistence.xml の provider を指定しなかった場合または空白文字を指定した場合に使用されます。	org.eclipse.persistence.jpa.PersistenceProvider	08-00
ejbserver.jpa.disable※	アプリケーションサーバの JPA 機能を使用する場合に指定します。 true を指定した場合： アプリケーションサーバの JPA 機能は無効になります。 false を指定した場合： アプリケーションサーバの JPA 機能は有効になります。	false	08-20
ejbserver.jpa.overrideJtaDsName	persistence.xml の jta-data-source に指定した値、および ejbserver.jpa.defaultJtaDsName に指定した値より優先して使用する JTA データソースの参照を指定します。	なし	08-00
ejbserver.jpa.overrideNonJtaDsName	persistence.xml の non-jta-data-source に指定した値、および ejbserver.jpa.defaultNonJtaDsName に指定した値より優先して使用する非 JTA データソースの参照を指定します。	なし	08-00
ejbserver.jpa.overrideProvider	persistence.xml の provider に指定した値、および ejbserver.jpa.defaultProviderClassName に指定した値より優先して使用する JPA プロバイダクラス名を指定します。	なし	08-00
ejbserver.jpa.emfprop.<property key>	JPA プロバイダ独自のプロパティのキーを指定します。すべての永続化ユニットのデプロイ時に、「ejbserver.jpa.emfprop.」プリフィックスを除去したプロパティが JPA プロバイダに渡されます。	なし	08-00
ejbserver.jpa.cdiEnabled	CDI Managed Bean（それに関連づくインターセプタ含む）からアプリケーションサーバの JPA 機能の利用可否を指定します。 true を指定した場合： CDI Managed Bean からアプリケーションサーバの JPA 機能を利用できます。なお、ejbserver.jpa.disable でアプリケーションサーバの JPA 機能を有効にしている必要があります。 false を指定した場合： CDI Managed Bean からアプリケーションサーバの JPA 機能を利用できません。	true	11-20

注※ ejbserver.jpa.disable=true を指定した場合の注意事項

アプリケーションに persistence.xml が含まれる場合、アプリケーションサーバはアプリケーション開始時に persistence.xml を読み込まなくなります。また、アプリケーションのリロード機能を利用している場合、persistence.xml が更新検知の対象ではなくなります。

アプリケーションがアプリケーションサーバの管理する永続化コンテキストまたは永続化ユニットを利用している場合、アプリケーションの開始ができなくなります。

2. J2EE サーバで使用するファイル

(u) ejbserver.jta から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.jta.TransactionManager.defaultTimeout	J2EE サーバ上で開始されるトランザクションのトランザクションタイムアウトのデフォルト値（単位：秒）を 1～2147483647 の整数で指定します。	180	—	「機能解説 基本・開発編(コンテナ共通機能)」の「3.15.8 トランザクションタイムアウトとステートメントキャンセル」

(凡例)

—：08-00 より前のバージョンを示します。

(v) ejbserver.logger から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.logger.access_log.nio_http.enabled	NIO HTTP サーバで扱う HTTP のアクセスログ機能の有効・無効を設定します。 true を指定した場合： HTTP のアクセスログ機能を有効にします。 false を指定した場合： HTTP のアクセスログ機能を無効にします。	true	11-00	
ejbserver.logger.access_log.nio_http.format	NIO HTTP サーバで扱う HTTP のアクセスログのフォーマットの形式を指定します。	%h %{X-Forwarded-For}i %l %u %d %rootap "%r" %s %b %D %S	11-00	

キー名称	内容	省略値	VR	関連情報
ejbserver.logger.access_log.websocket.enabled	NIO HTTP サーバで扱う WebSocket のアクセスログ機能の有効・無効を設定します。 true を指定した場合： WebSocket のアクセスログ機能を有効にします。 false を指定した場合： WebSocket のアクセスログ機能を無効にします。	false	11-00	
ejbserver.logger.access_log.websocket.format	NIO HTTP サーバで扱う WebSocket のアクセスログのフォーマットの形式を指定します。 フォーマット引数の%PAYLOADDATA(n)の n には 1～32768 の整数を指定します。 フォーマット引数の%PAYLOADDATA と%PAYLOADDATA(n)は同時には使用できません。	%TS %IO %OPCODE %ROOTAPP %URI %FIN %PAYLOADDATA LEN %CLIENTAP %CLOSE REASON	11-00	
ejbserver.logger.DevelopmentLogFile.level	J2EE サーバの開発調査ログ機能のログの出力レベルを指定します。 開発時には INFO レベルを設定することを推奨します。開発しているアプリケーションの障害調査のために INFO レベルよりさらに詳細な情報が必要な場合は、より詳細なログを出力するレベルを指定してください。 指定できる値は次のとおりです。出力情報はこの順番で多くなります。 • OFF (出力しない) • SEVERE • WARNING • INFO • CONFIG • FINE • FINER • FINEST • ALL (すべて出力)	OFF	09-00	
ejbserver.logger.channels.define.<チャンネル名>*.filenum	J2EE サーバのログファイルの面数を、1～16 の整数で指定します。	• チャンネル名が MaintenanceLogFile,	—	

キー名称	内容	省略値	VR	関連情報
		WebS ervletL ogFile または Devel opmen tLogFil e の 場合 4 • チャネ ル名が NIOH TTPA ccessL ogFile または WebS ocketA ccessL ogFile の場合 16 • チャネ ル名が 上記以 外の 場合 2		
ejbserver.logger.channels.define.<チャネル名>*.filesize	J2EE サーバのログファイルのサイズ（単位：バイト）を、4096～2147483647 の整数で指定します。	• チャネ ル名が Mainte nance LogFil e の 場合 16777 216 • チャネ ル名が WebS ervletL ogFile , NIOH TTPA ccessL ogFile	—	

キー名称	内容	省略値	VR	関連情報
		または WebS ocketA ccessL ogFile の場合 41943 04 • チャネ ル名が 上記以 外の 場合 10485 76		
ejbserver.logger.enabled.*	J2EE サーバのログレベルを指定します。 Error, Warning, Information, Debug の うち、一つまたは複数を指定します。一つだけ 指定した場合は、該当するログレベルのログだ けが出力されます。複数指定する場合は、レベ ル名の文字列の間をコンマ (,) で区切ります。 通常はデフォルトのまま利用してください。	Error	—	
ejbserver.logger.systemlog.enabled	J2EE サーバの起動、停止および異常終了に関 するメッセージを、イベントログ (UNIX の場 合は、syslog) へ出力するかどうかを指定しま す。 true を指定した場合： イベントログ (UNIX の場合は、syslog) へログを出力します。 false を指定した場合： イベントログ (UNIX の場合は、syslog) へログを出力しません。	true	—	
ejbserver.logger.rotationTime	J2EE サーバのログの出力先ファイルを切り替 える時刻を HHMMSS の形式で指定します。 指定がない場合は、ログのファイルサイズで出 力先を切り替えます。 なお、このキーを指定した場合も、ファイルサ イズの指定は有効になります。このキーに指定 した時刻になったとき、または出力先ファイル が指定したファイルサイズになったときに、出 力先ファイルが切り替わります。	なし	09-0 0	「機能解説 保守 ／移行編」の 「3. トラブル シューティング のための準備」
ejbserver.logger.rotationStyle	J2EE サーバのログの出力先ファイル切り替え 時のファイル名付与規則を指定します。 SHIFT を指定した場合： シフトモードでファイル名を付与します。	WRAP	09-0 0	「機能解説 保守 ／移行編」の 「3. トラブル

キー名称	内容	省略値	VR	関連情報
	ファイルの合計数は、指定した出力先ファイル面数+1（現在の出力先ファイル分）となります。 WRAP を指定した場合： ラップアラウンドモードでファイル名を付与します。			シューティングのための準備

(凡例)

－：08-00 より前のバージョンを示します。

空欄：関連情報はあります。

注※

チャンネル名称として次に示す名称を設定できます。

- アプリケーションサーバシステムのログ出力用のチャンネル

MessageLogFile, MaintenanceLogFile, ExceptionLogFile, ConsoleLogFile, EJBContainerLogFile, WebContainerLogFile, ServletLogFile, UserOutLogFile, UserErrLogFile, JPAOperationLogFile, JPAMaintenanceLogFile, DevelopmentLogFile, NIOHTTPAccessLogFile, WebSocketAccessLogFile

- リソース枯渇監視ログ出力用のチャンネル

MemoryWatchLogFile, FileDescriptorWatchLogFile, ThreadWatchLogFile, ThreaddumpWatchLogFile, RequestQueueWatchLogFile, HttpSessionWatchLogFile, ConnectionPoolWatchLogFile

資料の取得については、マニュアル「アプリケーションサーバ 機能解説 保守／移行編」の「2.3 資料の取得」を参照してください。

(w) ejbserver.management から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.management.JVM.stats_monitor.FullGCCCount.enabled	FullGC 回数の監視を有効にするかどうかを指定します。 true : 有効にします。 false : 無効にします。	true	－	「機能解説 運用／監視／連携編」の「3.4 イベントの発行機能」
ejbserver.management.JVM.stats_monitor.FullGCCCount.threshold	FullGC 回数の監視でのしきい値となる回数を、1～2147483647 の整数で指定します。	10	－	「機能解説 運用／監視／連携編」の「3.4 イベントの発行機能」

キー名称	内容	省略値	VR	関連情報
ejbserver.management.JVM.stats_monitor.FullGCCCount.interval	FullGC 回数の監視でのしきい値監視間隔を、1～2147483647 の整数（単位：秒）で指定します。	600	—	「機能解説 運用／監視／連携編」の「3.4 イベントの発行機能」
ejbserver.management.statistics.interval	稼働情報収集間隔を、1～86400 の整数（単位：秒）で指定します。	60	—	「機能解説 運用／監視／連携編」の「3.3 稼働情報ファイルの出力機能」
ejbserver.management.stats_file.base_time	稼働情報ファイルの時間切り替え運用での基点時刻を、ローカルタイムの 1970 年 1 月 1 日 0 時 0 分 0 秒からの増分として 0～1439 の整数（単位：分）で指定します。 ローカルタイムの 1970 年 1 月 1 日 h 時 m 分を基点にする場合には、h×60 + m を指定します。	0	—	
ejbserver.management.stats_file.dir	稼働情報ファイルの出力先ディレクトリを指定します。出力先ディレクトリを変更する場合は、J2EE サーバごとに別々のディレクトリを指定してください。	- Windows の場合 <製品の作業ディレクトリ>*ejb* <サーバ名称>*stats - UNIX の場合 <製品の作業ディレクトリ>/ejb/ <サーバ名称>/stats	—	
ejbserver.management.stats_file.enabled	稼働情報ファイル出力機能を有効にするかどうかを指定します。 true : 有効にします。 false : 無効にします。	true	—	「機能解説 運用／監視／連携編」の「3.3 稼働情報ファイルの出力機能」
ejbserver.management.stats_file.num	稼働情報ファイルの面数を、2～168 の整数で指定します。	7	—	「機能解説 運用／監視／連携編」の「3.3 稼働情報ファイルの出力機能」

キー名称	内容	省略値	VR	関連情報
				働情報ファイル の出力機能」
ejbserver.management.stats_file.period	稼働情報ファイルの時間切り替え運用での切り替え時間を、1～744 の整数（単位：時）で指定します。	24	—	
ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.enabled	Web コンテナ単位の全体実行待ちリクエスト数の監視を有効にするかどうかを指定します。 true : 有効にします。 false : 無効にします。	true	09-00	
ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.high_threshold	Web コンテナ単位の全体実行待ちリクエスト数の割合が警告域に達したことを知らせるメッセージを出力する際の割合を 1～100 の整数（単位：%）で指定します。 ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.low_threshold 以上の値を指定してください。 Web コンテナ単位の全体実行待ちリクエスト数の割合が指定した値以上になると、メッセージが出力されます。 100 を指定した場合、Web コンテナ単位の全体実行待ちリクエスト数の格納率が 100% になると、アラートメッセージが出力されます。	80	09-00	
ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.low_threshold	Web コンテナ単位の全体実行待ちリクエスト数の割合が正常域に達したことを知らせるメッセージを出力する際の割合を 0～99 の整数（単位：%）で指定します。 ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.high_threshold 以下の値を指定してください。 Web コンテナ単位の全体実行待ちリクエスト数の割合が指定した値以下になると、メッセージが出力されます。 0 を指定した場合、Web コンテナ単位の全体実行待ちリクエスト数の格納率が 0% になると、アラートメッセージが出力されます。	0	09-00	
ejbserver.management.webcontainer.stats_monitor.waiting_request_count.enabled	Web コンテナ単位の実行待ちリクエスト数の監視を有効にするかどうかを指定します。 true : 有効にします。 false : 無効にします。	true	09-00	

キー名称	内容	省略値	VR	関連情報
ejbserver.management.webcontainer.stats_monitor.waiting_request_count.high_threshold	Web コンテナ単位の実行待ちリクエスト数の割合が警告域に達したことを知らせるメッセージを出力する際の割合を 1～100 の整数（単位：%）で指定します。 ejbserver.management.webcontainer.stats_monitor.waiting_request_count.low_threshold 以上の値を指定してください。 Web コンテナ単位の実行待ちリクエスト数の割合が指定した値以上になると、メッセージが出力されます。 100 を指定した場合、Web コンテナ単位の実行待ちリクエスト数の格納率が 100%になると、アラートメッセージが出力されます。	80	09-00	
ejbserver.management.webcontainer.stats_monitor.waiting_request_count.low_threshold	Web コンテナ単位の実行待ちリクエスト数の割合が正常域に達したことを知らせるメッセージを出力する際の割合を 0～99 の整数（単位：%）で指定します。 ejbserver.management.webcontainer.stats_monitor.waiting_request_count.high_threshold 以下の値を指定してください。 Web コンテナ単位の実行待ちリクエスト数の割合が指定した値以下になると、メッセージが出力されます。 0 を指定した場合、Web コンテナ単位の実行待ちリクエスト数の格納率が 0%になると、アラートメッセージが出力されます。	0	09-00	

（凡例）

－：08-00 より前のバージョンを示します。

(x) ejbserver.manager から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.manager.agent.Agent.conf	Management Server との連携に必要な設定をします。なお、この内容は Management Server が J2EE サーバに自動で設定します。 Management Server が J2EE サーバに自動で設定する内容については、マニュアル「アプリケーションサーバ システム構築・運用ガイド」の「付録 E.4 Management Server が J2EE サーバに自動で設定する内容」を参照してください。	なし	－
ejbserver.manager.agent.Agent.enabled	Management Server との連携に必要な設定をします。なお、この内容は Management Server が J2EE サーバに自動で設定します。 Management Server が J2EE サーバに自動で設定する内容については、マニュアル「アプリケーションサーバ システム構築・運用ガイド」	false	－

キー名称	内容	省略値	VR
	ド」の「付録 E.4 Management Server が J2EE サーバに自動で設定する内容」を参照してください。		
ejbserver.manager.agent.JP1EventAgent.conf	JP1 連携に関する設定ファイルのパスを指定します。	なし	—
ejbserver.manager.agent.JP1EventAgent.enabled	JP1 と連携するかどうかを指定します。	false	—
ejbserver.manager.jp1event.event_server_name	使用する JP1/Base のイベントサービスのイベントサーバ設定ファイル (conf) に指定した、ports パラメタのアドレスと同じ値を指定します。ports パラメタに複数のアドレスを指定している場合は、指定したアドレスのうちのどれか一つを指定します。なお、ports パラメタのアドレスに「0.0.0.0」(デフォルト値)を指定したイベントサービスを使用する場合は、このキーを省略するか、自マシンのホスト名または localhost を指定します。	localhost	—
ejbserver.manager.agent.MEventAgent.conf	Management イベント発行用プロパティファイルを絶対パスで指定します。指定がない場合、または指定したファイルが存在しない場合は Management イベントを発行しません。 Management イベント発行用プロパティファイルの詳細は、 「 8.2.11 Management イベント発行用プロパティファイル 」を参照してください。	なし	—
ejbserver.manager.agent.MEventAgent.enabled	Management イベント発行機能を有効にするかどうかを指定します。 true を指定した場合： Management イベント発行機能を有効にします。 false を指定した場合： Management イベント発行機能を無効にします。 不正な値が指定された場合は、デフォルト値が設定されます。	false	—

(凡例)

— : 08-00 より前のバージョンを示します。

(y) ejbserver.naming から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.naming.exec.args	CORBA ネーミングサービスを自動起動モード (ejbserver.naming.startupMode=automatic または inprocess) で使用する場合に、起動させる CORBA ネーミングサービスに付加する実行時オプションを指定します。通常はこのプロパティに値を設定する必要はありません。CORBA ネーミングサービス自動起動モード使用時に CORBA ネーミングサービスで障害などが発生した場合に、その調査、対策および回避をするために使用します。	なし	—

キー名称	内容	省略値	VR
	実行時オプションは、nameserv のコマンドライン引数に指定する形式と同じ文字列を指定します（各プロパティの先頭に"-J"を付加して、スペース区切りで指定します。なお、実際は一行で指定します）。 (指定例) <pre>ejbserver.naming.exec.args=-J- Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=900 -J-Dvbroker.agent.enableLocator=false</pre> ただし、inprocess モードで CORBA ネーミングサービスを使用する場合は、次のプロパティを指定できません。 • javax.rmi.CORBA.StubClass • javax.rmi.CORBA.UtilClass • javax.rmi.CORBA.PortableRemoteObjectClass • vbroker.serverManager.name • vbroker.orb.enableServerManager • vbroker.agent.enableLocator • vbroker.se.iiop_tp.host このプロパティ値は、ejbserver.naming.port や vbroker.agent.enableLocator プロパティで指定された値よりも優先されます。ただし、inprocess モードで CORBA ネーミングサービスを起動する場合、ejbserver.naming.exec.args で指定されたプロパティと同一のプロパティがシステムプロパティで設定されているときは、システムプロパティに設定されている情報が優先されます。CORBA ネーミングサービスの実行時に指定できるプロパティについては、マニュアル「Borland(R) Enterprise Server VisiBroker(R) デベロッパーズガイド」、およびマニュアル「Borland(R) Enterprise Server VisiBroker(R) プログラマーズリファレンス」を参照してください。		
ejbserver.naming.host	J2EE サーバがネーミングサービスとして利用する CORBA ネーミングサービスが起動しているホスト名称または IP アドレスを指定します。ネーミングの切り替え機能を使用する場合、ホスト名称として"localhost"は使用しないでください。CORBA ネーミングサービスが起動しているホスト名称または IP アドレスを指定してください。CORBA ネーミングサービスを自動起動モード (ejbserver.naming.startupMode=automatic または inprocess) で使用する場合、デフォルト値 ("localhost"), または J2EE サーバを起動させるホスト名称もしくは IP アドレスを指定してください。	localhost	—
ejbserver.naming.name eroot	CORBA ネーミングサービスを自動起動モード (ejbserver.naming.startupMode=automatic または inprocess) で使用する場合に、起動させる CORBA ネーミングサービスに名称を付加するときに指定します。英数字 (A~Z, a~z, 0~9) またはアンダースコア (_) で構成される文字列で指定します。この名称は、VisiBroker の nsutil コマンドのパラメタなどに使用されます。 省略した場合、VisiBroker で自動的に"NameService"という名称が設定されます。	なし	—

キー名称	内容	省略値	VR
	なお、nsutil コマンドの使用方法、使用条件などについては、マニュアル「Borland(R) Enterprise Server VisiBroker(R) デベロッパーズガイド」を参照してください。		
ejbserver.naming.port	J2EE サーバがネーミングサービスとして利用する CORBA ネーミングサービスのポート番号を 1～65535 の整数で指定します。ポート番号には、すでにほかのアプリケーションで使用されているポート番号は指定できません。ほかのアプリケーションで使用されているポート番号を指定して J2EE サーバを起動した場合、起動処理が完了しないことがあります。	900	—
ejbserver.naming.protocol	J2EE サーバがネーミングサービスとして利用する CORBA ネーミングサービスへのアクセスプロトコルを指定します。現状、corbaname だけサポートしています。ただし、旧バージョンで使用されていたプロトコル (iioploc または iiopname) でも動作できます。	corbaname	—
ejbserver.naming.startupMode [※]	CORBA ネーミングサービスの起動モードを指定します。 manual を指定した場合： CORBA ネーミングサービスを手動起動モードで使用する場合に指定します。 J2EE サーバを開始する前に CORBA ネーミングサービスを手動で開始する必要があります。 J2EE サーバ開始時に ejbserver.naming.host, ejbserver.naming.port で指定した CORBA ネーミングサービスを手動で指定します。このプロパティ値が指定された場合には、J2EE サーバ起動処理からは CORBA ネーミングサービスの自動起動をしません。 automatic を指定した場合： CORBA ネーミングサービスを自動起動モードで使用する場合に指定します。 J2EE サーバ開始時に自動で、CORBA ネーミングサービスをアウトプロセスで開始します。 このモードでは、J2EE サーバ終了時に自動で CORBA ネーミングサービスも停止します。ただし、J2EE サーバ開始時に ejbserver.naming.port で指定した CORBA ネーミングサービスが自ホスト上ですでに開始されている場合には自動開始処理をしないで、その CORBA ネーミングサービスを利用します。また、この場合、J2EE サーバ終了時の CORBA ネーミングサービスの自動停止はしません。 inprocess を指定した場合： CORBA ネーミングサービスを自動起動モードで使用する場合に指定します。 J2EE サーバ開始時に自動で、CORBA ネーミングサービスをインプロセスで開始するモードです。J2EE サーバ開始時に ejbserver.naming.port で指定した CORBA ネーミングサービスが自ホスト上ですでに開始されている場合には、エラーメッセージを出力して J2EE サーバの起動処理が失敗します。	inprocess	—

2. J2EE サーバで使用するファイル

キー名称	内容	省略値	VR
ejbserver.naming.startupRetryCount	ejbserver.naming.startupWaitTime で指定した CORBA ネーミングサービスの起動待ち時間経過後、CORBA ネーミングサービスがまだ起動していない場合に、ejbserver.naming.startupWaitTime で指定された秒数だけ起動待ちを繰り返す回数を 0～2147483647 の整数で指定します。 0 が指定された場合は CORBA ネーミングサービス起動状態チェック処理のリトライをしません。 なお、ejbserver.naming.startupWaitTime に 0 が指定されている場合は、このプロパティ値は無効です。	9	—
ejbserver.naming.startupWaitTime	J2EE サーバ起動時に CORBA ネーミングサービスが利用可能になるまで待つ時間（単位：秒）を 0～2147483647 の整数で指定します。0 を指定した場合は、CORBA ネーミングサービスの起動待ちをしません。 ejbserver.naming.startupMode に inprocess を設定している場合には、0 を指定しないでください。	1	—

(凡例)

—：08-00 より前のバージョンを示します。

注※

Management Server を利用して J2EE サーバを起動する場合、automatic 以外を指定してください。automatic を指定した場合、CORBA Naming Service プロセスが TPBroker 運用支援機能のプロセス監視対象外となり、プロセスがダウンしても再起動されなくなります。

(z) ejbserver.rmi から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.rmi.localinvocation.scope	ローカル呼び出し最適化機能の適用範囲を指定します。 none を指定した場合： 適用範囲はありません。 app を指定した場合： 同一アプリケーション内が適用範囲になります。 all を指定した場合： 同一 J2EE サーバ内が適用範囲になります。	app	—	「機能解説 基本・開発編(EJB コンテナ)」の「2.13 EJB のリモートインタフェースの呼び出し」
ejbserver.rmi.logger.filenum	J2EE サーバの RMI 通信ログの面数を、2～16 の整数で指定します。	4	—	

キー名称	内容	省略値	VR	関連情報
	数値以外の文字列や範囲外の数値が指定された場合、または文字列が何も指定されていない場合は、メッセージが出力され、デフォルト値が設定されます。			
ejbserver.rmi.logger.filesize	J2EE サーバの RMI 通信ログのサイズ(単位：バイト)を、8192～2147483647 の整数で指定します。	1048576	—	
ejbserver.rmi.naming.host	マルチホームドホスト環境で、J2EE サーバが使用する RMI レジストリや MBean サーバのホスト名称、または IP アドレスを指定します。指定した半角英文字の大文字、小文字の区別はしません。	なし	—	
ejbserver.rmi.naming.port	J2EE サーバが利用する RMI レジストリのポート番号を、1～65535 の整数で指定します。ポート番号には、すでにほかのアプリケーションで使用されているポート番号は指定できません。ほかのアプリケーションで使用されているポート番号を指定して J2EE サーバを起動した場合、起動処理が完了しないことがあります。	23152	—	
ejbserver.rmi.passbyreference [※]	true を指定した場合、リモートインタフェースを持つ EJB のメソッド呼び出しで、引数および戻り値を値渡し (pass by value) ではなく参照渡し (pass by reference) にします。ただし、ejbserver.rmi.localinvocation.scope で設定された範囲を超えた呼び出しをした場合、参照渡しの機能は無効になります。 参照渡しでは、値をコピーして生成するコストを削減できますが、渡された参照を用いて元の値を変更できるため注意が必要です。	false	—	
ejbserver.rmi.remote.listener.port	任意の値を指定して、JMX クライアント (運用管理エージェントなど) による稼働情報取得時のリクエスト受付ポートを固定できます。0～65535 の整数を指定します。 ポート番号には、すでにほかのアプリケーションで使用されているポート番号は指定できません。ほかのアプリケーションで使用されているポート番号を指定して J2EE サーバを起動した場合、起動処理が完了しないことがあります。 このプロパティに 0 を指定した場合、J2EE サーバは稼働情報取得時のリクエスト受付ポートとして動的短命ポートを使用します。 0 の設定で Manager から J2EE サーバの停止処理を行うと、J2EE サーバの停止後に、J2EE サーバが使用していた動的短命ポートをほかのプロセスが新たに使用することがあります。	23550	—	

キー名称	内容	省略値	VR	関連情報
	このとき、運用管理エージェントがその動的短命ポートを新たに使用したプロセスに通信を行うと、受信したプロセスで問題が発生することがあるため、このパラメタには動的短命ポートの範囲外の値を設定することを推奨します。			
ejbserver.rmi.request.timeout	クライアントとサーバ間の通信タイムアウト時間（単位：秒）を 0～86400 の整数で指定します。 0 を指定した場合、またはこのプロパティを指定しない場合、タイムアウトしません。なお、86400 を超える値を設定した場合、警告メッセージを出力して、タイムアウトしません。	0（秒）	—	「機能解説 基本・開発編(EJB コンテナ)」の「2.11.5 RMI-IIOP 通信のタイムアウト」
ejbserver.rmi.stateless.unique_id.enabled	Stateless Session Bean の remove メソッドの呼び出しに対して、呼び出しを不要とするか、必要とするかを指定します。 true を指定した場合： Stateless Session Bean の EJBObject に対して remove メソッドの呼び出しを不要とします。 remove メソッドの呼び出し後にビジネスメソッドを呼び出した場合には、java.rmi.NoSuchObjectException 例外は発生しないで、ビジネスメソッドが実行されます。 false を指定した場合： Stateless Session Bean の EJBObject に対して remove メソッドの呼び出しを必要とします。 remove メソッドの呼び出し後にビジネスメソッドを呼び出した場合には、java.rmi.NoSuchObjectException 例外が発生します。	false	—	「機能解説 基本・開発編(EJB コンテナ)」の「4.3.1 Stateless Session Bean 実装時の注意事項」

(凡例)

—：08-00 より前のバージョンを示します。

空欄：関連情報はありません。

注※

サーバ管理コマンドから EJB 単位での参照渡し (Pass-by-reference) の設定をすることもできます。

(aa) ejbserver.security から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.security.digest.algorithm	セキュリティ管理機能での認証に使用するメッセージダイジェストアルゴリズムを指定します。指定可能値は MD5, SHA-1, SHA-256, またはその他の JDK がサポートしている MessageDigest アルゴリズムです。Java SE の API リファレンスの <code>java.security.MessageDigest</code> を参照してください。不正な値を指定した場合、デフォルト値が設定されます。 プロパティの値を変更した場合は、ユーザを削除し再作成する必要があります。この機能では、文字列が異なれば別のメッセージダイジェストアルゴリズムとみなします。 SHA-1 以外のアルゴリズムを使用し、かつ EJB クライアントでセキュリティの実装をする場合、EJB クライアントは 09-70 以降の必要があります。09-70 より前の EJB クライアントの場合、認証に失敗します。 09-70 より前の EJB クライアントを使用してセキュリティの実装をする場合、認証先の J2EE サーバのダイジェストアルゴリズムを SHA-1 にする必要があります。	SHA-1	09-70

(ab) ejbserver.server から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
ejbserver.server.eheap.httpsession.enabled	このキーは、JDK11 を使用する場合だけ指定できます。 HTTP セッションに格納するオブジェクトを Explicit ヒープに配置するかどうかを指定します。 true を指定した場合： HTTP セッションに格納するオブジェクトを Explicit ヒープに配置します。 false を指定した場合： HTTP セッションに格納するオブジェクトを Java ヒープ領域に配置します。 ただし、次の場合は、このプロパティは無効 (false 指定時と同じ挙動) となります。	true	08-00	

キー名称	内容	省略値	VR	関連情報
	- JavaVM オプション HitachiUseExplicitMemory を無効にした場合 - JDK17 以降の日立 Java VM を使用している場合			
ejbserver.server.mutex.invocation.timeout	同時に同一プライマリキーの Entity Bean にアクセスした場合に掛かる排他のタイムアウト時間（単位：秒）を、45～92233720368547758 の整数で指定します。タイムアウト時間は、デフォルト値（45 秒）以上 java.lang.Long.MAX_VALUE/1000（秒）以下の範囲で指定してください。デフォルト値未満の値を指定した場合、タイムアウト時間は 45（秒）で動作します。 java.lang.Long.MAX_VALUE/1000（秒）を超える値を指定した場合、タイムアウト時間は java.lang.Long.MAX_VALUE（ミリ秒）で動作します。さらに、java.lang.Long.MAX_VALUE を超える値を指定した場合は、デフォルト値になります。	45	—	
ejbserver.server.prf.PRFID	PRF 識別子を指定します。 PRF デーモン起動時に PRF 識別子を指定した場合に、同じ PRF 識別子を指定します。 PRF デーモン起動時に PRF 識別子を省略した場合は、指定しないでください。PRF 識別子が一致しないと、性能解析トレースが取得されません。	PRF_ID	—	
ejbserver.server.threaddump.fileenum	J2EE サーバが障害を検知して自発的にスレッドダンプを出力する場合のスレッドダンプのファイル数の上限を、1～2147483647 の整数で指定します。 範囲外の値を指定した場合は、デフォルト値が設定されます。 環境変数 JAVACOREDIR を指定している場合、環境変数 JAVACOREDIR で指定したディレクトリとデフォルトの出力先ディレクトリ（Windows の場合：<製品の作業ディレクトリ>¥ejb¥<サーバ名称>、UNIX の場合：<製品の作業ディレクトリ>/ejb/<サーバ名称>）の両方のスレッドダンプのファイルの個数の合計が上限になります。 また、外部からのスレッドダンプ出力要求に対しては、このオプションの設定に関係なくスレッドダンプのファイルが出力されます。外部からのスレッドダンプ出力要求の例を次に示します。	256	—	「機能解説 運用／監視／連携編」の「4. リソースの枯渇監視」

キー名称	内容	省略値	VR	関連情報
	• jheapprof コマンド • cjdumpsrv コマンド • cjstopsv コマンドの-fd 指定 • J2EE サーバ実行コンソールでの Ctrl+Break (Windows の場合) • J2EE サーバプロセスへの kill コマンド -3 (SIGTERM) 指定 (UNIX の場合)			

(凡例)

－：08-00 より前のバージョンを示します。

空欄：関連情報はあります。

(ac) ejbserver.stateful から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.stateful.passivate.switch	Stateful Session Bean の非活性化、活性化の機能を使用するかどうかを指定します。 true を指定した場合： Stateful Session Bean の非活性化、活性化の機能が動作します。 false を指定した場合： Stateful Session Bean の非活性化、活性化の機能が動作しません。	false	－

(凡例)

－：08-00 より前のバージョンを示します。

(ad) ejbserver.stdoutlog から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.stdoutlog.autoflush	ユーザ出力ログおよびユーザエラーログの自動フラッシュ機能を有効にするかどうかを指定します。 自動フラッシュ機能とは、ユーザ出力ログ (user_out[n]log)、ユーザエラーログ (user_err[n].log) の出力に使用している java.io.PrintStream の自動フラッシュオプションのことです。	false	08-00

キー名称	内容	省略値	VR
	true を指定した場合： 自動フラッシュ機能を有効にします。無効のケースよりも性能が劣化することがあります。 false を指定した場合： 自動フラッシュ機能を無効にします。 java.io.PrintStream.print() など、PrintStream クラスの自動フラッシュ対象外の処理がされた場合、ユーザ出力ログ、ユーザエラーログに出力される内容がバッファにたまり、メモリを圧迫する要因となります。		

(ae) ejbserver.watch から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.watch.defaultRequestQueue.enabled	デフォルトの実行待ちキューの場合の HTTP リクエスト実行待ちキュー監視のアラート出力を有効にするかどうかを指定します。 true を指定した場合： デフォルトの実行待ちキューの場合の HTTP リクエスト実行待ちキュー監視のアラート出力を有効にします。 false を指定した場合： デフォルトの実行待ちキューの場合の HTTP リクエスト実行待ちキュー監視のアラート出力を無効にします。	true	—
ejbserver.watch.defaultRequestQueue.interval	デフォルトの実行待ちキューの場合の HTTP リクエスト実行待ちキュー監視間隔を、1～2147483647 の整数（単位：秒）で指定します。	30	—
ejbserver.watch.defaultRequestQueue.threshold	アラートメッセージを出力するしきい値を、1～100 の整数（単位：%）で指定します。監視対象のリクエストのデフォルトの実行待ちキューの格納率が、このキーで指定された値以上になるとアラートメッセージを出力します。	80	—
ejbserver.watch.defaultRequestQueue.writefile.enabled	デフォルトの実行待ちキューの場合の HTTP リクエスト実行待ちキュー監視結果をファイル出力するかどうかを指定します。 true を指定した場合： デフォルトの実行待ちキューの場合の HTTP リクエスト実行待ちキュー監視結果をファイル出力します。 false を指定した場合： デフォルトの実行待ちキューの場合の HTTP リクエスト実行待ちキュー監視結果をファイル出力しません。	true	—

キー名称	内容	省略値	VR
ejbserver.watch.enabled	すべてのリソース枯渇監視を有効にするかどうかを指定します。 true を指定した場合： すべてのリソース枯渇監視を有効にします。 false を指定した場合： すべてのリソース枯渇監視を無効にします。この場合、各リソース枯渇監視が有効に設定されていても、すべてのリソース枯渇監視が無効になります。	true	—
ejbserver.watch.fileDescriptor.enabled	ファイルディスクリプタ監視のアラート出力を有効にするかどうかを指定します。なお、ファイルディスクリプタ監視は、Windows の場合、および AIX の場合は使用できません。 true を指定した場合： ファイルディスクリプタ監視のアラート出力を有効にします。 false を指定した場合： ファイルディスクリプタ監視のアラート出力を無効にします。	true	—
ejbserver.watch.fileDescriptor.interval	ファイルディスクリプタ監視間隔を 1～2147483647 の整数（単位：秒）で指定します。なお、ファイルディスクリプタ監視は、Windows の場合、および AIX の場合は使用できません。	60	—
ejbserver.watch.fileDescriptor.threshold	ファイルディスクリプタの使用状態を監視するしきい値を、1～2147483647 の整数で指定します。使用中のファイルディスクリプタ数がしきい値以上になった場合にアラートを出力します。指定するしきい値は次のように設定してください。 OS のプロセスに割り当て可能なファイルディスクリプタ数※＞ファイルディスクリプタ見積み式から求めた値＞ファイルディスクリプタしきい値 注※ プラットフォームによってシステムの上限が存在しない場合があります。 なお、ファイルディスクリプタ監視は、Windows の場合、および AIX の場合は使用できません。	2147483647	—
ejbserver.watch.fileDescriptor.writefile.enabled	ファイルディスクリプタ監視結果をファイル出力するかどうかを指定します。なお、ファイルディスクリプタ監視は、Windows の場合、および AIX の場合は使用できません。 true を指定した場合： ファイルディスクリプタ監視結果をファイル出力します。 false を指定した場合： ファイルディスクリプタ監視結果をファイル出力しません。	true	—
ejbserver.watch.memory.enabled	メモリ監視のアラート出力を有効にするかどうかを指定します。 true を指定した場合： メモリ監視のアラート出力を有効にします。 false を指定した場合： メモリ監視のアラート出力を無効にします。	true	—

キー名称	内容	省略値	VR
	true を指定する場合、次の JavaVM オプションを同じ値で設定する必要があります。 • -XX:MetaspaceSize • -XX:MaxMetaspaceSize 異なる値を設定した場合、アラートが出力されないで FullGC が発生する場合があります。		
ejbserver.watch.memory.interval	メモリ監視間隔を 1～2147483647 の整数（単位：秒）で指定します。	60	—
ejbserver.watch.memory.threshold	メモリの使用状態を監視するしきい値を、1～100 の整数（単位：%）で指定します。次の条件のどれかで、FullGC の予兆を検知した場合に、アラートを出力します。 SerialGC 有効時： • Tenured 領域消費サイズの Tenured 領域合計サイズに対する割合（リソース枯渇監視情報の Rate1 に出力される値）がしきい値以上 • New 領域合計サイズの Tenured 領域最大空きサイズに対する割合（リソース枯渇監視情報の Rate2 に出力される値）がしきい値以上 • Metaspace 領域消費サイズの Metaspace 領域最大サイズ（リソース枯渇監視情報の Rate3 に出力される値）に対する割合がしきい値以上 G1GC 有効時： • Java ヒープ領域消費サイズの Java ヒープ領域最大メモリサイズに対する割合（リソース枯渇監視情報の Rate1 に出力される値）がしきい値以上 • Metaspace 領域消費サイズの Metaspace 領域最大サイズに対する割合（リソース枯渇監視情報の Rate3 に出力される値）がしきい値以上 ZGC 有効時： • Metaspace 領域消費サイズの Metaspace 領域最大サイズに対する割合（Rate3 として出力される値）がしきい値以上の場合	80	—
ejbserver.watch.memory.writefile.enabled	メモリ監視結果をファイル出力するかどうかを指定します。 true を指定した場合： メモリ監視結果をファイル出力します。 false を指定した場合： メモリ監視結果をファイル出力しません。	true	—
ejbserver.watch.thread.enabled	スレッド監視のアラート出力を有効にするかどうかを指定します。なお、Linux ではスレッド数を監視できません。 true を指定した場合： スレッド監視のアラート出力を有効にします。 false を指定した場合： スレッド監視のアラート出力を無効にします。	true	—

キー名称	内容	省略値	VR
ejbserver.watch.thread.interval	スレッド監視間隔を、1～2147483647 の整数（単位：秒）で指定します。なお、Linux ではスレッド数を監視できません。	60	—
ejbserver.watch.thread.threshold	スレッドの使用状態を監視するしきい値を、1～2147483647 の整数で指定します。生成済みのスレッド数がしきい値以上になった場合にアラートを出力します。指定するしきい値は次のように設定してください。 OS のプロセスに割り当て可能なスレッド数※>スレッド見積もり式より求めた値>スレッドしきい値 注※ プラットフォームによってシステムの上限が存在しない場合があります。 なお、Linux ではスレッド数を監視できません。	2147483647	—
ejbserver.watch.thread.writefile.enabled	スレッド監視結果をファイル出力するかどうかを指定します。なお、Linux ではスレッド数を監視できません。 true を指定した場合： スレッド監視結果をファイル出力します。 false を指定した場合： スレッド監視結果をファイル出力しません。	true	—
ejbserver.watch.thread.dump.enabled	スレッドダンプ監視のアラート出力を有効にするかどうかを指定します。 true を指定した場合： スレッドダンプ監視のアラート出力を有効にします。 false を指定した場合： スレッドダンプ監視のアラート出力を無効にします。	true	—
ejbserver.watch.thread.dump.interval	スレッドダンプ監視時間間隔を、1～2147483647 の整数（単位：秒）で指定します。	30	—
ejbserver.watch.thread.dump.threshold	スレッドダンプ監視のしきい値を、上限値に対する割合 1～100 の整数（単位：%）で指定します。 スレッドダンプのファイル数の上限値に対する現在値の割合がしきい値以上になった場合にアラートを出力します。 環境変数 JAVACOREDIR を指定している場合、環境変数 JAVACOREDIR で指定したディレクトリとデフォルトの出力先ディレクトリの両方のスレッドダンプのファイルの個数の合計が現在値となります。	80	—
ejbserver.watch.thread.dump.writefile.enabled	スレッドダンプ監視結果をファイル出力するかどうかを指定します。 true を指定した場合： スレッドダンプ監視結果をファイル出力します。 false を指定した場合： スレッドダンプ監視結果をファイル出力しません。	true	—
ejbserver.watch.memory.rate2alert.enabled	リソース枯渇監視機能のメモリ枯渇監視情報の Rate2 のアラート出力を有効にするかどうかを指定します。	false	Cosminexus

キー名称	内容	省略値	VR
	true を指定した場合： Rate2 のアラート出力を有効にします。ただし、G1GC 有効時に Rate2 のアラート出力を有効にしても、Rate2 は常に-1 であるため、アラート出力されません。 false を指定した場合： Rate2 のアラート出力を無効にします。		Component Container 09-70-12 以降および 09-80-02 以降

(凡例)

－：08-00 より前のバージョンを示します。

(af) ejbserver.webj2ee から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
ejbserver.webj2ee.connectionAutoClose.enabled	自動クローズ機能を使用するかどうかを指定します。 true を指定した場合： 自動クローズ機能を有効にします。 false を指定した場合： 自動クローズ機能を無効にします。 不正な値を指定した場合は、デフォルト値が設定されます。	true	－

(凡例)

－：08-00 より前のバージョンを示します。

(ag) java から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
java.naming.factory.initial	JNDI の InitialContext がデレゲートするコンテキストの実装クラスのファクトリクラスを指定します。各設定値での動作を次に示します。 com.hitachi.software.ejb.jndi.InsContextFactory： JNDI ラウンドロビン検索機能を使用しないで、通常の検索機能を使用する場合に指定するコンテキストです。 com.hitachi.software.ejb.jndi.GroupContextFactory： J2EE サーバ内アプリケーション (EJB) の実行時に、JNDI ラウンドロビン検索機能を使用する場合に指定するコンテキストです。	com.hitachi.software.ejb.jndi.InsContextFactory	－

(凡例)

－：08-00 より前のバージョンを示します。

(ah) vbj から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
vbj.java2iop.jvm.maxHeapSize	J2EE アプリケーション開始時に起動される java2iop コマンドのヒープサイズの最大値を指定します。 指定する値は、java コマンドの-Xmx, -Xms オプションが受け付ける形式で指定してください。それ以外の形式で指定された場合の動作は保証されません。 次のように指定してください。 2048～4294966272 の値を指定してください。ただし、vbj.java2iop.jvm.minHeapSize に指定した以上の値を指定してください。 - キロバイト単位で指定する場合は、文字「k」または「K」を付けて指定してください。 - メガバイト単位で指定する場合は、文字「m」または「M」を付けて指定してください。	128m	－
vbj.java2iop.jvm.minHeapSize	J2EE アプリケーション開始時に起動される java2iop コマンドのヒープサイズの初期値を指定します。 指定する値は、java コマンドの-Xmx, -Xms オプションが受け付ける形式で指定してください。それ以外の形式で指定された場合の動作は保証されません。 次のように指定してください。 1024～-Xmx 指定値の値を指定してください。 - キロバイト単位で指定する場合は文字「k」または「K」を付けて指定してください。 - メガバイト単位で指定する場合は、文字「m」または「M」を付けて指定してください。	16m	－

(凡例)

－：08-00 より前のバージョンを示します。

(ai) vbroker から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

注意事項

ここに記載していない TPBroker のプロパティは指定できません（指定した場合の動作は保証されません）。

キー名称	内容	省略値	VR	関連情報
vbroker.agent.enableLocator	スマートエージェントを使用するかどうかを指定します。 通常は設定不要のプロパティです。デフォルトのままご使用ください。 また、CTM 連携機能を有効 (ejbserver.ctm.enabled キーに true を指定) にした場合は、J2EE サーバの起動時に自動的に true が設定されます。 この目的以外でスマートエージェントが必要な場合は、true を設定してください。 なお、このプロパティに true を指定した場合は、スマートエージェントを先に起動する必要があるため、また、併せて vbroker.orb.htc.oadNoUse=true も指定してください。 スマートエージェントの詳細については、マニュアル「Borland(R) Enterprise Server VisiBroker(R) プログラマーズリファレンス」を参照してください。	false	—	
vbroker.agent.port	スマートエージェントのポート番号を指定します。 詳細については、マニュアル「Borland(R) Enterprise Server VisiBroker(R) プログラマーズリファレンス」を参照してください。	14000	—	
vbroker.ce.iiop.ccm.htc.readPerConnection	リモートインタフェースとして定義された EJB メソッドの呼び出しでタイムアウトが発生した場合に、コネクションのクローズを抑制するかどうかを設定します。 true を指定した場合 タイムアウト発生時のコネクションのクローズを抑制します。 false を指定した場合 タイムアウト発生時のコネクションのクローズを抑制しません。	false	09-50	
vbroker.ce.iiop.ccm.htc.threadStarter	リプライ受信専用スレッドを管理するスレッドを起動するかどうかを設定します。	false	09-50	

キー名称	内容	省略値	VR	関連情報
	vbroker.ce.iiop.ccm.htc.readerPerConnection=true を設定する場合に, "true"を設定してください。			
vbroker.orb.htc.comt.entryCount	TPBroker 通信トレースファイル 1 個当たりのエントリ数の上限値を 100～300000000 の範囲で指定します。	120000	—	
vbroker.orb.htc.comt.fileCount	TPBroker 通信トレースファイルの個数の上限値を 1～256 の範囲で指定します。	3	—	
vbroker.orb.htc.tracePath	TPBroker のトレースファイル出力先のパスを 1～210 バイトの範囲で指定します。指定したパスのサブディレクトリとして comtrc と mdltrc をあらかじめ作成しておく必要があります。デフォルトの出力先の場合, サブディレクトリ comtrc と mdltrc はサーバ初回起動時に自動作成されます。ディレクトリのパスの区切り記号には, "/"を使用してください。 例えば, Windows の場合, work ディレクトリとして C:¥temp¥work を設定するときは, 次のように指定します。 (指定例) vbroker.orb.htc.tracePath=c:/temp/work	- Windows の場合 <製品の作業ディレクトリ>¥ejb¥<サーバ名称>¥logs¥TPB¥logj - UNIX の場合 <製品の作業ディレクトリ>/ejb/<サーバ名称>/logs/TPB/logj	—	
vbroker.se.iiop_tp.host	任意の値を指定して, J2EE サーバ単位で EJB コンテナの IP アドレスを固定します。	なし	—	「機能解説 基本・開発編 (EJB コンテナ)」の「2.14 EJB コンテナの通信ポートと IP アドレスの固定 (TPBroker のオプション)」

キー名称	内容	省略値	VR	関連情報
vbroker.se.iiop_tp.proxyHost ※	J2EE サーバおよび CORBA ネーミングサービスが配置してあるホスト名を指定します。ここで指定するホスト名は、EJB クライアントでは NAT の IP アドレスとして解決されるように設定してください。また、NAT を経由しない EJB クライアントがある場合は、そのホストでは J2EE サーバの IP アドレスとして解決されるように設定してください。	なし	09-50	
vbroker.se.iiop_tp.scm.iiop_tp.listener.port	任意の値を指定して、J2EE サーバ単位で通信ポートを固定できます。ポート番号は、ほかのプログラムなどと重複しないようにしてください。 このプロパティを設定しない場合は、TPBroker によってランダムな値が設定されます。 また、ejbserver.container.ejhome.sessionbean.reconnect.enabled に true を指定した場合は、必ず値を指定して、ポート番号を固定にしてください。	0	—	
vbroker.se.iiop_ts.proxyHost ※	J2EE サーバおよび CORBA ネーミングサービスが配置してあるホスト名を指定します。ここで指定するホスト名は、EJB クライアントでは NAT の IP アドレスとして解決されるように設定してください。また、NAT を経由しない EJB クライアントがある場合は、そのホストでは J2EE サーバの IP アドレスとして解決されるように設定してください。	なし	09-50	

(凡例)

— : 08-00 より前のバージョンを示します。

注※

EJB クライアントと J2EE サーバ間に NAT が設置されている場合に、J2EE サーバおよび CORBA ネーミングサービスに対して設定するキーです。設定方法は、CORBA ネーミングサービスの起動モードによって異なります。

CORBA ネーミングサービスの起動モードをインプロセスで自動起動するモードの場合、J2EE サーバ用ユーザプロパティファイルにこれらのキーを指定します。J2EE サーバを配置したマシンのホスト名が「MyJ2EEHost」の場合の設定例を示します。

```
# CORBAネーミングサービス起動モード設定
ejbserver.naming.startupMode=inprocess

# NATを介した構成でのEJBクライアント-J2EEサーバ間通信用設定
vbroker.se.iiopt.proxyHost=MyJ2EEHost
vbroker.se.iiopt.proxyHost=MyJ2EEHost
```

CORBA ネーミングサービスをアウトプロセスで自動起動するモードの場合、J2EE サーバ用ユーザプロパティファイルおよび J2EE サーバ用ユーザプロパティ ejbserver.naming.exec.args の両方にこれらのキーを指定します。J2EE サーバを配置したマシンのホスト名が「MyJ2EEHost」の場合の設定例

を示します。なお、ejbserver.naming.exec.args の設定は-J ごとにスペースで区切って一行で指定してください。また、既存のシステムで ejbserver.naming.exec.args に設定値がある場合は、そのコマンドに追加で指定してください。

```
# CORBAネーミングサービス起動モード設定
ejbserver.naming.startupMode=automatic

# NATを介した構成でのEJBクライアント-J2EEサーバ間通信用設定
vbroker.se.iiop_ts.proxyHost=MyJ2EEHost
vbroker.se.iiop_tp.proxyHost=MyJ2EEHost
ejbserver.naming.exec.args=-J-Dvbroker.se.iiop_ts.proxyHost=MyJ2EEHost -J-Dvbroker.se.iio
p_tp.proxyHost=MyJ2EEHost
```

CORAB ネーミングサービスをアウトプロセスで手動起動するモードの場合、J2EE サーバ用ユーザプロパティファイルおよび nameserv コマンド起動オプションの両方にこれらのキーを指定します。J2EE サーバを配置したマシンのホスト名が「MyJ2EEHost」の場合の設定例を示します。

```
# CORBAネーミングサービス起動モード設定
ejbserver.naming.startupMode>manual

# NATを介した構成でのEJBクライアント-J2EEサーバ間通信用設定
vbroker.se.iiop_ts.proxyHost=MyJ2EEHost
vbroker.se.iiop_tp.proxyHost=MyJ2EEHost
```

nameserv コマンドの実行例を示します。なお、nameserv コマンドの実行の際は-J ごとのスペース区切りによって一行で設定してください。また、既存のシステムで nameserv コマンドを実行している場合は、そのコマンドに追記してください。

```
nameserv -J-Dvbroker.agent.enableLocator=false
-J-Djava.security.policy=="<Application Serverのインストールディレクトリ>%CC%server%sysco
nf%cli.policy"
-J-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=<ポート番号>
-J-Dvbroker.se.iiop_ts.proxyHost=MyJ2EEHost
-J-Dvbroker.se.iiop_tp.proxyHost=MyJ2EEHost
```

(aj) webserver.application から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.application.lower_version	Web アプリケーションのバージョン設定機能で設定したい Web アプリケーションのバージョンを指定します。 2.4 を指定した場合は、web.xml で定義された Web アプリケーションバージョンが 2.4 以下の Web アプリケーションをバージョン 2.4 の Web アプリケーションとして実行します。 2.5 を指定した場合は、web.xml で定義された Web アプリケーションバージョンが 2.5 以下の Web アプリケーションをバージョン 2.5 の Web アプリケーションとして実行します。	なし	08-20

(ak) webserver.connector から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
webserver.connector.http.bind_host	管理用サーバで使用するローカル IP アドレス、または解決できるローカルホスト名称を指定します。 IP アドレスまたはホスト名称の前後の半角スペースは無視されます。値を指定しない場合は、ワイルドカードアドレスが使用されます。	なし	—	
webserver.connector.http.permitted.hosts	管理用サーバへのアクセスを許可するホストの IP アドレス（10 進表記）またはホスト名を指定します。 なお、ローカルホスト（"localhost"に関連づけられたアドレス※）は明記しなくても常にアクセスできます。したがって、このキーの指定を省略した場合、アクセスできるのはローカルホスト（"localhost"に関連づけられたアドレス※）だけになります。 複数指定する場合には IP アドレスまたはホスト名の間をコンマ（,）で区切ります。アクセス制限をしない場合はアスタリスク（*）だけを指定します。IP アドレスまたはホスト名の前後の半角スペースは無視されます。	なし	—	
webserver.connector.limit.max_parameter_count	リクエストパラメタ数の上限値を-1～2147483647 の整数で指定します。 リクエストパラメタ数の上限値の有効範囲は、リクエストのクエリストリングと、Content-Type ヘッダの値が application/x-www-form-urlencoded または multipart/form-data のフォームデータのパラメタ数の合計とします。 リクエストパラメタ数の上限値を設定しない場合は、-1 を指定します。 数値以外の文字列や範囲外の数値、空文字列または空白文字を指定した場合は、メッセージが出力され、デフォルト値が設定されます。 リクエストのパラメタ数が指定した値を超えた場合、API によって次のように動作し、KDJE39341-W、または KDJE39342-E のメッセージが出力されます。	10000	09-00	

キー名称	内容	省略値	VR	関連情報
	IllegalStateException をスローする API • javax.servlet.http.HttpServletRequest .getPart(String) • javax.servlet.http.HttpServletRequest .getParts() 読み込んだ分のリクエストパラメタを用いて戻り値を返す API • javax.servlet.ServletRequest.getParameter(String) • javax.servlet.ServletRequest.getParameterMap() • javax.servlet.ServletRequest.getParameterNames() • javax.servlet.ServletRequest.getParameterValues(String)			
webserver.connector.limit.max_post_form_data	POST リクエストのフォームデータの最大サイズを、-1～2147483639 の整数（単位：バイト）で指定します。最大サイズを設定しない場合は、-1 を指定します。 次の数値または文字列を指定した場合、メッセージが出力され、デフォルト値が適用されます。 • 数値以外の文字列 • 指定範囲外の数値 • 空文字列または空白文字 Content-Type ヘッダの値が application/x-www-form-urlencoded の場合： リクエストの Content-Length ヘッダの値が、指定した最大サイズを超える場合、ステータスコード 413（Request Entity Too Large）エラーが返却され、メッセージ KDJE39336-E が出力されます。 リクエストボディがチャンク形式の場合、Servlet API 呼び出しの延長で、受信した POST リクエストサイズが指定した最大値を超えると、java.lang.IllegalStateException をスローし、メッセージ KDJE39336-E が出力されます。 Content-Type ヘッダの値が multipart/form-data の場合： Servlet API 呼び出しの延長で、マルチパート形式で受信したフォームデータのうち、ファイル以外のすべてのデータについて、パラメタの名前や値のサイズを合計した値が指定された上限値を超えるかどうか確認します。上限値を超えた場合、	2097152	08-50	

キー名称	内容	省略値	VR	関連情報
	java.lang.IllegalStateException がスローされ、メッセージ KDJE39336-E が出力されます。			
webserver.connector.nio_http.backlog	Web クライアントからの接続要求を格納する TCP リンキューの長さ（単位：バイト）を、1～2147483647 の整数で指定します。 有効な指定値の最大値や実際に設定される TCP リンキューの長さは OS によって異なります。	511	11-00	
webserver.connector.nio_http.bind_host	NIO HTTP サーバで使用する IP アドレス、またはホスト名を指定します。IP アドレス、またはホスト名の前後の半角スペースは無視されます。値を指定しない場合は、ワイルドカードアドレスが使用されます。 指定されたホスト名、または IP アドレスが解決できない場合： KDJE39565-W を出力し 、ワイルドカードアドレスが使用されます。 ローカルでないホストのホスト名、または IP アドレスを指定した場合： KDJE39567-W を出力し、ワイルドカードアドレスが使用されます。	なし	11-00	
webserver.connector.nio_http.hostname_lookups	NIO HTTP サーバで受信したリクエストに対して、クライアントの IP アドレスをホスト名に変換するために、ホスト名のルックアップの逆引きをするかどうかを指定します。 ホスト名の逆引きを行うと、スループットが低下します。 ホスト名を解決しなかった場合、javax.servlet.ServletRequest インタフェースの getRemoteHost() メソッドの結果や、ログファイルに出力するクライアントの IP アドレスは、ドット (.) で区切られた書式の IP アドレスとなります。 true を指定した場合： IP アドレスをホスト名に変換します。 false を指定した場合： IP アドレスをホスト名に変換しません。	false	11-00	
webserver.connector.nio_http.idle_thread_timeout	NIO HTTP サーバのスレッドプール内のスレッドがアイドル状態を継続できる最長時間（単位：秒）を、1～2147483647 の整数で指定します。	60	11-00	
webserver.connector.nio_http.keep_alive.max_requests	TCP コネクションを維持したままの連続接続回数の上限を、0～2147483647 の整数で指定します。上限値を指定しない場合は 0 を指定します。	0	11-00	

キー名称	内容	省略値	VR	関連情報
webserver.connector.nio_http.keep_alive.timeout	TCP コネクションを持続した状態での、リクエスト待ち時間（単位：秒）を 0～3600 の整数で指定します。0 を指定した場合、タイムアウトは有効になりません。	0	11-00	
webserver.connector.nio_http.limit.max_headers	HTTP/1.0 および HTTP/1.1 での、HTTP リクエストに含まれる HTTP ヘッダの個数の上限を 0～32767 の整数で指定します。指定値 0 は個数 0 を表します。 このプロパティで設定した HTTP ヘッダの個数に満たない場合であっても、webserver.connector.nio_http.limit.max_request_header プロパティで指定したサイズを超えた場合はエラーとなります。	100	11-00	
webserver.connector.nio_http.limit.max_request_body	HTTP リクエストのリクエストボディの最大サイズ（単位：バイト）を、-1～9223372036854775807 の整数で指定します。上限値を指定しない場合は-1 を指定します。 またリクエストボディがチャンク形式で送信された場合、チャンクヘッダのサイズも、指定するサイズに含める必要があります。	-1	11-00	
webserver.connector.nio_http.limit.max_request_header	HTTP/1.0 および HTTP/1.1 での、HTTP リクエストのリクエストヘッダの最大サイズ（単位：バイト）を、7～65536 の整数で指定します。 このプロパティで設定したリクエストヘッダの最大サイズに満たない場合であっても、webserver.connector.nio_http.limit.max_headers プロパティで指定した HTTP ヘッダの個数を超えた場合はエラーとなります。 また、HTTP ヘッダの終わりを示す改行文字 (CR(0x0d)+LF(0x0A)の 2 バイト)も指定するサイズに含める必要があります。	16384	11-00	
webserver.connector.nio_http.limit.max_request_trailer	HTTP/1.1 リクエストのリクエストトレイラの最大サイズ（単位：バイト）を、1～65536 の整数で指定します。 このプロパティで設定したリクエストトレイラの最大サイズに満たない場合であっても、webserver.connector.nio_http.limit.max_trailers プロパティで指定した HTTP トレイラの個数を超えた場合はエラーとなります。 また、HTTP ヘッダの終わりを示す改行文字 (CR(0x0d)+LF(0x0A)の 2 バイト)も指定するサイズに含める必要があります。 webserver.connector.nio_http.limit.max_request_header とは別カウントになります。	16384	11-20	

キー名称	内容	省略値	VR	関連情報
webserver.connector.nio_http.limit.max_trailers	HTTP/1.1 リクエストに含まれる HTTP トレイラの個数の上限を 0～32767 の整数で指定します。指定値 0 は個数 0 を表します。 このプロパティで設定した HTTP ヘッダの個数に満たない場合であっても、webserver.connector.nio_http.limit.max_request_trailer プロパティで指定したサイズを超えた場合はエラーとなります。 webserver.connector.nio_http.limit.max_headers とは別カウントになります。	100	11-20	
webserver.connector.nio_http.max_connections	Web クライアントとの最大接続数を 1～2147483647 の整数で指定します。有効な最大値は実行するプラットフォームに依存します。	1024	11-00	
webserver.connector.nio_http.max_servlet_execute_threads	NIO HTTP サーバを介して Web コンテナが同期サーブレットを実行する場合の、Web コンテナ全体の最大同時実行スレッド数を、1～2147483647 の整数で指定します。 指定する値は、NIO HTTP サーバの処理スレッド数の最大値 (webserver.connector.nio_http.max_threads パラメタに指定した値) 以下である必要があります。NIO HTTP サーバの処理スレッド数の最大値を超える値を指定した場合は、メッセージ KDJE39010-W を出力し、処理スレッド数の最大値は、Web コンテナがリクエストを処理する同時実行数として設定されます。 このパラメタで制御できる最大同時実行スレッド数には、同期サーブレット以外の実行スレッド (非同期サーブレットの実行スレッドや非同期 I/O API によるコールバックスレッドなど) は含まれません。同期サーブレット以外の実行スレッドも含めたすべての処理スレッドの最大値は webserver.connector.nio_http.max_threads パラメタで指定します。	100	11-00	
webserver.connector.nio_http.max_threads	NIO HTTP サーバの処理スレッド数の最大値を 1～2147483647 の整数で指定します。	100	11-00	
webserver.connector.nio_http.min_threads	NIO HTTP サーバの処理スレッド数の最小値を 1～2147483647 の整数で指定します。 指定する値は、スレッド数の最大値 (webserver.connector.nio_http.max_threads パラメタに指定した値) 以下である必要があります。スレッド数の最大値を超える値を設定した場合は、メッセージ KDJE39010-W を出力し、スレッド数の最大値が、最小値として設定されます。	10	11-00	

キー名称	内容	省略値	VR	関連情報
	有効な最大値は実行するプラットフォームに依存します。			
webserver.connector.nio_http.permitted.hosts	NIO HTTP サーバへのアクセスを許可するホストの IP アドレス (10 進表記), またはホスト名を指定します。複数指定する場合には IP アドレス, またはホスト名の間をコンマ (,) で区切ります。アクセス制限をしない場合はアスタリスク (*) だけを指定します。 なお, ローカルホストは明記しなくても常にアクセスが許可されます。空文字列または空白文字を指定した場合は, メッセージ KDJE39009-W を出力し, デフォルト値が設定されます。 指定されたホスト名が解決できない場合, KDJE39563-W を出力し, ローカルホストからのアクセスだけを許可します。 IP アドレス, またはホスト名の前後のスペースは無視されます。	*	11-00	
webserver.connector.nio_http.port	NIO HTTP サーバが使用するポート番号を 1～65535 の整数で指定します。すでに他のアプリケーションで使用されているポート番号, または確保されていないポート番号は指定できません。すでに使用されているポート番号や確保されているポート番号を指定した場合は, メッセージ KDJE39566-E を出力して, J2EE サーバが起動されません。	8008	11-00	
webserver.connector.nio_http.receive_timeout	Web クライアントからのリクエスト受信で, タイムアウトするまでの時間 (単位: 秒) を 0～3600 の整数で指定します。0 を指定した場合, タイムアウトは有効になりません。	300	11-00	
webserver.connector.nio_http.response.header.server	レスポンスに自動的に付加する Server ヘッダの値を指定します。	CosminexusComponentContainer	11-00	
webserver.connector.nio_http.send_timeout	Web クライアントへのレスポンス送信で, タイムアウトするまでの時間 (単位: 秒) を 0～3600 の整数で指定します。0 を指定した場合, タイムアウトは有効になりません。	300	11-00	
webserver.connector.nio_http.setQueryEncoding.enable	NIO HTTP サーバでのリクエストボディのエンコード指定をクエリにも適用するかどうかを指定します。 true を指定した場合: リクエストボディのエンコード指定をクエリにも適用します。	false	11-20	

キー名称	内容	省略値	VR	関連情報
	false を指定した場合： リクエストボディのエンコード指定をクエリに適用しません。			

(凡例)

－：08-00 より前のバージョンを示します。

注※

J2EE サーバ起動時に"localhost"に関連づけられたアドレスになります。

(al) webserver.container から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
webserver.container.jaxws.webservice.no_webxml.enabled	Web サービスのクライアントから SOAP 要求メッセージが到着した際、WAR ファイルに web.xml が完全に、または部分的に含まれていなくても、この製品の JAX-WS 機能を利用して開発した Web サービス実装クラスまたはプロバイダ実装クラスが含まれていれば、SOAP 要求メッセージを受け付けて、Web サービス実装クラスに振り分けるかどうか指定します。 strict または true を指定した場合： WAR ファイルに web.xml が含まれない場合、@javax.jws.WebService アノテーション、または @javax.xml.ws.WebServiceProvider アノテーションの情報に基づき、SOAP 要求メッセージを Web サービス実装クラス、またはプロバイダ実装クラスに振り分けます。 lax を指定した場合： WAR ファイルに web.xml が完全に、または部分的に含まれない場合、@javax.jws.WebService アノテーション、または @javax.xml.ws.WebServiceProvider アノテーションの情報に基づき、SOAP 要求メッセージを Web サービス実装クラス、またはプロバイダ実装クラスに振り分けます。	none	08-00	

キー名称	内容	省略値	VR	関連情報
	none または false を指定した場合： WAR ファイルに web.xml が含まれない場合，SOAP 要求メッセージを Web サービス実装クラスに振り分けません。 意図しないクラスが誤って URL と関連づけられ，ユーザが外部からアクセス可能な状態になることを防ぐため，デフォルトは none(false) に設定されています。			
webserver.container.jaxws.webservice.wsee.no_webxml.enabled	Web サービスのクライアントから SOAP 要求メッセージが到着した際，設定用の WAR ファイルに web.xml が完全に，または部分的に含まれていなくても，この製品の JAX-WS 機能を利用して開発した Web サービス実装クラスが含まれていれば，SOAP 要求メッセージを受け付けて，Web サービス実装クラスに振り分けるかどうか指定します。 strict を指定した場合： 設定用の WAR ファイルに web.xml が含まれない場合，@javax.jws.WebService アノテーションの情報に基づき，SOAP 要求メッセージを Web サービス実装クラスに振り分けます。 lax を指定した場合： 設定用の WAR ファイルに web.xml が完全に，または部分的に含まれない場合，@javax.jws.WebService アノテーションの情報に基づき，SOAP 要求メッセージを Web サービス実装クラスに振り分けます。 none を指定した場合： 設定用の WAR ファイルに web.xml が含まれない場合，SOAP 要求メッセージを Web サービス実装クラスに振り分けません。	lax	08-70	
webserver.container.jaxws.webservice.wsee.warname	EJB-JAR ファイルに含まれる Web サービス実装クラスに対する，設定用の WAR ファイルのパス名を指定します※¹。 設定用 WAR ファイルは，ユーザが作成する場合とデプロイ時に自動生成する場合があります。デプロイ時に自動生成する場合は設定不要のプロパティです。	CosminexusWSEE.war	08-70	
webserver.container.redirect>WelcomeFile	welcome ファイルの転送を，HTTP リダイレクト（HTTP ステータスコード 302 でブラウザがリダイレクトする）するかどうかを示す互換オプションを指定します。	false	11-20-01	

キー名称	内容	省略値	VR	関連情報
	true を指定した場合（互換性のあるオプション）： welcome ファイルの転送を，HTTP リダイレクトします（HTTP ステータスコード 302 でブラウザがリダイレクトします）。 false を設定した場合： welcome ファイルを，HTTP リダイレクトしないで呼び出します。			
webserver.container.server_id.enabled	HTTP レスポンスに，サーバ ID を示す HTTP Cookie を追加するかどうかを指定します。 true を指定した場合： HTTP レスポンスに，サーバ ID を示す HTTP Cookie を追加します。 webserver.container.server_id.value で，サーバ ID を指定してください。 false を設定した場合： HTTP レスポンスに，サーバ ID を示す HTTP Cookie を追加しません。 true または false 以外の文字列を指定した場合，空文字列，または空白文字※2 を指定した場合は，メッセージが出力され，デフォルト値が設定されます。	false	—	
webserver.container.server_id.name	HTTP レスポンスに，サーバ ID を示す HTTP Cookie を追加する場合に，HTTP Cookie の名前を指定します。1～64 文字の文字列で指定します。半角英数字（A～Z，a～z，0～9）またはアンダースコア（_）が使用できます。ただし，次の名前は Web コンテナが使用するため，大文字・小文字の区別なく使用できません。 JSESSIONID 注意事項： webserver.container.server_id.enabled キーで true を指定し，このプロパティで Cookie の名前を指定，またはデフォルトの「ServerID」としている場合，webserver.session.cookie_config.name キーで同じ Cookie 名を指定しているとき，このプロパティの値が優先されます。なお，webserver.session.cookie_config.name キーで指定する Cookie 名にはデフォルトが使用されます。 不正な値を指定した場合は，警告メッセージが出力されて，デフォルト値が設定されます。	ServerID	—	
webserver.container.server_id.value	HTTP レスポンスに，サーバ ID を示す HTTP Cookie を追加する場合に，HTTP Cookie の	ホスト名と J2EE サーバ	—	

キー名称	内容	省略値	VR	関連情報
	値を指定します。1～64 文字の文字列で指定します。半角英数字（A～Z, a～z, 0～9）またはアンダースコア（_）が使用できます。 webserver.container.server_id.enabled に true を指定した場合に、このプロパティを省略したとき、または不正な値、空文字列、もしくは空白文字※2 を指定したときは、メッセージが出力され、デフォルト値が設定されます。	名／Web コンテナサーバ名（互換機能）からハッシュ関数を用いて生成されます。生成された値は、64 文字の 16 進数です。		
webserver.container.servlet.default_mapping.enabled	サーブレットのデフォルトマッピングを有効にするかどうかを指定します。 true を指定した場合： 有効になります。 false を指定した場合： 無効になります。	false	08-00	
webserver.container.thread_control.queue_size	Web アプリケーション単位の同時実行スレッド数の制御機能を使用する場合に、デフォルトの実行待ちキューサイズを 0～2147483647 の整数で指定します。J2EE サーバモードでだけ有効です。	8192	—	「機能解説 基本・開発編 (Web コンテナ)」の「2.15 Web アプリケーション単位の同時実行スレッド数の制御」

(凡例)

—：08-00 より前のバージョンを示します。

注※1

設定用 WAR ファイルをユーザが作成する場合は、このプロパティで指定した EAR ファイル内の相対パスに一致するように、設定用 WAR ファイルを作成します。なお、設定用 WAR ファイルには、EJB-JAR ファイルに含まれる Web サービスへの SOAP 要求メッセージを振り分けるための web.xml を含みます。

EJB-JAR ファイルに含まれる Web サービス実装クラスが存在し、設定用の WAR ファイルが存在しない展開ディレクトリ形式のアプリケーションをデプロイする場合、パス名は 1～255 バイトで指定します。なお、パス名の長さは次の A + B + C の合計です。

A：展開ディレクトリ形式のアプリケーションのアプリケーションディレクトリのパス長

B：パス区切り文字（1 バイト）

C：このプロパティに指定したパスの長さ

このプロパティの値を変更するには、EJB の Web サービスを含む Web アプリケーションを停止してください。Web アプリケーションを開始した状態で、プロパティの値を変更した場合、動作は保証されません。ほかのアプリケーションが不正となり、予期しない例外が発生するおそれがあります。

注※2

空白文字とは、半角スペース、タブ、LF (0x0a)、CR (0x0d) または FF (0x0c) のことを指します。

(am) webserver.context から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
webserver.context.check_interval	Web アプリケーションの更新検知インターバルを 0～2147483647 の範囲（単位：秒）の整数で指定します。 0 を指定した場合、更新の自動検知は行われません。 J2EE アプリケーションのリロード機能が無効の場合、このプロパティは設定できません。設定した場合無視されます。 数値以外の文字列、範囲外の数値または空文字列を指定した場合はメッセージを出力し、デフォルト値が設定されます。	ejbserver.deploy.context.check_interval に設定した値（デフォルト値：0）	—	「機能解説 基本・開発編(コンテナ共通機能)」の「18.8 J2EE アプリケーションの更新検知とリロード」
webserver.context.reload_delay_timeout	Web アプリケーションのリロードの遅延実行機能の最大遅延期間を、負の整数または 0～2147483647（単位：秒）で指定します。 0 を指定した場合は、リロードの遅延実行機能は使用されません。 J2EE アプリケーションのリロード機能が無効な場合、このプロパティは設定できません。設定した場合無視されます。 負の整数を指定した場合は、最大遅延期間を設定しないで、リロードの遅延実行機能が使用されます。	0	—	
webserver.context.stop_async_wait_timeout	アプリケーション停止時に非同期処理の完了を待つ最大時間（単位：秒）を、0～2147483647 の整数で指定します。 0 を指定した場合、0 秒として扱い非同期処理の完了を待ちません。指定した最大時間を経過した場合は、完了していない非同期処理があってもアプリケーション	30	11-00	

キー名称	内容	省略値	VR	関連情報
	ンを停止します。アプリケーション停止後に残っている非同期処理の挙動はアプリケーションの実装に依存します。停止中アプリケーションの非同期処理がすべて完了した場合は、指定した最大時間を経過する前でも、アプリケーションを停止します。			
webserver.context.update.interval	更新するファイルのコピーに必要な時間のインターバルを 0～2147483647 の範囲（単位：秒）の整数で指定します。なお、インターバルは余裕のある値にしてください。リソースの更新を検知してから指定したインターバルのあとに、処理中リクエストの監視が開始されます。 JSP のリロード機能が無効の場合、このプロパティは設定できません。設定した場合無視されます。 数値以外の文字列、範囲外の数値または空文字列を指定した場合はメッセージを出力し、デフォルト値が設定されます。	ejbserver.deploy.context.update.interval（デフォルト値：0）	—	

（凡例）

—：08-00 より前のバージョンを示します。

（an） webserver.dbsfo から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.dbsfo.attribute_data_size.max	グローバルセッション情報に含めることができる HTTP セッションの属性情報の最大サイズをバイト単位で設定します。	1024	08-00
webserver.dbsfo.check_size.mode	HTTP セッションの属性情報のサイズ見積もり機能（データベースセッションフェイルオーバー機能）を有効にするかどうかを指定します。 on を指定した場合： HTTP セッションの属性情報のサイズ見積もり機能が有効となります。 off を指定した場合： HTTP セッションの属性情報のサイズ見積もり機能が無効となります。	off	08-00

キー名称	内容	省略値	VR
webserver.dbsfo.connector.name	データベースセッションフェイルオーバー機能で使用する DB Connector の別名を設定します。	COSMINEX US_SFO_D BCONNECTOR	08-00
webserver.dbsfo.enabled	データベースセッションフェイルオーバー機能を有効にするかどうかを指定します。 true を指定した場合： J2EE サーバ上のすべての Web アプリケーションでデータベースセッションフェイルオーバー機能が有効となります。 false を指定した場合： データベースセッションフェイルオーバー機能が無効となります。	false	08-00
webserver.dbsfo.exception_type_backcompat	データベースセッションフェイルオーバー機能を抑止する対象リクエストのエラーページ内で HttpServletRequest#getSession メソッドを実行した場合に発生する例外の互換オプションを指定します。 true を指定した場合： com.hitachi.software.web.dbsfo.DatabaseAccessException 例外をスローします。 false を指定した場合： com.hitachi.software.web.dbsfo.SessionOperationException 例外をスローします。	false	08-70
webserver.dbsfo.exclude.extensions	データベースセッションフェイルオーバー機能を抑止する拡張子を指定します。 URL のパスが指定した拡張子と合致した場合、該当するリクエストでデータベースセッションフェイルオーバー機能は無効となります。指定した拡張子と URL の比較では大文字、小文字の区別はしません。 また、データベースセッションフェイルオーバー機能を抑止する拡張子を設定しない場合、空文字列、またはコンマ「,」を指定します。	txt,html,html .jpg,gif,js	08-00
webserver.dbsfo.exclude.uris	データベースセッションフェイルオーバー機能を抑止する URI を指定します。リクエスト URI のパスが指定した URI と一致した場合、該当するリクエストはデータベースセッションフェイルオーバー機能が無効となります。 指定した URI とリクエスト URI の比較では大文字、小文字が区別されます。 データベースセッションフェイルオーバー機能を抑止する URI を設定しない場合、空文字列を設定します。	なし	08-50
webserver.dbsfo.integrity_mode.enabled	データベースセッションフェイルオーバー機能の完全性保障モードを有効にするかどうか指定します。 true を指定した場合： 完全性保障モードを有効にします。 false を指定した場合： 完全性保障モードを無効にします。	false	08-70

キー名称	内容	省略値	VR
webserver.dbsfo.negotiation.high_level	データベースセッションフェイルオーバー機能を使用する場合、Web アプリケーション開始で実施するネゴシエーションが失敗したときに、Web アプリケーションの開始処理を続行するか中止するかを指定します。 true を指定した場合： Web アプリケーションの開始処理を中止します。 false を指定した場合： Web アプリケーションの開始処理を続行します。	false	08-70
webserver.dbsfo.session_read_only.uris	データベースセッションフェイルオーバー機能の参照専用リクエストとする URI を指定します。指定できる URI を次に示します。 - スラッシュ (/) から始まる文字列 - 正規化した URI 512 文字までの URI 複数の URI を指定する場合はセミコロン (;) で区切ります。参照専用リクエストを設定しない場合は、空文字列を指定します。 指定した URI とリクエスト URI が合致した場合、このリクエストは参照専用リクエストになります。指定した URI とリクエスト URI の比較では大文字と小文字を区別します。	なし	08-70
webserver.dbsfo.thread_control_queue.enabled	Web アプリケーション単位の同時実行スレッド数制御機能の有効時にデータベースセッションフェイルオーバー機能を使用する場合に、実行待ちキューの空きが不足したときの動作を指定します。 true を指定した場合： 実行待ちキューの空き不足時にクライアントに 503 エラーを返します。 false を指定した場合： 実行待ちキューの空き不足時にクライアントに 503 エラーを返しません。	false	08-70

(ao) webserver.errorpage から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.errorpage.stack_trace.enabled	例外発生時、デフォルトのエラーページにスタックトレースを出力するかどうかを指定します。 true を指定した場合： スタックトレースを出力します。 false を指定した場合： スタックトレースを出力しません。	false	—

キー名称	内容	省略値	VR
	true または false 以外の文字列を指定した場合、空文字列または空白文字※を指定した場合は、メッセージが出力され、デフォルト値が設定されます。		

(凡例)

－：08-00 より前のバージョンを示します。

注※

空白文字とは、半角スペース、タブ、LF (0x0a)、CR (0x0d) または FF (0x0c) のことを指します。

(ap) webserver.http から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.http.request.encoding	リクエストボディおよびクエリのデコードに使用する文字エンコーディングを指定します。 デフォルトの文字エンコーディングとして設定できる文字は、JavaVM がサポートしている文字エンコーディングとなります。JavaVM がサポートしている文字エンコーディングについては、JDK のドキュメントのサポートされているエンコーディングに関する説明を参照してください。また、指定できる文字列は、java.nio API 用の正準名と java.lang API 用の正準名に記載されている文字エンコーディング、およびそれらの別名になります。	なし	－
webserver.http.request.uri_decode.enabled	サブレットパス、追加のパス情報をデコードするかどうかを指定します。 true を指定した場合： デコードします。 false を指定した場合： デコードしません。	false	08-00
webserver.http.response.encoding	レスポンスボディのエンコードに使用する文字エンコーディングを指定します。 デフォルトの文字エンコーディングとして設定できる文字は、JavaVM がサポートしている文字エンコーディングとなります。JavaVM がサポートしている文字エンコーディングについては、JDK のドキュメントのサポートされているエンコーディングに関する説明を参照してください。また、指定できる文字列は、java.nio API 用の正準名と java.lang API 用の正準名に記載されている文字エンコーディング、およびそれらの別名になります。	なし	－

(凡例)

－：08-00 より前のバージョンを示します。

(aq) webserver.http2 から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.http2.enabled	HTTP/2 を有効にするかどうかを指定します。 true を指定した場合：有効になります。 false を指定した場合：無効になります。	false	11-20
webserver.http2.initialWindowSize	Web コンテナがクライアントに伝えるストリームの WindowSize の初期値（単位：バイト）を指定します。値は、1～2147483647 の整数を指定します。	65535	11-20
webserver.http2.keepAliveTimeout	HTTP/2 のアクティブなストリームがない場合に、Web コンテナが HTTP/2 のフレームを待機してから、接続を閉じる時間（単位：秒）を指定します。値は 0～3600 の整数で指定します。0 を指定した場合はタイムアウトしません。	20	11-20
webserver.http2.maxConcurrentStreamExecution	Web コンテナのスレッドプールから、スレッドを割り当てることができるストリームの最大数を指定します。使用可能なスレッドよりも多くのストリームがアクティブになった場合、ストリームは使用可能になるまで待機します。値は、1～2147483647 の整数を指定します。	20	11-20
webserver.http2.maxConcurrentStreams	1 つの接続で許可されるストリームの最大数を指定します。クライアントがこの値よりも多くのストリームを開こうとすると、STREAM_REFUSED エラーでストリームがリセットされます。値は、1～2147483647 の整数を指定します。	100	11-20
webserver.http2.max_headers	HTTP/2 リクエストに含まれる HTTP ヘッダの個数の上限を 0～32767 の整数で指定します。指定値 0 は個数 0 を表します。 このプロパティで設定した HTTP ヘッダの個数に満たない場合であっても、webserver.http2.max_request_header プロパティで指定したサイズを超えた場合はエラーとなります。 webserver.http2.max_trailers とは別カウントになります。	100	11-20
webserver.http2.max_request_header	HTTP/2 リクエストのリクエストヘッダの最大サイズ（単位：バイト）を、7～65536 の整数で指定します。 このプロパティで設定したリクエストヘッダの最大サイズに満たない場合であっても、webserver.http2.max_headers プロパティで指定した HTTP Trailer の個数を超えた場合はエラーとなります。 また、HTTP ヘッダの終わりを示す改行文字（CR(0x0d)+LF(0x0A) の 2 バイト）も指定するサイズに含める必要があります。 webserver.http2.max_request_trailer とは別カウントになります。	16384	11-20
webserver.http2.max_request_trailer	HTTP/2 リクエストの Trailer の最大サイズ（単位：バイト）を、1～65536 の整数で指定します。 このプロパティで設定した Trailer の最大サイズに満たない場合であっても、webserver.http2.max_trailers プロパティで指定した HTTP Trailer の個数を超えた場合はエラーとなります。	16384	11-20

キー名称	内容	省略値	VR
	また、HTTP ヘッダの終わりを示す改行文字 (CR(0x0d)+LF(0x0A) の 2 バイト) も指定するサイズに含める必要があります。 webserver.http2.max_request_header とは別カウントになります。		
webserver.http2.maxtrailers	HTTP/2 リクエストに含まれる HTTP トレイラの個数の上限を 0～32767 の整数で指定します。指定値 0 は個数 0 を表します。 このプロパティで設定した HTTP トレイラの個数に満たない場合であっても、webserver.http2.max_request_header プロパティで指定したサイズを超えた場合はエラーとなります。 webserver.http2.max_headers とは別カウントになります。	100	11-20
webserver.http2.readTimeout	HTTP/2 のフレームが受信された場合の、Web コンテナが追加のデータを待機する時間 (単位: 秒) を指定します。値は 0～3600 の整数で指定します。0 を指定した場合はタイムアウトしません。	5	11-20
webserver.http2.streamReadTimeout	アプリケーションがブロッキング I/O 読み取りを実行していて、追加のデータが必要な場合に、Web コンテナがストリームに追加の DATA フレームの到着を待機する時間 (単位: 秒) を指定します。値は 0～3600 の整数で、0 を指定した場合タイムアウトしません。	20	11-20
webserver.http2.streamWriteTimeout	アプリケーションがブロッキング I/O 書き込みを実行していて、ストリームやフロー制御ウィンドウが小さすぎて書き込みを完了できない場合に、Web コンテナが、追加の WINDOW_UPDATE フレームの到着を待機する時間 (単位: 秒) を指定します。値は 0～3600 の整数で指定します。0 を指定した場合タイムアウトしません。	20	11-20
webserver.http2.writeTimeout	HTTP/2 のフレームが送信された場合の、Web コンテナが追加のデータを待機する時間 (単位: 秒) を指定します。値は 0～3600 の整数で指定します。0 を指定した場合はタイムアウトしません。	5	11-20

(ar) webserver.jsp から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
webserver.jsp.additional.import.list	JSP コンパイル時に暗黙にインポートしたいクラス名 (完全修飾名のクラス名または、「パッケージ名.*」) を指定します。複数のクラス名を指定する場合は、クラス名とクラス名の間をコンマ (,) で区切ります。存在しないクラス名や、クラスパスの通っていないクラス名など不正な値を指定した場合、メッ	なし	08-70	

キー名称	内容	省略値	VR	関連情報
	セージを出力し、コンパイルエラーになります。			
webserver.jsp.check_interval	JSP の更新検知インターバルを 0～2147483647 の範囲（単位：秒）の整数で指定します。 0 を指定した場合、更新の自動検知は行われません。 JSP のリロード機能が無効の場合、このプロパティは設定できません。設定した場合無視されます。 数値以外の文字列、範囲外の数値または空文字列を指定した場合はメッセージを出力し、デフォルト値が設定されます。	ejbserver.deploy.context.check_interval に設定した値（デフォルト値：0）	—	「機能解説 基本・開発編(コンテナ共通機能)」の「18.8 J2EE アプリケーションの更新検知とリロード」
webserver.jsp.compile.bac kcompat	JSP から生成されたサーブレットの java ソースのコンパイル時に、Java コンパイラの Java 言語仕様のバージョンを指定します。このプロパティは、JSP ファイル内のスクリプティングで Java SE 11 以前の言語仕様に依存したコーディングをしており、JSP から生成されたサーブレットのソースファイルがコンパイルできない場合に指定します。JSP から生成されたサーブレットのソースファイルのコンパイルが正常にできている場合、このプロパティの指定は不要です。 指定できる値を次に示します。 11（JDK17 以降を使用する場合）： Java SE 11 の言語仕様に依存したソースコードを JSP ファイルのスクリプティングに記述している場合に指定します。 1.8 または 8： Java SE 8 の言語仕様に依存したソースコードを、JSP ファイルのスクリプティングに記述している場合に指定します。 1.7 または 7： Java SE 7 の言語仕様に依存したソースコードを JSP ファイルのスクリプティングに記述している場合に指定します。	false	—	

キー名称	内容	省略値	VR	関連情報
	1.6 または 6 (JDK11 を使用する場合) : Java SE 6 の言語仕様に依存したソースコードを JSP ファイルのスク립ティングに記述している場合に指定します。 false : JSP ファイル内にスク립ティングを記述していない場合、スク립ティングを記述している場合でも Java SE 7 以前のバージョンに依存しない場合に指定します。 Java SE 8 の Java 言語仕様に従って Java ソースファイルをコンパイルします。 指定可能なパラメタ値以外の文字列、空文字列または空白文字※を指定した場合は、メッセージが出力され、デフォルト値が設定されます。			
webserver.jsp.precompile.jsp_work_dir	JSP 事前コンパイル機能によって生成された JSP ファイルのコンパイル結果の格納先ディレクトリ名を指定します。ディレクトリ名は、半角の英数字 (A~Z, a~z, 0~9) またはアンダースコア (_) で構成される文字列で指定します。 不正な値を指定した場合は、メッセージが出力され、デフォルト値が設定されます。また、予約語である次の文字列を指定した場合は、メッセージが出力され、デフォルト値を使用します。 • classes • lib • tags	cosminexus_jsp_work	—	
webserver.jsp.keepgenerated	JSP ファイル、またはタグファイルのコンパイルで生成された java ファイルを保持するかどうかを指定します。 次の値を指定します。 true を指定した場合 : java ファイルを保持します。 false を指定した場合 : java ファイルを保持しません。	false	—	
webserver.jsp.pageEncoding	JSP の文字エンコーディングを指定します。デフォルトの文字エンコーディ	なし	—	

キー名称	内容	省略値	VR	関連情報
	ングとして設定できる文字は、JavaVM がサポートしている文字エンコーディングとなります。JavaVM がサポートしている文字エンコーディングについては、JDK のドキュメントのサポートされているエンコーディングに関する説明を参照してください。また、指定できる文字列は、java.nio API 用の正準名と java.lang API 用の正準名に記載されている文字エンコーディング、およびそれらの別名になります。			
webserver.jsp.translation.backcompat.customAction.declareVariable	複数のカスタムタグで、スクリプト変数名とスクリプト変数のスコープが重複する場合、JSP ファイルから生成された Java コードで 2 回目のカスタムタグに対応するスクリプト変数の変数宣言を出力するかどうかを指定します。 true : 2 回目以降のスクリプト変数を宣言します。 false : 2 回目以降のスクリプト変数を宣言しません。(*)スクリプト変数のスコープは、javax.servlet.jsp.tagext.TagExtraInfo クラスのサブクラス、または TLD ファイルの variable 要素内の scope 要素で指定できます。	false	—	
webserver.jsp.translation.backcompat.tag.noCheckRtexprvalue	Expression が指定できないタグの属性値に Expression が指定されているかどうか検証するかどうかを指定します。 true : Expression が指定されているかどうかを検証しません。 false : Expression が指定されているかどうかを検証します。	false	—	
webserver.jsp.translation.backcompat.tag.rtexprvalueTerminate	タグの属性値が、「<%=」または「<%=」で開始しており、「%>」(「<%=」で開始した場合は「%>」)で終了していない属性値の「」(または「」)で囲まれた値を文字列として扱うかどうかを指定します。	false	—	

キー名称	内容	省略値	VR	関連情報
	true : 「"」までを文字列として扱います。 false : 「%>"」までを属性値とします。			
webserver.jsp.translation.backcompat.taglib.noCheckPrefix	taglib ディレクティブの前に、taglib ディレクティブで指定した prefix を使用したカスタムタグを記述しているかチェックするかどうかを指定します。 true : チェックしない。 false : チェックする。	false	—	
webserver.jsp.translation.backcompat.useBean.noCheckClass	JSP トランスレーション時に <jsp:useBean>タグのクラス属性値のチェック処理を実行するかどうかを指定します。 true : クラス属性値をチェックしません。 false : クラス属性値をチェックします。	false	—	
webserver.jsp.update.interval	JSP のリロード監視対象ファイルのコピーに必要な時間のインターバルを 0～2147483647 の範囲（単位：秒）の整数で指定します。なお、インターバルは余裕のある値にしてください。JSP ファイルの更新を検知してから指定したインターバルのあとに、JSP のリロードが実行されます。 JSP のリロード機能が無効の場合、このプロパティは設定できません。設定した場合無視されます。 数値以外の文字列、範囲外の数値または空文字列を指定した場合はメッセージを出力し、デフォルト値が設定されます。	ejbserver.deploy.context.update.interval に設定した値（デフォルト値：0）	—	
webserver.jsp.translation.customAction.ignoreCaseAttributeName	カスタムタグで指定された属性が、TLD ファイルもしくはタグファイルで定義されているかどうかをチェックする際に、大文字小文字を区別するかどうかを指定します。 true を指定した場合： 大文字小文字を区別しません。	false	08-00	

キー名称	内容	省略値	VR	関連情報
	false を指定した場合： 大文字小文字を区別します。			
webserver.jsp.translation. useBean.noCheckDuplicateId	<jsp:useBean>タグの id 属性値が重複しているかどうかをチェックします。 true の場合、id 属性値が重複してもトランスレーションエラーは発生しません。false の場合、id 属性値が重複している時にトランスレーションエラーが発生します。	false	08-00	
webserver.jsp.tld.mapping. java_ee_tag_library.enabled	Servlet2.5 仕様以降の Web アプリケーションで、Java EE 仕様に含まれるタグライブラリ (JSTL, JSF) の TLD と URL のマッピングを自動的に実行するかどうかを指定します。 true を指定した場合： 自動的にマッピングします。 false を指定した場合： 自動的にマッピングしません。 また、Servlet2.5 仕様以降の Web アプリケーションでアプリケーションサーバが提供するタグライブラリ (JSTL, JSF) 以外を使用する場合は、false を指定してください。	true	08-00	
webserver.jsp.tld.validation	TLD バージョン 3.0 以降のスキーマ宣言を持つ TLD ファイルを解析する時に、スキーマ検証をするかどうかを指定します。 true を指定した場合： スキーマ検証をします。 false を指定した場合： スキーマ検証をしません。 TLD バージョン 2.1 以前のスキーマ宣言を持つ TLD ファイルには、このプロパティの指定は無効となり、このプロパティの指定内容に関係なく常にスキーマ検証をします。	false	11-50	
webserver.jsp.jsp_page.bom.enabled	Servlet2.5 仕様以降の Web アプリケーションの JSP ページで、BOM による文字エンコーディングの指定を有効にするかどうかを指定します。 true を指定した場合： BOM による文字エンコーディングの指定が有効になります。	true	08-00	

キー名称	内容	省略値	VR	関連情報
	false を指定した場合： BOM による文字エンコーディングの指定が無効になります。			
webserver.jsp.debugging.enabled	JSP デバッグ機能を有効にするかどうかを指定します。なお、このプロパティは J2EE サーバモードにだけ有効です。 true を指定した場合： JSP デバッグ機能が有効となり JSP ファイルから作成されたサープレットの class ファイルに SMAP を埋め込みます。 false を指定した場合： JSP デバッグ機能が無効になります。	false	08-00	

(凡例)

ー：08-00 より前のバージョンを示します。

注※

空白文字とは、半角スペース、タブ、LF (0x0a)、CR (0x0d) または FF (0x0c) のことを指します。

(as) webserver.prf から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.prf.output_httpheader	性能解析トレースのイベント ID 0x8236 のオペレーション名に出力する値の HTTP ヘッダ名を、文字列で指定します。指定しない場合、空文字列を指定した場合、および指定したヘッダ名に該当するヘッダが含まれなかった場合、イベント ID 0x8236 のオペレーション名には何も出力しません。	なし	11-00

(at) webserver.servlet から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.servlet_api.exception.getCause.backcompat	javax.servlet.ServletException のコンストラクタ ServletException (String, Throwable) および ServletException (Throwable) で指	false	08-00

キー名称	内容	省略値	VR
	定した根本原因の例外を <code>getCause()</code> により取得できないようにします。 同様に <code>javax.servlet.jsp.JspException</code> のコンストラクタ <code>JspException (String, Throwable)</code> および <code>JspException (Throwable)</code> で指定した根本原因の例外を <code>getCause()</code> により取得できないようにします。 true を指定した場合： 根本原因の例外を取得できません。 false を指定した場合： 根本原因の例外を取得できます。		
<code>webserver.servlet_api.unsupported.throwUnsupportedOperationException</code>	非サポートの API を呼び出したとき、<code>UnsupportedOperationException</code> をスローするかどうかを指定します。 true の場合 <code>UnsupportedOperationException</code> 例外をスローします。 false の場合 API ごとに以下の値を返します。 <code>javax.servlet.ServletContext#getJspConfigDescriptor</code> メソッド：null <code>javax.servlet.ServletContext#getVirtualServerName</code> メソッド：空文字 <code>javax.servlet.http.HttpServletRequest#changeSessionId</code> メソッド：現在の HTTP セッションのセッション ID, HTTP セッション未作成の場合は <code>IllegalStateException</code> <code>javax.servlet.http.HttpServletRequestWrapper#changeSessionId</code> メソッド： <code>javax.servlet.http.HttpServletRequest</code> の同名メソッドと同じです。	false	11-00

(au) webserver.ServletContainerInitializer_jar から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
<code>webserver.ServletContainerInitializer_jar.include.path</code>	<code>ServletContainerInitializer</code> の実装クラスを含む JAR ファイルのパスを絶対パスで指定します。複数のパスを指定する場合は、コンマ (,) で区切ります。ディレクトリの区切り文字はスラッシュ (/) を使用します。	なし	09-00

キー名称	内容	省略値	VR
	指定した JAR ファイルのパスは、usrconf.cfg (J2EE サーバ用オプション定義ファイル) の add.class.path に指定する必要があります。 注意事項： アプリケーションに含まれる WAR ファイル内の WEB-INF/lib ディレクトリにある JAR ファイルのパスを指定しないでください。WEB-INF/lib ディレクトリにある JAR ファイルを指定した場合の動作は保証しません。		

(av) webserver.session から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	省略値	VR	関連情報
webserver.session.cookie_config.http_only	Web アプリケーションで作成される HTTP セッションのセッション ID を示す HTTP Cookie に HttpOnly 属性を付けるかどうかを指定します。 true を指定した場合： Web アプリケーションで作成される HTTP セッションのセッション ID を示す HTTP Cookie に HttpOnly 属性を付けます。 false を指定した場合： Web アプリケーションで作成される HTTP セッションのセッション ID を示す HTTP Cookie に HttpOnly 属性を付けません。	false	09-50	
webserver.session.cookie_config.name	Web アプリケーションで作成される HTTP セッションのセッション ID を示す HTTP Cookie の名称、または URL のパスパラメタ名を指定します。 注意事項： webserver.container.server_id.enabled キーで true を指定している場合、 webserver.container.server_id.name キーと異なる値を指定してください。 webserver.container.server_id.name キーと同じ値を指定すると、	HTTP Cookie の名称の場合 JSESSIONID URL のパスパラメタ名の場合 jsessionid	09-50	

キー名称	内容	省略値	VR	関連情報
	webserver.container.server_id.name キーでの値が優先されます。 このキーで指定する Cookie 名にはデフォルトが使用され、ワーニングメッセージ (KDJE39008-W) が出力されます。			
webserver.session.delete_cookie.backcompat	HTTP セッションを無効にした場合、Web クライアントが保持するセッション ID を示す HTTP Cookie を削除するかどうかを設定します。 true を指定した場合： HTTP セッションを無効化した場合、Web クライアントが保持するセッション ID を示す HTTP Cookie を削除しません。 false を指定した場合： HTTP セッションを無効化した場合、HTTP レスポンスのヘッダに Web クライアントが保持するセッション ID を示す HTTP Cookie を削除するための HTTP Cookie を付加します。	false	08-00	
webserver.session.max.logging_interval	次に示すメッセージのインターバルを、0～2147483647 の整数（単位：秒）で指定します。 - KDJE34380-E データベースセッションフェイルオーバー機能使用時にデータベースのレコード数を超過して HttpSession オブジェクトを生成しようとした場合に出力されます。 - KDJE39225-E HttpSession 数の上限値を超過して HttpSession オブジェクトを生成しようとした場合に出力されます。 メッセージ KDJE39225-E が出力されてからこのプロパティで指定したインターバルの間、メッセージ KDJE39225-E の出力を抑止します。 0 を指定した場合、メッセージ KDJE39225-E の出力を抑止しません。 なお、この設定は Web アプリケーション単位に適用されます。 数値以外の文字列や範囲外の数値を指定した場合は、警告メッセージが出力され、デフォルト値が設定されます。	60	—	

キー名称	内容	省略値	VR	関連情報
webserver.session.max.threadHttpSessionLimitExceededException	HTTP セッションオブジェクト数が上限値を超えた時に、 com.hitachi.software.web.session.HttpSessionLimitExceededException 例外をスローするかどうかを指定します。 true を指定した場合： java.lang.IllegalStateException 例外の派生クラスである com.hitachi.software.web.session.HttpSessionLimitExceededException 例外をスローします。 false を指定した場合： java.lang.IllegalStateException 例外をスローします。	false	08-00	
webserver.session.server_id.enabled	セッション ID にサーバ ID を付加するかどうかを指定します。 true を指定した場合： セッション ID にサーバ ID を付加します。 false を指定した場合： セッション ID にサーバ ID を付加しません。 通常は true を指定し、 webserver.session.server_id.value で、サーバ ID を指定してください。 true または false 以外の文字列を指定した場合、空文字列または空白文字※を指定した場合は、メッセージが出力され、デフォルト値が設定されます。	true	—	「機能解説 基本・開発編(Web コンテナ)」の「2.7 セッション管理機能」
webserver.session.server_id.value	セッション ID に付加するサーバ ID を、1～64 文字の文字列で指定します。半角英数字 (A～Z, a～z, 0～9) またはアンダースコア (_) が使用できます。 webserver.session.server_id.enabled に true を指定し、このプロパティを省略した場合、またはプロパティに不正な値、空文字列、もしくは空白文字※を指定した場合、メッセージが出力され、デフォルト値が設定されます。	ホスト名と J2EE サーバ名／Web コンテナサーバ名 (互換機能) からハッシュ関数を用いて生成されます。生成される値は、64 文字の 16 進数です。	—	「機能解説 基本・開発編(Web コンテナ)」の「2.7 セッション管理機能」
webserver.session.tracking_mode	Web アプリケーションで作成される HTTP セッションの管理方法を指定します。COOKIE, URL, または両方を指定できます。両方のセッション管理	COOKIE, URL	09-50	

キー名称	内容	省略値	VR	関連情報
	方法を指定する場合は COOKIE と URL の間をコンマ (,) で区切ります。なお、COOKIE または URL の前後の半角スペースは無視されます。 COOKIE だけを指定した場合 HTTP Cookie によるセッション管理だけが有効となります。このとき、URL 書き換えにより生成される文字列にセッション ID を示す URL パスパラメタを含みません。 URL だけを指定した場合 URL 書き換えによるセッション管理だけが有効となります。このとき、レスポンスにセッション ID を示す HTTP Cookie の情報を含みません。 COOKIE と URL の両方を指定した場合： HTTP Cookie によるセッション管理と URL 書き換えによるセッション管理が有効となります。			

(凡例)

ー：08-00 より前のバージョンを示します。

注※

空白文字とは、半角スペース、タブ、LF (0x0a)、CR (0x0d) または FF (0x0c) のことを指します。

(aw) webserver.sessionManager から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.sessionManager.<コンテキストルート名>.className	セッションマネージャの指定機能で使用する、ライブラリのセッションマネージャのクラス名を指定します。<コンテキストルート名>には、セッションマネージャの指定機能を使用する war のコンテキストルートを指定します。サーバのすべての war のセッションマネージャを指定する場合は、*を指定します。	なし	11-10
webserver.sessionManager.<コンテキストルート名>.maxActiveSessions	セッションマネージャの指定機能で使用する、HttpSession オブジェクト数の上限を-1～2147483647の整数で指定します。<コンテキストルート名>には、セッションマネージャの指定機能を使用する war のコンテキストルートを指定します。サーバのすべての war に HttpSession オブジェクト数で同じ上限値を指定する場合は、*を指定します。	-1	11-10

キー名称	内容	省略値	VR
webserver.sessionManager.<コンテキストルート名>.properties.<プロパティ>	セッションマネージャの指定機能で使用する、ライブラリのセッションマネージャに設定するプロパティを指定します。<コンテキストルート名>には、セッションマネージャの指定機能を使用する war のコンテキストルート指定します。	なし	11-10
webserver.sessionManager.backgroundProcessor.threads	セッションマネージャの指定機能で、指定したセッションマネージャのバックグラウンド処理を実行するためのスレッド数を、サーバ単位で指定します。	0	11-10
webserver.sessionManager.backgroundProcessor.delay	セッションマネージャの指定機能で、指定したセッションマネージャのバックグラウンド処理の実行間隔を、サーバ単位で指定します。 バックグラウンド処理は、この値を 6 倍した間隔（単位：秒）で実行されます。例えば 10 を指定した場合、60 秒間隔でバックグラウンド処理が実行されます。	10	11-10

(ax) webserver.static から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.static_content.cache.enabled	静的コンテンツキャッシュ機能の有効／無効／強制的な無効を指定します。 true を指定した場合： 静的コンテンツキャッシュ機能を有効にします。 false を指定した場合： 静的コンテンツキャッシュ機能を無効にします。 forceoff を指定した場合： DD ファイル（web.xml）またはアプリケーション属性ファイルで静的コンテンツキャッシュが有効と設定されていても、キャッシュ機能を強制的に無効にします。 このプロパティが false および forceoff の場合、webserver.static_content.cache.size、webserver.static_content.cache.filesize.threshold の設定は無効となります。 DD ファイル（web.xml）またはアプリケーション属性ファイルで、静的コンテンツキャッシュ機能の有効／無効が設定されていない Web アプリケーションでは、このプロパティに指定した値が有効になります。また、静的コンテンツキャッシュ機能の有効／無効が設定されている Web アプリケーションではこのプロパティに指定した値は無効となります。ただし、forceoff を指定した場合は、DD ファイル（web.xml）またはアプリケーション属性ファイルの設定に関係なく、このプロパティに指定した値が有効になります。 true, false, forceoff 以外の文字列、空文字列または空白文字※を指定した場合は、メッセージが出力され、デフォルト値が設定されます。	false	—

キー名称	内容	省略値	VR
webserver.static_content.cache.size	静的コンテンツキャッシュ機能が有効な場合、メモリにキャッシュできるサイズ（単位：バイト）の上限を 0～2147483647 の整数で指定します。 Web アプリケーション単位で、キャッシュの合計サイズが指定した値を超えた場合は、アクセスされていない時間が最も長いキャッシュから削除していき、キャッシュの合計サイズが設定した値以下になるまでキャッシュの削除が繰り返されます。 0 を指定した場合、キャッシュできるサイズに上限を設けません。 DD（web.xml）またはアプリケーション属性ファイルで、キャッシュできるメモリサイズが設定されていない Web アプリケーションでは、このプロパティに指定した値が有効となります。また、キャッシュできるメモリサイズが設定されている Web アプリケーションでは、このプロパティに指定した値は無効となります。 無効な値、webserver.static_content.cache.filesize.threshold で指定した値よりも小さい値、空文字列または空白文字※を指定した場合は、メッセージが出力され、デフォルト値が設定されます。	10485760	—
webserver.static_content.cache.filesize.threshold	静的コンテンツキャッシュ機能が有効な場合、キャッシュできるファイルサイズ（単位：バイト）を 0～2147483647 の整数で指定します。指定した値を超えるサイズのファイルはキャッシュされません。 0 を指定した場合、キャッシュできるファイルのサイズに上限を設けません。 DD（web.xml）またはアプリケーション属性ファイルで、キャッシュできるファイルサイズが設定されていない Web アプリケーションでは、このプロパティに指定した値が有効になります。また、キャッシュできるファイルサイズが設定されている Web アプリケーションでは、このプロパティに指定した値は無効となります。 無効な値、webserver.static_content.cache.size で指定した値より大きい値、空文字列または空白文字※を設定した場合は、メッセージが出力され、デフォルト値が設定されます。	524288	—
webserver.static_content.encoding.extension	静的コンテンツに使用する文字エンコーディングを適用させる拡張子を指定します。webserver.http.response.encoding キーが指定されている場合に有効になります。 キーの値に何も設定しない場合、このキーは無効になります。	なし	—

（凡例）

—：08-00 より前のバージョンを示します。

注※

空白文字とは、半角スペース、タブ、LF（0x0a）、CR（0x0d）または FF（0x0c）のことを指します。

（ay） webserver.work から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.work.clean	JSP 用テンポラリディレクトリ下に生成したディレクトリをサーバ起動時、および終了時に削除するかどうかを指定します。 true が指定された場合： ディレクトリを削除します。 false が指定された場合： ディレクトリを削除しません。	false	—
webserver.work.directory	JSP 用テンポラリディレクトリを指定します。絶対パスによるディレクトリ名を指定します。※ ディレクトリのパスの区切り記号には、"/"を使用してください。 例えば、Windows の場合、work ディレクトリとして C:*temp*work を設定するときは、次のように指定します。 (指定例) webserver.work.directory=c:/temp/work	• Windows の場合 <Application Server のインストールディレクトリ>*CC*server*repository*<サーバ名称>*web • UNIX の場合 /opt/Cosminexus/CC/server/repository/<サーバ名称>/web	—

(凡例)

—：08-00 より前のバージョンを示します。

注※

使用する OS によってファイル名長に制限があります。例えば、Windows では 250 バイト程度となっています。このため、ディレクトリ階層の深い場所でサブレットや JSP を実行できません。特に JSP ファイルから生成されるファイル名は Web アプリケーションで配置されたディレクトリ階層の深さに比例して長くなり、JSP ファイルに対する Java ファイルを生成できないことがあります。これを回避するためには Web アプリケーションの配置ディレクトリの階層を浅くしたり、プロパティ webserver.work.directory に絶対パスで浅いディレクトリ階層を指定したりすることで回避してください。Application Server の作業ディレクトリについては、マニュアル「アプリケーションサーバ システム構築・運用ガイド」の「付録 C.1 J2EE サーバの作業ディレクトリ」を参照してください。

(az) webserver.xml から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
webserver.xml.validate	Servlet 2.3 の Web アプリケーションについて、タグライブラリ・ディスクリプタ（TLD ファイル）が、DTD で記述されたスキーマに沿って記述されているかチェックするかどうかを指定します。 J2EE サーバモードで起動した場合の web.xml、および Servlet 2.4 以降の Web アプリケーションの TLD ファイルには、このプロパティの指定は無効となります。また、Servlet 2.4 以降の Web アプリケーションに含まれる TLD ファイルは TLD のバージョンにかかわらずすべてチェックされます。 次の値を指定します。 true を指定した場合： タグライブラリ・ディスクリプタだけをチェックします。 false を指定した場合： チェックしません。	true	—

(凡例)

—：08-00 より前のバージョンを示します。

(ba) mail.mime から始まるキー

指定できるキーについて次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
mail.mime.charset	JavaMail で使用するデフォルトの文字セットを指定します。指定がない場合、J2SE の file.encoding プロパティが使用されます。このプロパティによってメッセージを送信する際、ファイルで使用されるデフォルトの文字セットと異なる文字セットをデフォルトとして使用できます。	なし	—
mail.mime.encodeeol.strict	データが次の条件をすべて満たす場合に、改行コードを CRLF へ変換するかどうかを選択できます。false が設定された場合、変換を行います。 • ASCII で表示可能な文字しか含まない。 • CR, LF を含む。	false	—
mail.mime.decodetext.strict	エンコードされたテキストの開始に区切り文字を含まない RFC2047 に違反した不正なメールヘッダに対してデコードを行うかどうかを切り替えます。RFC2047 に違反したエンコードの不正とは、次のものを指します。 • プロパティに true を設定した場合、デコードを行わない。 • プロパティに false を設定した場合、RFC に違反したメールヘッダであってもデコードする。	true	—
mail.mime.encodefilename	標準仕様書を参照してください。	false	09-50

キー名称	内容	省略値	VR
mail.mime.decodefilename	標準仕様書を参照してください。	false	09-50
mail.mime.multipart.ignoremissingendbound	標準仕様書を参照してください。	true	09-50
mail.mime.encodeemail.ignoremissingboundaryparameter	標準仕様書を参照してください。	true	09-50
mail.mime.encodeparameters	標準仕様書を参照してください。	false	09-50
mail.mime.decodeparameters	標準仕様書を参照してください。	false	09-50

(凡例)

－：08-00 より前のバージョンを示します。

(bb) manager.j2ee から始まるキー

この内容は Management Server が J2EE サーバに自動で設定します。なお「省略値」とは、キーの指定がない場合に仮定される値です。

内部的に利用するキーのため、何も指定されていない場合に指定したり設定を変更したりしないでください。

「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	省略値	VR
manager.j2ee.compat	Management Server との連携に必要な設定をします。なお、この内容は Management Server が自動で設定するため、手動での編集はできません。Management Server が自動で設定する内容については、マニュアル「アプリケーションサーバ システム構築・運用ガイド」の「付録 E.4 Management Server が J2EE サーバに自動で設定する内容」を参照してください。	none	11-00

(6) 記述例

```
ejbserver.deploy.stub.generation.scope=ejb
ejbserver.compiler.jvm.maxHeapSize=256m
ejbserver.compiler.jvm.minHeapSize=32m

ejbserver.naming.host=localhost
ejbserver.naming.port=900
ejbserver.naming.protocol=corbaname
ejbserver.naming.startupMode=inprocess
ejbserver.naming.startupWaitTime=1
ejbserver.naming.startupRetryCount=9

ejbserver.logger.enabled.*=Error,Warning,Information,Debug
ejbserver.logger.channels.define.MessageLogFile.filenum=2
```

```
ejbserver.logger.channels.define.MessageLogFile.filesize=1048576

ejbserver.application.InitTermProcessClasses=InitTermClass1,InitTermClass2

webserver.connector.http.permitted.hosts=host1,host2
webserver.work.directory=c:/work
```

2.2.4 server.policy (J2EE サーバ用セキュリティポリシーファイル)

(1) 形式

J2SE のセキュリティポリシーファイル形式に従います。

(2) ファイルの格納先

- Windows の場合
 <Application Server のインストールディレクトリ>%CC%server%usrconf%ejb%<サーバ名称>%
- UNIX の場合
 /opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/

(3) 機能

J2EE サーバを実行する JavaVM のセキュリティポリシーを指定します。

J2EE サーバの稼働中に、このファイルの内容を変更した場合、変更した内容は次に J2EE サーバを起動したときに反映されます。

(4) 記述例

使用されるポリシーファイルの内容を次に示します。

```
// (1)
// Grant all permissions to anything loaded from the
// EJB server itself

grant codeBase "file:${ejbserver.install.root}/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${ejbserver.install.root}/javaee/-" {
permission java.security.AllPermission;
};
grant codeBase "file:${tpbroker.java.home}/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/DABJ/*" {
permission java.security.AllPermission;
};
```

```

grant codeBase "file:${cosminexus.home}/manager/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/c4web/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/c4web/exlib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/jaxws/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/jaxrs/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/jaxp/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/CTM/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/PRF/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/wss/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/XMLSEC/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${ejbserver.install.root}/sfo/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${hntplib.home}/classes/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/common/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${ejbserver.install.root}/weld/lib/*" {
permission java.security.AllPermission;
};

// (2)
// Grant all permissions to the container generated stubs and
// implementation classes
grant codeBase "file:${ejbserver.http.root}/ejb/${ejbserver.serverName}/containers/-" {
permission java.security.AllPermission;
};

// (3)
// Grant all permissions to imported resource (datasource) implementations
// implementation classes
grant codeBase "http://*/ejb/${ejbserver.serverName}/import/resjars/-" {
permission java.security.AllPermission;
};

// (4)

```

```
// Grant permissions to resource adapters
//
grant codeBase "file:${ejbserver.http.root}/ejb/${ejbserver.serverName}/rarjars/-" {

// For uCosminexus TP1 Connector & TP1/Client/J
permission java.util.PropertyPermission "*", "read, write";

// For uCosminexus TP1 Connector & TP1/Client/J & Cosminexus Reliable Messaging
permission java.io.FilePermission "<<ALL FILES>>", "read, write, delete";
permission java.net.SocketPermission "*", "connect,listen,accept";

// For TP1/Message Queue - Access
permission java.lang.RuntimePermission "loadLibrary.*";

// For TP1/Message Queue - Access & Cosminexus Reliable Messaging
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.lang.RuntimePermission "modifyThread";

// For DB Connector
permission java.lang.reflect.ReflectPermission "suppressAccessChecks";

// For authentication (from J2EE RI server.policy file)
permission javax.security.auth.PrivateCredentialPermission "* * ¥"*¥", "read";

// For Cosminexus Reliable Messaging
permission javax.security.auth.AuthPermission "modifyPrivateCredentials";
permission java.lang.RuntimePermission "getenv.HRMDIR";

// For Cosminexus SOA FTP Inbound Adapter
permission java.lang.RuntimePermission "getClassLoader";
permission java.lang.RuntimePermission "setContextClassLoader";
permission java.lang.RuntimePermission "accessDeclaredMembers";
};

// (5)
// Grant permissions to JSP/Servlet
//
grant codeBase "file:${ejbserver.http.root}/web/${ejbserver.serverName}/-" {
permission java.lang.RuntimePermission "loadLibrary.*";
permission java.lang.RuntimePermission "queuePrintJob";
permission java.lang.RuntimePermission "modifyThread";
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.net.SocketPermission "*", "connect";
permission java.io.FilePermission "<<ALL FILES>>", "read, write";
permission java.util.PropertyPermission "*", "read";
permission javax.security.auth.AuthPermission "getSubject";
permission javax.security.auth.AuthPermission "createLoginContext.*";
};

// (6)
// Grant permissions to Cosminexus Service Coordinator
//
grant codeBase "file:${cosminexus.home}/CSC/lib/*" {
permission java.security.AllPermission;
};

// (7)
// Grant permissions to custom login modules
```



```
//
grant codeBase "file:${cosminexus.home}/manager/modules/-" {
permission java.io.FilePermission "<<ALL FILES>>", "read";
permission javax.security.auth.AuthPermission "modifyPrincipals";
permission javax.security.auth.AuthPermission "modifyPublicCredentials";
};

// (8)
// Grant minimal permissions to everything else:
// EJBs
// client implementation classes
grant {
permission java.util.PropertyPermission "*", "read";
permission java.lang.RuntimePermission "queuePrintJob";
permission java.net.SocketPermission "*", "connect";
};
```

記述例の (1) ~ (8) について説明します。

(1)

J2EE サーバが使用するクラスファイルに対して次の権限を許可します。

- すべてのアクセス権を許可

(2)

J2EE サーバが生成するスタブとスケルトンなどのクラスファイルに対して次の権限を許可します。

- すべてのアクセス権を許可

(3)

J2EE サーバが使用するリソースのクラスファイルに対して次の権限を許可します。

- すべてのアクセス権を許可

(4)

J2EE サーバが使用するリソースアダプタのクラスファイルに対して次の権限を許可します。

- すべてのプロパティ情報に対して読み取り，および書き込みを許可
- すべてのファイルに対して読み取り，書き込み，および削除を許可
- すべてのソケット通信に対してネットワークへの接続，接続での待機，および接続の受け付けを許可
- すべてのライブラリのロードを許可
- スレッドグループの変更を許可
- スレッドの変更を許可
- すべてのリフレクション操作を許可
- 任意の Subject が所有する，すべての非公開 Credential へのアクセスを許可
- Subject に関連づけられた非公開 Credential の Set の変更を許可
- 環境変数 HRMDIR の値の取得を許可
- クラスローダの取得を許可

- コンテキストクラスローダの設定を許可
- クラスの宣言されたメンバへのアクセスを許可

注意事項

- リソースアダプタ内の JAR ファイルの展開先である J2EE サーバ管理下のディレクトリが記載されています。
- J2EE サーバで動作するすべてのリソースアダプタが有効範囲です。

(5)

JSP／サーブレットのクラスファイルに対して次の権限を許可します。

- すべてのライブラリのロードを許可
- 印刷ジョブ要求を許可
- スレッドの変更を許可
- スレッドグループの変更を許可
- すべてのソケット通信に対してネットワーク接続を許可
- すべてのファイルに対して読み取り，および書き込みを許可
- すべてのプロパティ情報の読み取りを許可
- Subject の参照を許可
- あらゆる名称で LoginContext クラスのインスタンス化を許可

(6)

Service Coordinator のクラスファイルに対し次の権限を許可します。

- すべてのアクセス権を許可

(7)

統合ユーザ管理のカスタムログインモジュールに対し次の権限を許可します。

- すべてのファイルに対する読み込みを許可
- Subject に Principal および Credential の追加を許可

(8)

すべてのクラスファイルに対して次の権限を許可します。

- すべてのプロパティ情報の読み取りを許可
- 印刷ジョブ要求を許可
- すべてのソケット通信に対してネットワーク接続を許可

(5) 注意事項

- 構文が不正または適切なアクセス権限が設定されていない server.policy ファイルを使用した場合、java.lang.StackOverflowError または java.lang.OutOfMemoryError が発生して J2EE サーバが異常終了することがあります。
- サーバをセットアップするときに生成された server.policy ファイルには、J2EE サーバを動作させるために最低限必要な権限が記述されています。生成された server.policy ファイルの記述行の削除および変更はしないでください。

2.2.5 criticalList.cfg (保護区リストファイル)

保護区としてメソッドキャンセルを禁止するクラスを設定します。

この保護区リストファイルの設定が必要な場合は、アプリケーションサーバの各製品の構成ソフトウェアおよび関連製品で提供されるドキュメントの注意事項に記載されます。記載されている指示に従って設定してください。

(1) 形式

次の形式で指定します。

<要素>

<要素>には、次のどちらかを指定できます。

- <クラス名>

保護区に指定するクラス名称を指定します。

- <プリフィックス名>

保護区に指定するクラスが含まれるパッケージのプリフィックス名を指定します。指定したプリフィックスを持つクラスは、すべて保護区として指定されます。

指定方法

- 指定可能な要素を改行で区切って指定します。
- 空行は無視されます。
- シャープ (#) で始まる行はコメントとみなされます。
- <要素>の後ろに空白やコメントを追加することはできません。
(例) <クラス名> #<コメント>
- クラス名は、パッケージを含めて記述します。ただし、サフィックス".class"は記述しないでください。
(例) pacA.pacB.pacC.pacD.MyTestClass.class

この場合は、pacA.pacB.pacC.pacD.MyTestClass と記述します。

- クラス名称のプリフィックスは指定できません。

(例) pacA.pacB.pacC.pacD.My*

"My"をプリフィックスに持つクラス名称 (MyTestClass) を保護区指定する場合、クラス名称にアスタリスク (*) は使用できません。

この場合は、pacA.pacB.pacC.*と記述します。

- アスタリスク (*) だけの指定はできません。
- <要素>の前後にある半角空白は無視されます。
- <要素>の前後にある全角空白は文字とみなされ、不正な値と判断されます。

(2) ファイルの格納先

- Windows の場合

<Application Server のインストールディレクトリ>%CC%server%usrconf%

- UNIX の場合

/opt/Cosminexus/CC/server/usrconf/

(3) 記述例

プリフィックス名を指定する場合

```
#プリフィックス名指定  
pacA.pacB.*
```

パッケージ pacA.pacB 以下のクラスおよびパッケージに含まれるクラスが、すべて保護区になります。

例えば、パッケージ pacA.pacB.pacC 以下に、pacA.pacB.pacC.pacD がある場合、pacA.pacB.pacC.pacD 以下のクラスもすべて保護区になります。

クラス名を指定する場合

```
#クラス名指定  
pacX.pacY.pacZ.MyTestpacX.pacY.pacZ.CommonTest
```

(4) 注意事項

- 保護区リストファイルは製品単位の定義です。Component Container をインストールしたマシン上で動作するすべての J2EE サーバで有効になります。
- 保護区リストファイルは J2EE サーバの起動時に読み込まれます。変更内容を有効にしたい場合は、J2EE サーバを再起動する必要があります。
- J2EE サーバ起動時に、保護区リストファイルが存在しない、または設定が不正な場合、J2EE サーバの起動に失敗します。

3

バッチサーバで使用するファイル

この章では、バッチサーバで使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

3.1 バッチサーバで使用するファイルの一覧

バッチサーバで使用するファイルの一覧を、次の表に示します。

表 3-1 バッチサーバで使用するファイルの一覧

ファイル名	分類	概要	参照先
usrconf.cfg	バッチサーバ用オプション定義ファイル	バッチサーバを実行する JavaVM の起動オプションを指定します。	3.2.1
usrconf.properties	バッチサーバ用ユーザプロパティファイル	バッチサーバを実行する JavaVM のシステムプロパティを指定します。	3.2.2
server.policy	バッチサーバ用セキュリティポリシーファイル	バッチサーバを実行する JavaVM のセキュリティポリシーを指定します。	3.2.3
criticalList.cfg	保護区リストファイル	保護区としてメソッドキャンセルを禁止するクラスを設定します。	3.2.4
usrconf.cfg	バッチアプリケーション用オプション定義ファイル	cjexecjob コマンド、cjkilljob コマンドおよび cjlistjob コマンドを実行する JavaVM の起動オプションを指定します。	3.2.5
usrconf.properties	バッチアプリケーション用ユーザプロパティファイル	cjexecjob コマンド、および cjkilljob コマンドを実行する JavaVM のシステムプロパティを指定します。	3.2.6

3.2 バッチサーバで使用するファイルの詳細

3.2.1 usrconf.cfg (バッチサーバ用オプション定義ファイル)

(1) 形式

次のようにキーを指定します。

`<キー名称> = <値>`

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) `<キー名称> = <値> #<コメント>`

- 実行時の文字エンコーディングと異なる文字エンコーディングでは記述できません。

(2) ファイルの格納先

- Windows の場合
`<Application Server のインストールディレクトリ>%CC%server%usrconf%ejb%<サーバ名称>%`
- UNIX の場合
`/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/`

(3) 機能

バッチサーバを実行する JavaVM の起動オプションを指定します。

バッチサーバの稼働中に、このファイルの内容を変更した場合、変更した内容は次にバッチサーバを起動したときに反映されます。

(4) 指定できるキー

指定できるキーの概要とデフォルト値を次の表に示します。キーの詳細については、「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」を参照してください。なお、参照する場合は「J2EE サーバ」および「J2EE アプリケーション」を「バッチサーバ」および「バッチアプリケーション」と読み替えてください。

注意事項

ファイル内でマルチバイト文字を使用する場合は、次の個所で使用しているマルチバイト文字とエンコードを統一してください。

- usrconf.cfg (バッチアプリケーション用オプション定義ファイル)
- cjexecjob コマンド (バッチアプリケーション実行コマンド) の引数
- バッチアプリケーションのソースコード (java.lang.System.out または java.lang.System.err)

表 3-2 usrconf.cfg に指定できるキーとデフォルト値 (バッチサーバ)

キー名称	概要	デフォルト値
add.jvm.arg	指定されたオプションを使って JavaVM を起動します。 add.jvm.arg で指定できる JavaVM のオプションについては、「14.1 JavaVM 拡張オプションの一覧」および「14.5 Application Server で指定できる Java HotSpot VM のオプション」を参照してください。	デフォルト値については「3.2.1(5) バッチサーバでの JavaVM オプションのデフォルト値」を参照してください。
add.class.path	コンテナ拡張ライブラリ用のクラスパスを指定します。 スペースを含むパスを指定する場合、パスをダブルクォーテーション (") で囲む必要はありません。また、ダブルクォーテーションでの指定は有効となりません。 バッチアプリケーションクラスローダのクラスパスに JAR を追加する場合は、バッチアプリケーション用オプション定義ファイルで add.class.path キーを指定してください。	なし
add.library.path*	JNI 用の共有ライブラリを指定します。コンテナ拡張ライブラリが JNI を利用する場合だけ指定します。	なし
batch.service.enabled	このキーは、手動で編集しないでください。 cjsetup コマンドに -batch オプションを指定して実行した場合に、キーの値に true が表示されます。	true(-cjsetup コマンドに -batch オプションを指定して実行した場合)
cpp.library.version	プロセス内で使用する libstdc++ ライブラリのバージョンを指定します。なお、このキーは、Linux 用です。 指定できる文字列を次に示します。 6 : libstdc++.so.6 ライブラリを使用します。Red Hat Enterprise Linux 以外のプラットフォームで設定しても無効です。 09-00 以降では、設定できるのは「6」だけです。	6
ejb.public.directory	指定されたディレクトリをバッチサーバの作業ディレクトリとしてバッチサーバを起動します。	• Windows の場合 <Application Server のインストールディレクトリ >%CC%server*public • UNIX の場合

キー名称	概要	デフォルト値
		/opt/ Cosminexus/CC/ server/public
ejb.server.corefilenum (UNIX 用)	バッチサーバ再起動時に残す core ファイルの数を指定します。	1
ejb.server.log.directory	ログファイルの出力先ディレクトリを指定します。 Manager や J2EE サーバ、Web コンテナサーバ（互換機能）とログ出力先が、同じディレクトリにならないように注意してください。同じディレクトリを指定した場合、動作は保証されません。	• Windows の場合 ＜製品の作業ディレクトリ＞¥ejb¥＜サーバ名称＞¥logs • UNIX の場合 ＜製品の作業ディレクトリ＞/ejb/＜サーバ名称＞/logs
ejb.server.log.stdout.file size	cjstdout.log ファイルのサイズの上限值を指定します。	1048576
ejb.server.log.stderr.file size	cjstderr.log ファイルのサイズの上限值を指定します。	1048576
jvm.type	使用する JavaVM のタイプを指定します。	server

注※

cjstartsv コマンドまたは cjstartweb コマンド実行時、add.library.path と OS の環境変数（LIBPATH など）の両方が設定されている場合、add.library.path の設定が優先されます。

(5) バッチサーバでの JavaVM オプションのデフォルト値

バッチサーバでの Java VM オプションのデフォルト値を示します。

JDK11 を使用する場合

- -Xms256m
- -Xmx512m
- -XX:MetaspaceSize=128m
- -XX:MaxMetaspaceSize=128m
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:SurvivorRatio=8
- -XX:HitachiJavaLog:<ejb.server.log.directory>
- -XX:HitachiJavaLogFileSize=4m
- -XX:-HitachiUseExplicitMemory
- -XX:+HitachiVerboseGC

- -XX:+HitachiVerboseGCPrintCause
- -XX:+HitachiOutputMilliTime
- -XX:-HitachiThreadDumpToStdout
- -XX:+HitachiOutOfMemoryAbort
- -XX:+HitachiJavaClassLibTrace
- -XX:HitachiJavaClassLibTraceLineSize=1024
- -XX:+HitachiLocalsSimpleFormat
- -XX:+HitachiTrueTypeInLocals
- -XX:+HitachiLocalsInStackTrace
- -XX:HitachiExplicitHeapMaxSize=64m
- -XX:HitachiExplicitMemoryLogLevel:normal
- -XX:HitachiExplicitMemoryJavaLog:<ejb.server.log.directory>
- -XX:HitachiExplicitMemoryJavaLogFileSize=4m

JDK17 以降を使用する場合

- -Xms1g
- -Xmx1g
- -XX:MetaspaceSize=128m
- -XX:MaxMetaspaceSize=128m
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:SurvivorRatio=8
- -XX:HitachiJavaLog:<ejb.server.log.directory>
- -XX:HitachiJavaLogFileSize=4m
- -XX:+HitachiVerboseGC
- -XX:+HitachiVerboseGCPrintCause
- -XX:+HitachiOutputMilliTime
- -XX:-HitachiThreadDumpToStdout
- -XX:+HitachiOutOfMemoryAbort
- -XX:+HitachiJavaClassLibTrace
- -XX:HitachiJavaClassLibTraceLineSize=1024
- -XX:+HitachiLocalsSimpleFormat
- -XX:+HitachiTrueTypeInLocals

- -XX:+HitachiLocalsInStackTrace

(6) 注意事項

- add.class.path キーに指定されている値の中に、<cosminexus.home>タグを使用してコンテナ拡張ライブラリ用の JAR を指定しているものがあります。これは Application Server のインストール先を意味しているため、add.class.path キーの記述内容を変更する必要はありません。なお、add.class.path キーにコンテナ拡張ライブラリを追加する場合は、インストールおよびアンインストール時の動作を保証できなくなるおそれがあるので、<cosminexus.home>タグを使用しないでください。
- add.jvm.arg キーで--add-modules および--module-path オプションを指定しないでください。指定した場合は、動作を保証しません。

3.2.2 usrconf.properties (バッチサーバ用ユーザプロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

次のようにキーを指定します。

<キー名称> = <値>

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) <キー名称> = <値> #<コメント>

- 記載する文字は Java の仕様に従って、ISO 8859-1 文字エンコーディングを使用してください。
- CJBATCHUSRCONFDIR に指定するパスは絶対パスで指定してください。
- Java の仕様に従わない形式の場合、バッチサーバの起動に失敗することがあります。

(例) 不正な Unicode エスケープシーケンスを含む場合 ("¥u"に続く文字列が適切な Unicode 16 進数の値ではない)。

(2) ファイルの格納先

- Windows の場合

<Application Server のインストールディレクトリ>¥CC¥server¥usrconf¥ejb¥<サーバ名称>¥

- UNIX の場合
/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/

(3) 機能

バッチサーバを実行する JavaVM のシステムプロパティを指定します。

usrconf.properties と usrconf.cfg の add.jvm.arg の -D に同じキーが指定されている場合は、usrconf.properties の値が優先されます。

バッチサーバの稼働中に、このファイルの内容を変更した場合、変更した内容は次にバッチサーバを起動したときに反映されます。

(4) バッチサーバの予約済みキー

バッチサーバでは、次に示す接頭子で始まるキーを内部的に利用します。このため、アプリケーションでこれらの接頭子で始まるキーを利用してはいけません。

- ejbserver.*
- webserver.*
- appclient.*
- j2eeserver.*

(5) バッチサーバのカスタマイズ用キー

バッチサーバでは、次に示すシステムプロパティのキーに値を設定することで、バッチサーバの動作をカスタマイズできます。

ここでは、キーを次の分類に分けて説明します。

- ejbserver.application から始まるキー
- ejbserver.batch から始まるキー
- ejbserver.client から始まるキー
- ejbserver.connectionpool から始まるキー
- ejbserver.connector から始まるキー
- ejbserver.container から始まるキー
- ejbserver.ctm から始まるキー
- ejbserver.deploy から始まるキー
- ejbserver.ext から始まるキー
- ejbserver.http から始まるキー

- ejbserver.instrumentation から始まるキー
- ejbserver.jndi から始まるキー
- ejbserver.jta から始まるキー
- ejbserver.logger から始まるキー
- ejbserver.management から始まるキー
- ejbserver.manager から始まるキー
- ejbserver.naming から始まるキー
- ejbserver.rmi から始まるキー
- ejbserver.server から始まるキー
- ejbserver.stdoutlog から始まるキー
- ejbserver.watch から始まるキー
- java から始まるキー
- vbroker から始まるキー
- webserver.connector から始まるキー

それぞれに指定できるキーの概要とデフォルト値を次の表に示します。キーの詳細については、「[2.2.3 usrconf.properties \(J2EE サーバ用ユーザプロパティファイル\)](#)」を参照してください。なお、参照する場合は「J2EE サーバ」および「J2EE アプリケーション」を「バッチサーバ」および「バッチアプリケーション」と読み替えてください。

(a) ejbserver.application から始まるキー

キー名称	概要	デフォルト値
ejbserver.application.InitTermProcessClasses	コンテナ拡張ライブラリを使用する場合に、サーバ起動・停止フックのクラス名を指定します。	なし
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.appname	AppName のフィールドに出力するデフォルトのアプリケーション名を指定します。	user_app
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.count	ログファイルの面数を指定します。	2
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.encoding	出力する文字列のエンコーディングを指定します。	null
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.filter	使用するフィルタを指定します。	null

キー名称	概要	デフォルト値
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>.formatter	使用するフォーマッタを指定します。	null
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>.level	ログの取得レベルの上限を指定します。	SEVERE
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>.limit	ログファイルの容量を指定します。	1048576
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>.msgid	MsgID のフィールドに出力するデフォルトのメッセージ ID を指定します。	0001
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>.path	ログファイル名のプリフィックスを 1～255 バイトの範囲で指定します。	user_log
ejbserver.application.serlog.CJLogHandler.<ハンドラ名称>.separator	CJSimpleFormatter を利用して、1 文でメッセージを出力するためのセパレータを指定します。	(ストローク)
ejbserver.application.serlog.loggers	使用するロガーの名称を指定します。	なし
ejbserver.application.serlog.Logger.<ロガー名称>.filter	ロガーで使用するフィルタを指定します。	null
ejbserver.application.serlog.Logger.<ロガー名称>.handlers	指定のロガー名称に対するハンドラクラスを指定します。	なし
ejbserver.application.serlog.Logger.<ロガー名称>.level	ロガーのログ出力レベルを指定します。	SEVERE
ejbserver.application.serlog.Logger.<ロガー名称>.useParentHandlers	ログレコードを受け取ったロガーから、親ロガーの接続しているハンドラへログレコードを伝播させるかどうかを指定します。	true

(b) ejbserver.batch から始まるキー

キー名称	概要	デフォルト値
ejbserver.batch.application.exit.enabled	バッチアプリケーションから VM 終了メソッドを呼び出した時のバッチサーバの動作を指定します。 true を指定した場合： JavaVM を終了しないでバッチアプリケーションのスレッドを終了します。	true

キー名称	概要	デフォルト値
	false を指定した場合： バッチサーバを終了します。	
ejbserver.batch.gc.watch.threshold	GC を実行するメモリ使用量のしきい値を、0～100（単位：％）の整数で設定します。次の条件のどれかの場合に GC を実行します。0 を指定した場合は GC の監視を行いません。 • Tenured 領域消費サイズの Tenured 領域合計サイズに対する割合がしきい値以上（SerialGC 有効時） • New 領域合計サイズの Tenured 領域最大空きサイズに対する割合がしきい値以上（SerialGC 有効時） • Java ヒープ領域消費サイズの Java ヒープ領域サイズに対する割合がしきい値以上（G1GC 有効時） • Metaspace 領域消費サイズの Metaspace 領域最大サイズに対する割合がしきい値以上	0
ejbserver.batch.schedule.group.name	CTM によって管理されるバッチサーバ群のグループ名を、英数字、およびアンダースコア「_」を使って 63 文字以内で指定します。	JOBGROUP
ejbserver.batch.queue.length	CTM で作成されるスケジュールキューの長さを 1～32767 の間で指定します。	50

注 このキーは、バッチサーバ用ユーザプロパティファイル特有のキーです。J2EE サーバ用ユーザプロパティファイルには存在しません。

(c) ejbserver.client から始まるキー

キー名称	概要	デフォルト値
ejbserver.client.ctm.RequestPriority	CTM 内のキューに滞留したリクエストを引き出す時のプライオリティ（優先順位）を指定します。	4

(d) ejbserver.connectionpool から始まるキー

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	概要	デフォルト値	関連情報
ejbserver.connectionpool.applicationAuthentication.disabled	コンテナ管理のサインオンの最適化機能を有効にするかどうかを指定します。	false	
ejbserver.connectionpool.association.enabled	コネクションアソシエーション機能を有効にするかどうかを指定します。	false	「機能解説 基本・開発編(コンテナ共通機能)」の「3.14 パフォーマンスチューニングのための機能」

キー名称	概要	デフォルト値	関連情報
ejbserver.connectionpool.sharing OutsideTransactionScope.enabled	アプリケーションサーバが管理するトランザクションの外で複数回コネクションの取得を行ったときのコネクションシェアリングの動作を指定します。	false	
ejbserver.connectionpool.validation.timeout	コネクション障害検知機能のタイムアウト時間およびコネクション数調節機能によるコネクション削除処理のタイムアウト時間（単位：秒）を 1～2147483647 の整数で指定します。	5	

(凡例)

空欄：関連情報はありません。

(e) ejbserver.connector から始まるキー

キー名称	概要	デフォルト値
ejbserver.connector.log writer.filenum	リソースアダプタ用ログファイルの面数を指定します。	4
ejbserver.connector.log writer.filesize	リソースアダプタ用ログファイルのサイズを指定します。	2097152

(f) ejbserver.container から始まるキー

キー名称	概要	デフォルト値
ejbserver.container.audit_trail.enabled	データベース監査証跡連携機能を有効にするかどうかを指定します。	false

(g) ejbserver.ctm から始まるキー

キー名称	概要	デフォルト値
ejbserver.ctm.Activate TimeOut	スケジューリング機能が有効な場合、バッチサーバは起動時に CTM キューを活性化します。このときの待ち時間（単位：秒）を 0～2147483647 で指定します。0 を指定した場合、無限に待ち続けます。	180
ejbserver.ctm.CTMDomain	CTM 機能を使用する場合に必要なプロパティです。バッチサーバが属する CTM ドメイン名称を指定します。	CTMDOMAIN
ejbserver.ctm.CTMID	CTM 機能を使用する場合に必要なプロパティです。バッチサーバを制御する CTM デーモンの CTM 識別子を指定します。	<IP アドレス>
ejbserver.ctm.CTMMYHost	CTM 機能を使用する場合に必要なプロパティです。マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを指定します。	<hostname コマンドで取得されるホスト名称>
ejbserver.ctm.DeactivateTimeOut	スケジューリング機能が有効な場合、バッチサーバは停止時に CTM キューを非活性化します。このときの待ち時間（単位：秒）を 0～2147483647 で指定します。	180

キー名称	概要	デフォルト値
ejbserver.ctm.enabled	スケジューリング機能の使用の有無を指定します。 true を指定した場合： スケジューリング機能を使用します。バッチサーバ起動時に CTM との接続の確立、および初期化をします。これに成功するとスケジューリング機能が使用できる状態でバッチサーバが起動します。 CTM との接続および初期化に失敗した場合、バッチサーバの起動に失敗します。 false を指定した場合： スケジューリング機能を使用しません。スケジューリング機能を使用しない状態でバッチサーバが起動します。	false

(h) ejbserver.deploy から始まるキー

キー名称	概要	デフォルト値
ejbserver.deploy.exclusive.lockAliveInterval	サーバ処理終了後、コマンドから送信された次のレスポンスを受け取るまでの最大待ち時間を指定します。	60
ejbserver.deploy.resourcefile.scramble.enabled	作業ディレクトリ以下に展開される DataSource, JavaMail, ResourceAdapter に関する DD ファイルをスクランブルするかどうかを設定します。	false

(i) ejbserver.ext から始まるキー

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	概要	デフォルト値	関連情報
ejbserver.ext.method_observation.interval	このプロパティの設定には、次の二つの意味があります。 - J2EE アプリケーション実行時間監視機能を使用するかどうかを指定します。 - 動作中のリクエスト処理がタイムアウトしていないかを監視する時間間隔、タイムアウトしたリクエスト（メソッド）をキャンセルする時間間隔を指定します。	0	「機能解説 運用／監視／連携編」の「5. J2EE アプリケーションの運用」

(j) ejbserver.http から始まるキー

キー名称	概要	デフォルト値
ejbserver.http.port	管理用サーバのポート番号を指定します。	28080

(k) ejbserver.instrumentation から始まるキー

キー名称	概要	デフォルト値
ejbserver.instrumentation.enabled	Management Server と連携するかどうかを指定します。	true

(l) ejbserver.jndi から始まるキー

キー名称	概要	デフォルト値
ejbserver.jndi.cache	ネーミングでのキャッシングをするかどうかを指定します。	on
ejbserver.jndi.cache.interval	ネーミングでのキャッシングをする場合、キャッシュをクリアする間隔を指定します。	0
ejbserver.jndi.cache.interval.clear.option	インターバル経過後のネーミングでのキャッシュ領域に対する動作を決定します。	refresh
ejbserver.jndi.cache.reference	DataSource オブジェクトのキャッシング機能を使用するかどうかを指定します。	off
ejbserver.jndi.namingservice.group.list	ラウンドロビン検索をする場合に、検索対象になる論理ネーミングサービスのグループを定義します。	なし
ejbserver.jndi.namingservice.group.<Specify group name>.providerurls	各グループに属するネーミングサービスのルート位置をプロバイダ URL で指定します。	なし
ejbserver.jndi.request.timeout	ネーミングサービスとの通信タイムアウト時間を指定します。	0

(m) ejbserver.jta から始まるキー

キー名称	概要	デフォルト値
ejbserver.jta.TransactionManager.defaultTimeout	バッチサーバ上で開始されるトランザクションのトランザクションタイムアウトのデフォルト値を指定します。	180

(n) ejbserver.logger から始まるキー

キー名称	概要	デフォルト値
ejbserver.logger.channels.define.<チャンネル名>*.filenum	バッチサーバのログファイルの面数を指定します。	- チャンネル名が MaintenanceLogFile の場合 4 - チャンネル名が上記以外の場合

キー名称	概要	デフォルト値
		2
ejbserver.logger.channels.define.<チャンネル名> ※.filesize	バッチサーバのログファイルのサイズを指定します。	- チャンネル名が MaintenanceLogFile の場合 16777216 - チャンネル名が上記以外の場合 1048576
ejbserver.logger.enabled.*	バッチサーバのログレベルを指定します。	Error
ejbserver.logger.rotationStyle	J2EE サーバのログ出力ファイルのローテーション方式を指定します。 SHIFT を指定した場合： シフトモードでローテーションします。 WRAP を指定した場合： ラップアラウンドモードでローテーションします。	WRAP

注※

チャンネル名称として次に示す名称を設定できます。

- アプリケーションサーバのログ出力用のチャンネル
MessageLogFile, MaintenanceLogFile, ExceptionLogFile, ConsoleLogFile, EJBContainerLogFile, UserOutLogFile, UserErrLogFile
- リソース枯渇監視ログ出力用のチャンネル
MemoryWatchLogFile, FileDescriptorWatchLogFile, ThreadWatchLogFile, ThreaddumpWatchLogFile, ConnectionPoolWatchLogFile

資料の取得については、マニュアル「アプリケーションサーバ 機能解説 保守／移行編」の「2.3 資料の取得」を参照してください。

(o) ejbserver.management から始まるキー

キー名称	概要	デフォルト値
ejbserver.management.JVM.stats_monitor.FullGCCount.enabled	FullGC 回数の監視を有効にするかどうかを指定します。	true
ejbserver.management.JVM.stats_monitor.FullGCCount.threshold	FullGC 回数の監視でのしきい値となる回数を指定します。	10
ejbserver.management.JVM.stats_monitor.FullGCCount.interval	FullGC 回数の監視でのしきい値監視間隔を指定します。	600

キー名称	概要	デフォルト値
ejbserver.management.statistics.interval	稼働情報収集間隔を指定します。	60
ejbserver.management.stats_file.base_time	稼働情報ファイルの時間切り替え運用での基点時刻を指定します。	0
ejbserver.management.stats_file.dir	稼働情報ファイルの出力先ディレクトリを指定します。	- Windows の場合 <製品の作業ディレクトリ>*ejb*<サーバ名称>*stats - UNIX の場合 <製品の作業ディレクトリ>/ejb/<サーバ名称>/stats
ejbserver.management.stats_file.enabled	稼働情報ファイル出力機能を有効にするかどうかを指定します。	true
ejbserver.management.stats_file.num	稼働情報ファイルの面数を指定します。	7
ejbserver.management.stats_file.period	稼働情報ファイルの時間切り替え運用での切り替え時間を指定します。	24

(p) ejbserver.manager から始まるキー

キー名称	概要	デフォルト値
ejbserver.manager.agent.Agent.conf	Management Server との連携に必要な設定をします。	なし
ejbserver.manager.agent.Agent.enabled	Management Server との連携に必要な設定をします。	false
ejbserver.manager.agent.JP1EventAgent.conf	JP1 連携に関する設定ファイルのパスを指定します。	なし
ejbserver.manager.agent.JP1EventAgent.enabled	JP1 と連携するかどうかを指定します。	false
ejbserver.manager.agent.MEventAgent.conf	Management イベント発行用プロパティファイルを指定します。	なし
ejbserver.manager.agent.MEventAgent.enabled	Management イベント発行機能を有効にするかどうかを指定します。	false

(q) ejbserver.naming から始まるキー

キー名称	概要	デフォルト値
ejbserver.naming.host	バッチサーバがネーミングサービスとして利用する CORBA ネーミングサービスが起動しているホスト名称または IP アドレスを指定します。	localhost
ejbserver.naming.name root	CORBA ネーミングサービスを自動起動モードで使用する場合に、起動させる CORBA ネーミングサービスに名称を付加するときに指定します。	なし
ejbserver.naming.port	バッチサーバがネーミングサービスとして利用する CORBA ネーミングサービスのポート番号を指定します。	900
ejbserver.naming.startup pRetryCount	ejbserver.naming.startupWaitTime で指定した CORBA ネーミングサービスの起動待ち時間経過後、CORBA ネーミングサービスがまだ起動していない場合に、ejbserver.naming.startupWaitTime で指定された秒数だけ起動待ちを繰り返す回数を指定します。	9
ejbserver.naming.startup pWaitTime	バッチサーバ起動時に CORBA ネーミングサービスが利用可能になるまで待つ時間を指定します。	1

(r) ejbserver.rmi から始まるキー

キー名称	概要	デフォルト値
ejbserver.rmi.naming.host	マルチホームドホスト環境で、バッチサーバが使用する RMI レジストリや MBean サーバのホスト名称、または IP アドレスを指定します。	なし
ejbserver.rmi.naming.port	バッチサーバが利用する RMI レジストリのポート番号を指定します。	23152
ejbserver.rmi.remote.listener.port	バッチサーバ内でエクスポートするリモートオブジェクトのポート番号を指定します。	0
ejbserver.rmi.request.timeout	クライアントとサーバ間の通信タイムアウト時間を指定します。	0

(s) ejbserver.server から始まるキー

キー名称	概要	デフォルト値
ejbserver.server.prf.PR FID	PRF 識別子を指定します。	PRF_ID
ejbserver.server.thread dump.filenum	バッチサーバが障害を検知して自発的にスレッドダンプを出力する場合のスレッドダンプのファイル数の上限を指定します。	256

(t) ejbserver.stdoutlog から始まるキー

キー名称	概要	デフォルト値
ejbserver.stdoutlog.aut oflush	ユーザ出力ログおよびユーザエラーログの自動フラッシュ機能を有効にするかどうかを指定します。	false

(u) ejbserver.watch から始まるキー

キー名称	概要	デフォルト値
ejbserver.watch.enabled	すべてのリソース枯渇監視を有効にするかどうかを指定します。	true
ejbserver.watch.memory.enabled	メモリ監視のアラート出力を有効にするかどうかを指定します。	true
ejbserver.watch.memory.interval	メモリ監視間隔を指定します。	60
ejbserver.watch.memory.threshold	メモリの使用状態を監視するしきい値を指定します。	80
ejbserver.watch.memory.writefile.enabled	メモリ監視結果をファイル出力するかどうかを指定します。	true
ejbserver.watch.fileDescriptor.enabled	ファイルディスクリプタ監視のアラート出力を有効にするかどうかを指定します。なお、ファイルディスクリプタ監視は、Windows の場合、および AIX の場合は使用できません。	true
ejbserver.watch.fileDescriptor.interval	ファイルディスクリプタ監視間隔を指定します。なお、ファイルディスクリプタ監視は、Windows の場合、および AIX の場合は使用できません。	60
ejbserver.watch.fileDescriptor.threshold	ファイルディスクリプタの使用状態を監視するしきい値を指定します。なお、ファイルディスクリプタ監視は、Windows の場合、および AIX の場合は使用できません。	2147483647
ejbserver.watch.fileDescriptor.writefile.enabled	ファイルディスクリプタ監視結果をファイル出力するかどうかを指定します。なお、ファイルディスクリプタ監視は、Windows の場合、および AIX の場合は使用できません。	true
ejbserver.watch.thread.enabled	スレッド監視のアラート出力を有効にするかどうかを指定します。なお、Linux ではスレッド数を監視できません。	true
ejbserver.watch.thread.interval	スレッド監視間隔を指定します。なお、Linux ではスレッド数を監視できません。	60
ejbserver.watch.thread.threshold	スレッドの使用状態を監視するしきい値を指定します。なお、Linux ではスレッド数を監視できません。	2147483647
ejbserver.watch.thread.writefile.enabled	スレッド監視結果をファイル出力するかどうかを指定します。なお、Linux ではスレッド数を監視できません。	true
ejbserver.watch.thread.dump.enabled	スレッドダンプ監視のアラート出力を有効にするかどうかを指定します。	true
ejbserver.watch.thread.dump.interval	スレッドダンプ監視時間間隔を指定します。	30
ejbserver.watch.thread.dump.threshold	スレッドダンプ監視のしきい値を、上限値に対する割合で指定します。	80
ejbserver.watch.thread.dump.writefile.enabled	スレッドダンプ監視結果をファイル出力するかどうかを指定します。	true

キー名称	概要	デフォルト値
ejbserver.watch.memory.rate2alert.enabled	リソース枯渇監視機能のメモリ枯渇監視情報の Rate2 のアラート出力を有効にするかどうかを指定します。	false

(v) java から始まるキー

キー名称	概要	デフォルト値
java.naming.factory.initial	JNDI の InitialContext がデレゲートするコンテキストの実装クラスの実装クラスを指定します。	com.hitachi.software.ejb.jndi.InitialContextFactory

(w) vbroker から始まるキー

キー名称	概要	デフォルト値
vbroker.agent.enableLocator	スマートエージェントを使用するかどうかを指定します。スケジューリング機能を使用するバッチサーバを起動する場合、自動的に"true"が設定されます。	false
vbroker.agent.port	スマートエージェントのポート番号を指定します。	14000
vbroker.ce.iiop.ccm.htc.readerPerConnection	リモートインタフェースとして定義された EJB メソッドの呼び出しでタイムアウトが発生した場合に、コネクションのクローズを抑止するかどうかを設定します。 true を指定した場合 タイムアウト発生時のコネクションのクローズを抑止します。 false を指定した場合 タイムアウト発生時のコネクションのクローズを抑止しません。	false
vbroker.ce.iiop.ccm.htc.threadStarter	リプライ受信専用スレッドを管理するスレッドを起動するかどうかを設定します。vbroker.ce.iiop.ccm.htc.readerPerConnection=true を設定する場合に、"true"を設定してください。	false
vbroker.orb.htc.comt.entryCount	TPBroker 通信トレースファイル 1 個当たりのエントリ数の上限値を指定します。	120000
vbroker.orb.htc.comt.fileCount	TPBroker 通信トレースファイルの個数の上限値を指定します。	3
vbroker.orb.htc.tracePath	TPBroker のトレースファイル出力先のパスを 1~210 バイトの範囲で指定します。	- Windows の場合 ＜製品の作業ディレクトリ＞%ejb%＜サーバ名称＞%logs%TPB%logj - UNIX の場合 ＜製品の作業ディレクトリ＞

キー名称	概要	デフォルト値
		>/ejb/<サーバ名称>/logs/TPB/logj
vbroker.se.iiop_tp.host	任意の値を指定して、バッチサーバ単位で EJB コンテナの IP アドレスを固定します。	なし
vbroker.se.iiop_tp.scm.i iop_tp.listener.port	任意の値を指定して、バッチサーバ単位で通信ポートを固定します。	0

(x) webserver.connector から始まるキー

キー名称	概要	デフォルト値
webserver.connector.aj p13.bind_host	Web サーバ連携で使用する IP アドレスまたはホスト名称を指定します。	なし
webserver.connector.aj p13.port	Web サーバとの通信に使用するポート番号を指定します。	8007

3.2.3 server.policy (バッチサーバ用セキュリティポリシーファイル)

(1) 形式

J2SE のセキュリティポリシーファイル形式に従います。

(2) ファイルの格納先

- Windows の場合
<Application Server のインストールディレクトリ>%CC%server%usrconf%ejb%<サーバ名称>%
- UNIX の場合
/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/

(3) 機能

バッチサーバを実行する JavaVM のセキュリティポリシーを指定します。

バッチサーバの稼働中に、このファイルの内容を変更した場合、変更した内容は次にバッチサーバを起動したときに反映されます。

(4) 記述例

使用されるポリシーファイルの内容を次に示します。

```

// (1)
// Grant all permissions to anything loaded from the
// EJB server itself

grant codeBase "file:${ejbserver.install.root}/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${tpbroker.java.home}/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/DABJ/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/manager/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/c4web/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/c4web/exlib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/jaxp/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/CTM/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/PRF/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/wss/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/XMLSEC/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${ejbserver.install.root}/sfo/lib/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${hntplib.home}/classes/*" {
permission java.security.AllPermission;
};
grant codeBase "file:${cosminexus.home}/common/lib/*" {
permission java.security.AllPermission;
};

// (2)
// Grant all permissions to the container generated stubs and
// implementation classes
grant codeBase "file:${ejbserver.http.root}/ejb/${ejbserver.serverName}/containers/-" {
permission java.security.AllPermission;
};

// (3)
// Grant all permissions to imported resource (datasource) implementations
// implementation classes

```



```

grant codeBase "http://*/ejb/${ejbserver.serverName}/import/resjars/-" {
permission java.security.AllPermission;
};

// (4)
// Grant permissions to resource adapters
//
grant codeBase "file:${ejbserver.http.root}/ejb/${ejbserver.serverName}/rarjars/-" {

// For Cosminexus TP1 Connector & TP1/Client/J
permission java.util.PropertyPermission "*", "read, write";

// For Cosminexus TP1 Connector & TP1/Client/J & Cosminexus Reliable Messaging
permission java.io.FilePermission "<<ALL FILES>>", "read, write, delete";
permission java.net.SocketPermission "*", "connect,listen,accept";

// For TP1/Message Queue - Access
permission java.lang.RuntimePermission "loadLibrary.*";

// For TP1/Message Queue - Access & Cosminexus Reliable Messaging
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.lang.RuntimePermission "modifyThread";

// For DB Connector
permission java.lang.reflect.ReflectPermission "suppressAccessChecks";

// For authentication (from J2EE RI server.policy file)
permission javax.security.auth.PrivateCredentialPermission "* * ¥*¥", "read";

// For Cosminexus Reliable Messaging
permission javax.security.auth.AuthPermission "modifyPrivateCredentials";
permission java.lang.RuntimePermission "getenv.HRMDIR";

// (5)
// Grant permissions to JSP/Servlet
//
grant codeBase "file:${ejbserver.http.root}/web/${ejbserver.serverName}/-" {
permission java.lang.RuntimePermission "loadLibrary.*";
permission java.lang.RuntimePermission "queuePrintJob";
permission java.lang.RuntimePermission "modifyThread";
permission java.lang.RuntimePermission "modifyThreadGroup";
permission java.net.SocketPermission "*", "connect";
permission java.io.FilePermission "<<ALL FILES>>", "read, write";
permission java.util.PropertyPermission "*", "read";
};

// (6)
// Grant permissions to Cosminexus Service Coordinator
//
grant codeBase "file:${cosminexus.home}/CSC/lib/*" {
permission java.lang.security.AIIPermission;
};

// (7)
//
// Grant minimal permissions to everything else:
// Batch applications
// EJBs

```

```
// client implementation classes
grant {
    permission java.security.AllPermission;
};
```

記述例の (1) ~ (7) について説明します。

(1)

バッチサーバが使用するクラスファイルに対して次の権限を許可します。

- すべてのアクセス権を許可

(2)

バッチサーバが生成するスタブとスケルトンなどのクラスファイルに対して次の権限を許可します。

- すべてのアクセス権を許可

(3)

バッチサーバが使用するリソースのクラスファイルに対して次の権限を許可します。

- すべてのアクセス権を許可

(4)

バッチサーバが使用するリソースアダプタのクラスファイルに対して次の権限を許可します。

- すべてのプロパティ情報に対して読み取り，および書き込みを許可
- すべてのファイルに対して読み取り，書き込み，および削除を許可
- すべてのソケット通信に対してネットワークへの接続，接続での待機，および接続の受け付けを許可
- すべてのライブラリのロードを許可
- スレッドグループの変更を許可
- スレッドの変更を許可
- すべてのリフレクション操作を許可
- 任意の Subject が所有する，すべての非公開 Credential へのアクセスを許可
- Subject に関連づけられた非公開 Credential の Set の変更を許可
- 環境変数 HRMDIR の値の取得を許可

注意事項

- リソースアダプタ内の JAR ファイルの展開先であるバッチサーバ管理下のディレクトリが記載されています。
- バッチサーバで動作するすべてのリソースアダプタが有効範囲です。

(5)

JSP／サーブレットのクラスファイルに対して次の権限を許可します。

- すべてのライブラリのロードを許可
- 印刷ジョブ要求を許可

- スレッドの変更を許可
- スレッドグループの変更を許可
- すべてのソケット通信に対してネットワーク接続を許可
- すべてのファイルに対して読み取り，および書き込みを許可
- すべてのプロパティ情報の読み取りを許可

(6)

Service Coordinator のクラスファイルに対して次の権限を許可します。

- すべてのアクセス権を許可

(7)

すべてのクラスファイルに対して次の権限を許可します。

- すべてのアクセス権を許可

(5) 注意事項

- 構文が不正または適切なアクセス権限が設定されていない `server.policy` ファイルを使用した場合，`java.lang.StackOverflowError` または `java.lang.OutOfMemoryError` が発生してバッチサーバが異常終了することがあります。
- サーバをセットアップするときに生成された `server.policy` ファイルには，バッチサーバを動作させるために最低限必要な権限が記述されています。生成された `server.policy` ファイルの記述行の削除および変更はしないでください。ただし，`signedBy`，`codeBase`，および `principal` が指定されていない `grant` にある `java.security.AllPermission` は，次の権限を設定すれば削除できます。
 - `java.util.PropertyPermission "*" , "read"`;
 - `java.lang.RuntimePermission "queuePrintJob"`;
 - `java.net.SocketPermission "*" , "connect"`;

3.2.4 criticalList.cfg (保護区リストファイル)

保護区としてメソッドキャンセルを禁止するクラスを設定します。

この保護区リストファイルの設定が必要な場合は，アプリケーションサーバの各製品の構成ソフトウェアおよび関連製品で提供されるドキュメントの注意事項に記載されます。記載されている指示に従って設定してください。

(1) 形式

次の形式で指定します。

<要素>

<要素>には、次のどちらかを指定できます。

- <クラス名>

保護区に指定するクラス名称を指定します。

- <プリフィックス名>

保護区に指定するクラスが含まれるパッケージのプリフィックス名を指定します。指定したプリフィックスを持つクラスは、すべて保護区として指定されます。

指定方法

- 指定可能な要素を改行で区切って指定します。
- 空行は無視されます。
- シャープ（#）で始まる行はコメントとみなされます。
- <要素>の後ろに空白やコメントを追加することはできません。
(例) <クラス名> #<コメント>
- クラス名は、パッケージを含めて記述します。ただし、サフィックス".class"は記述しないでください。
(例) pacA.pacB.pacC.pacD.MyTestClass.class
この場合は、pacA.pacB.pacC.pacD.MyTestClass と記述します。
- クラス名称のプリフィックスは指定できません。
(例) pacA.pacB.pacC.pacD.My*
"My"をプリフィックスに持つクラス名称（MyTestClass）を保護区指定する場合、クラス名称にアスタリスク（*）は使用できません。
この場合は、pacA.pacB.pacC.*と記述します。
- アスタリスク（*）だけの指定はできません。
- <要素>の前後にある半角空白は無視されます。
- <要素>の前後にある全角空白は文字とみなされ、不正な値と判断されます。

(2) ファイルの格納先

- Windows の場合
<Application Server のインストールディレクトリ>%CC%server%usrconf%
• UNIX の場合
/opt/Cosminexus/CC/server/usrconf/

(3) 記述例

プリフィックス名を指定する場合

```
#プリフィックス名指定
pacA. pacB. *
```

パッケージ pacA.pacB 以下のクラスおよびパッケージに含まれるクラスが、すべて保護区になります。
例えば、パッケージ pacA.pacB.pacC 以下に、pacA.pacB.pacC.pacD がある場合、
pacA.pacB.pacC.pacD 以下のクラスもすべて保護区になります。

クラス名を指定する場合

```
#クラス名指定
pacX. pacY. pacZ. MyTest pacX. pacY. pacZ. CommonTest
```

(4) 注意事項

- 保護区リストファイルは製品単位の定義です。Component Container をインストールしたマシン上で動作するすべてのバッチサーバで有効になります。
- 保護区リストファイルはバッチサーバの起動時に読み込まれます。変更内容を有効にしたい場合は、バッチサーバを再起動する必要があります。
- バッチサーバ起動時に、保護区リストファイルが存在しない、または設定が不正な場合、バッチサーバの起動に失敗します。

3.2.5 usrconf.cfg (バッチアプリケーション用オプション定義ファイル)

(1) 形式

次のようにキーを指定します。

```
<キー名称> = <値>
```

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) <キー名称> = <値> #<コメント>

- 実行時の文字エンコーディングと異なる文字エンコーディングでは記述できません。

- CJBATCHUSRCONFDIR に指定するパスは絶対パスで指定してください。

(2) ファイルの格納先

環境変数 CJBATCHUSRCONFDIR に usrconf.cfg (バッチアプリケーション用オプション定義ファイル) の格納先のパスを設定するか、バッチアプリケーションで使用するコマンドを実行するディレクトリに usrconf.cfg (バッチアプリケーション用オプション定義ファイル) を格納してください。usrconf.cfg (バッチアプリケーション用オプション定義ファイル) は、バッチアプリケーションで使用するコマンドを実行する実行時に、次の順で検索されます。

1. 環境変数 (CJBATCHUSRCONFDIR) に設定したディレクトリ
2. cjexecjob コマンド (バッチアプリケーション実行コマンド) を実行したディレクトリ

(3) 機能

バッチアプリケーションを実行する JavaVM の起動オプションを指定します。

バッチアプリケーションの実行中に、このファイルの内容を変更した場合、変更した内容は次にバッチアプリケーションを実行したときに反映されます。

(4) 定義値

バッチアプリケーションで使用するコマンドを実行する場合、スケジューリング機能を使用するか、使用しないかによって有効となるオプションが異なります。

バッチアプリケーションで使用するコマンドごとに次の表に示します。

オプション名	cjexecjob		cckilljob		cjlistjob	
	スケジューリング機能未使用	スケジューリング機能使用	スケジューリング機能未使用	スケジューリング機能使用	スケジューリング機能未使用	スケジューリング機能使用
add.jvm.arg	○	○	—	—	—	—
add.class.path	○	○	—	—	—	—
batch.ctm.enabled	○	○	○	○	○	○
batch.log.directory	○	○	○	○	○	○
batch.log.lockInterval	○	○	○	○	○	○
batch.log.lockRetryCount	○	○	○	○	○	○
batch.log.maintenance.filenum	○	○	○	○	○	○
batch.log.maintenance.filesize	○	○	○	○	○	○
batch.log.message.filenum	○	○	○	○	○	○
batch.log.message.filesize	○	○	○	○	○	○

オプション名	cjexecjob		cjkiljob		cjlistjob	
	スケジューリング機能未使用	スケジューリング機能使用	スケジューリング機能未使用	スケジューリング機能使用	スケジューリング機能未使用	スケジューリング機能使用
batch.log.stdout.enabled	○	○	—	—	—	—
batch.request.timeout	—	○	—	○	—	○
batch.schedule.group.name	—	○	—	○	—	○
batch.vbroker.agent.addr	—	○	—	○	—	○
batch.vbroker.agent.port	—	○	—	○	—	○

(凡例)

○：オプションが有効になります。

—：オプションが無効になります。

(5) 指定できるキー

指定できるキーとデフォルト値を次の表に示します。

注意事項

ファイル内でマルチバイト文字を使用する場合は、次の個所で使用しているマルチバイト文字とエンコードを統一してください。

- usrconf.cfg (バッチサーバ用オプション定義ファイル)
- cjexecjob コマンド (バッチアプリケーション実行コマンド) の引数
- バッチアプリケーションのソースコード (java.lang.System.out または java.lang.System.err)

表 3-3 usrconf.cfg に指定できるキーとデフォルト値 (バッチアプリケーション)

キー名称	内容	デフォルト値
add.jvm.arg	指定された値をバッチサーバ上のバッチアプリケーションに渡します。次のオプションを指定できます。 • -D<property> 次に示すプロパティは指定できません。 • java.endorsed.dirs • java.security.policy • java.class.path • java.library.path なお、システムプロパティを指定したい場合は、"-D"を利用しないで、usrconf.properties ファイルに指定します。	なし
add.class.path	設定されている値をバッチサーバ上にあるバッチアプリケーションクラスローダのクラスパスに追加します。このキーに指定できる	なし

キー名称	内容	デフォルト値
	JavaVM のオプションは 1 個だけです。複数指定する場合は、新たに add.class.path を追加します。 なお、スペースを含むパスを指定する場合、パスをダブルクォーテーション (") で囲む必要はありません。また、ダブルクォーテーションでの指定は有効となりません。 コンテナ拡張ライブラリ用の JAR を指定する場合はバッチサーバ用オプション定義ファイルで add.class.path キーを指定してください。	
batch.ctm.enabled	スケジューリング機能を使用するかどうかを指定します。 true を指定した場合： スケジューリング機能を使用します。 false を指定した場合： スケジューリング機能を使用しません。	false
batch.log.directory	ログファイルの出力先ディレクトリを指定します。ログ出力先のパスを絶対パス、または相対パスで指定します。次の文字が指定できます。 - 半角英数字 - アンダースコア (_) - ハイフン (-) - パス区切り文字 180 バイトを超えるパス（相対パス指定の場合、カレントディレクトリまでの絶対パスと相対パスの合計）を指定した場合は、KDJE40059-W のメッセージが出力され、デフォルト値が使用されます。デフォルト値が 180 バイトを超える場合は、ログの初期化に失敗し、異常終了します。ディレクトリにアクセス権がない場合や、ファイル名を指定した場合は、KDJE55039-W のメッセージが出力され、コマンドが終了します。 Windows の場合、UNC 名を含むパスは指定できません。UNIX の場合、nfs マウントされたディスク上へのパスは指定できません。	<Application Server インストールディレクトリ>¥CC¥batch¥logs
batch.log.lockInterval	ログファイルの排他処理に失敗したときのリトライ間隔（単位：ミリ秒）を 0～2147483647 の整数値で指定します。0 を指定した場合は即時にエラーとなります。 バッチアプリケーションで使用するコマンドが KDJE55042-E メッセージを出力して終了する場合に使用されます。 Windows の場合、batch.lockRetryCount の値と batch.log.lockInterval の値を掛けた値が 0～2147483647 の範囲になるように指定します。 Windows の場合、排他処理の待ち時間として、batch.log.lockRetryCount の値と batch.log.lockInterval の値を掛けた値が使用されます。	10
batch.log.lockRetryCount	ログファイルの排他処理に失敗したときのリトライ回数（単位：回）を 1～2147483647 の整数値で指定します。 バッチアプリケーションで使用するコマンドが KDJE55042-E メッセージを出力して終了する場合に使用されます。	1000

キー名称	内容	デフォルト値
batch.log.maintenance. .filenum	バッチアプリケーションで使用するコマンドの保存用ログファイルの面数を指定します。 1～64 の整数値で指定します。面数を変更する場合には、該当するログファイルに出力しているバッチアプリケーションで使用するコマンド（cjexecjob, ckilljob, clistjob）をすべて停止し、ログファイルと mmap ディレクトリ以下のログ管理ファイルを別のディレクトリへ移動、または削除してください。	2
batch.log.maintenance. .filesize	バッチアプリケーションで使用するコマンドの保守用ログファイルのファイルサイズ（単位：バイト）を指定します。 4096～16777216 の整数値で指定します。サイズを変更する場合には、該当するログファイルに出力しているバッチアプリケーションで使用するコマンドをすべて停止し、ログファイルと mmap ディレクトリ以下のログ管理ファイルを別のディレクトリへ移動、または削除してください。	1048576
batch.log.message.file num	バッチアプリケーションで使用するコマンドのメッセージログファイルの面数を指定します。 1～64 の整数値で指定します。面数を変更する場合には、該当するログファイルに出力しているバッチアプリケーションで使用するコマンドをすべて停止し、ログファイルと mmap ディレクトリ以下のログ管理ファイルを別のディレクトリへ移動、または削除してください。	2
batch.log.message.files ize	バッチアプリケーションで使用するコマンドのメッセージログファイルのファイルサイズ（単位：バイト）を指定します。 4096～16777216 の整数値で指定します。サイズを変更する場合には、該当するログファイルに出力しているバッチアプリケーションで使用するコマンドをすべて停止し、ログファイルと mmap ディレクトリ以下のログ管理ファイルを別のディレクトリへ移動、または削除してください。	1048576
batch.log.stdout.enabl ed	バッチ実行コマンドの標準出力に、バッチジョブ投入コマンドが出すメッセージを出力するかを指定します。 true を指定した場合： メッセージを標準出力に出力します。 false を指定した場合： メッセージを標準出力に出力しません。	true
batch.request.timeout	バッチ実行コマンドとバッチサーバ間の通信タイムアウト時間を 0～2147483647（秒単位）で指定します。0 を指定すると無限に待ち続けます。	0
batch.schedule.group. name	バッチサーバのスケジュールグループ名を英数字、およびアンダースコア「_」を使った 1～63 文字の文字列で指定します。	JOBGROUP
batch.vbroker.agent.ad dr	OSAgent が使用している IP アドレスを指定します。複数の IP アドレスが振り分けられている環境で指定します。	なし
batch.vbroker.agent.po rt	OSAgent に接続するためのポート番号を 1～65535 の整数で指定します。	14000

3. バッチサーバで使用するファイル

(6) 注意事項

- add.jvm.arg キーで--add-modules および--module-path オプションを指定しないでください。指定した場合は、動作を保証しません。

3.2.6 usrconf.properties (バッチアプリケーション用ユーザプロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

環境変数 CJBATCHUSRCONFDIR に usrconf.properties (バッチアプリケーション用ユーザプロパティファイル) の格納先のパスを設定するか、cjexecjob コマンド (バッチアプリケーション実行コマンド) を実行するディレクトリに usrconf.properties (バッチアプリケーション用ユーザプロパティファイル) を格納してください。usrconf.cfg (バッチアプリケーション用オプション定義ファイル) は、cjexecjob コマンド (バッチアプリケーション実行コマンド) 実行時に次の順で検索されます。

1. 環境変数 (CJBATCHUSRCONFDIR) に設定したディレクトリ
2. cjexecjob コマンド (バッチアプリケーション実行コマンド) を実行したディレクトリ

(3) 機能

cjexecjob コマンド、および ckilljob コマンドを実行する JavaVM のシステムプロパティを指定します。

(4) サーバ管理コマンドの予約済みキー

cjexecjob コマンド、および ckilljob コマンドでは、次に示す接頭子で始まるキーを内部的に利用します。このため、アプリケーションでこれらの接頭子で始まるキーを利用してはいけません。

- ejbserver.*
- webserver.*
- appclient.*
- j2eeserver.*

(5) cjexecjob コマンド、および ckilljob コマンドのカスタマイズ用キー

cjexecjob コマンド、および ckilljob コマンドでは、次に示すシステムプロパティのキーに値を設定することで、バッチサーバの動作をカスタマイズできます。

キーの概要とデフォルト値を次の表に示します。キーの詳細については、「[2.2.3 usrconf.properties \(J2EE サーバ用ユーザプロパティファイル\)](#)」を参照してください。なお、参照する場合は「J2EE サーバ」および「J2EE アプリケーション」を「バッチサーバ」および「バッチアプリケーション」と読み替えてください。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	概要	デフォルト値	関連情報
ejbserver.client.ctm.RequestPriority	CTM 内のキューに滞留したリクエストを引き出す時のプライオリティ（優先順位）を指定します。	4	
ejbserver.container.rebindpolicy	EJB メソッドの呼び出し時に通信障害が発生した場合、EJB クライアント側での接続の再接続動作とリクエストの再送動作を指定します。	VB_TRANSPARENT	「機能解説 基本・開発編(EJB コンテナ)」の「2.13 EJB のリモートインタフェースの呼び出し」
ejbserver.jndi.request.timeout	ネーミングサービスとの通信タイムアウト時間を指定します。	0	
java.naming.provider.url ^{※1}	バッチアプリケーションが利用する CORBA ネーミングサービスのホスト名称とポート番号を指定します。 なお、旧バージョンで使用されていたプロトコル（iioploc または iiopname）と互換性があります。 (指定形式) corbaname::[ホスト名称]:[ポート番号]	バッチアプリケーションが実行されるバッチサーバが利用する、CORBA ネーミングサービスと通信する値。	
ejbserver.rmi.request.timeout	クライアントとサーバ間の通信タイムアウト時間を指定します。	0	

(凡例)

空欄：関連情報はありません。

注※1 このキーは、バッチアプリケーション用ユーザプロパティファイル特有のキーです。J2EE サーバ用ユーザプロパティファイルには存在しません。

4

Smart Composer 機能で使用するファイル

この章では、Smart Composer 機能で使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

4.1 Smart Composer 機能で使用するファイルの一覧

Smart Composer 機能で使用するファイルについて説明します。使用するファイルは J2EE アプリケーションを実行するシステムとバッチアプリケーションを実行するシステムで異なります。システムごとに使用するファイルについて説明します。

4.1.1 J2EE アプリケーションを実行するシステムで使用するファイル

J2EE アプリケーションを実行するシステムで使用するファイルの一覧を次の表に示します。

表 4-1 J2EE アプリケーションを実行するシステムで使用するファイルの一覧

分類	ファイル名称	概要
コマンドの動作環境設定ファイル	cmxserver.properties (サーバ設定プロパティファイル)	Smart Composer 機能のコマンドの実行環境を設定します。
	.cmxrc (クライアント設定プロパティファイル)	Smart Composer 機能のコマンドの共通引数のデフォルト値を設定できます。 クライアントごとに、異なるデフォルト値を設定したい場合に使用します。
	cmxclient.properties (クライアント共通設定プロパティファイル)	Smart Composer 機能のコマンドの共通引数のデフォルト値を設定できます。 すべてのクライアントに、共通のデフォルト値を設定したい場合に使用します。
	lb.properties (負荷分散機定義プロパティファイル)	負荷分散機へのアクセスに必要な接続情報を設定します。
Web システム構築用定義ファイル	簡易構築定義ファイル	Smart Composer 機能のコマンドで構築する Web システムを定義します。
	構成変更定義ファイル	Smart Composer 機能のコマンドで構築した Web システムの構成変更内容を定義します。
	論理サーバ参照定義ファイル※	Smart Composer 機能のコマンドで構築した Web システム中のサービスユニットに追加する、論理ユーザサーバを定義します。
	スケールアウト用ホスト定義ファイル	Smart Composer 機能のコマンドで構築した、ホスト単位管理モデルの Web システムをスケールアウトする場合に、複製先の Web システムの構成内容を定義します。

注※ 論理サーバ参照定義ファイルは旧バージョンとの互換用ファイルです。論理ユーザサーバを設定する場合は、簡易構築定義ファイルを使用してください。

4.1.2 バッチアプリケーションを実行するシステムで使用するファイル

バッチアプリケーションを実行するシステムで使用するファイルの一覧を次の表に示します。

表 4-2 バッチアプリケーションを実行するシステムで使用するファイルの一覧

分類	ファイル名称	概要
コマンドの動作環境設定 ファイル	cmxserver.properties (サーバ設定プロパティ ファイル)	Smart Composer 機能のコマンドの実行環境を 設定します。
	.cmxrc (クライアント設定プロパティファイル)	Smart Composer 機能のコマンドの共通引数の デフォルト値を設定できます。 クライアントごとに、異なるデフォルト値を設 定したい場合に使用します。
	cmxclient.properties (クライアント共通設定 プロパティファイル)	Smart Composer 機能のコマンドの共通引数の デフォルト値を設定できます。 すべてのクライアントに、共通のデフォルト値 を設定したい場合に使用します。
Web システム構築用定義 ファイル	簡易構築定義ファイル	Smart Composer 機能のコマンドで構築する Web システムを定義します。
	構成変更定義ファイル	Smart Composer 機能のコマンドで構築した Web システムの構成変更内容を定義します。
	スケールアウト用ホスト定義ファイル	Smart Composer 機能のコマンドで構築した、 ホスト単位管理モデルの Web システムをスケ ールアウトする場合に、複製先の Web システム の構成内容を定義します。

4.2 Smart Composer 機能のコマンドの動作環境設定ファイル

Windows 環境で JDK21 以降の場合、Smart Composer で使用するファイルにマルチバイト文字は使用できません。

4.2.1 cmxserver.properties（サーバ設定プロパティファイル）

(1) 形式

Java プロパティファイル形式です。

Smart Composer 機能のサーバ設定プロパティファイルです。

(2) ファイルの格納先

- Windows の場合
＜Application Serverのインストールディレクトリ＞¥manager¥config¥
- UNIX の場合
/opt/Cosminexus/manager/config/

(3) 機能

Smart Composer 機能のコマンドの実行環境を設定します。

(4) 設定できるキー

指定できるキーとデフォルト値を次に示します。

キー名称	内容	デフォルト
cmx.max_wait_time	共通オプション-wait を指定して実行したコマンドに有効で、すでに実行しているコマンドの完了を待つときの最大待ち時間を、1～2147483647 の整数値（単位：秒）で指定します。ここで設定した時間を超えても、すでに実行しているコマンドが完了しない場合は、あとから実行したコマンドはエラーリターンします。	180

(5) 記述例

```
cmx.max_wait_time=360
```

(6) 注意事項

- パラメタが存在しない、または誤っている場合はデフォルト値を仮定します。
- Management Server 起動中に設定ファイルを更新した場合、Management Server の再起動後に更新情報が反映されます。

4.2.2 .cmxrc (クライアント設定プロパティファイル)

(1) 形式

Java プロパティ形式です。

(2) ファイルの格納先

コマンドを実行する OS ユーザのホームディレクトリに格納します。

(3) 機能

Smart Composer 機能で提供するコマンドの共通引数のデフォルト値を設定します。このファイルはクライアントごとに個別にデフォルト値を設定する場合に使用します。

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config%templates%.cmxrc
- UNIX の場合
 /opt/Cosminexus/manager/config/templates/.cmxrc

(4) 設定できるキー

指定できるキーについて次に示します。なお「省略値」とは、パラメタの指定がない場合に仮定される値です。

「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
cmx.connect.host ^{※1}	共通引数の-m オプション省略時に仮定するホスト名、および IP アドレスを設定します。 IP アドレスは、メタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で指定された場合、利用できるローカル IP アドレスのうちマッチする IP アドレスを採用します。 ^{※2}	指定できる値を次に示します。 - ホスト名 - IPv4 アドレス	なし	06-50

キー名称	内容	指定可能値	省略値	VR
cmx.connect.userid	共通引数の-u オプション省略時に仮定するユーザ ID を設定します。 管理ユーザアカウントの省略機能が有効の場合、このキーの指定は無視されます。	Management Server の管理ユーザアカウントのユーザ ID	なし	06-50
cmx.connect.passwd	共通引数の-p オプション省略時に仮定するパスワードを設定します。 管理ユーザアカウントの省略機能が有効の場合、このキーの指定は無視されます。	Management Server の管理ユーザアカウントのパスワード	なし	06-50
cmx.wait	共通引数の-wait オプションをデフォルトにする場合は、true を、-nowait オプションをデフォルトにする場合は、false を設定します。	指定できる文字列を次に示します。 • true • false	false	06-50
cmx.verbose	共通引数の-verbose オプションをデフォルトにする場合は、true を、-noverbose オプションをデフォルトにする場合は、false を設定します。	指定できる文字列を次に示します。 • true • false	false	06-50
cmx.websystem.name	次に示すコマンドの-s オプション省略時に、Web システム名を設定します。 • cmx_build_system ^{※3} • cmx_delete_system • cmx_list_model ^{※5} • cmx_list_status • cmx_resume_lb ^{※4} • cmx_start_target • cmx_stop_target • cmx_test_lb ^{※4}	Web システム名	なし	07-50
cmx.lb ^{※4}	cmx_delete_system コマンドの-lb オプション省略時に、負荷分散機の仮想サーバに対する動作を設定します。 • keepvs：仮想サーバを削除しません。 • deletevs：仮想サーバを削除します。	指定できる文字列を次に示します。 • keepvs • deletevs	なし	07-10
cmx.encoding	cmx_export_model コマンドの-encoding オプション省略時に仮定する文字エンコーディングを指定します。 指定できる値については、マニュアル「アプリケーションサーバ リファレンス コマンド編」の「cmx_export_model（簡易構築定義ファイルの出力）」を参照してください。	文字エンコーディング名	UTF-8	07-50
cmx.indent	cmx_export_model コマンドの-indent オプション省略時に仮定するインデントサイズを指定します。 指定できる値については、マニュアル「アプリケーションサーバ リファレンス コマンド編」の	0～8 の整数	2	07-50

キー名称	内容	指定可能値	省略値	VR
	「cmx_export_model (簡易構築定義ファイルの出力)」を参照してください。			
cmx.myhost	cmx_export_model コマンドの-myhost オプション省略時に仮定する運用管理サーバマシンのホスト名を指定します。 指定できる値については、マニュアル「アプリケーションサーバリファレンス コマンド編」の「cmx_export_model (簡易構築定義ファイルの出力)」を参照してください。	ホスト名	なし	07-50

注※1

次の場合に mserver.properties の webserver.connector.http.bind_host に値を指定しているときは、cmx.connect.host キーにも同じ値を指定してください。

- J2EE アプリケーションを実行するシステムをホスト単位管理モデルで運用する場合
- バッチアプリケーションを実行するシステムの場合

注※2

運用管理ドメイン内で運用管理エージェントとの通信に使用される IP アドレスのサブネットが一つに決められている場合、指定する IP アドレスを"192¥¥.168¥¥.0¥¥..+"のように記述しておけば、前方が"192.168.0."である IP アドレス ("192.168.0.32"や"192.168.0.128"など) にマッチするため、設定ファイルを修正しないで、すべてのホストに配布して利用できます。正規表現についての詳細は、Java の java.util.regex.Pattern クラスの仕様を確認してください。

指定した正規表現にマッチする IP アドレスが複数個検出された場合は、最も小さい値の IP アドレスを採用します。例えば、"192.168.0.32"と"192.168.0.128"が検出された場合は"192.168.0.32"を採用します。この場合、採用された IP アドレスが意図する IP アドレスになるとは限らないため、"192¥¥.168¥¥.0¥¥.1.."のように必ず 1 個だけマッチする正規表現を記述してください。

IP アドレスを正規表現で指定する場合は、次の点に注意してください。

- ¥を指定する場合は 2 文字連続して (¥¥) 指定してください。¥¥は 1 文字の¥に置き換えられます。
- コロン (:) を使用しないでください。コロン以降はポート番号とみなされます。

mserver.properteis の webserver.connector.http.bind_host に値を指定しているときは、同じ値を指定してください。

注※3

cmx_build_system コマンド実行時に、-f オプションを指定している場合は、キーの設定は無効になります。

注※4

バッチアプリケーションを実行するシステムでは使用できません。

注※5

cmx.websystem.name キーを設定している場合に、cmx_list_model コマンドを使用して、登録されているすべての Web システムの情報モデルを取得するときは、-all オプションを指定してください。

(5) 記述例

```
cmx.connect.host=localhost:28080
cmx.connect.userid=admin
cmx.connect.passwd=p1574809
cmx.wait=true
#cmx.verbose=false
cmx.websystem.name=MyWebSystem
cmx.lb=deletevs
```

(6) 注意事項

- このファイルには、ユーザ ID やパスワードなどを記載するため、適切にファイルのアクセス権を設定してください。
- ASCII 文字以外の文字を記述する場合は、設定ファイルの文字エンコーディングと Smart Composer 機能のコマンド実行時の OS ロケールの文字エンコーディングを合わせてください。また、日本語を使用する場合に Unicode エンコードされた文字 (Unicode-encoded character) を含めた文字列で指定しないでください。
- 各パラメタが存在しない、または誤っている場合はデフォルト値を仮定します。
- 設定値 (true, false) は、大文字、小文字の区別はしません。

4.2.3 cmxclient.properties (クライアント共通設定プロパティファイル)

(1) 形式

Java プロパティ形式です。

(2) ファイルの格納先

- Windows の場合
＜Application Serverのインストールディレクトリ＞¥manager¥config¥
- UNIX の場合
/opt/Cosminexus/manager/config/

(3) 機能

Smart Composer 機能で提供するコマンドの共通引数のデフォルト値を設定します。このファイルはすべてのクライアントで共通のデフォルト値を設定する場合に使用します。

(4) 設定できるキー

指定できるキーについて次に示します。なお「省略値」とは、パラメタの指定がない場合に仮定される値です。

「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
cmx.apply_user	このファイルを適用するクライアントの OS のユーザ ID を設定します。ユーザ ID はコンマ「,」で区切って指定します。また、Windows のサービスから Smart Composer 機能のコマンドが呼び出される場合に、この定義ファイルを適用するには、「<ホスト名>\$」（<ホスト名>は、Management Server が稼働するホストのホスト名）を指定してください。	OS に登録されているユーザ名	なし	06-70
cmx.connect.host ^{※1}	共通引数の-m オプション省略時に仮定するホスト名、および IP アドレスを設定します。 IP アドレスは、メタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で指定された場合、利用できるローカル IP アドレスのうちマッチする IP アドレスを採用します。 ^{※2}	指定できる値を次に示します。 • ホスト名 • IPv4 アドレス	なし	06-50
cmx.connect.userid	共通引数の-u オプション省略時に仮定するユーザ ID を設定します。 管理ユーザアカウントの省略機能が有効の場合、このプロパティの指定は無視されます。	Management Server の管理ユーザアカウントのユーザ ID	なし	06-50
cmx.connect.passwd	共通引数の-p オプション省略時に仮定するパスワードを設定します。 管理ユーザアカウントの省略機能が有効の場合、このプロパティの指定は無視されます。	Management Server の管理ユーザアカウントのパスワード	なし	06-50
cmx.wait	共通引数の-wait オプションをデフォルトにする場合は、true を、-nowait オプションをデフォルトにする場合は、false を設定します。	指定できる文字列を次に示します。 • true • false	false	06-50
cmx.verbose	共通引数の-verbose オプションをデフォルトにする場合は、true を、-noverbose オプションをデフォルトにする場合は、false を設定します。	指定できる文字列を次に示します。 • true • false	false	06-50
cmx.websystem.name	次に示すコマンドの-s オプション省略時に、Web システム名を設定します。 • cmx_build_system^{※3} • cmx_delete_system • cmx_list_model^{※5} • cmx_list_status • cmx_resume_lb^{※4}	Web システム名	なし	07-50

キー名称	内容	指定可能値	省略値	VR
	• cmx_start_target • cmx_stop_target • cmx_test_lb※4			
cmx.lb※4	cmx_delete_system コマンドの-lb オプション省略時に、負荷分散機の仮想サーバに対する動作を設定します。 • keepvs：仮想サーバを削除しません。 • deletevs：仮想サーバを削除します。	指定できる文字列を次に示します。 • keepvs • deletevs	なし	07-10
cmx.encoding	cmx_export_model コマンドの-encoding オプション省略時に仮定する文字エンコーディングを指定します。指定できる値については、マニュアル「アプリケーションサーバ リファレンス コマンド編」の「cmx_export_model（簡易構築定義ファイルの出力）」を参照してください。	文字エンコーディング名	UTF-8	07-50
cmx.indent	cmx_export_model コマンドの-indent オプション省略時に仮定するインデントサイズを指定します。指定できる値については、マニュアル「アプリケーションサーバ リファレンス コマンド編」の「cmx_export_model（簡易構築定義ファイルの出力）」を参照してください。	0～8 の整数	2	07-50
cmx.myhost	cmx_export_model コマンドの-myhost オプション省略時に仮定する運用管理サーバマシンのホスト名を指定します。指定できる値については、マニュアル「アプリケーションサーバ リファレンス コマンド編」の「cmx_export_model（簡易構築定義ファイルの出力）」を参照してください。	ホスト名	なし	07-50

注※1

J2EE アプリケーションを実行するシステムをホスト単位管理モデルで運用する場合、またはバッチアプリケーションを実行するシステムの場合、次の点に注意してください。

mserver.properties の webserver.connector.http.bind_host に値を指定しているときは、cmx.connect.host キーにも同じ値を指定してください。

注※2

運用管理ドメイン内で運用管理エージェントとの通信に使用される IP アドレスのサブネットが一つに決められている場合、指定する IP アドレスを"192.168.0.32"のように記述しておけば、前方が"192.168.0."である IP アドレス（"192.168.0.32"や"192.168.0.128"など）にマッチするため、設定ファイルを修正しないで、すべてのホストに配布して利用できます。正規表現についての詳細は、Java の java.util.regex.Pattern クラスの仕様を確認してください。

指定した正規表現にマッチする IP アドレスが複数個検出された場合は、最も小さい値の IP アドレスを採用します。例えば、"192.168.0.32"と"192.168.0.128"が検出された場合は"192.168.0.32"を採用し

ます。この場合、採用された IP アドレスが意図する IP アドレスになるとは限らないため、"192¥¥.168¥¥.0¥¥.1.."のように必ず 1 個だけマッチする正規表現を記述してください。

IP アドレスを正規表現で指定する場合は、次の点に注意してください。

- ¥を指定する場合は 2 文字連続して (¥¥) 指定してください。¥¥は 1 文字の¥に置き換えられます。
- コロン (:) を使用しないでください。コロン以降はポート番号とみなされます。

mserver.properteis の webserver.connector.http.bind_host に値を指定しているときは、同じ値を指定してください。

注※3

cmx_build_system コマンド実行時に、-f オプションを指定している場合は、キーの設定は無効になります。

注※4

バッチアプリケーションを実行するシステムでは使用できません。

注※5

cmx.websystem.name キーを設定している場合に、cmx_list_model コマンドを使用して、登録されているすべての Web システムの情報モデルを取得するときは、-all オプションを指定してください。

(5) 記述例

```
cmx.apply_user=Administrator,muser1
cmx.connect.host=localhost:28080
cmx.connect.userid=admin
cmx.connect.passwd=p1574809
cmx.wait=true
#cmx.verbose=false
cmx.websystem.name=MyWebSystem
cmx.lb=deletevs
```

(6) 注意事項

- このファイルには、ユーザ ID やパスワードなどを記載するため、適切にファイルのアクセス権を設定してください。
- ASCII 文字以外の文字を記述する場合は、設定ファイルの文字エンコーディングと Smart Composer 機能のコマンド実行時の OS ロケールの文字エンコーディングを合わせてください。また、日本語を使用する場合に Unicode エンコードされた文字 (Unicode-encoded character) を含めた文字列で指定しないでください。
- 各パラメタが存在しない、または誤っている場合はデフォルト値を仮定します。
- 設定値 (true, false) は、大文字、小文字の区別はされません。
- cmx.apply_user キーに指定する内容は、UNIX の場合、大文字、小文字が区別されます。Windows の場合は、大文字と小文字の区別はされません。

- .cmxrc ファイルと cmxclient.properties ファイルの両方を使用している場合は、.cmxrc ファイルが適用されます。cmxclient.properties ファイルは読み込まれません。
- .cmxrc ファイルを使用していないクライアントで、cmxclient.properties ファイルにアクセス権のないクライアントは、cmxclient.properties ファイルの読み込み時にエラーが発生します。このため、Smart Composer 機能のコマンドを使用するクライアントには、cmx.apply_user キーにユーザ ID を設定して、cmxclient.properties ファイルへのアクセス権を与えてください。

4.2.4 lb.properties（負荷分散機定義プロパティファイル）

(1) 形式

Java プロパティ形式です。

(2) ファイルの格納先

- Windows の場合
`<Application Serverのインストールディレクトリ>%manager%config`
- UNIX の場合
`/opt/Cosminexus/manager/config`

(3) 機能

負荷分散機へのアクセスに必要な接続情報を設定します。Application Server から負荷分散機を制御する場合に使用します。

(4) 設定できるキー

設定できるキーとデフォルト値を次に示します。なお、負荷分散機の制御方法によって、設定できるキーが異なるので注意してください。API を使用した直接接続を使用する負荷分散機で設定できるキーについては、マニュアル「アプリケーションサーバ 機能解説 セキュリティ管理機能編」の「18.2.1 lb.properties（負荷分散機定義プロパティファイル）」を参照してください。

キー名称	内容	デフォルト
lb.list	負荷分散機の管理 IP アドレスをドット記法（xxx.xxx.xxx.xxx）で設定します。xxx には 0～255 の整数を指定します。なお、複数の負荷分散機を使用する場合は、コンマ「,」で区切り、複数の管理 IP アドレスを設定します。	なし
lb.connect_type.<IP アドレス>※1※2	負荷分散機への接続形態を設定します。 • Telnet を直接使用する場合 「telnet」を指定してください。 • ssh を直接使用する場合	jpl_nc

キー名称	内容	デフォルト
	「ssh」を指定してください。 何も指定しない場合、jpl_nc(旧 VR 互換)が指定されます。	
lb.enable_passwd.<IP アドレス>	負荷分散機に設定した Privileged EXEC レベルに必要なパスワードを負荷分散機ごとに設定します。 なお、このプロパティは ACOS の場合に設定します。	なし
lb.ssh.port.<IP アドレス>*1	負荷分散機の ssh サーバのポート番号を設定します。指定できる値は 1～65535 の整数です。	22
lb.ssh.user.<IP アドレス>*1	負荷分散機に ssh で接続するときのユーザ名を設定します。	なし
lb.ssh.passwd.<IP アドレス>*1	負荷分散機に ssh で接続するときのユーザのパスワードを設定します。	なし
lb.ssh.timeout.<IP アドレス>*1*2	負荷分散機への ssh コマンド実行（接続処理）時のタイムアウト時間（単位：秒）を設定します。 このキーで設定した時間内に、負荷分散機への接続処理が完了しない場合は、Smart Composer 機能のコマンド（cmx_build_system など）がタイムアウトエラーで異常終了します。 なお、設定できる値は 1～2147483 の整数です。	10
lb.telnet.command_timeout.<IP アドレス>	負荷分散機へのコマンド送信時のタイムアウト時間（単位：秒）を設定します。 このキーで設定した時間内に、負荷分散機に対して発行した CLI コマンドが完了しない場合は、Smart Composer 機能のコマンド（cmx_build_system など）がタイムアウトエラーで異常終了します。 なお、設定できる値は 1～2147483 の整数です。	10
lb.telnet.login_prompt.<IP アドレス>	負荷分散機に接続するときのログイン完了後プロンプトに表示される文字列を設定します。 最後に空白を指定したい場合、” ” で囲んで指定します。	なし
lb.telnet.login_timeout.<IP アドレス>	負荷分散機に接続するときのログイン処理時のタイムアウト時間（単位：秒）を設定します。 このキーで設定した時間内に、負荷分散機への接続処理が完了しない場合は、Smart Composer 機能のコマンド（cmx_build_system など）がタイムアウトエラーで異常終了します。 なお、設定できる値は 1～2147483 の整数です。	10
lb.telnet.passwd.<IP アドレス>	負荷分散機に Telnet でログインするときのユーザのパスワードを設定します。	なし
lb.telnet.passwd_prompt.<IP アドレス>	負荷分散機に接続するときのパスワード入力プロンプトに表示される文字列を設定します。 最後に空白を指定したい場合、” ” で囲んで指定します。	なし
lb.telnet.port.<IP アドレス>	負荷分散機のポート番号を設定します。指定できる値は 1～65535 の整数です。	23
lb.telnet.user.<IP アドレス>	負荷分散機に Telnet でログインするときのユーザ名を設定します。	なし

キー名称	内容	デフォルト
lb.telnet.user_prompt.<IP アドレス>	負荷分散機に接続するときのユーザ名入力プロンプトに表示される文字列を設定します。 最後に空白を指定したい場合, ” ” で囲んで指定します。	なし

注※1

lb.list に指定した負荷分散機の管理 IP アドレスを指定します。

注※2

指定できる値以外の値が指定された場合は, cmx_test_lb コマンド, cmx_build_system コマンド, cmx_delete_system コマンド, cmx_start_target コマンド, および cmx_stop_target コマンドを実行したときにエラーになるので注意してください。

(5) 記述例

```
lb.list=192.168.1.100
lb.enable_passwd.192.168.1.100=adminpw

lb.connect_type.192.168.1.100=telnet
#lb.telnet.port.192.168.1.100=23
lb.telnet.user.192.168.1.100=user01
lb.telnet.passwd.192.168.1.100=user01pw
lb.telnet.user_prompt.192.168.1.100="AX login: "
lb.telnet.passwd_prompt.192.168.1.100="Password: "
lb.telnet.login_prompt.192.168.1.100=>
#lb.telnet.login_timeout.192.168.1.100=10
#lb.telnet.command_timeout.192.168.1.100=10
```

(6) 注意事項

- このファイルには, パスワードなどの情報が含まれているため, 適切にファイルのアクセス権を設定してください。
- 各パラメタが存在しない, または誤っている場合はデフォルト値を仮定します。
- Management Server 起動中に設定ファイルを更新した場合, Management Server の再起動後に更新情報が反映されます。
- このファイルを更新した場合, または Management Server マシンと負荷分散機との接続構成を変更した場合は, cmx_test_lb コマンドを使用して, 負荷分散機への接続を確認してください。cmx_test_lb コマンドの詳細については, マニュアル「アプリケーションサーバリファレンス コマンド編」の「8. Smart Composer 機能で使用するコマンド」を参照してください。負荷分散機へ接続できない場合は, cmx_test_lb コマンドが出力したメッセージを基に, 負荷分散機の設定および簡易構築定義ファイルの負荷分散機の定義 (<load-balancer>タグの定義) の設定内容を確認してください。負荷分散機の設定および簡易構築定義ファイルの負荷分散機の定義 (<load-balancer>タグの定義) については, マニュアル「アプリケーションサーバシステム構築・運用ガイド」の「4.7.5 負荷分散機へ接続する環境を設定する」を参照してください。

4.3 簡易構築定義ファイル

簡易構築定義ファイルについて説明します。

4.3.1 概要

簡易構築定義ファイルには、Web システムの属性、負荷分散機、物理ティア、サービスユニット、およびサービスユニットを構成するホストを定義します。Web システムの属性、物理ティア、サービスユニット、およびホストは、定義ファイル内に必要な数だけ定義し、負荷分散機は Web システム内に一つ定義します。

ただし、アプリケーションサーバから負荷分散機を制御する場合（Smart Composer 機能で負荷分散機の設定、および実サーバポートの開始・閉塞をしない場合）は、負荷分散機の定義を記述する必要はありません。また、バッチアプリケーションを実行するシステムの場合、負荷分散機を使用しないため、負荷分散機の定義を記述する必要はありません。

なお、次のテンプレートファイルをコピーして利用してください。

- 互換モードを使用する場合

```
<Application Server のインストールディレクトリ  
>%manager%config%templates%cmxdefcombinedmodel_V9.xml
```

- 互換モードを使用しない場合

```
<Application Server のインストールディレクトリ  
>%manager%config%templates%cmxdefcombinedmodel.xml
```

参考

テンプレートファイルの文字エンコーディングは、UTF-8 です。

注意事項

システムを構築したあとにシステムの構成変更をする場合は、構成変更定義ファイルを使用してください。簡易構築定義ファイルの編集で、システムの構成変更はできません。

システムを構成変更する方法は、サービスユニット・ホスト追加用の構成変更定義ファイルを利用した `cmx_change_model` コマンドの実行だけです。

(1) 形式

xml 形式です。

(2) 格納先

任意の作業場所に格納してください。

4.3.2 簡易構築定義ファイルの指定内容

簡易構築定義ファイルの指定内容で説明する項目と、その項目の意味を記します。

項目名	説明
タグ名	設定するタグを示します。
出現パターン	タグが出現する回数を示します。
内容	キーに定義する内容を示します。
指定可能値	タグに指定できる値を示します。指定できる値がない場合はハイフン「-」を示します。
タグの省略	定義形式で示すキーが省略できるかどうかを示します。
値の省略	定義形式で示すキーの値が省略されるかどうかを示します。
デフォルト	キーに指定する内容が省略されている場合に有効になる指定値を示します。デフォルト値がない場合は「なし」を示します。
バージョン	定義できるアプリケーションサーバのバージョンを示します。

タグの省略、値の省略および内容の詳細については、4.6.3 以降の簡易構築定義ファイルの各定義の詳細を参照してください。

簡易構築定義ファイルについて、次の表に示します。

タグ名	出現パターン	内容	指定可能値	省略値	VR
<web-system>	0 回以上	Web システムを定義します。 複数の Web システムを定義する場合は、<web-system>タグを複数指定します。	-	なし	07-00
ト <name>	1 回	必須定義。 Web システムを識別するための名称を指定します。	英数字または、アンダースコア「_」およびハイフン「-」で指定した 32 文字以内の文字列を指定します。	なし	06-50

タグ名		出現 パ ター ン	内容	指定可能値	省略値	VR
	<display-name>	0 ま たは 1 回	Web システムの表示名を指定 します。 指定する文字に制限はありま せん。	128 文字以内の 任意の文字列を 指定します。	なし	06-50
	<description>	0 ま たは 1 回	Web システムのコメントを指 定します。 指定する文字に制限はありま せん。	1024 文字以内 の任意の文字列 を指定します。	なし	06-50
	<load-balancer>	0 ま たは 1 回	負荷分散機を定義します。	—	なし	06-50
	<name>	0 ま たは 1 回	負荷分散機の名称を指定しま す。	英数字または、 アンダースコア 「_」 およびハイ フン「-」で指 定した 32 文字 以内の文字列を 指定します。	なし	06-50
	<load-balancer-type>	1 回	負荷分散機の種類を指定しま す。	指定できる文字 列を次に示しま す。 • BIG-IPv9 • BIG- IPv10.1 • BIG- IPv10.2 • BIG-IPv11 • ACOS	なし	06-50
	<display-name>	0 ま たは 1 回	負荷分散機の表示名を指定し ます。 指定する文字に制限はありま せん。	128 文字以内の 任意の文字列を 指定します。	なし	06-50
	<description>	0 ま たは 1 回	負荷分散機のコメントを指定 します。 指定する文字に制限はありま せん。	1024 文字以内 の任意の文字列 を指定します。	なし	06-50
	<cookie-switching>	0 ま たは 1 回	負荷分散機の Cookie スイッ チング機能を使用する場合に 指定します。	—	なし	06-51

タグ名				出現 パ ター ン	内容	指定可能値	省略値	VR
		ト	<cookie-switching-enabled>	0 ま たは 1 回	Cookie スイッチング機能を使用する場合に指定します。	指定できる文字列を次に示します。 • true • false	false	06-51
		ト	<cookie-name>	0 ま たは 1 回	cookie の名称を英数字またはアンダースコア「_」で指定します。 ただし、次の名前は Web コンテナが使用するため、大文字・小文字の区別なく使用できません。 • JSESSIONID	• BIG-IP の場合 64 文字以内の文字列を指定します。 なお、先頭 1 文字は大文字または小文字の英字を指定する必要があります。 • ACOS の場合 63 文字以内の文字列を指定します。	CMX_SERVER_ID	06-51
		└	<server-id-rule>	0 ま たは 1 回	08-53 以降は、このタグに値を指定しないでください。値を指定した場合、指定した値は無効になります。	—	なし	06-51
	ト		<management-host>	1 回	負荷分散機に設定した管理 IP アドレスまたはホスト名を指定します。	指定できる文字列を次に示します。 • ホスト名 • IPv4 アドレス	なし	07-50
	ト		<redundant-management-host>	1 回	負荷分散機を冗長化したシステムの場合に、2 台目の負荷分散機に設定した管理 IP アドレスまたはホスト名を指定します。	指定できる文字列を次に示します。 • ホスト名 • IPv4 アドレス	なし	07-50
	└		<virtual-server>	1 回	仮想サーバを定義します。	—	なし	06-50

タグ名			出現 パ ター ン	内容	指定可能値	省略値	VR
		ト <virtual-server-name>	1 回	仮想サーバ名を指定します。 複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで仮想サーバ名が重複しないように指定してください。	英数字またはアンダースコア「_」、ピリオド「.」およびハイフン「-」で指定した 31 文字以内の文字列を指定します。	なし	06-50
		ト <display-name>	0 または 1 回	仮想サーバの表示名を指定します。	128 文字以内の任意の文字列を指定します。	なし	06-50
		ト <description>	0 または 1 回	仮想サーバのコメントを指定します。	1024 文字以内の任意の文字列を指定します。	なし	06-50
		ト <ip-address>	1 回	仮想サーバの IP アドレスを指定します。 複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで仮想サーバの IP アドレスが重複しないように指定してください。	指定できる文字列を次に示します。 • ホスト名 • IPv4 アドレス	なし	06-50
		└ <http-port>	1 回	仮想サーバの http ポート番号を 1～65535 の間で指定します。	1～65535 の整数を指定します。	なし	06-50
ト		<tier>	1 回以上	必須定義。 物理ティアを定義します。	—	なし	06-50
	ト	<tier-type>	1 回	必須定義。 物理ティアの種類を指定します。	指定できる文字列を次に示します。 • combined-tier • http-tier • j2ee-tier • ctm-tier • free-tier	なし	06-50
	ト	<j2ee-server-count>	0 または 1 回	<tier-type>タグで ctm-tier を指定した場合に、1 ホストに	0～32 の整数を指定します。	1	07-10

タグ名			出現 パ ター ン	内容	指定可能値	省略値	VR
				配置する J2EE サーバの数を指定します。			
		└ <configuration>	0 回 以上	物理ティアに属するすべての論理サーバに適用するコンフィグレーションを、論理サーバの種類ごとに定義します。	—	なし	06-50
		└ <logical-server-type>	1 回	必須定義。 コンフィグレーションを定義する論理サーバの種類を指定します。	指定できる文字列を次に示します。 • web-server • j2ee-server • performance-tracer • ctm-domain-manager • component-transaction-monitor • smart-agent	なし	06-50
		└ <server-no>	0 または 1 回	物理ティア内の特定の J2EE サーバで、共通のコンフィグレーションを指定する場合に、J2EE サーバを識別するための番号を指定します。 物理ティア内で番号が重複しないように指定してください。	1～<j2ee-server-count>の指定値	なし	07-10
		└ <param>	1 回 以上	必須定義。 論理サーバの環境を設定するパラメタを、パラメタ名と値で定義します。	—	なし	06-50
		└└ <param-name>	1 回	パラメタ名を指定します。	指定できる値については 「4.8 システム	なし	06-50

タグ名					出現 パ ター ン	内容	指定可能値	省略値	VR	
							の構成パターンと定義する論理サーバ」の説明を参照してください。			
				└	<param-value>	1 回以上	パラメタの設定値を指定します。	指定できる値については「4.8 システムの構成パターンと定義する論理サーバ」の説明を参照してください。	なし	06-50
				└	<unit>	1 回以上	必須定義。 サービスユニットを定義します。	—	なし	06-50
				└	<name>	1 回	必須定義。 サービスユニットを識別するための名称を指定します。	英数字またはアンダースコア「_」、およびハイフン「-」で指定した 32 文字以内の文字列を指定します。	なし	06-50
				└	<display-name>	0 または 1 回	サービスユニットの表示名を指定します。指定する文字に制限はありません。	128 文字以内の任意の文字列を指定します。	なし	06-50
				└	<description>	0 または 1 回	サービスユニットのコメントを指定します。指定する文字に制限はありません。	1024 文字以内の任意の文字列を指定します。	なし	06-50
				└	<allocated-host>	1 回以上	必須定義。 サービスユニットを構成するホストのリファレンスを定義します。	—	なし	06-50
				└	<host-ref>	1 回	必須定義。 ホスト定義の<name>タグに指定したホスト名、構築済みの Web システムで定義されているホスト名、または @myhost を指定します。	指定できる値を次に示します。 • ホスト名 • IPv4 アドレス • @myhost	なし	06-50

タグ名				出現 パ ター ン	内容	指定可能値	省略値	VR
			ト 	1 回	必須定義。 <host-ref>タグで指定したホ ストが属する物理ティアの種 類を指定します。	指定できる文字 列を次に示しま す。 - combined- tier - http-tier - j2ee-tier - ctm-tier - free-tier	なし	06-50
			└	0 回 以上	<hosts-for>タグで指定した 物理ティアに属する論理サー バごとに、コンフィグレーショ ンを定義します。	—	なし	07-00
			ト 	0 ま たは 1 回	論理サーバまたはクラスタを 識別するための名称を指定し ます。	英数字またはアン ダースコア「_」、およびハ イフン「-」で 指定した 128 文字以内の文字 列を指定しま す。	cmx_< Web シ ステム名 >_< サービス ユニット 名>_< 種別 >_<通 番>	07-00
			ト 	0 ま たは 1 回	論理サーバの表示名を指定し ます。指定する文字に制限は ありません。	128 文字以内の 任意の文字列を 指定します。	なし	07-00
			ト 	0 ま たは 1 回	論理サーバのコメントを指定 します。指定する文字に制限 はありません。	1024 文字以内 の任意の文字列 を指定します。	なし	07-00
			ト 	1 回	必須定義。 コンフィグレーションを定義 する論理サーバの種類を指定 します。	指定できる文字 列を次に示しま す。 - web- server - j2ee-server - performan ce-tracer	なし	07-00

タグ名					出現 パ ター ン	内容	指定可能値	省略値	VR
							- ctm-domain-manager - component-transaction-monitor - smart-agent - naming-service - user-server		
			└	<cluster-ref>	0 または 1 回	論理サーバをクラスタのメンバにする場合に、クラスタ名を指定します。	英数字またはアンダースコア「_」、およびハイフン「-」で指定した 128 文字以内の文字列を指定します。	なし	07-50
			└	<server-no>	0 または 1 回	サービスユニット内の特定の J2EE サーバのコンフィグレーションを指定する場合に、J2EE サーバを識別するための番号を指定します。1 ホスト内で番号が重複しないように指定してください。	1～<j2ee-server-count>の指定値	なし	07-10
			└	<configuration>	0 または 1 回	論理サーバごとに適用するコンフィグレーションを定義します。	—	なし	07-00
			└	<param>	1 回以上	必須定義。 論理サーバの環境を設定するパラメタを、パラメタ名と値で定義します。	—	なし	07-00
			└	<param-name>	1 回	パラメタ名を指定します。	指定できる値については [4.8 システムの構成パターン	なし	07-00

タグ名							出現 パ ター ン	内容	指定可能値	省略値	VR
						 			と定義する論理サーバ」の説明を参照してください。		
						└ <param-value>	1 回 以上	パラメタの設定値を指定します。	指定できる値については「4.8 システムの構成パターンと定義する論理サーバ」の説明を参照してください。	なし	07-00
<host>							0 回 以上	ホストを定義します。	—	なし	06-50
┌ 	<host-name>						1 回	ホストを識別するための名称または IP アドレスを指定します。 ホストが複数のネットワークセグメントに接続されていて、複数の IP アドレスを持つ場合は、次のどちらかを設定してください。 - サーバ間、または負荷分散機とサーバ間の通信に使用するネットワークの IP アドレス - サーバ間、または負荷分散機とサーバ間の通信に使用するネットワークのホスト名	指定できる文字列を次に示します。 - ホスト名 - IPv4 アドレス @myhost	なし	07-50
┌ 	<display-name>						0 または 1 回	ホストの表示名を指定します。指定する文字に制限はありません。	128 文字以内の任意の文字列を指定します。	なし	06-50
┌ 	<description>						0 または 1 回	ホストのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	1024 文字以内の任意の文字列を指定します。	なし	06-50
┌ 	<agent-host>						0 または 1 回	運用管理エージェントのホスト名、IP アドレスまたは @myhost を指定します。 ホストが複数のネットワークセグメントに接続されていて、	指定できる文字列を次に示します。 ホスト名	host-name で指定した値	07-50

タグ名		出現 パ ター ン	内容	指定可能値	省略値	VR
I			複数の IP アドレスを持つ場合は、次のどちらかを設定してください。 • Management Server とサーバ間の通信に使用するネットワークの IP アドレス • Management Server とサーバ間の通信に使用するネットワークのホスト名	• IPv4 アドレス • @myhost		
L	<agent-port>	0 または 1 回	運用管理エージェントのポート番号を指定します。	1～65535 の整数を指定します。	20295	06-50

(凡例) - : 該当なし

4.3.3 Web システムの定義

(1) <web-system>

内容

Web システムを定義します。複数の Web システムを定義する場合は、<web-system> タグを複数指定します。

指定可能値

—

タグの省略

省略できません。

値の省略

省略できません。

(2) <name>

内容

必須定義。

Web システムを識別するための名称を指定します。

運用管理ドメイン内でユニークな名称を指定する必要があります。Smart Composer 機能のコマンドの -s オプションには、ここで指定した名称を指定します。

指定可能値

英数字またはアンダースコア「_」、およびハイフン「-」で指定した 32 文字以内の文字列を指定します。

タグの省略

省略できません。

値の省略

省略できません。

(3) <display-name>

内容

Web システムの表示名を指定します。指定する文字に制限はありません。

指定可能値

128 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(4) <description>

内容

Web システムのコメントを指定します。指定する文字に制限はありません。

指定可能値

1024 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

4.3.4 負荷分散機の定義

(1) <load-balancer>

内容

負荷分散機を定義します。

Smart Composer 機能で負荷分散機の設定，および実サーバポートの開始・閉塞をしない場合は，負荷分散機の定義（<load-balancer>タグから</load-balancer>タグまで）は必要ありません。なお，バッチアプリケーションを実行するシステムでは負荷分散機の定義は不要です。

指定可能値

—

タグの省略

省略できます。

値の省略

省略できます。

(2) <name>

内容

負荷分散機の名称を 32 文字以下で指定します。

指定可能値

英数字またはアンダースコア「_」，およびハイフン「-」で指定した 32 文字以内の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(3) <load-balancer-type>

内容

負荷分散機の種類を指定します。負荷分散機の種類を次に示します。

BIG-IPv9

BIG-IP v9 を使用する場合に指定します。

BIG-IPv10.1

BIG-IP v10.1 を使用する場合に指定します。

BIG-IPv10.2

BIG-IP v10.2 を使用する場合に指定します。

BIG-IPv11

BIG-IP v11 を使用する場合に指定します。

ACOS

ACOS を使用する場合に指定します。

Smart Composer 機能で負荷分散機の設定をする場合は，必ず指定してください。

指定可能値

指定できる文字列を次に示します。

- BIG-IPv9
- BIG-IPv10.1
- BIG-IPv10.2
- BIG-IPv11
- ACOS

タグの省略

省略できません。

値の省略

省略できません。

(4) <display-name>

内容

負荷分散機の表示名を 128 文字以下で指定します。指定する文字に制限はありません。

指定可能値

128 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(5) <description>

内容

負荷分散機のコメントを 1024 文字以下で指定します。指定する文字に制限はありません。

指定可能値

1024 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

4.3.5 cookie スイッチングの定義

(1) <cookie-switching>

内容

負荷分散機の Cookie スイッチング機能を使用する場合に指定します。

この設定によって、一連の HTTP リクエストが一つの Web サーバまたは J2EE サーバで処理されます。

指定可能値

—

タグの省略

省略できます。

値の省略

省略できます。

(2) <cookie-switching-enabled>

内容

Cookie スイッチング機能を使用する場合に指定します。

指定可能値

指定できる文字列を次に示します。

- true
Cookie スイッチング機能を使用する場合に指定します。
- false
Cookie スイッチング機能を使用しない場合に指定します。

タグの省略

省略できます。

値の省略

省略できます。

(3) <cookie-name>

内容

cookie の名称を 1～64 文字の文字列で指定します。

指定できる文字は、半英数字またはアンダースコア「_」の組み合わせです。

ただし、次の名前は Web コンテナが使用するため、大文字・小文字の区別なく使用できません。

- JSESSIONID

注

<load-balancer-type>に ACOS を指定し、かつ lb.properties の lb.connect_type.<IP アドレス>に API を指定すると、この指定は無効となります。

指定可能値

- BIG-IP の場合

英数字またはアンダースコア「_」で指定した 64 文字以内の文字列を指定します。なお、先頭 1 文字は大文字または小文字の英字を指定する必要があります。

- ACOS の場合

英数字またはアンダースコア「_」で指定した 63 文字以内の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(4) <server-id-rule>

内容

08-53 以降は、このタグに値を指定しないでください。値を指定した場合、指定した値は無効になります。

指定可能値

08-53 以降は、このタグに値を指定しないでください。値を指定した場合、指定した値は無効になります。

タグの省略

省略できます。

値の省略

省略できます。

(5) <management-host>

内容

負荷分散機に設定した管理 IP アドレスまたはホスト名を指定します。

複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで同じ管理 IP アドレスまたはホスト名を指定してください。

指定可能値

指定できる文字列を次に示します。

- ホスト名

管理 IP アドレスはドット記法 (xxx.xxx.xxx.xxx) で指定します。xxx には 0~255 の整数を指定します。

- IPv4 アドレス

ホスト名に指定できる文字は、英数字またはアンダースコア「_」、ハイフン「-」の組み合わせです。

タグの省略

省略できません。

値の省略

省略できません。

(6) <redundant-management-host>

内容

負荷分散機を冗長化したシステムの場合に、2 台目の負荷分散機に設定した管理 IP アドレスまたはホスト名を指定します。

複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで同じ管理 IP アドレスまたはホスト名を指定してください。

なお、定義ごとに指定できるタグについては、「[4.7 簡易構築定義ファイルおよび構成変更定義ファイルに指定できるタグ](#)」を参照してください。

指定可能値

指定できる文字列を次に示します。

- ホスト名

ホスト名に指定できる文字は、英数字またはアンダースコア「_」、ハイフン「-」の組み合わせです。

- IPv4 アドレス

管理 IP アドレスはドット記法 (xxx.xxx.xxx.xxx) で指定します。xxx には 0~255 の整数を指定します。

タグの省略

省略できます。

値の省略

省略できます。

4.3.6 仮想サーバの定義

(1) <virtual-server>

内容

仮想サーバを定義します。

指定可能値

—

タグの省略

省略できません。

値の省略

省略できません。

(2) <virtual-server-name>

内容

仮想サーバ名を 1～31 文字で指定します。

複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで仮想サーバ名が重複しないように指定してください。

指定可能値

英数字またはアンダースコア「_」、ピリオド「.」およびハイフン「-」で指定した 31 文字以内の文字列を指定します。

タグの省略

省略できません。

値の省略

省略できません。

(3) <display-name>

内容

仮想サーバの表示名を指定します。

指定可能値

128 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(4) <description>

内容

仮想サーバのコメントを指定します。

指定可能値

1024 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(5) <ip-address>

内容

仮想サーバのホスト名，または IP アドレスを指定します。

複数の Web システムで負荷分散機を共有する場合は，共有する Web システムで仮想サーバの IP アドレスが重複しないように指定してください。

指定可能値

指定できる文字列を次に示します。

- ホスト名
- IPv4 アドレス

タグの省略

省略できません。

値の省略

省略できません。

(6) <http-port>

内容

仮想サーバの http ポート番号を 1～65535 の間で指定します。

BIG-IP，ACOS の場合で，複数の Web システムで負荷分散機を共有するときは，共有する Web システムで仮想サーバの http ポート番号が重複しないように指定してください。

指定可能値

1～65535 の整数を指定します。

タグの省略

省略できません。

値の省略

省略できません。

4.3.7 物理ティアの定義

(1) <tier>

内容

必須定義。

物理ティアを定義します。

物理ティアには、combined-tier, http-tier, j2ee-tier, ctm-tier, および free-tier の 5 種類があります。それぞれの物理ティアは、単独または複数を組み合わせて、Web システムを構成します。

指定できる物理ティアの構成を次に示します。

- combined-tier 構成
- http-tier と j2ee-tier の構成※¹
- http-tier 構成※²
- j2ee-tier 構成
- ctm-tier 構成
- free-tier 構成※¹

注※¹ Smart Composer 機能で構築するシステムで使用できます。仮想化システムでは使用できません。

注※² 仮想化システムで使用できます。Smart Composer 機能で構築するシステムでは使用できません。

combined-tier 構成の場合

サービスユニット内の Web サーバと J2EE サーバを 1 ホストで構成し、一つの<tier>タグで定義します。

http-tier と j2ee-tier の構成の場合

サービスユニット内の Web サーバと J2EE サーバを別々のホストで構成し、二つの<tier>タグで定義します。

http-tier 構成の場合

サービスユニット内の Web サーバを 1 ホストで構成し、一つの<tier>タグで定義します。http-tier 構成を使用する場合は、仮想サーバグループ情報ファイルで関連づけられた j2ee-tier 構成の仮想サーバを用意しておく必要があります。

ctm-tier 構成の場合

ctm-tier は、combined-tier, http-tier, j2ee-tier を定義する Web システムとは別の Web システムで定義します。ctm-tier は、サービスユニット内の CTM 関連の論理サーバ (CTM ドメインマネージャ, CTM およびスマートエージェント) と J2EE サーバを 1 ホストで構成し、一つの<tier>タグで定義します。

また、統合ネーミングスケジューラサーバを配置する場合は、統合ネーミングスケジューラサーバ用および CTM 用の Web システムに分けて、それぞれ定義します。

統合スケジューラサーバ用の Web システムは、サービスユニット内の CTM 関連の論理サーバ (CTM ドメインマネージャ、CTM およびスマートエージェント) を 1 ホストで構成し、一つの<tier> タグで定義します。

仮想化システムで ctm-tier 構成を使用する場合は、ctm-tier 構成の管理ユニットに属する仮想サーバに加えて、combined-tier 構成の管理ユニットに属する仮想サーバを用意しておく必要があります。

free-tier 構成の場合

ほかのどの物理ティアの定義にも当てはまらない構成が、一つの<tier>タグで定義されます。free-tier は、cmx_export_model 実行時に生成する簡易構築定義ファイル内でだけ使用されます。

cmx_export_model 以外では、free-tier を含む簡易構築定義ファイルを作成しないでください。

指定可能値

—

タグの省略

省略できません。

値の省略

省略できません。

(2) <tier-type>

内容

必須定義。

物理ティアの種類を指定します。物理ティアの種類は次のとおりです。

- combined-tier
- http-tier
- j2ee-tier
- ctm-tier
- free-tier

指定可能値

指定できる文字列を次に示します。

- combined-tier
- http-tier
- j2ee-tier
- ctm-tier
- free-tier

タグの省略

省略できません。

値の省略

省略できません。

(3) <j2ee-server-count>

内容

<tier-type>タグで ctm-tier を指定した場合に、1 ホストに配置する J2EE サーバの数を 0～32 の整数で指定します。

指定可能値

0～32 の整数で指定します。

統合ネーミングスケジューラサーバ用の Web システムの場合

0 を指定します。

CTM 用の Web システムの場合

Web システムごとに、1 ホスト内に配置する J2EE サーバの数を指定します。

なお、Web システムの構成変更時、J2EE サーバの数は変更できません。

タグの省略

省略できます。

値の省略

省略できます。

(4) <configuration>

内容

物理ティアに属するすべての論理サーバに適用するコンフィグレーションを、論理サーバの種類ごとに定義します。

例えば、Web サーバと J2EE サーバの二つのコンフィグレーションを定義する場合は、<configuration>タグを二つ定義し、一つを Web サーバ用、もう一つを J2EE サーバ用として定義します。

指定可能値

—

タグの省略

省略できます。

値の省略

省略できます。

(5) <logical-server-type>

内容

必須定義。

コンフィグレーションを定義する論理サーバの種類を指定します。論理サーバの種類を次に示します。

- web-server : Web サーバ
- j2ee-server : J2EE サーバ
- performance-tracer : パフォーマンストレーサ
- ctm-domain-manager : CTM ドメインマネージャ
- component-transaction-monitor : CTM
- smart-agent : スマートエージェント

物理ティアの種類によって、コンフィグレーションが定義できる論理サーバが異なります。物理ティアの種類を次に示します。

combined-tier の場合

Web サーバ, J2EE サーバとパフォーマンストレーサ

http-tier の場合

Web サーバとパフォーマンストレーサ

j2ee-tier の場合

J2EE サーバとパフォーマンストレーサ

ctm-tier の場合

Web システムごとに定義できる論理サーバが異なります。

- 統合ネーミングスケジューラサーバ用の Web システムの場合
CTM ドメインマネージャ, CTM, スマートエージェントとパフォーマンストレーサ
- CTM 用の Web システムの場合
CTM ドメインマネージャ, CTM, スマートエージェント, J2EE サーバとパフォーマンストレーサ

注

物理ティアの定義では、ユーザサーバ、および free-tier 構成に含まれる論理サーバのコンフィグレーションが設定できません。これらの論理サーバのコンフィグレーションはユニットの定義で設定してください。

指定可能値

指定できる文字列を次に示します。

- web-server
- j2ee-server
- performance-tracer

- ctm-domain-manager
- component-transaction-monitor
- smart-agent

タグの省略

省略できません。

値の省略

省略できません。

(6) <server-no>

内容

物理ティア内の特定の J2EE サーバで、共通のコンフィグレーションを指定する場合に、J2EE サーバを識別するための番号を指定します。

指定できる範囲は、1～<j2ee-server-count>の指定値です。物理ティア内で番号が重複しないように指定してください。

このタグは、<tier-type>タグで ctm-tier を指定して、<j2ee-server-count>タグを指定した場合にだけ指定できます。

物理ティア内のすべての J2EE サーバで、共通のコンフィグレーションを指定する場合は、このタグを指定する必要はありません。

指定可能値

1～<j2ee-server-count>の指定値

タグの省略

省略できます。

値の省略

省略できます。

(7) <param>

内容

必須定義。

論理サーバの環境を設定するパラメタを、パラメタ名と値で定義します。

1 種類のパラメタごとに<param>タグで囲んで定義します。

パラメタによっては、複数の値を設定する場合、<param-value>タグを複数指定します。

指定可能値

—

タグの省略

省略できません。

値の省略

省略できません。

(8) <param-name>

内容

パラメタ名を指定します。

指定可能値

指定可能値については、「[4.8 システムの構成パターンと定義する論理サーバ](#)」の説明を参照してください。

タグの省略

省略できません。

値の省略

省略できません。

(9) <param-value>

内容

パラメタの設定値を指定します。

指定可能値

指定可能値については、「[4.8 システムの構成パターンと定義する論理サーバ](#)」の説明を参照してください。

タグの省略

省略できません。

値の省略

省略できません。

4.3.8 サービスユニットの定義

(1) <unit>

内容

必須定義。

サービスユニットを定義します。

複数のサービスユニットを定義する場合は、<unit>タグを複数指定します。

指定可能値

—

タグの省略

省略できません。

値の省略

省略できません。

(2) <name>

内容

必須定義。

サービスユニットを識別するための名称を指定します。Web システム内でユニークな名称を指定する必要があります。Smart Composer 機能のコマンドの-unit オプションには、ここで指定した名称を指定します。

指定可能値

英数字またはアンダースコア「_」およびハイフン「-」で指定した 32 文字以内の文字列を指定します。

タグの省略

省略できません。

値の省略

省略できません。

(3) <display-name>

内容

サービスユニットの表示名を指定します。指定する文字に制限はありません。

指定可能値

128 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(4) <description>

内容

サービスユニットのコメントを指定します。指定する文字に制限はありません。

指定可能値

1024 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(5) <allocated-host>

内容

必須定義。

サービスユニットを構成するホストのリファレンスを定義します。<allocated-host>タグは一つだけ指定します。free-tier 構成の Web システムでは、<allocated-host>タグを複数指定できます。

指定可能値

—

タグの省略

省略できません。

値の省略

省略できません。

(6) <host-ref>

内容

必須定義。

ホスト定義の<name>タグに指定したホスト名、構築済みの Web システムで定義されているホスト名、IP アドレス、または@myhost を指定します。

@myhost を指定していると、システム構築時に、ホストの定義の<host-name>タグの値が自動で設定されます。

なお、複数ホストがある場合、一つのサービスユニット内で同じホストを共有することはできません。

指定可能値

指定できる文字列を次に示します。

- ホスト名
- IPv4 アドレス
- @myhost

タグの省略

省略できません。

値の省略

省略できません。

(7) <hosts-for>

内容

必須定義。

<host-ref>タグで指定したホストが属する物理ティアの種類を指定します。物理ティアの種類を次に示します。

- combined-tier
- http-tier
- j2ee-tier
- ctm-tier
- free-tier

指定可能値

指定できる文字列を次に示します。

- combined-tier
- http-tier
- j2ee-tier
- ctm-tier
- free-tier

タグの省略

省略できません。

値の省略

省略できません。

4.3.9 論理サーバの定義

(1) <define-server>

内容

<hosts-for>タグで指定した物理ティアに属する論理サーバごとに、コンフィグレーションを定義します。

例えば、<hosts-for>タグが http-tier の場合は、Web サーバ用と、パフォーマンストレーサ用にそれぞれ一つずつ論理サーバを定義できます。複数の論理サーバを定義する場合は、<define-server>タグを複数指定します。

指定可能値

—

タグの省略

省略できます。

値の省略

省略できます。

(2) <logical-server-name>

内容

論理サーバまたはクラスタを識別するための名称を指定します。

指定可能値

英数字またはアンダースコア「_」およびハイフン「-」で指定した 128 文字以内の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(3) <display-name>

内容

論理サーバの表示名を指定します。指定する文字に制限はありません。

指定可能値

128 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(4) <description>

内容

論理サーバのコメントを指定します。指定する文字に制限はありません。

指定可能値

1024 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(5) <logical-server-type>

内容

必須定義。

コンフィグレーションを定義する論理サーバの種類を指定します。

論理サーバの種類を次に示します。

- web-server : Web サーバ
- j2ee-server : J2EE サーバ
- performance-tracer : パフォーマンストレーサ
- ctm-domain-manager : CTM ドメインマネージャ
- component-transaction-monitor : CTM
- smart-agent : スマートエージェント
- naming-service : ネーミングサービス
- user-server : ユーザサーバ

ホストが属する物理ティアの種類によって、コンフィグレーションが定義できる論理サーバが異なります。物理ティアの種類を次に示します。

combined-tier の場合

Web サーバ、J2EE サーバ、パフォーマンストレーサとユーザサーバ

http-tier の場合

Web サーバ、パフォーマンストレーサとユーザサーバ

j2ee-tier の場合

J2EE サーバ、パフォーマンストレーサとユーザサーバ

ctm-tier の場合

Web システムごとに定義できる論理サーバが異なります。

- 統合ネーミングスケジューラサーバ用の Web システムの場合
CTM ドメインマネージャ、CTM、スマートエージェント、パフォーマンストレーサとユーザサーバ
- CTM 用の Web システムの場合

CTM ドメインマネージャ, CTM, スマートエージェント, J2EE サーバ, パフォーマンストレーサとユーザサーバ

free-tier の場合

Web サーバ, J2EE サーバ, CTM ドメインマネージャ, CTM, スマートエージェント, ネーミングサービス, パフォーマンストレーサ, Web サーバクラスタ, J2EE サーバクラスタとユーザサーバ

指定可能値

指定できる文字列を次に示します。

- web-server
- j2ee-server
- performance-tracer
- ctm-domain-manager
- component-transaction-monitor
- smart-agent
- naming-service
- user-server

タグの省略

省略できません。

値の省略

省略できません。

(6) <cluster-ref>

内容

論理サーバをクラスタのメンバにする場合に、クラスタ名を指定します。

クラスタのメンバにする論理サーバが J2EE サーバの場合は J2EE サーバクラスタ名を、Web サーバの場合は Web サーバクラスタ名を指定します。

J2EE サーバおよび Web サーバ以外の論理サーバは、クラスタのメンバとして設定できません。

free-tier 構成の場合だけに定義されます。

指定可能値

英数字またはアンダースコア「_」およびハイフン「-」で指定した 128 文字以内の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(7) <server-no>

内容

サービスユニット内の特定の J2EE サーバのコンフィグレーションを指定する場合に、J2EE サーバを識別するための番号を指定します。

1 ホスト内で番号が重複しないように指定してください。このタグは、物理ティアの定義で<j2ee-server-count>タグを指定した場合に、J2EE サーバのコンフィグレーションを指定するときは、必ず指定してください。

指定可能値

1 ~ <j2ee-server-count>の指定値

タグの省略

省略できます。

値の省略

省略できます。

(8) <configuration>

内容

論理サーバごとに適用するコンフィグレーションを定義します。

なお、物理ティアの定義で定義した環境設定値で運用する場合は、<configuration>タグの定義は不要です。

<logical-server-type>で J2EE サーバクラスタまたは Web サーバクラスタを指定していた場合は指定できません。

指定可能値

—

タグの省略

省略できます。

値の省略

省略できます。

4.3.10 論理サーバのコンフィグレーション定義

(1) <param>

内容

必須定義。

論理サーバの環境を設定するパラメタを、パラメタ名と値で定義します。1 種類のパラメタごとに `<param>` タグで囲んで定義します。

パラメタによっては、複数の値を設定する場合、`<param-value>` タグを複数指定します。

指定可能値

—

タグの省略

省略できません。

値の省略

省略できません。

(2) `<param-name>`

内容

パラメタ名を指定します。

指定可能値

指定可能値については、「[4.8 システムの構成パターンと定義する論理サーバ](#)」の説明を参照してください。

タグの省略

省略できません。

値の省略

省略できません。

(3) `<param-value>`

内容

パラメタの設定値を指定します。

指定可能値

指定可能値については、「[4.8 システムの構成パターンと定義する論理サーバ](#)」の説明を参照してください。

タグの省略

省略できません。

値の省略

省略できません。

4.3.11 ホストの定義

(1) <host>

内容

ホストを定義します。

複数のホストを定義する場合は、host タグを複数指定します。ホストは複数の Web システムで共有できます。

指定可能値

—

タグの省略

省略できます。

値の省略

省略できます。

(2) <host-name>

内容

ホストを識別するための名称または IP アドレスを指定します。

IP アドレス変換できないホスト名を指定した場合、システム構築時にエラーとなります。

<host-name>タグには@myhost も指定できます。@myhost を指定すると、システム構築時に、構築先の環境の運用管理サーバマシンのホスト名に自動で置換されます。

指定可能値

指定できる文字列を次に示します。

- ホスト名
- IPv4 アドレス
- @myhost

タグの省略

省略できません。

値の省略

省略できません。

(3) <display-name>

内容

ホストの表示名を 128 文字以下で指定します。指定する文字に制限はありません。

指定可能値

128 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(4) <description>

内容

ホストのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。

指定可能値

1024 文字以内の任意の文字列を指定します。

タグの省略

省略できます。

値の省略

省略できます。

(5) <agent-host>

内容

運用管理エージェントのホスト名、または IP アドレスを指定します。

<agent-host>タグには@myhost も指定できます。@myhost を指定していると、システム構築時に、ホストの定義の<host-name>タグの値が自動で設定されます。

指定可能値

指定できる文字列を次に示します。

- ホスト名
- IPv4 アドレス
- @myhost

タグの省略

省略できます。

値の省略

省略できます。

(6) <agent-port>

内容

運用管理エージェントのポート番号を指定します。

指定可能値

1～65535 の整数を指定します。

タグの省略

省略できます。

値の省略

省略できます。

4.4 構成変更定義ファイル

4.4.1 構成変更定義ファイルの詳細

(1) 形式

XML 形式です。

(2) ファイルの格納先

任意の作業場所に格納してください。

(3) 機能

Smart Composer 機能のコマンドで構築した Web システムの構成を変更する場合に使用する定義ファイルです。構成変更定義ファイルには、論理サーバのパラメタ変更用とサービスユニット・ホスト追加用の二つの定義ファイルがあります。

次のテンプレートファイルをコピーして利用してください。

論理サーバのパラメタ変更用

- Windows の場合
 <Application Serverのインストールディレクトリ
 >%manager%config%templates%cmxmodcombinedmodel.xml
- UNIX の場合
 /opt/Cosminexus/manager/config/templates/cmxmodcombinedmodel.xml

注

論理サーバのパラメタ変更用の構成変更定義ファイルは、前バージョンとの互換用のファイルです。
論理サーバのパラメタを変更する場合は、簡易構築定義ファイルを使用してください。

サービスユニット・ホスト追加用

- Windows の場合
 <Application Serverのインストールディレクトリ
 >%manager%config%templates%cmxaddcombinedmodel.xml
- UNIX の場合
 /opt/Cosminexus/manager/config/templates/cmxaddcombinedmodel.xml

参考

テンプレートファイルの文字エンコーディングは、UTF-8 です。

(4) 設定できる要素

構成変更定義ファイルには、簡易構築定義ファイルに指定するタグの一部が指定できます。

次に、簡易構築定義ファイルとの違いを中心に説明します。

(a) 論理サーバのパラメタ変更用の構成変更定義ファイル

論理サーバのパラメタ変更用の構成変更定義ファイルの構造を次に示します。<model-modification>タグおよび<web-system>タグは、このままの形式で指定します。

論理サーバのパラメタ変更用の構成変更定義ファイルの構造

```
<model-modification xmlns="http://www.cosminexus.com/mngsvr/schema/ModelModification-2.5">
  <!-- 構成変更対象のWebシステムの定義 -->
  <web-system>
    <!-- 構成変更対象のWebシステムの属性定義 -->
    <name>Webシステム名</name>

    <!-- パラメタ変更する物理ティアの論理サーバの定義 -->
    <tier>
      <tier-type>物理ティアの種類</tier-type>
      <configuration>
        <logical-server-type>論理サーバの種類</logical-server-type>
        <param>
          <param-name>パラメタ名</param-name>
          <param-value>パラメタ値1</param-value>
          :
          <param-value>パラメタ値 n</param-value>
        </param>
        <param>
          :
        </configuration>
      </configuration>
      :
    </tier>
    <tier>
      :
    </tier>

    <!-- パラメタ変更するサービスユニットの論理サーバの定義 -->
    <unit>
      <name>サービスユニット名</name>
      <allocated-host>
        <host-ref>サービスユニットの構成ホストのホスト名またはIPアドレス</host-ref>
        <hosts-for>構成ホストが属する物理ティアの種類</hosts-for>
        <define-server>
          <logical-server-type>論理サーバの種類</logical-server-type>
          <configuration>
            <param>
              <param-name>パラメタ名</param-name>
              <param-value>パラメタ値1</param-value>
              :
              <param-value>パラメタ値 n</param-value>
            </param>
            <param>
              :
            </param>
          </configuration>
        </define-server>
      </allocated-host>
    </unit>
  </web-system>

```

```

        :
        </configuration>
        <configuration>
        :
        </allocated-host>
        <allocated-host>
        :
        </unit>
        <unit>
        :
        </web-system>
        <web-system>
        :
        </model-modification>

```

各定義での簡易構築定義ファイルとの違いを次に示します。

- Web システムの属性定義

- <name>タグの Web システム名には、構成変更対象の Web システムの名称を指定します。

- 物理ティアの定義

物理ティア内の論理サーバの種類ごとにコンフィグレーションを変更する場合に指定します。

- <configuration>タグは必ず定義します。ただし、cmx_trans_param コマンドを使用し、抽象パラメタを展開して物理ティアのコンフィグレーションを生成する場合は省略できます。
- <param-value>タグの指定が省略できます。省略した場合、設定済みのパラメタ値をすべて削除し、デフォルトの状態に戻します。

- サービスユニットの定義

サービスユニット内の特定の論理サーバごとにコンフィグレーションを変更する場合に指定します。

- CTM 用の Web システムで、物理ティアの定義に<j2ee-server-count>タグを指定している場合、特定の J2EE サーバのコンフィグレーションを変更するときは、<server-no>タグを必ず指定してください。
- <param-value>タグの指定が省略できます。省略した場合、設定済みのパラメタ値をすべて削除し、デフォルトの状態に戻します。

定義ごとに指定できるタグについては、「[4.7 簡易構築定義ファイルおよび構成変更定義ファイルに指定できるタグ](#)」を参照してください。

(b) サービスユニット・ホスト追加用の構成変更定義ファイル

サービスユニット・ホスト追加用の構成変更定義ファイルの構造を次に示します。<unit-addition>タグおよび<web-system>タグは、このままの形式で指定します。

サービスユニット・ホスト追加用の構成変更定義ファイルの構造

```

<unit-addition xmlns="http://www.cosminexus.com/mngsvr/schema/UnitAddition-2.5">
  <!-- 構成変更対象のWebシステムの定義 -->
  <web-system>

```



```

<!-- 構成変更対象のWebシステムの属性定義 -->
<name>Webシステム名</name>
<addition-mode>追加モード</addition-mode>

<!-- 追加するサービスユニットの定義 -->
<unit>
  <name>サービスユニット名</name>
  <display-name>サービスユニットの表示名</display-name>
  <description>サービスユニットのコメント</description>
  <allocated-host>
    <host-ref>サービスユニットの構成ホストのホスト名またはIPアドレス</host-ref>
    <hosts-for>構成ホストが属する物理ティアの種類</hosts-for>
    <define-server>
      <logical-server-name>論理サーバ名</logical-server-name>
      <display-name>論理サーバの表示名</display-name>
      <description>論理サーバのコメント</description>
      <logical-server-type>論理サーバの種類</logical-server-type>
      <configuration>
        <param>
          <param-name>パラメタ名</param-name>
          <param-value>パラメタ値1</param-value>
          :
          <param-value>パラメタ値 n</param-value>
        </param>
        <param>
          :
        </configuration>
      </configuration>
    </allocated-host>
  </allocated-host>
  :
</unit>
<unit>
  :
</web-system>
<web-system>
  :

<!-- 追加するホストの定義 -->
<host>
  <host-name>ホスト名またはIPアドレス</host-name>
  <display-name>ホストの表示名</display-name>
  <description>ホストのコメント</description>
  <agent-host>運用管理エージェントのホスト名またはIPアドレス</agent-host>
  <agent-port>運用管理エージェントのポート番号</agent-port>
</host>
<host>
  :
</unit-addition >

```

各定義での簡易構築定義ファイルとの違いを次に示します。

- Web システムの属性定義
 - <name>タグの Web システム名には、構成変更対象の Web システムの名称を指定します。

- JP1/SC/DPM を使用してスケールアウトを実施する場合は、<addition-mode>タグに「DEPLOYED」を指定してください。<addition-mode>タグは、サービスユニット・ホスト追加用の構成変更定義ファイルでだけ指定できます。
- サービスユニットの定義
 - 簡易構築定義ファイルと同じです。
- ホストの定義
 - 簡易構築定義ファイルと同じです。

定義ごとに指定できるタグについては、「[4.7 簡易構築定義ファイルおよび構成変更定義ファイルに指定できるタグ](#)」を参照してください。

4.5 論理サーバ参照定義ファイル

4.5.1 論理サーバ参照定義ファイルの詳細

(1) 形式

XML 形式です。

(2) ファイルの格納先

任意の作業場所に格納してください。

(3) 機能

Smart Composer 機能のコマンドで構築した Web システム中のサービスユニットに、ユーザ定義の論理サーバ（論理ユーザサーバ）を追加する場合に使用する定義ファイルです。論理ユーザサーバをサービスユニットに関連づけるための情報を定義します。

次のテンプレートファイルをコピーして利用してください。

- Windows の場合

<Application Serverのインストールディレクトリ>%manager%config%templates%cmxaddserverref.xml

- UNIX の場合

/opt/Cosminexus/manager/config/templates/cmxaddserverref.xml

なお、論理サーバ参照定義ファイルは、旧バージョンとの互換用のファイルです。構築済みのシステムに論理ユーザサーバを追加する場合に使用します。論理ユーザサーバを追加する場合は、簡易構築定義ファイルで設定することをお勧めします。

(4) 設定できる要素

論理サーバ参照定義ファイルには、追加する論理サーバ名、追加先の Web システムおよびサービスユニットを定義します。Web システム、サービスユニット、および論理サーバは、必要な数だけ定義します。

論理サーバ参照定義ファイルの構造を次に示します。<server-reference>タグおよび<web-system>タグは、このままの形式で指定します。

論理サーバ参照定義ファイルの構造

```
<server-reference xmlns="http://www.cosminexus.com/mngsvr/schema/ServerReference-2.0">
  <!-- Webシステムの定義 -->
  <web-system>
    <!-- Webシステムの属性定義 -->
    <name>Webシステム名</name>
```

```

<!-- サービスユニットの定義 -->
<unit>
  <name>サービスユニット名1</name>
  <logical-server-ref>論理ユーザサーバ名1</logical-server-ref>
  :
  <logical-server-ref>論理ユーザサーバ名n</logical-server-ref>
</unit>
<unit>
  :
</web-system>
<web-system>
  :
</server-reference>

```

各タグの説明とデフォルト値を次に示します。

タグ名称	内容	デフォルト
web-system	Web システムを定義します。 複数の Web システムを定義する場合は、<web-system>タグを複数指定します。	なし
name	Web システムを識別するための名称を 32 文字以下で指定します。ドメイン内でユニークな名称を指定する必要があります。Smart Composer 機能のコマンドの-s オプションには、ここで指定した名称を指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。	なし
unit	サービスユニットを定義します。 複数のサービスユニットを定義する場合は、<unit>タグを複数指定します。	なし
name	サービスユニットを識別するための名称を 32 文字以下で指定します。Web システム内でユニークな名称を指定する必要があります。Smart Composer 機能のコマンドの-unit オプションには、ここで指定した名称を指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。	なし
logical-server-ref	サービスユニットに追加する論理ユーザサーバの名称を 128 文字以下で指定します。 論理ユーザサーバ定義ファイルの<logical-server-name>タグで指定した名称を指定します。mngsvrutil add userserver コマンドで追加済みの論理ユーザサーバを指定してください。 複数の論理ユーザサーバを定義する場合は、<logical-server-ref>タグを複数指定します。	なし

(5) 記述例

サービスユニット unit1 に論理ユーザサーバ usersv1 および usersv2 を、サービスユニット unit2 に論理ユーザサーバ usersv3 および usersv4 を追加する例を次に示します。

```

<?xml version="1.0" encoding="UTF-8"?>
<server-reference xmlns="http://www.cosminexus.com/mngsvr/schema/ServerReference-2.0">

```

```
<web-system>
  <name>MyWebSystem</name>
  <unit>
    <name>unit1</name>
    <logical-server-ref>usersv1</logical-server-ref>
    <logical-server-ref>usersv2</logical-server-ref>
  </unit>
  <unit>
    <name>unit2</name>
    <logical-server-ref>usersv3</logical-server-ref>
    <logical-server-ref>usersv4</logical-server-ref>
  </unit>
</web-system>
</server-reference>
```

4.6 スケールアウト用ホスト定義ファイル

4.6.1 スケールアウト用ホスト定義ファイルの詳細

(1) 形式

XML 形式です。

(2) ファイルの格納先

任意の作業場所に格納してください。

(3) 機能

Smart Composer 機能のコマンドで構築した、ホスト単位管理モデルの Web システムをスケールアウトする場合に使用する定義ファイルです。複製先のホストの、Web システムの情報を定義します。

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
 <Application Serverのインストールディレクトリ>
 >%manager%config%templates%cmxscaleouthostdef.xml
- UNIX の場合
 /opt/Cosminexus/manager/config/templates/cmxscaleouthostdef.xml

■ 参考

テンプレートファイルの文字エンコーディングは、UTF-8 です。

(4) 設定できる要素

スケールアウト用ホスト定義ファイルには、追加するホスト、および複製先の Web システムで使用する負荷分散機を定義します。

スケールアウト用ホスト定義ファイルの構造を次に示します。<host-scaleout>タグおよび<web-system>タグは、このままの形式で指定します。

スケールアウト用ホスト定義ファイルの構造

```
<host-scaleout xmlns="http://www.cosminexus.com/mngsvr/schema/HostScaleOut-2.5">
  <!-- 複製先のWebシステムの定義 -->
  <web-system>
    <!-- Webシステムの属性定義 -->
    <name>Webシステム名</name>
```

```
</web-system>

<!-- 複製先のホストの定義 -->
<host>
  <host-name>ホスト名またはIPアドレス</host-name>
  <display-name>ホストの表示名</display-name>
  <description>ホストのコメント</description>
  <agent-host>運用管理エージェントのホスト名またはIPアドレス</agent-host>
</host>
</host-scaleout>
```

各タグの説明とデフォルト値を次に示します。

タグ名称	内容	デフォルト
web-system	複製先の Web システムを定義します。 複製先の Web システムの定義を変更する場合に指定します。複数の Web システムを定義する場合は、<web-system>タグを複数指定します。	なし
name	設定を変更する複製先の Web システムの名称を指定します。	なし
host*	複製先のホストを定義します。 ホストは複数の Web システムで共有できます。	なし
host-name	運用用のホスト名または IP アドレスを指定します。ホスト名を指定する場合は、32 文字以下で指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。IP アドレスを指定する場合は、ドット記法（xxx.xxx.xxx.xxx）で指定します。xxx には 0～255 の整数を指定します。	なし
display-name	ホストの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	なし
description	ホストのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	なし
agent-host	運用管理エージェントのホスト名、または IP アドレスを指定します。ホスト名を指定する場合は、32 文字以下で指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。IP アドレスを指定する場合は、ドット記法（xxx.xxx.xxx.xxx）で指定します。xxx には 0～255 の整数を指定します。	<host-name>タグに指定した値

注※
ホストの定義を省略した場合は、複製先のホストのホスト名が<host-name>タグに仮定され、ホストが定義されます。

(5) 記述例

複製元のホストと複製先のホストの設定項目例を次に示します。

表 4-3 複製元のホストと複製先のホストの設定項目例

設定項目	複製元のホスト	複製先のホスト
Web システム名	MyWebSystem	MyWebSystem
ホスト名	apsvA	apsvA
IP アドレス	192.168.1.20	192.168.1.21
管理 IP アドレス	192.168.100.20	192.168.100.21

ホスト apsvA を複製して作成したディスクイメージを使用してスケールアウトする例を次に示します。この例は、J2EE アプリケーションを実行するシステムをスケールアウトする場合の記述例です。

```
<?xml version="1.0" encoding="UTF-8"?>
<host-scaleout xmlns="http://www.cosminexus.com/mngsvr/schema/HostScaleOut-2.1">
  <!-- 複製先のWebシステムの定義 -->
  <web-system>
    <!-- Webシステムの属性定義 -->
    <name>MyWebSystem</name>
  </web-system>
  <!-- 複製先のホストの定義 -->
  <host>
    <host-name>apsvA</host-name>
    <agent-host>192.168.100.21</agent-host>
  </host>
</host-scaleout>
```


4.7 簡易構築定義ファイルおよび構成変更定義ファイルに指定できるタグ

簡易構築定義ファイル，論理サーバのパラメタ変更用の構成定義変更ファイル，およびサービスユニット・ホスト追加用の構成変更定義ファイルで指定できるタグについて説明します。

J2EE アプリケーションを実行するシステムで使用するタグ，およびバッチアプリケーションを実行するシステムで使用するタグについて，システムごとに説明します。

なお，簡易構築定義ファイルの構造については「[4.3 簡易構築定義ファイル](#)」を，構成定義変更ファイルの構造については「[4.4 構成変更定義ファイル](#)」を参照してください。

4.7.1 J2EE アプリケーションを実行するシステムで指定できるタグ

ここでは，J2EE アプリケーションを実行するシステムで指定できるタグについて説明します。

(1) Web システムの属性定義

Web システムの属性として定義できる内容を次の表に示します。

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
web-system	Web システムを定義します。 複数の Web システムを定義する場合は，<web-system>タグを複数指定します。	△※1	△	△※2	なし
name	Web システムを識別するための名称を 32 文字以下で指定します。運用管理ドメイン内でユニークな名称を指定する必要があります。Smart Composer 機能のコマンドの-s オプションには，ここで指定した名称を指定します。指定できる文字は，英数字，アンダースコア「_」，ハイフン「-」の組み合わせです。	◎	◎	◎	なし
addition-mode	追加モードを指定します。 - DEPLOYED※3 JP1/SC/DPM を使用してディスクイメージのスケールアウトをする場合に指定します。この指定によって，システム構築時の J2EE アプリケーションおよびリソースアダプタでのデプロイが不要になります。 - NORMAL 通常のスケールアウトの場合に指定します。	—	—	△	NORMAL
display-name	Web システムの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	△	—	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
description	Web システムのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	△	—	なし

(凡例)

P：論理サーバのパラメタ変更用の構成定義変更ファイルを示します。

U：サービスユニット・ホスト追加用の構成定義変更ファイルを示します。

◎：Web システムを定義する場合は、必ず指定します。

△：省略できます。

—：該当しません。

注※1

Smart Composer 機能で定義済みの Web システムを利用する場合は省略できます。

注※2

サービスユニットを追加しない場合は省略できます。

注※3

この設定を有効にする場合、Smart Composer 機能のコマンドを使用したインポートと一括開始をしておく必要があります。詳細については、マニュアル「アプリケーションサーバ システム構築・運用ガイド」の「4.1.26 システムを起動する（CUI 利用時）」を参照してください。

(2) 負荷分散機の定義

負荷分散機の定義は、<load-balancer>以下のタグで定義します。定義できる内容を次の表に示します。

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
load-balancer	負荷分散機を定義します。 Smart Composer 機能で負荷分散機の設定、および実サーバポートの開始・閉塞をしない場合は、負荷分散機の定義（<load-balancer>タグから</load-balancer>タグまで）は必要ありません。	△	—	—	なし
name	負荷分散機の名称を 32 文字以下で指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。	△	—	—	なし
load-balancer-type ^{※1}	負荷分散機の種類を次のどれかで指定します。 - BIG-IPv9 - BIG-IPv10.1 - BIG-IPv10.2 - BIG-IPv11 - ACOS	◎	—	—	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	Smart Composer 機能で負荷分散機の設定をする場合は、必ず指定してください。				
display-name	負荷分散機の表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	—	—	なし
description	負荷分散機のコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	—	—	なし
cookie-switching	Cookie スイッチング機能を使用する場合に定義します。この設定によって、一連の HTTP リクエストが一つの Web サーバまたは J2EE サーバで処理されます。	△※2	—	—	なし
cookie-switching-enabled	Cookie スイッチング機能を使用する場合は true を、使用しない場合は false を指定します。	△	—	—	false
cookie-name	cookie の名称を指定します。指定できる文字は、英数字、アンダースコア「_」の組み合わせです。ただし、次の名前は Web コンテナが使用するため、大文字・小文字の区別なく使用できません。 JSESSIONID BIG-IP の場合、1～64 文字の文字列で指定します。なお、先頭 1 文字は大文字または小文字の英字を指定する必要があります。 ACOS の場合、1～63 文字の文字列で指定します。	△	—	—	CMX_SERVER_ID
server-id-rule	08-53 以降は、このタグに値を指定しないでください。値を指定した場合、指定された値は無効になります。	△	—	—	なし
management-host	負荷分散機に設定した管理 IP アドレスまたはホスト名を指定します。管理 IP アドレスはドット記法 (xxx.xxx.xxx.xxx) で指定します。xxx には 0～255 の整数を指定します。ホスト名に指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。 複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで同じ管理 IP アドレスまたはホスト名を指定してください。	◎	—	—	なし
redundant-management-host	負荷分散機を冗長化したシステムの場合に、2 台目の負荷分散機に設定した管理 IP アドレスまたはホスト名を指定します。管理 IP アドレスはドット記法 (xxx.xxx.xxx.xxx) で指定します。xxx には 0～255 の整数を指定します。ホスト名に指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。 複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで同じ管理 IP アドレスまたはホスト名を指定してください。	△	—	—	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
virtual-server	仮想サーバを定義します。	◎	—	—	なし
virtual-server-name	仮想サーバ名を 1～31 文字で指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」、ピリオド「.」の組み合わせです。 複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで仮想サーバ名が重複しないように指定してください。	◎	—	—	なし
display-name	仮想サーバの表示名を指定します。	△	—	—	なし
description	仮想サーバのコメントを指定します。	△	—	—	なし
ip-address	仮想サーバの IP アドレスを指定します。 複数の Web システムで負荷分散機を共有する場合は、共有する Web システムで仮想サーバの IP アドレスが重複しないように指定してください。	◎	—	—	なし
http-port	仮想サーバの http ポート番号を 1～65535 の間で指定します。 BIG-IP v9, ACOS の場合で、複数の Web システムで負荷分散機を共有するときは、共有する Web システムで仮想サーバの http ポート番号が重複しないように指定してください。	◎	—	—	なし

(凡例)

- P：論理サーバのパラメタ変更用の構成定義変更ファイルを示します。
- U：サービスユニット・ホスト追加用の構成定義変更ファイルを示します。
- ◎：負荷分散機を定義する場合は、必ず指定します。
- △：省略できます。
- ：該当しません。

注※1

アプリケーションサーバのバージョンによって、サポートしている負荷分散機の種類が異なります。アプリケーションサーバのバージョンごとに、サポートしている負荷分散機を次の表に示します。

アプリケーションサーバのバージョン	サポートしている負荷分散機
09-50	BIG-IPv9, BIG-IPv10.1, BIG-IPv10.2, BIG-IPv11, ACOS
09-00, 08-70 および 08-53	BIG-IPv9, BIG-IPv10.1, BIG-IPv10.2, ACOS
08-50	BIG-IPv9, ACOS
08-00	Loadflowbal, BIG-IP, BIG-IPv9

注※2

負荷分散機の Cookie スイッチング機能を使用しない場合は省略できます。

4. Smart Composer 機能で使用するファイル

(3) 物理ティアの定義

構築するシステムの構成に合わせて、物理ティアを定義します。物理ティアは<tier>以下のタグで定義します。定義できる内容を次の表に示します。

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
tier	物理ティアを定義します。 物理ティアには、combined-tier、http-tier、j2ee-tier、ctm-tier、および free-tier の 5 種類があります。それぞれの物理ティアは、単独、または複数を組み合わせて、Web システムを構成します。 指定できる物理ティアの構成を次に示します。 • combined-tier 構成 • http-tier と j2ee-tier の構成 • j2ee-tier 構成 • ctm-tier 構成 • free-tier 構成 combined-tier 構成の場合 サービスユニット内の Web サーバと J2EE サーバを 1 ホストで構成し、一つの<tier>タグで定義します。 http-tier と j2ee-tier の構成の場合 サービスユニット内の Web サーバと J2EE サーバを別々のホストで構成し、二つの<tier>タグで定義します。 ctm-tier 構成の場合 ctm-tier は、combined-tier、http-tier、j2ee-tier を定義する Web システムとは別の Web システムで定義します。ctm-tier は、統合ネーミングスケジューラサーバ用および CTM 用の Web システムに分けて、それぞれ定義します。 • 統合ネーミングスケジューラサーバ用の Web システム サービスユニット内の CTM 関連の論理サーバ ※1 を 1 ホストで構成し、一つの<tier>タグで定義します。 • CTM 用の Web システム サービスユニット内の CTM 関連の論理サーバ ※1 と J2EE サーバを 1 ホストで構成し、一つの<tier>タグで定義します。 free-tier 構成の場合 ほかのどの物理ティアの定義にも当てはまらない構成が、一つの<tier>タグで定義されます。	◎	△※2	—	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
tier-type	物理ティアの種類を次のどれかで指定します。 • combined-tier • http-tier • j2ee-tier • ctm-tier • free-tier	◎	◎	—	なし
j2ee-server-count	<tier-type>タグで ctm-tier を指定した場合に、1 ホストに配置する J2EE サーバの数を 0～32 の整数で指定します。 • 統合ネーミングスケジューラサーバ用の Web システムの場合 0 を指定します。 • CTM 用の Web システムの場合 Web システムごとに、1 ホスト内に配置する J2EE サーバの数を指定します。 なお、Web システムの構成変更時、J2EE サーバの数は変更できません。	△	—	—	1
configuration	物理ティアに属するすべての論理サーバに適用するコンフィグレーションを、論理サーバの種類ごとに定義します。例えば、Web サーバと J2EE サーバの二つのコンフィグレーションを定義する場合は、<configuration>タグを二つ定義し、一つを Web サーバ用、もう一つを J2EE サーバ用として定義します。	△※3	◎※4	—	なし
logical-server-type	コンフィグレーションを定義する論理サーバの種類を指定します。指定できる論理サーバの種類を次に示します。 • web-server：Web サーバ • j2ee-server：J2EE サーバ • performance-tracer：パフォーマンストレーサ • ctm-domain-manager：CTM ドメインマネージャ • component-transaction-monitor：CTM • smart-agent：スマートエージェント 物理ティアの種類によって、コンフィグレーションが定義できる論理サーバが異なります。 combined-tier の場合 Web サーバ、J2EE サーバとパフォーマンストレーサ http-tier の場合 Web サーバとパフォーマンストレーサ	◎	◎	—	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	j2ee-tier の場合 J2EE サーバとパフォーマンストレーサ ctm-tier の場合 Web システムごとに定義できる論理サーバが異なります。 - 統合ネーミングスケジューラサーバ用の Web システムの場合 CTM ドメインマネージャ, CTM, スマートエージェントとパフォーマンストレーサ - CTM 用の Web システムの場合 CTM ドメインマネージャ, CTM, スマートエージェント, J2EE サーバとパフォーマンストレーサ				
server-no	物理ティア内の特定の J2EE サーバで、共通のコンフィグレーションを指定する場合に、J2EE サーバを識別するための番号を指定します。指定できる範囲は、1～<j2ee-server-count>の指定値です。物理ティア内で番号が重複しないように指定してください。このタグは、<tier-type>タグで ctm-tier を指定して、<j2ee-server-count>タグを指定した場合にだけ指定できます。 物理ティア内のすべての J2EE サーバで、共通のコンフィグレーションを指定する場合は、このタグを指定する必要はありません。	△	△	—	なし
param ^{*5}	論理サーバの環境を設定するパラメタを、パラメタ名と値で定義します。1 種類のパラメタごとに<param>タグで囲んで定義します。	◎	◎	—	なし
param-name ^{*5}	論理サーバの環境を設定するパラメタの名称を指定します。	◎	◎	—	なし
param-value ^{*5}	<param-name>タグで指定したパラメタへの設定値を指定します。パラメタによっては、複数の値を設定する場合、<param-value>タグを複数指定します。	◎	△ ^{*6}	—	なし

(凡例)

- P：論理サーバのパラメタ変更用の構成定義変更ファイルを示します。
- U：サービスユニット・ホスト追加用の構成定義変更ファイルを示します。
- ◎：Web システムを定義する場合は、必ず指定します。
- △：省略できます。
- ：該当しません。

注

物理ティアの定義では、ユーザサーバ、および free-tier 構成に含まれる論理サーバのコンフィグレーションが設定できません。これらの論理サーバのコンフィグレーションはユニットの定義で設定してください。

- 注※1
CTM 関連の論理サーバには、CTM ドメインマネージャ、CTM およびスマートエージェントがあります。
- 注※2
物理ティアの定義を変更しない場合は省略できます。
- 注※3
Management Server が論理サーバ作成時に仮定する、環境設定値で運用する場合は省略できます。
- 注※4
cmx_trans_param コマンドを使用して、抽象パラメタを展開して物理ティアのコンフィグレーションを生成する場合は省略できます。
- 注※5
<param>タグに指定できるパラメタは論理サーバの種類によって異なります。また、定義する論理サーバはシステムの構成パターンによって異なります。「[4.8 システムの構成パターンと定義する論理サーバ](#)」を参照して、定義する論理サーバを確認してください。それぞれの論理サーバに設定できるパラメタについては、使用する論理サーバの種類に応じて 4.12 以降の節を参照してください。
- 注※6
<param-name>タグで指定したパラメタを削除する場合は省略できます。

(4) サービスユニットの定義

サービスユニットは<unit>以下のタグに定義します。定義できる内容を次の表に示します。

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
unit	サービスユニットを定義します。 複数のサービスユニットを定義する場合は、<unit>タグを複数指定します。	◎	△※1	◎	なし
name	サービスユニットを識別するための名称を 32 文字以下で指定します。Web システム内でユニークな名称を指定する必要があります。Smart Composer 機能のコマンドの-unit オプションには、ここで指定した名称を指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。	◎	△※2	◎	なし
display-name	サービスユニットの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	△	△	なし
description	サービスユニットのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	△	△	なし
allocated-host	サービスユニットを構成するホストのリファレンスを定義します。<allocated-host>タグは一つだけ指定します。free-tier 構成の Web システムでは、<allocated-host>タグを複数指定できます。	◎	◎	◎	なし
host-ref	ホスト定義の<name>タグに指定したホスト名、構築済みの Web システムで定義されているホスト名、または@myhost を指定します。	◎	◎	◎	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	@myhost を指定していると、システム構築時に、ホストの定義の<host-name>タグの値が自動で設定されます。 なお、複数ホストがある場合、一つのサービスユニット内で同じホストを共有することはできません。				
hosts-for	<host-ref>タグで指定したホストが属する物理ティアの種類を、次のどれかで指定します。 • combined-tier • http-tier • j2ee-tier • ctm-tier • free-tier	◎	◎	◎	なし
define-server	<hosts-for>タグで指定した物理ティアに属する論理サーバごとに、コンフィグレーションを定義します。 例えば、<hosts-for>タグが http-tier の場合は、Webサーバ用と、パフォーマンストレーサ用にそれぞれ一つずつ論理サーバを定義できます。複数の論理サーバを定義する場合は、<define-server>タグを複数指定します。	△※3	◎	△※3	なし
logical-server-name	論理サーバまたはクラスタを識別するための名称を 128 文字以下で指定します。Smart Composer 機能のコマンドの-s オプションには、ここで指定した名称を指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。	△	—	△	cmx_<Web システム名>_<サービスユニット名>_<種別>_<通番>
display-name	論理サーバの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	△	△	なし
description	論理サーバのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	△	△	なし
logical-server-type	コンフィグレーションを定義する論理サーバの種類を指定します。指定できる論理サーバの種類を次に示します。 • web-server：Web サーバ • j2ee-server：J2EE サーバ • performance-tracer：パフォーマンストレーサ • ctm-domain-manager：CTM ドメインマネージャ • component-transaction-monitor：CTM • smart-agent：スマートエージェント • naming-service：ネーミングサービス • user-server：ユーザサーバ※6	◎	◎	◎	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	ホストが属する物理ティアの種類によって、コンフィグレーションが定義できる論理サーバが異なります。 combined-tier の場合 Web サーバ、J2EE サーバ、パフォーマンスストレサとユーザサーバ http-tier の場合 Web サーバ、パフォーマンスストレサとユーザサーバ j2ee-tier の場合 J2EE サーバ、パフォーマンスストレサとユーザサーバ ctm-tier の場合 Web システムごとに定義できる論理サーバが異なります。 - 統合ネーミングスケジューラサーバ用の Web システムの場合 CTM ドメインマネージャ、CTM、スマートエージェント、パフォーマンスストレサとユーザサーバ - CTM 用の Web システムの場合 CTM ドメインマネージャ、CTM、スマートエージェント、J2EE サーバ、パフォーマンスストレサとユーザサーバ free-tier の場合 Web サーバ、J2EE サーバ、CTM ドメインマネージャ、CTM、スマートエージェント、ネーミングサービス、パフォーマンスストレサ、Web サーバクラスタ、J2EE サーバクラスタとユーザサーバ				
cluster-ref	論理サーバをクラスタのメンバにする場合に、クラスタ名を指定します。 クラスタのメンバにする論理サーバが J2EE サーバの場合は J2EE サーバクラスタ名を、Web サーバの場合は Web サーバクラスタ名を指定します。 J2EE サーバおよび Web サーバ以外の論理サーバは、クラスタのメンバとして設定できません。 free-tier 構成の場合だけに定義されます。	△	×	×	なし
server-no	サービスユニット内の特定の J2EE サーバのコンフィグレーションを指定する場合に、J2EE サーバを識別するための番号を指定します。指定できる範囲は、1～<j2ee-server-count>の指定値です。1 ホスト内で番号が重複しないように指定してください。このタグは、物理ティアの定義で<j2ee-server-count>タグを指定	△	△	△	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	した場合に、J2EE サーバのコンフィグレーションを指定するときは、必ず指定してください。				
configuration	論理サーバごとに適用するコンフィグレーションを定義します。 なお、物理ティアの定義で定義した環境設定値で運用する場合は、<configuration>タグの定義は不要です。 <logical-server-type>で J2EE サーバクラスタまたは Web サーバクラスタを指定していた場合は指定できません。	△	△	△	なし
param ^{※4}	論理サーバの環境を設定するパラメタを、パラメタ名と値で定義します。1 種類のパラメタごとに<param>タグで囲んで定義します。	◎	◎	◎	なし
param-name ^{※4}	論理サーバの環境を設定するパラメタの名称を指定します。指定できるパラメタ名については、「 4.8 システムの構成パターンと定義する論理サーバ 」を参照してください。	◎	◎	◎	なし
param-value ^{※4}	<param-name>タグで指定したパラメタへの設定値を指定します。パラメタによっては、複数の値を設定する場合、<param-value>タグを複数指定します。それぞれのパラメタの設定値については、「 4.8 システムの構成パターンと定義する論理サーバ 」を参照してください。	◎	△ ^{※5}	◎	なし

(凡例)

P：論理サーバのパラメタ変更用の構成定義変更ファイルを示します。

U：サービスユニット・ホスト追加用の構成定義変更ファイルを示します。

◎：Web システムを定義する場合は、必ず指定します。

△：省略できます。

－：該当しません。

注

構成変更定義ファイルには、free-tier 構成に含まれる論理サーバのコンフィグレーションが設定できません。

注※1

サービスユニットの定義を変更しない場合は省略できます。

注※2

クライアント設定プロパティファイルまたはクライアント共通設定プロパティファイルに cmx.websystem.name を指定している場合は省略できます。

注※3

論理サーバごとにコンフィグレーションを定義しない場合は省略できます。

注※4

<param>タグに指定できるパラメタは論理サーバの種類によって異なります。また、定義する論理サーバはシステムの構成パターンによって異なります。「[4.8 システムの構成パターンと定義する論理サーバ](#)」を参照して、定義する論理サーバを確

認してください。それぞれの論理サーバに設定できるパラメタについては、使用する論理サーバの種類に応じて 4.12 以降の節を参照してください。

注※5

<param-name>タグで指定したパラメタを削除する場合は省略できます。

注※6

簡易構築定義ファイルと、サービスユニットの追加の構成定義変更ファイルに user-server を指定できます。サービスユニットの変更の構成定義変更ファイルには user-server を指定できません。

(5) ホストの定義

使用するホストの情報は<host>以下のタグに定義します。定義できる内容を次の表に示します。

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
host	ホストを定義します。 複数のホストを定義する場合は、host タグを複数指定します。ホストは複数の Web システムで共有できます。	△※1	—	△※2	なし
host-name	ホストを識別するための名称または IP アドレスを 32 文字以下で指定します。 ホスト名に指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。管理 IP アドレスはドット記法（xxx.xxx.xxx.xxx）で指定します。xxx には 0～255 の整数を指定します。 IP アドレス変換できないホスト名を指定した場合、システム構築時にエラーとなります。 <host-name>タグには@myhost も指定できます。 @myhost を指定すると、システム構築時に、構築先の環境の運用管理サーバマシンのホスト名に自動で置換されます。	◎	—	◎	なし
display-name	ホストの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	—	△	なし
description	ホストのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	—	△	なし
agent-host	運用管理エージェントのホスト名、または IP アドレスを指定します。ホスト名を指定する場合は、32 文字以下で指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。IP アドレスを指定する場合は、ドット記法（xxx.xxx.xxx.xxx）で指定します。xxx には 0～255 の整数を指定します。 <agent-host>タグには@myhost も指定できます。 @myhost を指定していると、システム構築時に、ホス	△	—	△	host-name で指定した 値

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	トの定義の<host-name>タグの値が自動で設定されま す。				
agent-port	運用管理エージェントのポート番号を 1～65535 の間 で指定します。	△	—	△	20295

(凡例)

P：論理サーバのパラメタ変更用の構成定義変更ファイルを示します。

U：サービスユニット・ホスト追加用の構成定義変更ファイルを示します。

◎：ホストを定義する場合は、必ず指定します。

△：省略できます。

—：該当しません。

注

ユニットの定義の<host-ref>タグで@myhost を指定して、ホストの定義の<display-name>タグ、<description>タグ、<agent-host>タグ、または<agent-port>タグの値を省略している場合、ホストの定義は出力されません。この場合、ホストの定義の<host-name>タグの値は、Management Server が必要に応じて自動生成します。Management Server が自動作成するホスト名は、Windows の場合はコンピュータ名、UNIX の場合は、hostname コマンドで設定したホスト名です。ホストの中に論理サーバが一つも含まれなくなったとき、Management Server が、そのホストを自動で削除します。

注※1

Smart Composer 機能で定義済みのホストを利用する場合は省略できます。

注※2

ホストの定義を変更しない場合は省略できます。

4.7.2 バッチアプリケーションを実行するシステムで指定できるタグ

ここでは、バッチアプリケーションを実行するシステムで指定できるタグについて説明します。

(1) Web システムの属性定義

Web システムの属性として定義できる内容を次の表に示します。

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
web-system	Web システムを定義します。 複数の Web システムを定義する場合は、<web- system>タグを複数指定します。	△※1	△	△※2	なし
name	Web システムを識別するための名称を 32 文字以下で 指定します。運用管理ドメイン内でユニークな名称を 指定する必要があります。Smart Composer 機能のコ	◎	◎	◎	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	マンドの-s オプションには、ここで指定した名称を指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。				
addition-mode	追加モードを指定します。 - DEPLOYED※3 JP1/SC/DPM を使用してディスクイメージのスケールアウトをする場合に指定します。この指定によって、システム構築時のリソースアダプタでのデプロイが不要になります。 - NORMAL 通常のスケールアウトの場合に指定します。	—	—	△	NORMAL
display-name	Web システムの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	△	—	なし
description	Web システムのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	△	—	なし

(凡例)

- P：論理サーバのパラメタ変更用の構成定義変更ファイルを示します。
- U：サービスユニット・ホスト追加用の構成定義変更ファイルを示します。
- ◎：Web システムを定義する場合は、必ず指定します。
- △：省略できます。
- ：該当しません。

注※1

Smart Composer 機能で定義済みの Web システムを利用する場合は省略できます。

注※2

サービスユニットを追加しない場合は省略できます。

注※3

この設定を有効にする場合、Smart Composer 機能のコマンドを使用したインポートと一括開始をしておく必要があります。詳細については、マニュアル「アプリケーションサーバ システム構築・運用ガイド」を参照してください。

(2) 物理ティアの定義

物理ティアを<tier>以下のタグで定義します。

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
tier	物理ティアを定義します。 バッチアプリケーションを実行するシステムでは物理ティアに j2ee-tier または ctm-tier のどちらかを指定	◎	△※1	—	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	してください。それぞれの物理ティアは、一つの<tier>タグで定義します。				
tier-type	物理ティアの種類を指定します。バッチアプリケーションを実行するシステムでは、j2ee-tier または ctm-tier のどちらかを指定します。	◎	◎	—	なし
configuration	物理ティアに属するすべての論理サーバに適用するコンフィグレーションを、論理サーバの種類ごとに定義します。	△※2	◎※3	—	なし
logical-server-type	コンフィグレーションを定義する論理サーバの種類を指定します。バッチアプリケーションを実行するシステムではバッチサーバを論理 J2EE サーバとして定義します。タグに指定できる論理サーバの種類を次に示します。 • j2ee-server : J2EE サーバ • performance-tracer : パフォーマンストレーサ • ctm-domain-manager : CTM ドメインマネージャ • component-transaction-monitor : CTM • smart-agent : スマートエージェント 物理ティアの種類によって、コンフィグレーションが定義できる論理サーバが異なります。 j2ee-tier の場合 J2EE サーバとパフォーマンストレーサ ctm-tier の場合 CTM ドメインマネージャ, CTM, スマートエージェント, J2EE サーバおよびパフォーマンストレーサ	◎	◎	—	なし
param※4	論理サーバの環境を設定するパラメタを、パラメタ名と値で定義します。1 種類のパラメタごとに<param>タグで囲んで定義します。	◎	◎	—	なし
param-name※4	論理サーバの環境を設定するパラメタの名称を指定します。	◎	◎	—	なし
param-value※4	<param-name>タグで指定したパラメタへの設定値を指定します。パラメタによっては、複数の値を設定する場合、<param-value>タグを複数指定します。	◎	△※5	—	なし

(凡例)

P : 論理サーバのパラメタ変更用の構成定義変更ファイルを示します。

U : サービスユニット・ホスト追加用の構成定義変更ファイルを示します。

◎ : Web システムを定義する場合は、必ず指定します。

△ : 省略できます。

— : 該当しません。

注

物理ティアの定義では、ユーザサーバ、および free-tier 構成に含まれる論理サーバのコンフィグレーションが設定できません。これらの論理サーバのコンフィグレーションはユニットの定義で設定してください。

注※1

物理ティアの定義を変更しない場合は省略できます。

注※2

Management Server が論理サーバ作成時に仮定する、環境設定値で運用する場合は省略できます。

注※3

cmx_trans_param コマンドを使用して、抽象パラメタを展開して物理ティアのコンフィグレーションを生成する場合は省略できます。

注※4

<param>タグに指定できるパラメタは論理サーバの種類によって異なります。また、定義する論理サーバはシステムの構成パターンによって異なります。「[4.8 システムの構成パターンと定義する論理サーバ](#)」を参照して、定義する論理サーバを確認してください。それぞれの論理サーバに設定できるパラメタについては、使用する論理サーバの種類に応じて 4.12 以降の節を参照してください。

注※5

<param-name>タグで指定したパラメタを削除する場合は省略できます。

(3) サービスユニットの定義

サービスユニットは<unit>以下のタグに定義します。定義できる内容を次の表に示します。

タグ名称	内容	簡易構築 定義ファイル	構成定義変更 ファイル		デフォルト
			P	U	
unit	サービスユニットを定義します。 複数のサービスユニットを定義する場合は、<unit>タグを複数指定します。	◎	△※1	◎	なし
name	サービスユニットを識別するための名称を 32 文字以下で指定します。Web システム内でユニークな名称を指定する必要があります。Smart Composer 機能のコマンドの-unit オプションには、ここで指定した名称を指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。	◎	△※2	◎	なし
display-name	サービスユニットの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	△	△	なし
description	サービスユニットのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	△	△	なし
allocated-host	サービスユニットを構成するホストのリファレンスを定義します。<allocated-host>タグは一つだけ指定します。	◎	◎	◎	なし
host-ref	ホスト定義の<name>タグに指定したホスト名、構築済みの Web システムで定義されているホスト名、または@myhost を指定します。	◎	◎	◎	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	@myhost を指定していると、システム構築時に、ホストの定義の<host-name>タグの値が自動で設定されます。 なお、複数ホストがある場合、一つのサービスユニット内で同じホストを共有することはできません。				
hosts-for	<host-ref>タグで指定したホストが属する物理ティアの種類を指定します。バッチアプリケーションを実行するシステムでは j2ee-tier または ctm-tier のどちらかを指定します。	◎	◎	◎	なし
define-server	<hosts-for>タグで指定した物理ティアに属する論理サーバごとに、コンフィグレーションを定義します。例えば、<hosts-for>タグが j2ee-tier の場合は、バッチサーバ用と、パフォーマンストレーサ用にそれぞれ一つずつ論理サーバを定義できます。複数の論理サーバを定義する場合は、<define-server>タグを複数指定します。	△※3	◎	△※3	なし
logical-server-name	論理サーバまたはクラスタを識別するための名称を 128 文字以下で指定します。Smart Composer 機能のコマンドの-s オプションには、ここで指定した名称を指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。	△	—	△	cmx_<Web システム名>_<サービスユニット名>_<種別>_<通番>
display-name	論理サーバの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	△	△	なし
description	論理サーバのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	△	△	なし
logical-server-type	コンフィグレーションを定義する論理サーバの種類を指定します。バッチアプリケーションを実行するシステムでは、バッチサーバを論理 J2EE サーバとして定義します。タグに指定できる論理サーバの種類を次に示します。 • j2ee-server : J2EE サーバ • performance-tracer : パフォーマンストレーサ • ctm-domain-manager : CTM ドメインマネージャ • component-transaction-monitor : CTM • smart-agent : スマートエージェント • user-server : ユーザサーバ ホストが属する物理ティアの種類によって、コンフィグレーションが定義できる論理サーバが異なります。	◎	◎	◎	なし

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
	j2ee-tier の場合 J2EE サーバ, パフォーマンストレーサおよびユー ザサーバ ctm-tier の場合 CTM ドメインマネージャ, CTM, スマートエージェ ント, J2EE サーバ, パフォーマンストレーサおよ びユーザサーバ				
configuration	論理サーバごとに適用するコンフィグレーションを定 義します。 なお, 物理ティアの定義で定義した環境設定値で運用 する場合は, <configuration>タグの定義は不要です。	△	△	△	なし
param※4	論理サーバの環境を設定するパラメタを, パラメタ名 と値で定義します。1 種類のパラメタごとに<param> タグで囲んで定義します。	◎	◎	◎	なし
param-name※4	論理サーバの環境を設定するパラメタの名称を指定し ます。指定できるパラメタ名については, 「4.8 シス テムの構成パターンと定義する論理サーバ」を参照し てください。	◎	◎	◎	なし
param-value※4	<param-name>タグで指定したパラメタへの設定値 を指定します。パラメタによっては, 複数の値を設定 する場合, <param-value>タグを複数指定します。 それぞれのパラメタの設定値については, 「4.8 シス テムの構成パターンと定義する論理サーバ」を参照し てください。	◎	△※5	◎	なし

(凡例)

P: 論理サーバのパラメタ変更用の構成定義変更ファイルを示します。

U: サービスユニット・ホスト追加用の構成定義変更ファイルを示します。

◎: Web システムを定義する場合は, 必ず指定します。

△: 省略できます。

–: 該当しません。

注

構成変更定義ファイルには, free-tier 構成に含まれる論理サーバのコンフィグレーションを設定できません。

注※1

サービスユニットの定義を変更しない場合は省略できます。

注※2

クライアント設定プロパティファイルまたはクライアント共通設定プロパティファイルに cmx.websystem.name を指定して
いる場合は省略できます。

注※3

論理サーバごとにコンフィグレーションを定義しない場合は省略できます。

注※4

<param>タグに指定できるパラメタは論理サーバの種類によって異なります。また、定義する論理サーバはシステムの構成パターンによって異なります。「4.8 システムの構成パターンと定義する論理サーバ」を参照して、定義する論理サーバを確認してください。それぞれの論理サーバに設定できるパラメタについては、使用する論理サーバの種類に応じて 4.12 以降の節を参照してください。

注※5

<param-name>タグで指定したパラメタを削除する場合は省略できます。

(4) ホストの定義

タグ名称	内容	簡易構築 定義ファイル	構成定義変更 ファイル		デフォルト
			P	U	
host	ホストを定義します。 複数のホストを定義する場合は、host タグを複数指定します。ホストは複数の Web システムで共有できます。	△※1	—	△※2	なし
host-name	ホストを識別するための名称または IP アドレスを 32 文字以下で指定します。 ホスト名に指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。管理 IP アドレスはドット記法 (xxx.xxx.xxx.xxx) で指定します。xxx には 0～255 の整数を指定します。 IP アドレス変換できないホスト名を指定した場合、システム構築時にエラーとなります。 <host-name>タグには@myhost も指定できます。 @myhost を指定すると、システム構築時に、構築先の環境の運用管理サーバマシンのホスト名に自動で置換されます。	◎	—	◎	なし
display-name	ホストの表示名を 128 文字以下で指定します。指定する文字に制限はありません。	△	—	△	なし
description	ホストのコメントを 1,024 文字以下で指定します。指定する文字に制限はありません。	△	—	△	なし
agent-host	運用管理エージェントのホスト名、または IP アドレスを指定します。ホスト名を指定する場合は、32 文字以下で指定します。指定できる文字は、英数字、アンダースコア「_」、ハイフン「-」の組み合わせです。IP アドレスを指定する場合は、ドット記法 (xxx.xxx.xxx.xxx) で指定します。xxx には 0～255 の整数を指定します。 <agent-host>タグには@myhost も指定できます。 @myhost を指定していると、システム構築時に、ホストの定義の<host-name>タグの値が自動で設定されます。	△	—	△	host-name で指定した 値

タグ名称	内容	簡易構築 定義ファ イル	構成定義変更 ファイル		デフォルト
			P	U	
agent-port	運用管理エージェントのポート番号を 1～65535 の間で指定します。	△	－	△	20295

(凡例)

P：論理サーバのパラメタ変更用の構成定義変更ファイルを示します。

U：サービスユニット・ホスト追加用の構成定義変更ファイルを示します。

◎：ホストを定義する場合は、必ず指定します。

△：省略できます。

－：該当しません。

注

ユニットの定義の<host-ref>タグで@myhost を指定して、ホストの定義の<display-name>タグ、<description>タグ、<agent-host>タグ、または<agent-port>タグの値を省略している場合、ホストの定義は出力されません。この場合、ホストの定義の<host-name>タグの値は、Management Server が必要に応じて自動生成します。Management Server が自動作成するホスト名は、Windows の場合はコンピュータ名、UNIX の場合は、hostname コマンドで設定したホスト名です。ホストの中に論理サーバが一つも含まれなくなったとき、Management Server が、そのホストを自動で削除します。

注※1

Smart Composer 機能で定義済みのホストを利用する場合は省略できます。

注※2

ホストの定義を変更しない場合は省略できます。

4.8 システムの構成パターンと定義する論理サーバ

構築するシステムの構成パターンによって、定義する論理サーバの種類が異なります。ここでは、論理サーバに指定できるパラメタについて説明します。

4.8.1 J2EE アプリケーションを実行するシステムを構築する場合

簡易構築定義ファイルおよび構成変更定義ファイルの「物理ティアの定義」および「サービスユニットの定義」では、各論理サーバのコンフィグレーション（<configuration>タグの定義）を指定します。定義する論理サーバごとに、コンフィグレーションで指定できるパラメタが異なります。

また、それぞれの論理サーバに共通で指定できるパラメタがあります。論理サーバ共通のパラメタについては、「[4.9 各論理サーバ共通で指定できるパラメタ](#)」を参照してください。

定義する論理サーバの種類、および指定できるパラメタを表で示します。

表 4-4 定義する論理サーバの種類、および指定できるパラメタの一覧（J2EE アプリケーションの場合）

定義する論理サーバの種類	指定できるパラメタ	参照先
論理 Web サーバ	論理 Web サーバで指定できるパラメタ	4.10
論理 J2EE サーバ	J2EE サーバ用ユーザプロパティを設定するパラメタ	4.11.2
	J2EE サーバ用オプション定義を設定するパラメタ	4.11.3
	J2EE サーバの JavaVM のシステムプロパティに設定できるパラメタ	4.11.4
	J2EE サーバの拡張パラメタ	4.11.5
	運用監視エージェントプロパティを設定するパラメタ	4.11.10
	Management イベント発行用プロパティを設定するパラメタ	4.11.11
	JP1 連携を設定するパラメタ	4.11.12
	SecurityManager の使用を設定するパラメタ	4.11.13
論理パフォーマンストレーサ	論理パフォーマンストレーサで指定できるパラメタ	4.12
論理 CTM ドメインマネージャ	論理 CTM ドメインマネージャで指定できるパラメタ	4.13
論理 CTM	論理 CTM で指定できるパラメタ	4.14
論理スマートエージェント	論理スマートエージェントで指定できるパラメタ	4.15
論理ユーザサーバ	論理ユーザサーバで指定できるパラメタ	4.16

cmx_export_model コマンドを使用して出力した簡易構築定義ファイルの物理ティアの種別が free-tier の場合、論理ネーミングサービスが出力されることがあります。論理ネーミングサービスで出力されるパラメタについては、「[4.17 論理ネーミングサービスで指定できるパラメタ](#)」を参照してください。

4.8.2 バッチアプリケーションを実行するシステムを構築する場合

バッチアプリケーションを実行するシステムでは、次の論理サーバを定義します。

- 論理 J2EE サーバ※
- 論理パフォーマンストレーサ
- 論理 CTM ドメインマネージャ
- 論理 CTM
- 論理スマートエージェント

注※ バッチサーバは論理 J2EE サーバとして定義します。

また、定義する論理サーバごとに、コンフィグレーション（<configuration>タグの定義）で指定できるパラメタが異なります。

また、それぞれの論理サーバに共通で指定できるパラメタがあります。論理サーバ共通のパラメタについては、「[4.9 各論理サーバ共通で指定できるパラメタ](#)」を参照してください。

定義する論理サーバの種類、および指定できるパラメタを表で示します。

表 4-5 定義する論理サーバの種類、および指定できるパラメタの一覧（バッチアプリケーションの場合）

定義する論理サーバの種類	指定できるパラメタ	参照先
バッチサーバとしての論理 J2EE サーバ	バッチサーバ用ユーザプロパティを設定するパラメタ	4.11.6
	バッチサーバ用オプション定義を設定するパラメタ	4.11.7
	バッチサーバの JavaVM のシステムプロパティに設定できるパラメタ	4.11.8
	バッチサーバの拡張パラメタ	4.11.9
	運用監視エージェントプロパティを設定するパラメタ	4.11.10
	Management イベント発行用プロパティを設定するパラメタ	4.11.11
	JP1 連携を設定するパラメタ	4.11.12
	SecurityManager の使用を設定するパラメタ※	4.11.13
論理パフォーマンストレーサ	論理パフォーマンストレーサで指定できるパラメタ	4.12
論理 CTM ドメインマネージャ	論理 CTM ドメインマネージャで指定できるパラメタ	4.13

定義する論理サーバの種類	指定できるパラメタ	参照先
論理 CTM	論理 CTM で指定できるパラメタ	4.14
論理スマートエージェント	論理スマートエージェントで指定できるパラメタ	4.15

注※ バッチサーバの場合、このパラメタは必ず指定してください。

4.9 各論理サーバ共通で指定できるパラメタ

ここでは、次に示す論理サーバで共通に指定できるパラメタについて説明します。

- 論理 Web サーバ
- 論理 J2EE サーバ
- 論理パフォーマンストレーサ
- 論理 CTM ドメインマネージャ
- 論理 CTM
- 論理スマートエージェント
- 論理ユーザサーバ
- 論理ネーミングサービス

4.9.1 論理サーバ共通で指定できるパラメタ

各論理サーバ共通で指定できるパラメタについて次の表に示します。

「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-6 各論理サーバ共通で指定できるパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
additional.startcmd ^{*1}	起動コマンドに追加するオプションを指定します。	英数字、および記号 (\$ %'^{}+_.@:./) を使って指定します。 もしくはハイフン「-」、または (¥¥) が 1～256 回連続する文字列を指定します。	なし	06-50 07-00
mstartup.force.watchtime	論理サーバの強制停止監視時間を秒単位で指定します。0 を指定した場合監視しません。0, 60, 300, 600, 1800 のどれかを指定します。 J2EE サーバの強制停止時にスレッドダンプを出力する場合、強制停止監視時間を、スレッドダンプの出力が終わる時間よりも短く設定すると、スレッドダンプの出力が完全に終わる前に J2EE サーバのプロセスが終了してしまうので注意してください。	0, 60, 300, 600, 1800 のどれかを指定します。	60	07-60
mstartup.no	一括起動時のサーバの起動順序を整数で指定します。一括起動をしない場合は-1 を指定してください。	-1～999 の整数で指定します。	• パフォーマンス：0	07-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
			• スマート エージェント：10 • ネーミング サービス：20 • CTM ドメインマネージャ：30 • CTM：40 • J2EE サーバ：80 • Web サーバ：90 • ユーザサーバ：900	
mstartup.premised.serverName ^{※2}	論理サーバの前提となる論理サーバを指定します。	英数字、アンダースコア「_」、およびハイフン「-」を使って 128 文字以内で指定します。	なし	07-50
mstartup.restartcnt	論理サーバの自動再起動回数を指定します。 0, 1, 5, 10, 1000 のどれかを指定します。	0, 1, 5, 10, 1000 のどれかを指定します。	1	06-50
mstartup.retrywait	論理サーバの自動再起動リトライ間隔を秒単位で指定します。0, 60, 300, 600, 1800 のどれかを指定します。	0, 60, 300, 600, 1800 のどれかを指定します。	60	06-50
mstartup.start.watchtime	論理サーバの起動監視時間を秒単位で指定します。0, 60, 300, 600, 3600 のどれかを指定します。0 を指定した場合は監視しません。	0, 60, 300, 600, 3600 のどれかを指定します。	600	06-50 07-00
mstartup.wathtime	論理サーバの停止監視時間を秒単位で指定します。0, 60, 300, 600, 1800 のどれかを指定します。0 を指定した場合監視しません。 HWSGracefulStopTimeout ディレクティブの設定が必要になる場合があります。マニュアル「アプリケーションサーバ システム構築・運用ガイド」の「4.8.4 HWSGracefulStopTimeout ディレクティブと Manager の停止監視時間の設定に関する注意点」を参照してください。 ^{※6}	0, 60, 300, 600, 1800 のどれかを指定します。	60	06-50
realservername ^{※3}	実サーバ名を指定します。実サーバ名は、同一ホスト内でユニークになるように指定してください。cmx_build_system コマンド実行後は、変更できません。	英数字で始まり、英数字およびアンダースコア「_」、ハイフン「-」で構成された 128 文字以内の文字列を指定します。	論理サーバ名	06-50 07-00

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
user.env.variable	サーバ起動時の環境変数を指定します。※4 値を複数指定する際は、複数の<param-value>で指定します。 環境変数の値は、定義済みの環境変数名を含めて指定できます。定義済みの環境変数名を含めて指定する場合の指定方法の例を次に示します。 PATH=%{PATH};C:¥userlib 定義済みの環境変数は、「%{変数名}」の形式で記述します。「%{変数名}」は、論理サーバの起動時に、定義済みの環境変数の値に置き換えられます。 Windows の場合、環境変数名はサーバ起動時にすべて大文字に変換されます。	任意の文字列を指定します。	なし※5	07-50

注※1

論理ユーザサーバでは指定できません。

複数の起動オプションを追加する場合には、一つの param-value タグ内に複数のオプションを空白で区切って指定します。

(例)

```
<param-name>additional.startcmd</param-name>
```

```
<param-value>-CTMClientConnectCount 128 -CTMEntryCount 256</param-value>
```

注※2

論理パフォーマンストレーサでは指定できません。

注※3

論理 Web サーバおよび論理 J2EE サーバだけで指定できます。

注※4

値を複数指定する際は、複数の<param-value>で指定します。

(例)

```
<param-name>user.env.variable</param-name>
```

```
<param-value>AAA=1024</param-value>
```

```
<param-value>BBB=2048</param-value>
```

論理 Web サーバでリダイレクタ機能を使用する場合、Linux では「LD_LIBRARY_PATH」、Windows では「PATH」環境変数に次のパスを追加する必要があります。

Linux の場合：

```
/opt/Cosminexus/common/lib
```

Windows の場合：

```
<Application Serverのインストールディレクトリ>¥common¥lib
```

注※5

user.env.variable を指定しない場合、論理サーバにはデフォルトの環境変数が設定されます。論理サーバにデフォルトで設定されている環境変数については、マニュアル「アプリケーションサーバ システム構築・運用ガイド」の「4.1.15 論理サーバの環境変数を設定するときの確認事項」を参照してください。

注※6

Web サーバに 0 以外を指定する場合は、Web サーバ用定義の KeepAliveTimeout パラメタの値よりも大きい値を指定してください。Web サーバ用定義については、「[4.10.2 Web サーバ用定義を設定するパラメタ](#)」を参照してください。

4.10 論理 Web サーバで指定できるパラメタ

ここでは、論理 Web サーバで指定できるパラメタについて説明します。

4.10.1 Web サーバの J2EE サーバとの連携方法を設定するパラメタ

Web サーバの J2EE サーバとの連携方法を設定するパラメタについて、次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-7 Web サーバの J2EE サーバとの連携方法を設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
manager.web.send_request_method ^{※1}	Web サーバの J2EE サーバとの連携方法を指定します。次のどちらかを指定します。 reverseproxy: リバースプロキシを使用します。V9 互換モード以外の J2EE サーバと連携する場合に指定します。連携する J2EE サーバが V9 互換モードの場合、リクエストを J2EE サーバに転送できず、リクエスト処理がエラーになります。V9 互換モードの J2EE サーバと連携したい場合は、redirector を指定し、リダイレクタによって連携してください。 redirector: リダイレクタを使用します。V9 互換モードの J2EE サーバと連携する場合に指定します。連携する J2EE サーバが V9 互換モードではない場合、リクエストを J2EE サーバに転送できず、リクエスト処理がエラーになります。V9 互換モードではない J2EE サーバと連携したい場合は、reverseproxy を指定し、リバースプロキシによって連携してください。 reverseproxy を指定し、かつ SetBy パラメタに item を指定した場合、設定の配布時に「4.10.4 リバースプロキシを設定するパラメタ」に記載のパラメタが有効になります。	指定できる文字列を次に示します。 • reverseproxy • redirector	reverseproxy	11-00

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	reverseproxy を指定した場合、SetBy パラメタに text を指定することはできません。 設定ファイルの内容を直接設定したい場合は、redirector を指定する必要があります。※2 Web サーバが稼働するホストの OS が UNIX の場合、reverseproxy 指定時 worker MPM モジュールが組み込まれ、redirector 指定時 prefork MPM モジュールが組み込まれます。worker MPM、prefork MPM の詳細はマニュアル「HTTP Server」を参照してください。 Web サーバが稼働するホストの OS が UNIX の場合、設定後に値を変更する際は事前に「hwsserveredit -delete」コマンドによって Web サーバの環境を削除してください。			

注※1

Web サーバ構築後にパラメタの内容は変更できません。

このパラメタを指定することで次に示す項目に影響があります。

- このパラメタの指定内容によっては、Web サーバ用定義を設定するパラメタの指定内容が、有効になるものとならないものがあります。Web サーバ用定義については「[4.10.2 Web サーバ用定義を設定するパラメタ](#)」を参照してください。
- 運用管理ポータル「論理サーバの環境設定」に表示される画面が異なります。

reverseproxy:

[リダイレクタの設定] 画面、[マッピングの定義] 画面、[ワーカの設定] 画面が表示されません。

redirector:

[リバースプロキシの設定] 画面が表示されません。

運用管理ポータル「論理サーバの環境設定」に表示される画面については、マニュアル「アプリケーションサーバ運用管理ポータル操作ガイド」の「10.9 論理 Web サーバの定義」を参照してください。

注※2

redirector を指定し、J2EE サーバが V9 互換モードでない場合でも、次の方法でリバースプロキシによって連携できます。

- SetBy パラメタに text を指定します。
- AllText パラメタに、リバースプロキシに関する設定内容を含んだ httpsd.conf ファイル (HTTP Server 定義ファイル) の内容を指定します。

リバースプロキシに関する設定内容については、マニュアル「HTTP Server」の「4.7 リバースプロキシの設定」を参照してください。

4.10.2 Web サーバ用定義を設定するパラメタ

Web サーバ用定義を設定するパラメタについて、次の表に示します。

なお、「param-name 指定値」に対応する param-value の指定内容については、マニュアル「HTTP Server」のディレクティブ一覧についての説明を参照してください。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-8 Web サーバ用定義を設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
SetBy	Web サーバの設定方法を指定します。 item：項目ごとに設定します。 text：httpsd.conf（HTTP Server 定義ファイル）の内容を直接設定します。 J2EE サーバ連携方法（manager.web.send_request_method）がリバースプロキシの場合は、「text」を指定しないでください。設定ファイルの内容を直接設定したい場合は、J2EE サーバ連携方法としてリダイレクタを選択してください。 リダイレクタを選択した場合でも、次の方法でリバースプロキシによって連携することができます。 - SetBy パラメタに text を指定します。 - AllText パラメタに、リバースプロキシに関する設定内容を含んだ httpsd.conf ファイル（HTTP Server 定義ファイル）の内容を指定します。 リバースプロキシに関する設定内容については、マニュアル「HTTP Server」の「4.7 リバースプロキシの設定」を参照してください。	指定できる文字列を次に示します。 - item - text	item	08-70
CoreDump Directory	コアをダンプするディレクトリを指定します。絶対パス、または ServerRoot ディレクティブの指定値からの相対パスが指定できます。なお、指定したディレクトリには、User、Group ディレクティブで指定したユーザ、グループからの書き込み権限を付与する必要があります。 なお、この指定値は UNIX の場合だけ有効となります。	任意の文字列を指定します。	&{hws.home}/servers/HWS_<サーバ名称>	06-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
Group	サーバプロセスが動作するときのグループ名を指定します。 なお、この指定値は UNIX の場合だけ有効となります。	英数字、およびアンダースコア「_」を使って 16 文字以内で指定します。	bin	06-50 07-00
HWSKeepStartServers	サーバプロセスの稼働数を StartServers ディレクティブに指定した数だけ維持するかどうかを指定します。 On を指定した場合： StartServers ディレクティブに指定した数だけ、稼働しているサーバプロセスが維持されます。サーバプロセス数が StartServers ディレクティブ指定値より小さくなった場合、新しいプロセスが生成されます。 • prefork MPM この機能は、プロセス数に関する各ディレクティブの指定値が、次の関係にある場合に有効です。 $\text{MinSpareServers} < \text{StartServers} \leq \text{MaxClients}$ かつ $\text{MinSpareServers} < \text{MaxSpareServers} \leq \text{MaxClients}$ StartServers ディレクティブ設定値が、MinSpareServers ディレクティブ設定値より小さい場合は、MinSpareServers ディレクティブの値でサーバプロセス数が維持されます。 • worker MPM この機能は、プロセス数およびスレッド数に関する各ディレクティブの指定値が、次の関係にある場合に有効です。 $\text{MinSpareThreads} < \text{StartServers} \times \text{ThreadsPerChild} \leq \text{MaxClients}$ かつ $\text{MinSpareThreads} < \text{MaxSpareThreads} \leq \text{MaxClients}$ StartServers ディレクティブ $\times$ ThreadsPerChild ディレクティブの値が、MinSpareThreads ディレクティブ設定値より小さい場合は、MinSpareThreads ディレクティブ設定値に従ってサーバプロセス数が維持されます。 Off を指定した場合： StartServers ディレクティブに指定した数の稼働しているサーバプロセスは維持されません。	指定できる文字列を次に示します。 • On • Off	Off	06-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	プロセス数に関連するほかのディレクティブについては、マニュアル「HTTP Server」を参照してください。 なお、この指定値は UNIX の場合だけ有効となります。			
HWSLogTimeVerbose	エラーログとリクエストログの時刻、アクセスログのリクエスト処理に掛かった時間 (%T)、およびリクエスト処理を開始した時刻 (%t) をミリ秒まで表示するかどうかを指定します。 On を指定した場合： 時刻および時間をミリ秒まで表示します。 Off を指定した場合： 時刻および時間を秒まで表示します。 なお、エラーログは ErrorLog ディレクティブで指定するエラーログが対象になります。ScriptLog ディレクティブで指定する CGI スクリプトのエラーログは対象になりません。	指定できる文字列を次に示します。 • On • Off	On	07-50
KeepAliveTimeout	KeepAlive 接続時の要求待ち時間を秒単位で指定します。この時間以上経過しても、クライアントから次のリクエストが来ない場合、コネクションを切断します。KeepAlive はサーバプロセスが特定のクライアントに占有されます。ある Web ページから次の Web ページへ移る場合に必要とする標準的な時間以上は、タイムアウトにしてコネクションを切断し、サーバプロセスをほかのリクエストの処理に当てるようにします。 ディレクティブについては、マニュアル「HTTP Server」を参照してください。	0~65535 の整数で指定します。	3	06-50
ServerName	HTTP Server のサーバ名およびポート番号を指定します。 サーバ名は、FQDN（完全修飾ドメイン名）または IP アドレスで指定します。 指定内容の詳細については、マニュアル「HTTP Server」を参照してください。	サーバ名およびポート番号	www.example.com	08-70
Listen	リクエストを受け付ける IP アドレス、およびポート番号を指定します。Port ディレクティブと異なり、複数指定できます。バーチャルホストを定義する場合に指定します。Listen ディレクティブを指定すると、Port ディレクティブ、および BindAddress ディレクティブの指定は無視されます。 IP アドレスには IPv6 アドレスも指定できます。IPv6 アドレスを指定する場合は、IPv6 アドレスを [] で囲んでください。ただし、IP アドレスを省略してポート番号だけを指定した場合は、IPv4 アドレス	指定できる値を次に示します。 • IPv4 アドレス※3:ポート番号 • ホスト名※3:ポート番号 • @myhost:ポート番号	なし	07-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	スを使用したリクエストだけを受け付けます。このため、IPv6 アドレスを使用する場合は、必ず Listen ディレクティブに IPv6 アドレスを指定してください。			
LogLevel	エラーログに出力するエラーのレベルを指定します。指定したレベルの上位レベルのログを出力します。ただし、notice レベルのログはこの指定に関係なく出力されます。また、HTTP Server 起動時など、レベル指定の解析終了前に出力されるメッセージは、この指定に関係なく出力される場合があります。次にエラーレベルを上位順に示します。 • emerg：緊急メッセージ • alert：即時処理要求メッセージ • crit：致命的な状態のメッセージ • error：一般的エラーメッセージ • warn：警告レベルメッセージ • notice：標準的だが重要なメッセージ • info：インフォメーションメッセージ、外部モジュールと CGI プログラム実行時のモジュールトレース※¹ • debug：デバッグレベルメッセージ、内部モジュールトレース、および info 相当のモジュールトレース※¹	指定できる文字列を次に示します。 • emerg • alert • crit • error • warn • notice • info • debug	info	06-50
MaxClients	同時に接続できるクライアントの最大数を指定します。 サーバを起動すると、StartServer ディレクティブで指定した数のプロセスが起動されリクエストを待ちます。多くのリクエストが同時に発生した場合、複数のプロセスでリクエストを処理することになります。リクエスト待ちの残りプロセス数が MinSpareServers ディレクティブで指定した数より少なくなると、徐々に新規プロセスを生成します。このとき、プロセス数がこのディレクティブで指定した数になるまでプロセスが生成されます。その後、リクエストの処理が終了しリクエスト待ちプロセスが増加すると、MaxSpareServers ディレクティブで指定した数までプロセスを終了させます。 プロセス数に関連するほかのディレクティブについては、マニュアル「HTTP Server」を参照してください。 なお、この指定値は UNIX の場合、かつ manager.web.send_request_method パラメタが redirector の場合だけ有効となります。	1～1024 の整数で指定します。	150	06-50
StartServers	Web サーバ起動時のサーバプロセス数を指定します。プロセス数に関連するほかのディレクティブに	0～1024 の整数で指定します。	5	06-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	については、マニュアル「HTTP Server」を参照してください。 なお、この指定値は UNIX の場合、かつ manager.web.send_request_method パラメタが redirector の場合だけ有効となります。			
ThreadsPerChild	サーバとして起動するスレッド数を指定します。指定したスレッド数はサーバの最大同時接続数を示します。 ディレクティブについては、マニュアル「HTTP Server」を参照してください。 なお、この指定値は Windows の場合だけ有効となります。	1～15000 の整数で指定します。	50	06-50
User	サーバプロセスが動作するときのユーザ名を指定します。 なお、この指定値は UNIX の場合だけ有効となります。	英数字、およびアンダースコア「_」を使って16文字以内で指定します。	bin	06-50 07-00
DocumentRoot	コンテンツを格納するドキュメントルートディレクトリを絶対パスで指定します。 ディレクティブについては、マニュアル「HTTP Server」を参照してください。	任意の文字列を指定します。	&{hws.home}/htdocs	08-00
AppendDirectives	Web サーバを項目ごとに設定する（SetBy パラメタに item を指定した）場合に、追加するディレクティブの内容を指定します。※2 マルチバイト文字は使用できません。指定できるディレクティブについては、マニュアル「HTTP Server」を参照してください。 また、PidFile ディレクティブは追加ディレクティブに指定しないようにしてください。指定した場合、論理 Web サーバの起動が失敗します。なお、ここに指定した内容は、Management Server によって生成されるファイルに出力されます。また、httpsd.conf ファイル（HTTP Server 定義ファイル）の最後には、次の Include ディレクティブが Management Server によって追加されます。 • Include < HTTP Server のインストールディレクトリ > /servers/HWS_ < 論理 Web サーバの実サーバ名 > /conf/httpsd_manager.conf 複数指定できないディレクティブを指定した場合も上記ファイルに追加されます。 httpsd.conf ファイル（HTTP Server 定義ファイル）と httpsd_manager.conf の内容を確認し、追加したディレクティブが重複していて重複指定できないディレクティブがある場合は Web サーバ用定義を変更して該当ディレクティブを削除してください。	任意の文字列を指定します。	なし	08-70

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	い。複数指定できないディレクティブについては、マニュアル「HTTP Server」を参照してください。			
AllText※4	Web サーバのファイルを直接設定する（SetBy パラメタに text を指定した）場合に、httpsd.conf（HTTP Server 定義ファイル）の内容を指定します。※2httpsd.conf（HTTP Server 定義ファイル）については、マニュアル「HTTP Server」を参照してください。	任意の文字列を指定します。	なし	08-70
HWSPrfId	PRF デーモン起動時に PRF 識別子に指定した文字列を指定します。ただし、リダイレクタを使用する場合はこのディレクティブを指定しても無視されます。	英数字+記号<0～31>「CTM」,「ctm」,「TSC」,または「tsc」で始まる文字列を指定した場合、エラーになります。記号は_（アンダーバー）だけ指定できます。	<この Web サーバと同一サービスユニットの PRF デーモンの PRF 識別子>	11-00
MaxClients.worker	同時に接続できるクライアントの最大数（MaxClients ディレクティブ）を指定します。 サーバを起動すると、StartServers ディレクティブで指定した数のプロセスが起動されリクエストを待ちます。 多くのリクエストが同時に発生した場合、複数のスレッドでリクエストを処理することになります。リクエスト待ちの残りスレッド数が MinSpareThreads ディレクティブで指定した数より少なくなると、新しいプロセスを生成します。このとき、スレッド数がこのディレクティブで指定した数になるまでプロセスが生成されます。その後、リクエストの処理が終了しリクエスト待ちスレッドが増加すると、MaxSpareThreads ディレクティブで指定した数以下になるまでプロセスを終了させます。 ディレクティブについては、マニュアル「HTTP Server」を参照してください。 なお、この指定値は UNIX の場合、かつ manager.web.send_request_method パラメタが reverseproxy の場合だけ有効となります。	ThreadsPerChild.worker ~ (1000× ThreadsPerChild.worker) の整数で指定します。	400	11-00
ThreadsPerChild.worker	一つのサーバプロセスに生成するサーバスレッド数（ThreadsPerChild ディレクティブ）を指定します。ディレクティブについては、マニュアル「HTTP Server」を参照してください。 なお、この指定値は UNIX の場合、かつ manager.web.send_request_method パラメタが reverseproxy の場合だけ有効となります。	1～1000 の整数で指定します。	40	11-00

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
StartServers. worker	Web サーバ起動時のサーバプロセス数 (StartServers ディレクティブ) を指定します。 ディレクティブについては、マニュアル「HTTP Server」を参照してください。 なお、この指定値は UNIX の場合、かつ manager.web.send_request_method パラメタが reverseproxy の場合だけ有効となります。	0～ (MaxClients.worker ÷ ThreadsPerChild.worker) の整数で指定します。	2	11-00

注※1

モジュールトレースは、エラーログではなくリクエストログに出力するよう設定できます。詳細は、マニュアル「HTTP Server」を参照してください。

注※2

param-value の値は CDATA セクションで指定してください。

(例)

```
<param-name>AppendDirectives</param-name>
<param-value>
  <![CDATA[
    <Location /server-status>
      :
    </Location>
  ]]>
</param-value>
```

注※3

IPv4 アドレスまたはホスト名を指定する場合は、ホストの定義の<host-name>タグの値を指定してください。異なる値を指定した場合、警告メッセージ (KEOS24186-W) が出力され、意図しない設定となることがあります。

ホスト名は、英数字およびアンダースコア「_」、ピリオド「.」、ハイフン「-」で構成された 255 文字以内の文字列です。

ポート番号は 1～65535 の半角数字です。

注※4

複数の論理 Web サーバを構築した場合、PidFile ディレクティブは各論理 Web サーバごとにユニークな値を指定する必要があります。

4.10.3 HTTP Server のログの出力方法を設定するパラメタ

HTTP Server のログの出力方法を設定するパラメタについて、次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-9 HTTP Server のログの出力方法を設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
CustomDivideFileNum	HttpsCustomMethod で「Div」を設定したときに有効になります。 分割したログファイルのファイル数を指定します。分割したファイル数がここで指定した数を超えた場合、最も古いファイルから削除されます。0 を指定した場合、ファイルは削除されません。	0～256 の整数で指定します。	8	07-50
CustomDivideTimeDifference	HttpsCustomMethod で「Div」を設定したときに有効になります。 ログファイルを分割する基準となる時間（単位：分）を、GMT に対する差として-1439～1439 の範囲で指定します。0 を指定すると、1970 年 1 月 1 日 0 時 0 分 0 秒（GMT）が基準時間となります。	-1439～1439 の整数で指定します。	540	07-50
CustomDivideTimeInterval	HttpsCustomMethod で「Div」を設定したときに有効になります。 一つのログファイルで採取する時間間隔を 1～31536000 の範囲で指定します。	1～31536000 の整数で指定します。	86400	07-50
CustomWraparoundFileNum	HttpsCustomMethod で「Wrap」を設定したときに有効になります。 出力するログファイルの最大数を 1～256 の範囲で指定します。 ログファイルのサイズが CustomWraparoundFilesize で指定したサイズを超えた場合、ログの出力先は次のログファイルに移ります。このとき、すでにこのパラメタに指定した最大数と同じ数のログファイルが作成されていた場合、「.001」という名称がついているファイルから再度使用されます。	1～256 の整数で指定します。	5	07-50
CustomWraparoundFilesize	HttpsCustomMethod で「Wrap」を設定したときに有効になります。 ログファイルの最大サイズ（単位：キロバイト）を 1～2097151 の範囲で指定します。 ログを出力するときに、ログファイルのサイズがこのパラメタで指定した最大サイズを超えた場合、ログの出力先は次のログファイルに移ります。このとき、移動先のログファイルの内容はクリアされます。	1～2097151 の整数で指定します。	8192	07-50
ErrorDivideFileNum	HttpsErrorMethod で「Div」を設定したときに有効になります。 分割したログファイルのファイル数を 0～256 の範囲で指定します。分割したファイル数がここで指定した数を超えた場合、最も古いファイルから削除されます。0 を指定した場合、ファイルは削除されません。	0～256 の整数で指定します。	8	07-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
ErrorDivideTimeDifference	HttpsErrorMethod で「Div」を設定したときに有効になります。 ログファイルを分割する基準となる時間（単位：分）を、GMT に対する差として-1439～1439 の範囲で指定します。0 を指定すると、1970 年 1 月 1 日 0 時 0 分 0 秒（GMT）が基準時間となります。	-1439～1439 の整数で指定します。	540	07-50
ErrorDivideTimeInterval	HttpsErrorMethod で「Div」を設定したときに有効になります。 一つのログファイルで採取する時間間隔を 1～31536000 の範囲で指定します。	1～31536000 の整数で指定します。	86400	07-50
ErrorWraparoundFileNum	HttpsErrorMethod で「Wrap」を設定したときに有効になります。 出力するログファイルの最大数を 1～256 の範囲で指定します。 ErrorWraparoundFilesize で指定したサイズを超えた場合、ログの出力先は次のログファイルに移ります。このとき、すでにこのパラメタに指定した最大数と同じ数のログファイルが作成されていた場合、「.001」という名称がついているファイルから再度使用されます。	1～256 の整数で指定します。	5	07-50
ErrorWraparoundFilesize	HttpsErrorMethod で「Wrap」を設定したときに有効になります。 ログファイルの最大サイズ（単位：キロバイト）を 1～2097151 の範囲で指定します。 ログを出力するときに、ログファイルのサイズがこのパラメタで指定した最大サイズを超えた場合、ログの出力先は次のログファイルに移ります。このとき、移動先のログファイルの内容はクリアされます。	1～2097151 の整数で指定します。	8192	07-50
HttpsCustomLogFileDir	Web サーバが出力するアクセスログの出力先ディレクトリを指定します。	任意の文字列を指定します。	logs	07-50
HttpsCustomLogFormat	HTTP Server が提供するラベル名または任意のフォーマットを指定します。 HTTP Server が提供するラベル名と対応するフォーマットは次のとおりです。 • common • combined • combinedio • hws_std • hws_trace これ以外のフォーマットを指定する場合は、任意のフォーマットを指定してください。フォーマットで指定した内容が、httpsd.conf（HTTP Server 定義	任意の文字列を 1024 文字以内で指定します。	hws_std	07-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	ファイル) の CustomLog ディレクティブの引数に設定されます。 フォーマットの指定内容の詳細については、マニュアル「HTTP Server」の CustomLog ディレクティブに関する説明を参照してください。 UNIX の場合、かつ manager.web.send_request_method パラメタが reverseproxy の場合、直接 httpsd.conf を編集する際、デフォルトで指定されている LogFormat ディレクティブを削除しないでください。			
HttpsdcustomMethod	アクセスログの出力方式として、次のどれかを指定します。 Off : httpsd.conf (HTTP Server 定義ファイル) の設定を変更しません。 httpsd.conf を編集していない場合、省略値が設定されます。 On : 単調増加でログを出力します。 Div : 一定時間単位にログを分割して出力します。 Wrap : ファイルサイズでログを分割してラップアラウンドに出力します。	指定できる文字列を次に示します。 • Off • On • Div • Wrap	Div	07-50
HttpsdcustomLogFileDir	Web サーバの出力するエラーログの出力先ディレクトリを指定します。	任意の文字列を指定します。	logs	07-50
HttpsdcustomLogMethod	エラーログの出力方式として、次のどれかを指定します。 Off : httpsd.conf (HTTP Server 定義ファイル) の設定を変更しません。httpsd.conf を編集していない場合、「Wrap」が設定されます。 On : 単調増加でログを出力します。 Div : 一定時間単位にログを分割して出力します。 Wrap : ファイルサイズでログを分割してラップアラウンドに出力します。	指定できる文字列を次に示します。 • Off • On • Div • Wrap	Wrap	07-50
HttpsdcustomLogFileDir	Web サーバの出力するログの出力先ディレクトリを指定します。	任意の文字列を指定します。	logs	06-50 07-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
HttpsRequestLogFileDir	Web サーバの出力するリクエストログの出力先ディレクトリを指定します。	任意の文字列を指定します。	logs	07-50
HttpsRequestMethod	リクエストログの出力方式として、次のどれかを指定します。 Off : httpsd.conf (HTTP Server 定義ファイル) の設定を変更しません。httpsd.conf を編集していない場合、省略値が設定されます。 On : 単調増加でログを出力します。 Div : 一定時間単位にログを分割して出力します。 Wrap : ファイルサイズでログを分割してラップアラウンドに出力します。	指定できる文字列を次に示します。 • Off • On • Div • Wrap	Div	07-50
HWSRequestIoFilter	HWSRequestLogLevel で「on」を設定したときに有効になります。 モジュールが実装している入出力フィルタ関数の、実行契機を示す I/O フィルタトレースを出力する場合に指定します。出力量が多いため、実運用での指定は推奨しません。 次のどちらかを指定します。 true : 出力します。 false : 出力しません。	指定できる文字列を次に示します。 • true • false	false	07-50
HWSRequestLogLevel	リクエストログに出力するかどうかを指定します。次のどちらかを指定します。 on : 出力します。 none : 出力しません。 また、このパラメタで「on」を設定した場合、次に示すパラメタの設定が有効になります。 • HWSRequestIoFilter • HWSRequestModuleDebug • HWSRequestModuleInfo • HWSRequestRequest デフォルトでは、次に示すパラメタが出力されます。 • HWSRequestModuleInfo	指定できる文字列を次に示します。なお、大文字と小文字は区別されません。 • on • none	On	07-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	HWSRequestRequest			
HWSRequestModuleDebug	HWSRequestLogLevel で「on」を設定したときに有効になります。 内部モジュールに対するモジュールトレースと、外部モジュール相当のトレースを出力する場合に指定します。出力量が多いため、実運用での指定は推奨しません。 次のどちらかを指定します。 true : 出力します。 false : 出力しません。	指定できる文字列を次に示します。 - true - false	false	07-50
HWSRequestModuleInfo	HWSRequestLogLevel で「on」を設定したときに有効になります。 外部モジュールに対するモジュールトレースを出力する場合に指定します。 次のどちらかを指定します。 true : 出力します。 false : 出力しません。	指定できる文字列を次に示します。 - true - false	true	07-50
HWSRequestRequest	HWSRequestLogLevel で「on」を設定したときに有効になります。 次の場合に指定します。 - コネクション確立後とレスポンス完了後にトレースを出力したい場合 - KeepAlive 接続の場合に、次のリクエストライン受信後にもトレースを出力したい場合 次のどちらかを指定します。 true : 出力します。 false : 出力しません。	指定できる文字列を次に示します。 - true - false	true	07-50
HWSRequestProxy	HWSRequestLogLevel で「on」を設定したときに有効になります。 リバースプロキシ機能を使用している場合にリバースプロキシの処理状況を把握するためのプロキシトレースを出力するときに指定します。 次のどちらかを指定します。 true : 出力します。	指定できる文字列を次に示します。 - true - false	true	11-00

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	false : 出力しません。			
RequestDivideFileNum	HttpsRequestMethod で「Div」を設定したときに有効になります。 分割したログファイルのファイル数を 0～256 の範囲で指定します。分割したファイル数がここで指定した数を超えた場合、最も古いファイルから削除されます。0 を指定した場合、ファイルは削除されません。	0～256 の整数で指定します。	8	07-50
RequestDivideTimeDifference	HttpsRequestMethod で「Div」を設定したときに有効になります。 ログファイルを分割する基準となる時間（単位：分）を、GMT に対する差として-1439～1439 の範囲で指定します。0 を指定すると、1970 年 1 月 1 日 0 時 0 分 0 秒（GMT）が基準時間となります。	-1439～1439 の整数で指定します。	540	07-50
RequestDivideTimeInterval	HttpsRequestMethod で「Div」を設定したときに有効になります。 一つのログファイルで採取する時間間隔を 1～31536000 の範囲で指定します。	1～31536000 の整数で指定します。	86400	07-50
RequestWraparoundFileNum	HttpsRequestMethod で「Wrap」を設定したときに有効になります。 出力するログファイルの最大数を 1～256 の範囲で指定します。 ログファイルのサイズが RequestWraparoundFilesize で指定したサイズを超えた場合、ログの出力先は次のログファイルに移ります。このとき、すでにこのパラメタに指定した最大数と同じ数のログファイルが作成されていた場合、「.001」という名称がついているファイルから再度使用されます。	1～256 の整数で指定します。	5	07-50
RequestWraparoundFilesize	HttpsRequestMethod で「Wrap」を設定したときに有効になります。 ログファイルの最大サイズ（単位：キロバイト）を 1～2097151 の範囲で指定します。 ログを出力するときに、ログファイルのサイズがこのパラメタで指定した最大サイズを超えた場合、ログの出力先は次のログファイルに移ります。このとき、移動先のログファイルの内容はクリアされます。	1～2097151 の整数で指定します。	8192	07-50

4.10.4 リバースプロキシを設定するパラメタ

リバースプロキシを設定するパラメタについて、次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

注

リバースプロキシを設定するパラメタは、manager.web.send_request_method パラメタが reverseproxy かつ SetBy パラメタが item の場合だけ有効です。

表 4-10 リバースプロキシを設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
manager.web.reverseproxy.mapping	リバースプロキシのリクエスト転送元と転送先のマッピングを指定します。 値は次のように空白で区切り指定します。 <転送元パス> <転送先プロトコル> <転送先 J2EE サーバ> [<通信タイムアウトの設定>] 転送元パス: J2EE サーバへ転送するリクエストのパスを「/ (スラッシュ)」から始まる URL で指定します。 同一のパスを複数指定することはできません。同一のパスを複数指定した場合はメッセージ KEOS24506-W を出力し、指定は無視します。 転送先プロトコル: J2EE サーバにリクエストを転送する際のプロトコルを指定します。「http」または「ws」を指定します。 転送先 J2EE サーバ: リクエスト転送先 J2EE サーバの論理サーバ名を指定します。存在しない J2EE サーバの論理サーバ名を指定した場合はメッセージ KEOS24507-W を出力し、指定は無視します。 値を複数指定した場合、KEOS24507-W が出力されていても、正しい値については有効になります。 存在する J2EE サーバの論理サーバ名の指定が一つもない場合、省略値が設定されます。	指定できる文字列を次に示します。 - 転送元パス 「/ (スラッシュ)」から始まる文字列 - 転送先プロトコル 「http」または「ws」 - 転送先 J2EE サーバ J2EE サーバの論理サーバ名 - 通信タイムアウトの設定 次の形式で指定します。 timeout=<通信タイムアウト> <通信タイムアウト>は 1~65535 の整数で指定します。	/ http <この Web サーバと同一サービスユニットの J2EE サーバの論理サーバ名>	11-00

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	通信タイムアウトの設定: J2EE サーバとの送受信時の待ち時間（単位は秒）を次の形式で指定します。 timeout=<通信タイムアウト> 指定は省略できます。 設定した内容は、次の Web サーバのディレクティブに展開されます。 ProxyPass <パス名> <URL> <通信タイムアウトの設定> HWSPProxyPassReverseCookie <パス名> パス名: 「転送元パス」で指定した値。 URL: <プロトコル>://<ホスト>:<ポート><パス> ここで、プロトコル、ホスト、ポート、およびパスは次の値。 ・プロトコル: 「転送先プロトコル」で指定した値。 ・ホスト: この Web サーバと同一サービスユニットの J2EE サーバのホスト。 ・ポート: この Web サーバと同一サービスユニットの J2EE サーバの NIO HTTP サーバのポート番号。 ・パス: 「転送元パス」で指定した値。 通信タイムアウトの設定: timeout=<「通信タイムアウトの設定」> 「通信タイムアウトの設定」の指定を省略した場合は、timeout キーを出力しません。 システム構築時に、リバースプロキシ設定ファイル (httpd_reverseproxy.conf) が、httpd.conf と同じディレクトリに生成され、リバースプロキシに関する設定が書き出されます。httpd.conf には、Include ディレクティブによって httpd_reverseproxy.conf をインクルードする設定が追加されます。			

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	リバースプロキシ設定ファイル (httpsd_reverseproxy.conf) に出力されるディレクティブの詳細を次に示します。 LoadModule proxy_module modules/mod_proxy.so (Windows 版) LoadModule proxy_http_module modules/mod_proxy_http.so (Windows 版) LoadModule proxy_wstunnel_module modules/mod_proxy_wstunnel.so (Windows 版) LoadModule proxy_module libexec/mod_proxy.so (UNIX 版) LoadModule proxy_http_module libexec/mod_proxy_http.so (UNIX 版) LoadModule proxy_wstunnel_module libexec/mod_proxy_wstunnel.so (UNIX 版) HWSPrfId <PRFID> ProxyErrorOverride <On Off> ProxyPreserveHost On ProxyPass <パス名> <URL> <通信タイムアウトの設定> HWSProxyPassReverseCookie <パス名> これらのディレクティブは定義ファイルの順序どおりに出力されます。リクエストのマッピングを複数定義する場合は、順序に意味があるため注意が必要となります。詳細については、マニュアル「HTTP Server」を参照してください。			
ProxyErrorOverride	J2EE サーバからのレスポンスヘッダとレスポンスボディをオーバーライドするかどうかを指定します。詳細については、マニュアル「HTTP Server」を参照してください。	指定できる文字列を次に示します。 • On • Off	なし	11-00

4.11 論理 J2EE サーバで指定できるパラメタ

ここでは、論理 J2EE サーバで指定できるパラメタについて説明します。

論理 J2EE サーバでは、J2EE サーバとバッチサーバのどちらを使用するかで指定できるパラメタが異なります。指定できるパラメタとパラメタの参照先について、サーバごとに次の表に示します。

表 4-11 使用するサーバと指定するパラメタの参照先の対応

使用するサーバ	指定するパラメタ
J2EE サーバ	J2EE サーバの互換モードを設定するパラメタ (4.11.1 参照)
	J2EE サーバ用ユーザプロパティを設定するパラメタ (4.11.2 参照)
	J2EE サーバ用オプション定義を設定するパラメタ (4.11.3 参照)
	J2EE サーバの JavaVM のシステムプロパティに設定できるパラメタ (4.11.4 参照)
	J2EE サーバの拡張パラメタ (4.11.5 参照)
	統合ユーザ管理の使用を設定するパラメタ (4.11.14 参照)
バッチサーバ	バッチサーバ用ユーザプロパティを設定するパラメタ (4.11.6 参照)
	バッチサーバ用オプション定義を設定するパラメタ (4.11.7 参照)
	バッチサーバの JavaVM のシステムプロパティに設定できるパラメタ (4.11.8 参照)
	バッチサーバの拡張パラメタ (4.11.9 参照)
J2EE サーバとバッチサーバ	運用監視エージェントプロパティを設定するパラメタ (4.11.10 参照)
	Management イベント発行用プロパティを設定するパラメタ (4.11.11 参照)
	JP1 連携を設定するパラメタ (4.11.12 参照)
	SecurityManager の使用を設定するパラメタ (4.11.13 参照)
	JavaVM の起動パラメタで使用するファイルを設定するパラメタ (4.11.15 参照)

4.11.1 J2EE サーバの互換モードを設定するパラメタ

J2EE サーバの互換モードを設定するパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-12 J2EE サーバの互換モードを設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
manager.j2ee.compat	J2EE サーバの互換モードを使用するかどうかを指定します。 none: V9 互換モードを使用しない V9: V9 互換モードを使用する JDK17 以降の場合、V9 互換モードは非サポートのため、V9 を指定できません。 JDK17 以降で V9 を指定した場合、エラーメッセージ KEOS24015-E を出力し、リターンコード 64 でコマンドを終了します。 設定後に値を変更する場合は、事前に「cjsetup -d」コマンドで、J2EE サーバの環境を削除します。	指定できる文字列を次に示します。 • none • V9 (JDK17 以降の場合、指定できません)	none	11-00

4.11.2 J2EE サーバ用ユーザプロパティを設定するパラメタ

J2EE サーバ用ユーザプロパティを設定するパラメタについて説明します。

「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3 usrconf.properties \(J2EE サーバ用ユーザプロパティファイル\)](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

なお、「param-value の指定内容」が記載されているパラメタは、usrconf.properties (J2EE サーバ用ユーザプロパティファイル) と指定方法が異なるパラメタです。

J2EE サーバ用ユーザプロパティを設定するパラメタを指定する場合の指定形式を次に示します。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
```

usrconf.properties (J2EE サーバ用ユーザプロパティファイル) に指定できるパラメタのうち、「[4.11.4 J2EE サーバの JavaVM のシステムプロパティに設定できるパラメタ](#)」の表と、この項の表の両方に記載していないパラメタは、次の形式で指定してください。ただし、統合ユーザ管理の使用を設定するパラメタについては、「[4.11.14 統合ユーザ管理の使用を設定するパラメタ](#)」を参照して設定してください。

指定形式

```
<param-name>ex.properties</param-name>
<param-value>パラメタ=値</param-value>
```

値を複数指定する場合は、複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>ex.properties</param-name>
<param-value>パラメタ=値</param-value>
<param-value>パラメタ=値</param-value>
```

なお、この項に記載があるパラメタのうち、次のパラメタは、この形式で指定することもできます。それ以外のパラメタをこの形式で指定した場合、動作が保証されません。また、<param-name>パラメタ</param-name>の形式とこの形式の両方で同じパラメタを指定した場合、動作が保証されません。

- 運用管理ポータル画面に対応しないパラメタ
- 運用管理ポータル画面に対応し、画面名が「システムプロパティの設定」、設定個所が「システムプロパティ」であるパラメタ

パラメタと運用管理ポータル画面の対応については、マニュアル「アプリケーションサーバ 運用管理ポータル操作ガイド」を参照してください。

(1) ejbserver.application から始まるパラメタ

ejbserver.application から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(a\) ejbserver.application から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-13 ejbserver.application から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.application.InitTermProcessClasses ^{※1}	任意の文字列を指定します。	なし	06-50
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.appname	次に示す文字を使って 0～16 バイトの半角文字セットで指定します。 • 英数字 • ピリオド「.」 • ドル「\$」 • アンダースコア「_」 • ストローク「 」 • コロン「:」 • アンパサンド「&」 • 単価記号「@」 • シャープ「#」 • パーセント「%」 • ハイフン「-」	user_app	06-50 07-00
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.count	【CJMessageFileHandler を使用した場合】 2～16	2	06-50 07-00

param-name 指定値	指定可能値	省略値	VR
	【CJMPMessageFileHandler を使用した場合】 2～64		
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.encoding	次に示す文字を使って 0～1024 バイトで指定します。 - 英数字 - プラス「+」 - コロン「:」 - ピリオド「.」 - アンダースコア「_」 - ハイフン「-」	なし	06-50 07-00
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.filter	次に示す文字を使って 0～4096 バイトのクラス文字セットで指定します。 - 英数字 - ピリオド「.」 - ドル「\$」 - アンダースコア「_」	なし	06-50 07-00
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.formatter	次に示す文字を使って 0～4096 バイトのクラス文字セットで指定します。 - 英数字 - ピリオド「.」 - ドル「\$」 - アンダースコア「_」	なし	06-50 07-00
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.level	指定できる文字列を次に示します。 - OFF - SEVERE - WARNING - INFO - CONFIG - FINE - FINER - FINEST - ALL	SEVERE	06-50 07-00
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.limit	【CJMessageFileHandler を使用した場合】 8192～2147483647 【CJMPMessageFileHandler を使用した場合】 8192～16777216	1048576	06-50 07-00 07-50
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.msgid	次に示す文字を使って 0～21 バイトの半角文字セットで指定します。 - 英数字 - ピリオド「.」 - ドル「\$」 - アンダースコア「_」	0001	06-50 07-00

param-name 指定値	指定可能値	省略値	VR
	• ストローク「 」 • コロン「:」 • アンパサンド「&」 • 単価記号「@」 • シャープ「#」 • パーセント「%」 • ハイフン「-」		
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.path	任意の文字列を 1～255 バイトの範囲で指定します。	user_log	06-50
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.separator	次に示す文字を使って 0～1024 バイトの半角文字セットで指定します。 • 英数字 • ピリオド「.」 • ドル「\$」 • アンダースコア「_」 • ストローク「 」 • コロン「:」 • アンパサンド「&」 • 単価記号「@」 • シャープ「#」 • パーセント「%」 • ハイフン「-」	ストローク「 」	06-50 07-00
ejbserver.application.userlog.Logger.<ロガー名称>.filter	次に示す文字を使って 0～4096 バイトのクラス文字セットで指定します。 • 英数字 • ピリオド「.」 • ドル「\$」 • アンダースコア「_」	なし	06-50 07-00
ejbserver.application.userlog.Logger.<ロガー名称>.handlers ^{※1}	英数字で始まる 1～1024 バイトのクラス文字セットで指定します。 また、複数指定する場合はコンマ「,」で区切って指定します。	なし	06-50 07-00
ejbserver.application.userlog.Logger.<ロガー名称>.level	指定できる文字列を次に示します。 • OFF • SEVERE • WARNING • INFO • CONFIG • FINE • FINER • FINEST	SEVERE	06-50

param-name 指定値	指定可能値	省略値	VR
	• ALL • null		
ejbserver.application.userlog.Logger.<ロガー名称>.useParentHandlers	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.application.userlog.loggers ^{※1}	1024 文字以内のクラスパスを指定します。 複数指定する場合は、コンマ「,」で区切ります。 クラスパスに指定できる文字を示します。 • 英数字 • ドル「\$」 • アンダースコア「_」 • ピリオド「.」 ただし、先頭が英数字で始まる必要があります。	なし	06-50 07-00
ejbserver.application.userlog.menu.handlers. ^{※1※2}	ハンドラの名称一覧を指定します。 複数指定する場合は、コンマ「,」で区切ります。	なし	06-50 07-00

注※1

param-value に値を複数指定する際は、コンマ「,」で区切って指定します。

(例)

<param-value>値 1,値 2,値 3</param-value>

注※2

このパラメタからハンドラ名を削除する場合は、関連するパラメタ（ejbserver.application.userlog.CJLogHandler.<削除するハンドラ名称>.XXX）も削除してください。

なお、このパラメタに対応するキーは「2.4(5)(b).ejbserver.application から始まるキー」にはありません。

(2).ejbserver.client から始まるパラメタ

ejbserver.client から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(c\).ejbserver.client から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-14.ejbserver.client から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.client.ctm.RequestPriority	1～8 の整数で指定します。	4	06-50 07-10

(3).ejbserver.commonj から始まるパラメタ

ejbserver.commonj から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(d\).ejbserver.commonj から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-15 ejbserver.commonj から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.commonj.WorkManager.non_daemon_work_threads	1～65535 の整数で指定します。	10	08-50

(4) ejbserver.compiler から始まるパラメタ

ejbserver.compiler から始まるパラメタについて次の表に示します。

表 4-16 ejbserver.compiler から始まるパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
ejbserver.compiler.jvm.maxHeapSize※	J2EE アプリケーション開始時に起動される javac コマンドのヒープサイズの最大サイズを指定します。 ejbserver.compiler.jvm.minHeapSize ≤ ejbserver.compiler.jvm.maxHeapSize の関係が成り立つ値を設定してください。 このパラメタは 09-00-01 以前で有効です。09-00-02 以降は指定しても無視されます。	1～4095 の整数で指定します。	256	06-50 07-00
ejbserver.compiler.jvm.minHeapSize※	J2EE アプリケーション開始時に起動される javac コマンドのヒープサイズの初期サイズを 1～4095 の範囲（単位：メガバイト）で指定します。 ejbserver.compiler.jvm.minHeapSize ≤ ejbserver.compiler.jvm.maxHeapSize の関係が成り立つ値を設定してください。 このパラメタは 09-00-01 以前で有効です。09-00-02 以降は指定しても無視されます。	1～4095 の整数で指定します。	32	06-50 07-00

注※

メガバイト単位で指定する場合に、文字「m」または「M」は不要です。

また、文字「k」または「K」を付けてキロバイト単位で指定することはできません。

(5) ejbserver.connectionpool から始まるパラメタ

ejbserver.connectionpool から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、[2.2.3\(5\)\(f\) ejbserver.connectionpool から始まるキー](#)についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-17 ejbserver.connectionpool から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.connectionpool.association.enabled	指定できる文字列を次に示します。 • true	false	07-50

param-name 指定値	指定可能値	省略値	VR
	false		
ejbserver.connectionpool.validation.timeout	1～2147483647 の整数で指定します。	5	08-00

(6) ejbserver.connector から始まるパラメタ

ejbserver.connector から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(g\) ejbserver.connector から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-18 ejbserver.connector から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.connector.logwriter.filename	1～16 の整数で指定します。	4	06-50 07-00
ejbserver.connector.logwriter.filesize	4096～2147483647（単位：バイト）の整数で指定します。	2097152	06-50 07-00

(7) ejbserver.container から始まるパラメタ

ejbserver.container から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(h\) ejbserver.container から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-19 ejbserver.container から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.container.audit_trail.enabled	指定できる文字列を次に示します。 - true - false	false	07-60
ejbserver.container.passivate.scan.interval	0～2147483（単位：秒）の整数で指定します。	0	06-50
ejbserver.container.remove.scan.interval	0～153722867280912（単位：分）の整数で指定します。	5	06-50

(8) ejbserver.ctm から始まるパラメタ

ejbserver.ctm から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(i\) ejbserver.ctm から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-20 ejbserver.ctm から始まるキーパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.ctm.ActivateTimeOut	0～2147483647（単位：秒）の整数で指定します。	180	06-50 07-10
ejbserver.ctm.CTMID	指定できる文字列を次に示します。 「ctm」または「CTM」で始まらない、英数字、アンダースコア「_」およびピリオド「.」で指定した 31 文字以内の文字列 - IP アドレス	ctm-tier： <使用する CTM の IP アドレスのピリオドをアンダースコア（_）に置き換えたもの>_<使用する CTM のポート番号> それ以外： <IP アドレス>	07-50
ejbserver.ctm.DeactivateTimeOut	0～2147483647（単位：秒）の整数で指定します。	180	06-50 07-10
ejbserver.ctm.QueueLength	1～32767 の整数で指定します。	なし	06-50 07-10
ejbserver.ctm.useGlobalJNDI	指定できる文字列を次に示します。 - true - false	false	09-00

(9) ejbserver.deploy から始まるパラメタ

ejbserver.deploy から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(j\) ejbserver.deploy から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-21 ejbserver.deploy から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.deploy.annotations.load_libjars.enabled	指定できる文字列を次に示します。 - true - false	true	09-50
ejbserver.deploy.annotations.load_check.enabled	指定できる文字列を次に示します。 - true - false	false	08-00

param-name 指定値	指定可能値	省略値	VR
ejbserver.deploy.context.check_interval	0～2147483647（単位：秒）の整数で指定します。	0	07-00
ejbserver.deploy.context.reload_scope	指定できる文字列を次に示します。 • app • jsp • none • web	app	07-00
ejbserver.deploy.context.update_interval	0～2147483647（単位：秒）の整数で指定します。	0	07-00
ejbserver.deploy.resourcefile.scramble.enabled	指定できる文字列を次に示します。 • true • false	false	06-50
ejbserver.deploy.session.work.directory	任意の文字列を指定します。	• Windows の場合 &{cosminexus.home}\%CC%\server\repository\{server.name} • UNIX の場合 &{cosminexus.home}/CC/server/repository/{server.name}	07-00
ejbserver.deploy.LibraryConflictAvoid.<J2EE アプリケーション名>.enabled	指定できる文字列を次に示します。 • true • false (ignore)	false	11-10
ejbserver.deploy.LibraryConflictAvoid.<J2EE アプリケーション名>.package	任意の文字列を指定します。	なし	11-10

param-name 指定値	指定可能値	省略値	VR
ejbserver.deploy.LibraryConflictAv oid.<J2EE アプリケーション名 >.resource	任意の文字列を指定します。	なし	11-10

(10) ejbserver.distributedtx から始まるパラメタ

ejbserver.distributedtx から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(k\) ejbserver.distributedtx から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-22 ejbserver.distributedtx から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.distributedtx.ots.recoverFai lMessageCount	0～2147483647 の整数で指定します。	0	09-00
ejbserver.distributedtx.ots.status.dire ctory1	任意の文字列を 200 文字以内で指定します。	otsstatus	06-70
ejbserver.distributedtx.ots.status.dire ctory2	任意の文字列を 200 文字以内で指定します。	なし	06-70
ejbserver.distributedtx.recovery.port	1～65535 の整数で指定します。	20302	06-50 07-00
ejbserver.distributedtx.rollbackClient TxOnSystemException	指定できる文字列を次に示します。 • true • false	false	06-50
ejbserver.distributedtx.XATransactio n.enabled	指定できる文字列を次に示します。 • true • false	false	06-50

(11) ejbserver.DynamicStubLoading から始まるパラメタ

ejbserver.DynamicStubLoading から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(l\) ejbserver.DynamicStubLoading から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-23 ejbserver.DynamicStubLoading から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.DynamicStubLoading.En abled	指定できる文字列を次に示します。 • true • false	false	06-50

(12) ejbserver.ejb から始まるパラメタ

ejbserver.ejb から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(m\) ejbserver.ejb から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-24 ejbserver.ejb から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.ejb.timerservice.maxCall backThreads	1～100 の整数で指定します。	1	07-00
ejbserver.ejb.timerservice.retryCou nt	0～2147483646 の整数で指定します。	1	07-00
ejbserver.ejb.timerservice.retryInter val	1～604800（単位：秒）の整数で指定します。	5	07-00

(13) ejbserver.ext から始まるパラメタ

ejbserver.ext から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(n\) ejbserver.ext から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-25 ejbserver.ext から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.ext.method_observation.i nterval	0～86400（単位：秒）の整数で指定します。	0	06-50

(14) ejbserver.http から始まるパラメタ

ejbserver.http から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(o\) ejbserver.http から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-26 ejbserver.http から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.http.port	1～65535 の整数で指定します。	28008	07-50 11-00

(15) ejbserver.instrumentation から始まるパラメタ

ejbserver.instrumentation から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(p\) ejbserver.instrumentation から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-27 ejbserver.instrumentation から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.instrumentation.enabled	指定できる文字列を次に示します。 • true • false	true	06-50 07-00

(16) ejbserver.jca から始まるパラメタ

ejbserver.jca から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(r\) ejbserver.jca から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-28 ejbserver.jca から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.jca.adapter.tp1.bind_host	指定できる値を次に示します。 • IPv4 アドレス • ホスト名	システムによって自動的に選択された有効なローカルアドレス	08-50

(17) ejbserver.jndi から始まるパラメタ

ejbserver.jndi から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(s\) ejbserver.jndi から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-29 ejbserver.jndi から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.jndi.cache	指定できる文字列を次に示します。 • on • off	on	06-50
ejbserver.jndi.cache.interval	0～2147483647（単位：秒）の整数で指定します。	0	06-50
ejbserver.jndi.cache.interval.clear.option	指定できる文字列を次に示します。 • refresh	refresh	06-50

param-name 指定値	指定可能値	省略値	VR
	• check		
ejbserver.jndi.global.enabled	指定できる文字列を次に示します。 • true • false	true	09-00
ejbserver.jndi.namingservice.group.<Specify group name>.providerurls	次の順に並ぶ文字列を指定します。 「corbaname::」[英数字 1～255 文字, または@myhost] 「:」[0～65535 の数字] が 1 回以上 「;」 区切りで繰り返される文字列を指定します。	なし	07-50
ejbserver.jndi.namingservice.group.list	論理ネーミングサービスのグループをセミコロン 「;」 で区切って指定します。 論理ネーミングサービスのグループは、30 文字以内の英数字またはアンダースコア 「_」 で構成されます。	なし	07-50
ejbserver.jndi.request.timeout	0～86400（単位：秒）の整数で指定します。	0	06-50

(18) ejbserver.jta から始まるパラメタ

ejbserver.jta から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(u\) ejbserver.jta から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-30 ejbserver.jta から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.jta.TransactionManager.defaultTimeOut	1～2147483647（単位：秒）の整数で指定します。	180	06-50

(19) ejbserver.logger から始まるパラメタ

ejbserver.logger から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(v\) ejbserver.logger から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-31 ejbserver.logger から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.logger.access_log.nio_http.format	フォーマット形式を指定します。 このプロパティに指定できる値の最大長は 1024 文字です。 フォーマット形式に使用できる文字は、アスキーコードの 32(10 進数)以上、127(10 進数)未満の文字です。	%h %{X-Forwarded-For}i %l %u %d %rootap "%r" %s %b %D %S	11-00

param-name 指定値	指定可能値	省略値	VR
ejbserver.logger.access_log.websocket.enabled	指定できる文字列を次に示します。 • true • false	false	11-00
ejbserver.logger.access_log.websocket.format	フォーマット形式を指定します。 このプロパティに指定できる値の最大長は 1024 文字です。 フォーマット形式に使用できる文字は、アスキーコードの 32(10 進数)以上、127(10 進数)未満の文字です。	%TS %IO %OPCODE %ROOTAP %URI %FIN %PAYLOAD %DATALEN %CLIENT %AP %CLOSE %REASON	11-00
ejbserver.logger.channels.define.ConnectionPoolWatchLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.ConnectionPoolWatchLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.ConsoleLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.ConsoleLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.DevelopmentLogFile.filenum	1～16 の整数で指定します。	4	09-50
ejbserver.logger.channels.define.DevelopmentLogFile.filesize	4096～2147483647 の整数で指定します。	1048576	09-50
ejbserver.logger.DevelopmentLogFile.level	指定できる値を次に示します。 • OFF（出力しない） • SEVERE • WARNING • INFO • CONFIG • FINE • FINER • FINEST • ALL（すべて出力）	OFF	09-50
ejbserver.logger.channels.define.EJBContainerLogFile.filenum	1～16 の整数で指定します。	2	07-50

param-name 指定値	指定可能値	省略値	VR
ejbserver.logger.channels.define.EJBContainerLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.ExceptionLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.ExceptionLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.FileDescriptorWatchLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.FileDescriptorWatchLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.HttpSessionWatchLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.HttpSessionWatchLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.MaintenanceLogFile.filenum	1～16 の整数で指定します。	4	07-50
ejbserver.logger.channels.define.MaintenanceLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	16777216	07-50
ejbserver.logger.channels.define.MemoryWatchLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.MemoryWatchLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.MessageLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.MessageLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.NIOHTTPAccessLogFile.filenum	1～16 の整数で指定します。	16	11-00
ejbserver.logger.channels.define.NIOHTTPAccessLogFile.filesize	4096～2147483647 の整数で指定します。	4194304	11-00
ejbserver.logger.channels.define.RequestQueueWatchLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.RequestQueueWatchLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.ThreadDumpWatchLogFile.filenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.ThreadDumpWatchLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50

param-name 指定値	指定可能値	省略値	VR
ejbserver.logger.channels.define.ThreadWatchLogFile.fileenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.ThreadWatchLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.UserErrLogFile.fileenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.UserErrLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.UserOutLogFile.fileenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.UserOutLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.WebContainerLogFile.fileenum	1～16 の整数で指定します。	2	07-50
ejbserver.logger.channels.define.WebContainerLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	1048576	07-50
ejbserver.logger.channels.define.WebServletLogFile.fileenum	1～16 の整数で指定します。	4	07-50
ejbserver.logger.channels.define.WebServletLogFile.filesize	4096～2147483647（単位：バイト）の整数で指定します。	4194304	07-50
ejbserver.logger.channels.define.WebSocketAccessLogFile.fileenum	1～16 の整数で指定します。	16	11-00
ejbserver.logger.channels.define.WebSocketAccessLogFile.filesize	4096～2147483647 の整数を指定します。	4194304	11-00
ejbserver.logger.enabled.*	次に示す文字列をコンマ「,」で区切って指定します。 • Error • Warning • Information • Debug	Error	07-50
ejbserver.logger.rotationTime	HHMMSS の形式で指定します。	なし	09-50
ejbserver.logger.rotationStyle	指定できる文字列を次に示します。 • SHIFT • WRAP	WRAP	09-50
ejbserver.logger.systemlog.enabled	指定できる文字列を次に示します。 • true • false	true	06-50

(20) ejbserver.management から始まるパラメタ

ejbserver.management から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(w\) ejbserver.management から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-32 ejbserver.management から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.management.JVM.stats_monitor.FullGCCCount.enabled	指定できる文字列を次に示します。 • true • false	true	07-00
ejbserver.management.JVM.stats_monitor.FullGCCCount.interval	1～2147483647（単位：秒）の整数で指定します。	600	07-00
ejbserver.management.JVM.stats_monitor.FullGCCCount.threshold	1～2147483647 の整数で指定します。	10	07-00
ejbserver.management.statistics.interval	1～86400（単位：秒）の整数で指定します。	60	07-00
ejbserver.management.stats_file.base_time	0～1439（単位：分）の整数で指定します。	0	07-00
ejbserver.management.stats_file.dir	任意の文字列を指定します。	【Windows の場合】 <製品の作業ディレクトリ >¥ejb¥< サーバ名称 >¥stats 【UNIX の場合】 <製品の作業ディレクトリ >/ejb/< サーバ名称 >/stats	07-00
ejbserver.management.stats_file.enabled	指定できる文字列を次に示します。 • true • false	true	07-00
ejbserver.management.stats_file.num	2～168 の整数で指定します。	7	07-00
ejbserver.management.stats_file.period	1～744（単位：時）の整数で指定します。	24	07-00

param-name 指定値	指定可能値	省略値	VR
ejbserver.management.webcontainer.stats_monitor.waiting_request_count.enabled	指定できる文字列を次に示します。 • true • false	true	09-00
ejbserver.management.webcontainer.stats_monitor.waiting_request_count.high_threshold	1～100（単位：％）の整数で指定します。	80	09-00
ejbserver.management.webcontainer.stats_monitor.waiting_request_count.low_threshold	0～99（単位：％）の整数で指定します。	0	09-00
ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.enabled	指定できる文字列を次に示します。 • true • false	true	09-00
ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.high_threshold	1～100（単位：％）の整数で指定します。	80	09-00
ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.low_threshold	0～99（単位：％）の整数で指定します。	0	09-00

(21) ejbserver.manager から始まるパラメタ

ejbserver.manager から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(x\) ejbserver.manager から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-33 ejbserver.manager から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.manager.agent.JP1EventAgent.conf	任意の文字列を指定します。	<Application Server のインストールディレクトリ>manager/config/<manager.<実サーバ名>.properties	07-50
ejbserver.manager.agent.JP1EventAgent.enabled	指定できる文字列を次に示します。 • true • false	false	07-50

param-name 指定値	指定可能値	省略値	VR
ejbserver.manager.agent.MEventAgent.conf	任意の文字列を指定します。	<Application Server のインストールディレクトリ >manager/config/mevent.< 実サーバ名 >.properties	07-50
ejbserver.manager.agent.MEventAgent.enabled	指定できる文字列を次に示します • true • false	true	06-50 07-00

(22) ejbserver.naming から始まるパラメタ

ejbserver.naming から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(y\) ejbserver.naming から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-34 ejbserver.naming から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.naming.host	指定できる文字列を次に示します。 • ホスト名 • IPv4 アドレス • @myhost	J2EE サーバと同じホスト	07-50
ejbserver.naming.port※ ¹	1～65535 の整数で指定します。	900	07-50
ejbserver.naming.startupMode※ ²	指定できる文字列を次に示します。 • manual • inprocess	inprocess	07-50
ejbserver.naming.startupRetryCount※ ¹	0～2147483647 の整数で指定します。	9	07-50
ejbserver.naming.startupWaitTime※ ¹	0～2147483647（単位：秒）の整数で指定します。	1	07-50

注※¹

ejbserver.naming.startupMode に「inprocess」以外を指定した場合、指定は無視されます。

注※²

「automatic」は指定できません。

(23) ejbserver.rmi から始まるパラメタ

ejbserver.rmi から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(z\) ejbserver.rmi から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-35 ejbserver.rmi から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.rmi.localinvocation.scope	指定できる文字列を次に示します。 • none • app • all	app	06-50
ejbserver.rmi.naming.host	指定できる文字列を次に示します。 • ホスト名※ • IPv4 アドレス※ • @myhost	なし	07-50
ejbserver.rmi.naming.port	1～65535 の整数で指定します。	23152	07-00
ejbserver.rmi.passbyreference	指定できる文字列を次に示します。 • true • false	false	06-50
ejbserver.rmi.remote.listener.port	0～65535 の整数で指定します。	23550	07-00
ejbserver.rmi.request.timeout	0～86400（単位：秒）の整数で指定します。	0	06-50 07-00

注※
ホスト名または IPv4 アドレスを指定する場合は、ホストの定義の<host-name>タグの値を指定してください。異なる値を指定した場合、警告メッセージ（KEOS24186-W）が出力され、意図しない設定となることがあります。

(24) ejbserver.server から始まるパラメタ

ejbserver.server から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ab\) ejbserver.server から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-36 ejbserver.server から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.server.prf.PRFID	指定できる文字列を次に示します。 • 「ctm」、 「CTM」、 「tsc」、 および 「TSC」 で始まらない、英数字およびアンダースコア「_」で指定した 31 文字以内の文字列 • CTMDOMAIN	<使用する PRF の PRF 識別子>	07-50

param-name 指定値	指定可能値	省略値	VR
ejbserver.server.threaddump.filename	1～2147483647 の整数で指定します。	256	06-50
ejbserver.server.eheap.httpsession.enabled	このパラメタは、JDK11 を使用する場合だけ指定できます。 指定できる文字列を次に示します。 • true • false	true	08-00

(25) ejbserver.stateful から始まるパラメタ

ejbserver.stateful から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ac\) ejbserver.stateful から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-37 ejbserver.stateful から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.stateful.passivate.switch	指定できる文字列を次に示します。 • true • false	false	06-50

(26) ejbserver.stdoutlog から始まるパラメタ

ejbserver.stdoutlog から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ad\) ejbserver.stdoutlog から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-38 ejbserver.stdoutlog から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.stdoutlog.autoflush	指定できる文字列を次に示します。 • true • false	false	08-00

(27) ejbserver.watch から始まるパラメタ

ejbserver.watch から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ae\) ejbserver.watch から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-39 ejbserver.watch から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.watch.defaultRequestQueue.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.defaultRequestQueue.interval	1～2147483647（単位：秒）の整数で指定します。	30	06-50
ejbserver.watch.defaultRequestQueue.threshold	1～100（単位：%）の整数で指定します。	80	06-50
ejbserver.watch.defaultRequestQueue.writefile.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.fileDescriptor.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.fileDescriptor.interval	1～2147483647（単位：秒）の整数で指定します。	60	06-50
ejbserver.watch.fileDescriptor.threshold	1～2147483647 の整数で指定します。	2147483647	06-50
ejbserver.watch.fileDescriptor.writefile.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.memory.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.memory.interval	1～2147483647（単位：秒）の整数で指定します。	60	06-50
ejbserver.watch.memory.threshold	1～100（単位：%）の整数で指定します。	80	06-50
ejbserver.watch.memory.writefile.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.thread.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.thread.interval	1～2147483647（単位：秒）の整数で指定します。	60	06-50

param-name 指定値	指定可能値	省略値	VR
ejbserver.watch.thread.threshold	1～2147483647 の整数で指定します。	2147483647	06-50
ejbserver.watch.thread.writefile.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.threaddump.enabled	指定できる文字列を次に示します。 • true • false	true	06-50
ejbserver.watch.threaddump.interval	1～2147483647（単位：秒）の整数で指定します。	30	06-50
ejbserver.watch.threaddump.threshold	1～100（単位：%）の整数で指定します。	80	06-50
ejbserver.watch.threaddump.writefile.enabled	指定できる文字列を次に示します。 • true • false	true	06-50

(28) ejbserver.webj2ee から始まるパラメタ

ejbserver.webj2ee から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(af\) ejbserver.webj2ee から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-40 ejbserver.webj2ee から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.webj2ee.connectionAutoClose.enabled	指定できる文字列を次に示します。 • true • false	true	06-50 07-00

(29) java から始まるパラメタ

java から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ag\) java から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-41 java から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
java.naming.factory.initial	指定できる文字列を次に示します。 • com.hitachi.software.ejb.jndi.InsContextFactory	com.hitachi.software.ejb.jndi.Ins	06-50

param-name 指定値	指定可能値	省略値	VR
	com.hitachi.software.ejb.jndi.GroupContextFactory	ContextFactory	

(30) vbj から始まるパラメタ

vbj から始まるパラメタについて次の表に示します。

表 4-42 vbj から始まるパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
vbj.java2iio.jvm.maxHeapSize※	J2EE アプリケーション開始時に起動される java2iio コマンドのヒープサイズの最大サイズを指定します。 vbj.java2iio.jvm.minHeapSize ≤ vbj.java2iio.jvm.maxHeapSize の関係が成り立つ値を設定してください。	1～4095 の整数で指定します。	128	06-50 07-00
vbj.java2iio.jvm.minHeapSize※	J2EE アプリケーション開始時に起動される java2iio コマンドのヒープサイズの初期サイズを 1～4095 の範囲（単位：メガバイト）で指定します。 vbj.java2iio.jvm.minHeapSize ≤ vbj.java2iio.jvm.maxHeapSize の関係が成り立つ値を設定してください。	1～4095 の整数で指定します。	16	06-50 07-00

注※

メガバイト単位で指定する場合に、文字「m」または「M」は不要です。

また、文字「k」または「K」を付けてキロバイト単位で指定することはできません。

(31) vbroker から始まるパラメタ

vbroker から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ai\) vbroker から始まるキー](#)」についての説明を参照してください。
また、参照する場合はキーをパラメタに読み替えてください。

表 4-43 vbroker から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
vbroker.agent.port	1～65535 の整数で指定します。	14000	07-50
vbroker.se.iio_tp.host	指定できる文字列を次に示します。 - ホスト名※ - IPv4 アドレス※ @myhost	なし	07-50
vbroker.se.iio_tp.scm.iio_tp.listener.port	0～65535 の整数で指定します。	0	06-50 07-00

注※

ホスト名または IPv4 アドレスを指定する場合は、ホストの定義の<host-name>タグの値を指定してください。異なる値を指定した場合、警告メッセージ（KEOS24186-W）が出力され、意図しない設定となることがあります。

(32) webserver.application から始まるパラメタ

webserver.application から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(aj\) webserver.application から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-44 webserver.application から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.application.lower_version	指定できる文字列を次に示します。 • 2.4 • 2.5	なし	08-50

(33) webserver.connector から始まるパラメタ

webserver.connector から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ak\) webserver.connector から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-45 webserver.connector から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.connector.http.bind_host	次のどれかを指定します。 • ホスト名※² • IPv4 アドレス※² • @myhost	なし	07-50
webserver.connector.http.permitted_hosts※ ¹	指定できる値を次に示します。 • IPv4 アドレス • ホスト名 • アスタリスク「*」	なし	06-50 07-00
webserver.connector.limit.max_parameter_count	-1～2147483647 の整数で指定します。	10000	09-00
webserver.connector.limit.max_post_form_data	-1～2147483647 の整数で指定します。	2097152	08-50
webserver.connector.nio_http.port	1～65535 の整数で指定します。	8008	11-00
webserver.connector.nio_http.backlog	1～2147483647 で整数で指定します。	511	11-00
webserver.connector.nio_http.bind_host	指定できる値を次に示します。 • ホスト名※²	なし	11-00

param-name 指定値	指定可能値	省略値	VR
	IP アドレス※2		
webserver.connector.nio_http.idle_thread_timeout	1～2147483647 の整数で指定します。	60	11-00
webserver.connector.nio_http.keep_alive.max_requests	0～2147483647 の整数で指定します。	0	11-00
webserver.connector.nio_http.keep_alive.timeout	0～3600 の整数で指定します。	0	11-00
webserver.connector.nio_http.limit_max_headers	0～32767 の整数で指定します。	100	11-00
webserver.connector.nio_http.limit_max_request_body	-1～2147483647 の整数で指定します。	-1	11-00
webserver.connector.nio_http.limit_max_request_header	7～65536 の整数で指定します。	16384	11-00
webserver.connector.nio_http.max_connections	1～2147483647 の整数で指定します。	1024	11-00
webserver.connector.nio_http.max_servlet_execute_threads	1～2147483647 の整数で指定します。	100	11-00
webserver.connector.nio_http.max_threads	1～2147483647 の整数で指定します。	100	11-00
webserver.connector.nio_http.min_threads	1～2147483647 の整数で指定します。	10	11-00
webserver.connector.nio_http.permitted_hosts	コンマ区切りの IP アドレスもしくはホスト名、またはアスタリスク「*」を指定します。 IP アドレスおよびホスト名に指定できる文字は英数字、アンダースコア「_」、ハイフン「-」およびピリオド「.」です。コンマの前後にスペースを入れることはできません。	*	11-00
webserver.connector.nio_http.receive_timeout	0～3600 の整数で指定します。	300	11-00
webserver.connector.nio_http.response.header.server	任意の文字列	CosminexusComponentContainer	11-00
webserver.connector.nio_http.send_timeout	0～3600 の整数で指定します。	300	11-00

注※1

param-value に値を複数指定する際は、コンマ「,」で区切って指定します。

(例)

<param-value>値 1,値 2,値 3</param-value>

ホスト名または IP アドレスを指定する場合は、ホストの定義の<host-name>タグの値を指定してください。異なる値を指定した場合、警告メッセージ（KEOS24186-W）が出力され、意図しない設定となることがあります。

(34) webserver.container から始まるパラメタ

webserver.container から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(al\) webserver.container から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-46 webserver.container から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.container.server_id.enabled	指定できる文字列を次に示します。 • true • false	false	06-51
webserver.container.server_id.name	英数字またはアンダースコア「_」を使った 1～64 文字の文字列を指定します。なお、「JSESSIONID」という文字列を指定するとエラーとなります。	空文字（負荷分散機の設定で指定した <cookie-name>の値が自動的に設定されます。）	06-51 07-00 07-50
webserver.container.server_id.value	英数字、およびアンダースコア「_」を使った 1～64 文字の文字列を指定します。	ホスト名と J2EE サーバ名／Web コンテナサーバ名（互換機能）からハッシュ関数を用いて生成する文字列	06-51 07-10 07-50
webserver.container.thread_control.queue_size	0～2147483647 の整数で指定します。	8192	06-50
webserver.container.jaxws.webservice.no_webxml.enabled	指定できる文字列を次に示します。 • strict • true • lax • none • false	none	08-00
webserver.container.jaxws.webservice.wsee.no_webxml.enabled	指定できる文字列を次に示します。 • strict	lax	08-70

param-name 指定値	指定可能値	省略値	VR
	• lax • none		
webserver.container.jaxws.webservice.wsee.warname	EAR ファイル内の設定用 WAR ファイルの相対パスを指定します。	CosminexusWSEE.war	08-70
webserver.container.servlet.default_mapping.enabled	指定できる文字列を次に示します。 • true • false	false	08-00

(35) webserver.context から始まるパラメタ

webserver.context から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(am\) webserver.context から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-47 webserver.context から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.context.check_interval	0～2147483647（単位：秒）の整数で指定します。	なし	07-00
webserver.context.reload_delay_timeout	-2147483648～2147483647（単位：秒）の整数で指定します。	0	07-00
webserver.context.update.interval	0～2147483647（単位：秒）の整数で指定します。	なし	07-00

(36) webserver.dbsfo から始まるパラメタ

webserver.dbsfo から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(an\) webserver.dbsfo から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-48 webserver.dbsfo から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.dbsfo.enabled	指定できる文字列を次に示します。 • true • false	false	08-00
webserver.dbsfo.exclude.uris	スラッシュ「/」から始まる文字列で、正規化した URI を指定します。	なし	08-50
webserver.dbsfo.connector.name	任意の文字列を指定します。	COSMINEXUS_SFO_DBCONNECTOR	08-00

param-name 指定値	指定可能値	省略値	VR
webserver.dbsfo.attribute_data_size.max	128～10485760 の整数で指定します。	1024	08-00
webserver.dbsfo.exclude.extensions	次に示す文字を使って、512 文字以内で指定します。 • 英数字 • 括弧「()」 • エクスクラメーション「!」 • パーセント「%」 • ドル「\$」 • アンパサンド「&」 • シングルクォート「'」 • プラス「+」 • ハイフン「-」 • イコール「=」 • 単価記号「@」 • アンダースコア「_」 • チルダ「~」 • ピリオド「.」 • コンマ「,」 複数の拡張子を指定する場合、コンマ「,」で区切って指定します。	txt,html,html,jpg,gif,js	08-00
webserver.dbsfo.check_size.mode	指定できる文字列を次に示します。 • on • off	off	08-00
webserver.dbsfo.exception_type_backcompat	指定できる文字列を次に示します。 • true • false	false	08-70
webserver.dbsfo.integrity_mode_enabled	指定できる文字列を次に示します。 • true • false	false	08-70
webserver.dbsfo.negotiation_high_level	指定できる文字列を次に示します。 • true • false	false	08-70
webserver.dbsfo.session_read_only_uri	スラッシュ「/」から始まる文字列で、正規化した URI を指定します。	なし	08-70
webserver.dbsfo.thread_control_queue.enabled	指定できる文字列を次に示します。 • true • false	false	08-70

(37) webserver.http から始まるパラメタ

webserver.http から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ap\) webserver.http から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-49 webserver.http から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.http.request.encoding	任意の文字列を 1024 文字以内で指定します。	なし	07-10
webserver.http.request.uri_decode.enabled	指定できる文字列を次に示します。 • true • false	false	08-50
webserver.http.response.encoding	任意の文字列を 1024 文字以内で指定します。	なし	07-10

(38) webserver.jsp から始まるパラメタ

webserver.jsp から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ar\) webserver.jsp から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-50 webserver.jsp から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.jsp.additional.import.list	任意の文字列を指定します。	なし	08-70
webserver.jsp.check_interval	0～2147483647（単位：秒）の整数で指定します。	なし	07-00
webserver.jsp.compile.backcompat	指定できる値および文字列を次に示します。 • 17（JDK21 以降を使用する場合） • 11（JDK17 以降を使用する場合） • 8 • 1.8 • 7（JDK11 または JDK17 を使用する場合） • 1.7（JDK11 または JDK17 を使用する場合） • 6（JDK11 を使用する場合） • 1.6（JDK11 を使用する場合） • false	false	11-40
webserver.jsp.debugging.enabled	指定できる文字列を次に示します。 • true • false	false	08-00
webserver.jsp.jsp_page.bom.enabled	指定できる文字列を次に示します。 • true • false	true	08-00

param-name 指定値	指定可能値	省略値	VR
webserver.jsp.keepgenerated	指定できる文字列を次に示します。 • true • false	false	06-50
webserver.jsp.pageEncoding	任意の文字列を 1024 文字以内で指定します。	なし	07-10
webserver.jsp.tld.mapping.java_ee_tag_library.enabled	指定できる文字列を次に示します。 • true • false	true	08-00
webserver.jsp.translation.customAction.ignoreCaseAttributeName	指定できる文字列を次に示します。 • true • false	false	08-00
webserver.jsp.translation.backcompat.customAction.declareVariable	指定できる文字列を次に示します。 • true • false	false	07-60
webserver.jsp.translation.backcompat.tag.noCheckRtexprvalue	指定できる文字列を次に示します。 • true • false	false	07-60
webserver.jsp.translation.backcompat.tag.rtexprvalueTerminate	指定できる文字列を次に示します。 • true • false	false	07-60
webserver.jsp.translation.backcompat.taglib.noCheckPrefix	指定できる文字列を次に示します。 • true • false	false	07-60
webserver.jsp.translation.backcompat.useBean.noCheckClass	指定できる文字列を次に示します。 • true • false	false	07-60
webserver.jsp.translation.useBean.noCheckDuplicateId	指定できる文字列を次に示します。 • true • false	false	08-00
webserver.jsp.update.interval	0～2147483647 の整数で指定します。	なし	07-00

(39) webserver.servlet から始まるパラメタ

webserver.servlet から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(at\) webserver.servlet から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-51 webserver.servlet から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.servlet_api.exception.getCause.backcompat	指定できる文字列を次に示します。 • true • false	false	08-00
webserver.ServletContainerInitializer.jar.include.path	指定できる文字に制限はありません。 複数指定する場合は、コンマ「,」で区切ります。ディレクトリはスラッシュ「/」で区切ります。	なし	09-00

(40) webserver.session から始まるパラメタ

webserver.session から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(av\) webserver.session から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-52 webserver.session から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.session.cookie_config.http_only	指定できる文字列を次に示します。 • true • false	false	09-50
webserver.session.cookie_config.name	任意の文字列を指定します。	JSESSIONID	09-50
webserver.session.server_id.value	英数字、およびアンダースコア「_」を使って 64 文字以内で指定します。	ホスト名と J2EE サーバ名／Web コンテナサーバ名（互換機能）からハッシュ関数を用いて生成する文字列※	06-50 07-10 07-50
webserver.session.max.throwHttpSessionLimitExceededException	指定できる文字列を次に示します。 • true • false	false	08-00
webserver.session.tracking_mode	指定できる文字列を次に示します。 • COOKIE • URL	COOKIE と URL の両方を指定	09-50

注※

free-tier 以外の場合、Management Server によって 8 けたの 16 進数が自動設定されます。

(41) webserver.static から始まるパラメタ

webserver.static から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ax\) webserver.static から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-53 webserver.static から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.static_content.cache.enabled	指定できる文字列を次に示します。 • true • false • forceoff	false	06-70
webserver.static_content.cache.file.size.threshold	0～2147483647 の整数で指定します。	524288	06-70
webserver.static_content.cache.size	0～2147483647 の整数で指定します。	10485760	06-70 07-00
webserver.static_content.encoding.extension	任意の文字列を指定します。	なし	07-60

(42) webserver.work から始まるパラメタ

webserver.work から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(ay\) webserver.work から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-54 webserver.work から始まるパラメタ

param-name 指定値	指定可能値	省略値	VR
webserver.work.clean	指定できる文字列を次に示します。 • true • false	false	06-50
webserver.work.directory	任意の文字列を指定します。	【Windows の場合】 & {cosminexus.home} ¥CC¥server ¥repository ¥& {server.name}¥web 【UNIX の場合】	06-50

param-name 指定値	指定可能値	省略値	VR
		&{cosminexus.home}/C/C/server/repository/&{server.name}/web	

(43) webserver.xml から始まるパラメタ

webserver.xml から始まるパラメタについて次の表に示します。「param-name 指定値」に対応する param-value の指定内容については、「[2.2.3\(5\)\(az\) webserver.xml から始まるキー](#)」についての説明を参照してください。また、参照する場合はキーをパラメタに読み替えてください。

表 4-55 webserver.xml から始まるキー

param-name 指定値	指定可能値	省略値	VR
webserver.xml.validate	指定できる文字列を次に示します。 - true - false	true	06-50

4.11.3 J2EE サーバ用オプション定義を設定するパラメタ

J2EE サーバ用オプション定義を設定するパラメタについて、次の表に示します。なお、「param-name 指定値」に対応する param-value の指定内容については、「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」を参照してください。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

J2EE サーバ用オプション定義を設定するパラメタを指定する場合の指定形式を次に示します。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
```

値を複数指定する場合は、複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
<param-value>値</param-value>
```

usrconf.cfg (J2EE サーバ用オプション定義ファイル) に指定できるパラメタのうち、「[4.11.5 J2EE サーバの拡張パラメタ](#)」の表と、この項の表の両方に記載していないパラメタは、次の形式で指定してください。

指定形式

```
<param-name>ex.param</param-name>
<param-value>パラメタ=値</param-value>
```

値を複数指定する場合は、複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>ex.param</param-name>
<param-value>パラメタ=値</param-value>
<param-value>パラメタ=値</param-value>
```

この項に記載があるパラメタをこの形式で指定した場合、動作が保証されません。

表 4-56 J2EE サーバ用オプション定義を設定するパラメタ

param-name 指定値	指定可能値	省略値	VR
add.jvm.arg [※]	指定されたオプションを使って JavaVM を起動します。 システムプロパティを指定したい場合は、"-D"を利用しないで、usrconf.properties ファイルに指定します。 複数のオプションを指定する場合、同じキー名称を使用して、次のように複数回指定してください。 (指定例) <pre>add.jvm.arg= -Xms128m add.jvm.arg= -Xmx256m</pre> なお、次のように複数のオプションを指定できません。指定をした場合、JavaVM の初期化に失敗します。 (指定例) <pre>add.jvm.arg=-Xms128m -Xmx256m</pre> add.jvm.arg で指定できる JavaVM のオプションについては、「14.5 Application Server で指定できる Java HotSpot VM のオプション」を参照してください。 なお、JavaVM 拡張オプションの詳細については、「14.1 JavaVM 拡張オプションの一覧」を参照してください。	デフォルト値については「 2.2.2(5) J2EE サーバでの JavaVM オプションのデフォルト値 」を参照してください。	07-50
ejb.public.directory	任意の文字列を指定します。	【Windows の場合】	06-50

param-name 指定値	指定可能値	省略値	VR
		&{cosminexus.home} ¥CC¥server¥public 【UNIX の場合】 &{cosminexus.home}/CC/ server/public	
ejb.server.log.directory	任意の文字列を 200 文字以内で指定します。	【Windows の場合】 <製品の作業ディレクトリ>¥ejb¥< サーバ名称>¥logs 【UNIX の場合】 <製品の作業ディレクトリ>/ejb/< サーバ名称>/logs	06-50

注※

値の指定形式について、次に示します。

(例)

<param-name>add.jvm.arg</param-name>

<param-value>-Xms256m</param-value>

<param-value>-Xmx512m</param-value>

:

4.11.4 J2EE サーバの JavaVM のシステムプロパティに設定できるパラメタ

J2EE サーバの JavaVM のシステムプロパティを設定するパラメタについて、次の表に示します。「param-value 指定値」に指定する値の詳細については、「[2.2.3 usrconf.properties \(J2EE サーバ用ユーザプロパティファイル\)](#)」を参照してください。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

J2EE サーバの JavaVM のシステムプロパティを設定するパラメタを指定する場合の指定形式を次に示します。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
```

値を複数指定する場合は、複数の<param-value>を指定します。

値を複数指定する場合の指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
<param-value>値</param-value>
```

usrconf.properties (J2EE サーバ用ユーザプロパティファイル) に指定できるパラメタのうち、「[4.11.2 J2EE サーバ用ユーザプロパティを設定するパラメタ](#)」の表と、この項の表の両方に記載していないパラメ

タは、次の形式で指定してください。ただし、統合ユーザ管理の使用を設定するパラメタについては、[「4.11.14 統合ユーザ管理の使用を設定するパラメタ」](#)を参照して設定してください。

指定形式

```
<param-name>ex.properties</param-name>
<param-value>パラメタ=値</param-value>
```

値を複数指定する場合は、複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>ex.properties</param-name>
<param-value>パラメタ=値</param-value>
<param-value>パラメタ=値</param-value>
```

なお、この項の表で示すパラメタは、この形式で指定することもできます。

表 4-57 J2EE サーバの JavaVM のシステムプロパティに設定できるパラメタ

param-name 指定値	指定可能値	省略値	VR
ejbserver.connectionpool.applicationAuthentication.disabled	指定できる文字列を次に示します。 • true • false	false	07-50
ejbserver.connectionpool.association.enabledDespiteUnshareableSetting	指定できる文字列を次に示します。 • true • false	false	07-50
ejbserver.connectionpool.sharingOutsideTransactionScope.enabled	指定できる文字列を次に示します。 • true • false	false	07-50
ejbserver.container.bmp.backcompatible	指定できる文字列を次に示します。 • true • false	false	07-50
ejbserver.container.ejbhome.sessionbean.reconnect.enabled	指定できる文字列を次に示します。 • true • false	false	07-50
ejbserver.container.rebindpolicy	指定できる文字列を次に示します。 • VB_TRANSPARENT • NO_RECONNECT	VB_TRANSPARENT	07-50
ejbserver.container.security.disabled	指定できる文字列を次に示します。 • true • false	false	07-50
ejbserver.ctm.CTMDomain	指定できる文字列を次に示します。	CTMDOMAIN	07-50

param-name 指定値	指定可能値	省略値	VR
	「ctm」, 「CTM」, 「tsc」, および 「TSC」 で始まらない, 英数字およびアンダースコア 「_」 で指定した 31 文字以内の文字列 - CTMDOMAIN		
ejbserver.ctm.CTMMMyHost	次に示す文字を使って 1～255 文字で指定します。 - 英数字 - アンダースコア 「_」 - ピリオド 「.」 - ハイフン 「-」	<hostname コマンドで 取得される ホスト名称 >	07-50
ejbserver.ctm.enabled	指定できる文字列を次に示します。 - true - false	ctm-tier : true それ以外 : false	07-50
ejbserver.deploy.app.stopforcibly.disabled	指定できる文字列を次に示します。 - true - false	false	07-50
ejbserver.deploy.exclusive.lockAliveInterval	1～2147483647（単位：秒）の整数で指定します。	60	07-50
ejbserver.deploy.stub.generation.scope	指定できる文字列を次に示します。 - ejb - app	ejb	07-50
ejbserver.distributedtx.recovery.completionCheckOnStopping.timeout	-1～2147483647（単位：秒）の整数で指定します。	-1	07-50
ejbserver.ejb.cmp20.cmr.use.existing_table	指定できる文字列を次に示します。 - true - false	false	07-50
ejbserver.jndi.cache.reference	指定できる文字列を次に示します。 - on - off	off	07-50
ejbserver.manager.agent.Agent.conf※1	任意の文字列を指定します。	<Application Server のインス トールディ レクトリ >manager/ config/ mngagent. <実サーバ 名 >.properties	07-50

param-name 指定値	指定可能値	省略値	VR
ejbserver.manager.agent.Agent.enabled ^{*1}	指定できる文字列を次に示します。 • true • false	true	07-50
ejbserver.manager.jpl.event.event_server_name	任意の文字列を指定します。	localhost	07-50
ejbserver.naming.exec.args	任意の文字列を指定します。	なし	07-50
ejbserver.naming.nameroot	英数字, およびアンダースコア「_」を使った文字列を指定します。	なし	07-50
ejbserver.naming.protocol	指定できる文字列を次に示します。 • corbaname • iioploc • iiopname	corbaname	07-50
ejbserver.rmi.stateless.unique_id.enabled	指定できる文字列を次に示します。 • true • false	false	07-50
ejbserver.server.mutex.invocation.timeout	45～92233720368547758（単位：秒）の整数で指定します。	45	07-50
vbroker.agent.enableLocator ^{*2}	指定できる文字列を次に示します。 • true • false	false	07-50
vbroker.orb.htc.comt.entryCount	100～30000000 の整数で指定します。	120000	07-50
vbroker.orb.htc.comt.fileCount	1～256 の整数で指定します。	3	07-50
vbroker.orb.htc.tracePath	任意の文字列を 1～210 バイトの範囲で指定します。	【Windows の場合】 <製品の作業ディレクトリ >¥ejb¥< サーバ名称 >¥logs¥TPB¥logj 【UNIX の場合】 <製品の作業ディレクトリ >/ejb/< サーバ名称 >/logs/TPB/logj	07-50

param-name 指定値	指定可能値	省略値	VR
webserver.errorpage.stack_trace.enabled	指定できる文字列を次に示します。 • true • false	false	07-50
webserver.jsp.precompile.jsp_work_dir	英数字およびアンダースコア「_」の文字列を指定します。	cosminexus_jsp_work	07-50
webserver.session.max.log_interval	0～2147483647 の整数で指定します。	60	07-50
webserver.session.server_id.enabled	指定できる文字列を次に示します。 • true • false	true	07-50
webserver.session.delete_cookie.batchcompat	指定できる文字列を次に示します。 • true • false	false	08-00

注※1

Management Server が自動で設定するため、指定は非推奨です。

注※2

通常は設定不要です。

4.11.5 J2EE サーバの拡張パラメタ

J2EE サーバの拡張パラメタを設定するパラメタについて、次の表に示します。なお、「param-value 指定値」に指定する値の詳細については、「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」を参照してください。また、batch.service.enabled の詳細については「[3.2.1 usrconf.cfg \(バッチサーバ用オプション定義ファイル\)](#)」を参照してください。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

J2EE サーバの拡張パラメタを指定する場合の指定形式を次に示します。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
```

値を複数指定する場合は、複数の<param-value>を指定します。

値を複数指定する場合の指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
<param-value>値</param-value>
```

usrconf.cfg (J2EE サーバ用オプション定義ファイル) に指定できるパラメタのうち,「4.11.3 J2EE サーバ用オプション定義を設定するパラメタ」の表と, この項の表の両方に記載していないパラメタは, 次の形式で指定してください。

指定形式

```
<param-name>ex.param</param-name>
<param-value>パラメタ=値</param-value>
```

値を複数指定する場合は, 複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>ex.param</param-name>
<param-value>パラメタ=値</param-value>
<param-value>パラメタ=値</param-value>
```

なお, この項の表で示すパラメタは, この形式で指定することもできます。

表 4-58 J2EE サーバの拡張パラメタに設定できるパラメタ

param-name 指定値	指定可能値	省略値	VR
add.class.path	任意の文字列を指定します。	なし	07-50
add.library.path	任意の文字列を指定します。	なし	07-50
cpp.library.version	6 を指定します。09-00 以降では, 設定できるのは「6」だけです。	6	07-50
ejb.server.corefilenum (UNIX 用)	0～16 の整数で指定します。	1	07-50
ejb.server.log.stdout.filesize	4096～2147483647 (単位: バイト) の整数で指定します。	1048576	07-50
ejb.server.log.stderr.filesize	4096～2147483647 (単位: バイト) の整数で指定します。	1048576	07-50
jvm.type	指定できる文字列を次に示します。 - server - client	server	07-50
batch.service.enabled	false※	FALSE	07-60

注※

batch.service.enabled パラメタに true を指定しないでください。なお, デフォルトでは false になっています。また, このパラメタは Web システムの初期構築時だけ指定できます。Web システムの構成変更では値を変更しないでください。

4.11.6 バッチサーバ用ユーザプロパティを設定するパラメタ

バッチサーバ用ユーザプロパティを設定するパラメタについて説明します。

「param-name 指定値」に対応する param-value の指定内容については、「[3.2.2 usrconf.properties \(バッチサーバ用ユーザプロパティファイル\)](#)」を参照してください。また、参照する場合はキーをパラメタに読み替えてください。「省略値」とは、パラメタの指定がない場合に仮定される値です。

バッチサーバ用ユーザプロパティを設定するパラメタを指定する場合の指定形式を次に示します。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
```

usrconf.properties（バッチサーバ用ユーザプロパティファイル）に指定できるパラメタのうち、「[4.11.8 バッチサーバの JavaVM のシステムプロパティに設定できるパラメタ](#)」の表と、この項の表の両方に記載していないパラメタは、次の形式で指定してください。

指定形式

```
<param-name>ex. properties</param-name>
<param-value>パラメタ=値</param-value>
```

値を複数指定する場合は、複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>ex. properties</param-name>
<param-value>パラメタ=値</param-value>
<param-value>パラメタ=値</param-value>
```

なお、この項に記載があるパラメタのうち、次のパラメタは、この形式で指定することもできます。それ以外のパラメタをこの形式で指定した場合、動作が保証されません。また、<param-name>パラメタ</param-name>の形式とこの形式の両方で同じパラメタを指定した場合、動作が保証されません。

- 運用管理ポータル画面に対応しないパラメタ
- 運用管理ポータル画面に対応し、画面名が「システムプロパティの設定」、設定個所が「システムプロパティ」であるパラメタ

パラメタと運用管理ポータル画面の対応については、マニュアル「アプリケーションサーバ 運用管理ポータル操作ガイド」を参照してください。

(1) ejbserver.application から始まるパラメタ

ejbserver.application から始まるパラメタについて次の表に示します。

表 4-59 ejbserver.application から始まるパラメタ

param-name 指定値	省略値
ejbserver.application.InitTermProcessClasses* ¹	なし

param-name 指定値	省略値
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.appname	user_app
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.count	2
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.encoding	なし
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.filter	なし
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.formatter	なし
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.level	SEVERE
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.limit	1048576
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.msgid	0001
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.path	user_log
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.separator	
ejbserver.application.userlog.Logger.<ロガー名称>.filter	なし
ejbserver.application.userlog.Logger.<ロガー名称>.handlers ^{※1}	なし
ejbserver.application.userlog.Logger.<ロガー名称>.level	SEVERE
ejbserver.application.userlog.Logger.<ロガー名称>.useParentHandlers	true
ejbserver.application.userlog.loggers ^{※1}	なし
ejbserver.application.userlog.menu.handlers. ^{※1} ^{※2}	なし

注※1

param-value に値を複数指定する際は、コンマ「,」で区切って指定します。

(例)

<param-value>値 1,値 2,値 3</param-value>

注※2

このパラメタには、ハンドラの名称一覧を指定します。

このパラメタからハンドラ名を削除する場合は、関連するパラメタ (ejbserver.application.userlog.CJLogHandler.<削除するハンドラ名称>.XXX) も削除してください。

なお、このパラメタに対応するキーは「3.3 usrconf.properties (バッチサーバ用ユーザプロパティファイル)」にはありません。

(2) ejbserver.batch から始まるパラメタ

ejbserver.batch から始まるパラメタについて次の表に示します。

表 4-60 ejbserver.batch から始まるパラメタ

param-name 指定値	省略値
ejbserver.batch.application.exit.enabled	true
ejbserver.batch.gc.watch.threshold	0

param-name 指定値	省略値
ejbserver.batch.queue.length	50
ejbserver.batch.schedule.group.name	JOBGROUP

(3) **ejbserver.client** から始まるパラメタ

ejbserver.client から始まるパラメタについて次の表に示します。

表 4-61 **ejbserver.client** から始まるパラメタ

param-name 指定値	省略値
ejbserver.client.ctm.RequestPriority	4

(4) **ejbserver.connectionpool** から始まるパラメタ

ejbserver.connectionpool から始まるパラメタについて次の表に示します。

表 4-62 **ejbserver.connectionpool** から始まるパラメタ

param-name 指定値	省略値
ejbserver.connectionpool.association.enabled	false
ejbserver.connectionpool.validation.timeout	5

(5) **ejbserver.connector** から始まるパラメタ

ejbserver.connector から始まるパラメタについて次の表に示します。

表 4-63 **ejbserver.connector** から始まるパラメタ

param-name 指定値	省略値
ejbserver.connector.logwriter.filenum	4
ejbserver.connector.logwriter.filesize	2097152

(6) **ejbserver.container** から始まるパラメタ

ejbserver.container から始まるパラメタについて次の表に示します。

表 4-64 **ejbserver.connector** から始まるパラメタ

param-name 指定値	省略値
ejbserver.container.audit_trail.enabled	false

(7) ejbserver.deploy から始まるパラメタ

ejbserver.deploy から始まるパラメタについて次の表に示します。

表 4-65 ejbserver.deploy から始まるパラメタ

param-name 指定値	省略値
ejbserver.deploy.resourcefile.scramble.enabled	false

(8) ejbserver.ext から始まるパラメタ

ejbserver.ext から始まるパラメタについて次の表に示します。

表 4-66 ejbserver.ext から始まるパラメタ

param-name 指定値	省略値
ejbserver.ext.method_observation.interval	0

(9) ejbserver.http から始まるパラメタ

ejbserver.http から始まるパラメタについて次の表に示します。

表 4-67 ejbserver.http から始まるパラメタ

param-name 指定値	省略値
ejbserver.http.port	28080

(10) ejbserver.instrumentation から始まるパラメタ

ejbserver.instrumentation から始まるパラメタについて次の表に示します。

表 4-68 ejbserver.instrumentation から始まるパラメタ

param-name 指定値	省略値
ejbserver.instrumentation.enabled	true

(11) ejbserver.jndi から始まるパラメタ

ejbserver.jndi から始まるパラメタについて次の表に示します。

表 4-69 ejbserver.jndi から始まるパラメタ

param-name 指定値	省略値
ejbserver.jndi.cache	on
ejbserver.jndi.cache.interval	0

param-name 指定値	省略値
ejbserver.jndi.cache.interval.clear.option	refresh
ejbserver.jndi.namingservice.group.<Specify group name>.providerurls	なし
ejbserver.jndi.namingservice.group.list	なし
ejbserver.jndi.request.timeout	0

(12) ejbserver.jta から始まるパラメタ

ejbserver.jta から始まるパラメタについて次の表に示します。

表 4-70 ejbserver.jta から始まるパラメタ

param-name 指定値	省略値
ejbserver.jta.TransactionManager.defaultTimeOut	180

(13) ejbserver.logger から始まるパラメタ

ejbserver.logger から始まるパラメタについて次の表に示します。

表 4-71 ejbserver.logger から始まるパラメタ

param-name 指定値	省略値
ejbserver.logger.channels.define.ConnectionPoolWatchLogFile.filenum	2
ejbserver.logger.channels.define.ConnectionPoolWatchLogFile.filesize	1048576
ejbserver.logger.channels.define.ConsoleLogFile.filenum	2
ejbserver.logger.channels.define.ConsoleLogFile.filesize	1048576
ejbserver.logger.channels.define.EJBContainerLogFile.filenum	2
ejbserver.logger.channels.define.EJBContainerLogFile.filesize	1048576
ejbserver.logger.channels.define.ExceptionLogFile.filenum	2
ejbserver.logger.channels.define.ExceptionLogFile.filesize	1048576
ejbserver.logger.channels.define.FileDescriptorWatchLogFile.filenum	2
ejbserver.logger.channels.define.FileDescriptorWatchLogFile.filesize	1048576
ejbserver.logger.channels.define.MaintenanceLogFile.filenum	4
ejbserver.logger.channels.define.MaintenanceLogFile.filesize	16777216
ejbserver.logger.channels.define.MemoryWatchLogFile.filenum	2
ejbserver.logger.channels.define.MemoryWatchLogFile.filesize	1048576
ejbserver.logger.channels.define.MessageLogFile.filenum	2

param-name 指定値	省略値
ejbserver.logger.channels.define.MessageLogFile.filesize	1048576
ejbserver.logger.channels.define.ThreaddumpWatchLogFile.filenum	2
ejbserver.logger.channels.define.ThreaddumpWatchLogFile.filesize	1048576
ejbserver.logger.channels.define.ThreadWatchLogFile.filenum	2
ejbserver.logger.channels.define.ThreadWatchLogFile.filesize	1048576
ejbserver.logger.channels.define.UserErrLogFile.filenum	2
ejbserver.logger.channels.define.UserErrLogFile.filesize	1048576
ejbserver.logger.channels.define.UserOutLogFile.filenum	2
ejbserver.logger.channels.define.UserOutLogFile.filesize	1048576
ejbserver.logger.enabled.*	Error, Warning

(14) ejbserver.management から始まるパラメタ

ejbserver.management から始まるパラメタについて次の表に示します。

表 4-72 ejbserver.management から始まるパラメタ

param-name 指定値	省略値
ejbserver.management.JVM.stats_monitor.FullGCCCount.enabled	true
ejbserver.management.JVM.stats_monitor.FullGCCCount.interval	600
ejbserver.management.JVM.stats_monitor.FullGCCCount.threshold	10
ejbserver.management.statistics.interval	60
ejbserver.management.stats_file.base_time	0
ejbserver.management.stats_file.dir	【Windows の場合】 <製品の作業ディレクトリ>¥ejb¥<サーバ名称>¥stats 【UNIX の場合】 <製品の作業ディレクトリ>/ejb/<サーバ名称>/stats
ejbserver.management.stats_file.enabled	true
ejbserver.management.stats_file.num	7
ejbserver.management.stats_file.period	24

(15) ejbserver.manager から始まるパラメタ

ejbserver.manager から始まるパラメタについて次の表に示します。

表 4-73 ejbserver.manager から始まるパラメタ

param-name 指定値	省略値
ejbserver.manager.agent.JP1EventAgent.conf	なし
ejbserver.manager.agent.JP1EventAgent.enabled	false
ejbserver.manager.agent.MEventAgent.conf	なし
ejbserver.manager.agent.MEventAgent.enabled	true

(16) ejbserver.naming から始まるパラメタ

ejbserver.naming から始まるパラメタについて次の表に示します。

表 4-74 ejbserver.naming から始まるパラメタ

param-name 指定値	省略値
ejbserver.naming.host	バッチサーバと同じホスト
ejbserver.naming.port	900
ejbserver.naming.startupRetryCount	9
ejbserver.naming.startupWaitTime	1

(17) ejbserver.rmi から始まるパラメタ

ejbserver.rmi から始まるパラメタについて次の表に示します。

表 4-75 ejbserver.rmi から始まるパラメタ

param-name 指定値	省略値
ejbserver.rmi.naming.host	なし
ejbserver.rmi.naming.port	23152
ejbserver.rmi.remote.listener.port	0
ejbserver.rmi.request.timeout	0

(18) ejbserver.server から始まるパラメタ

ejbserver.server から始まるパラメタについて次の表に示します。

表 4-76 ejbserver.server から始まるパラメタ

param-name 指定値	省略値
ejbserver.server.prf.PRfid	なし

param-name 指定値	省略値
ejbserver.server.threaddump.filenum	256

(19) `ejbserver.stdoutlog` から始まるパラメタ

`ejbserver.stdoutlog` から始まるパラメタについて次の表に示します。

表 4-77 `ejbserver.stdoutlog` から始まるパラメタ

param-name 指定値	省略値
ejbserver.stdoutlog.autoflush	false

(20) `ejbserver.watch` から始まるパラメタ

`ejbserver.watch` から始まるパラメタについて次の表に示します。

表 4-78 `ejbserver.watch` から始まるパラメタ

param-name 指定値	省略値
ejbserver.watch.enabled	true
ejbserver.watch.fileDescriptor.enabled	true
ejbserver.watch.fileDescriptor.interval	60
ejbserver.watch.fileDescriptor.threshold	2147483647
ejbserver.watch.fileDescriptor.writefile.enabled	true
ejbserver.watch.memory.enabled	true
ejbserver.watch.memory.interval	60
ejbserver.watch.memory.threshold	80
ejbserver.watch.memory.writefile.enabled	true
ejbserver.watch.thread.enabled	true
ejbserver.watch.thread.interval	60
ejbserver.watch.thread.threshold	2147483647
ejbserver.watch.thread.writefile.enabled	true
ejbserver.watch.threaddump.enabled	true
ejbserver.watch.threaddump.interval	30
ejbserver.watch.threaddump.threshold	80
ejbserver.watch.threaddump.writefile.enabled	true

(21) java から始まるパラメタ

java から始まるパラメタについて次の表に示します。

表 4-79 java から始まるパラメタ

param-name 指定値	省略値
java.naming.factory.initial	com.hitachi.software.ejb.jndi.InsContextFactory

(22) vbroker から始まるパラメタ

vbroker から始まるパラメタについて次の表に示します。

表 4-80 vbroker から始まるパラメタ

param-name 指定値	省略値
vbroker.agent.port	14000
vbroker.se.iiop_tp.host	なし
vbroker.se.iiop_tp.scm.iiop_tp.listener.port	0

(23) webserver.connector から始まるパラメタ

webserver.connector から始まるパラメタについて次の表に示します。

表 4-81 webserver.connector から始まるパラメタ

param-name 指定値	省略値
webserver.connector.http.bind_host	なし
webserver.connector.http.permitted.hosts※	なし

注※

param-value に値を複数指定する際は、コンマ「,」で区切って指定します。

(例)

<param-value>値 1,値 2,値 3</param-value>

4.11.7 バッチサーバ用オプション定義を設定するパラメタ

バッチサーバ用オプション定義を設定するパラメタについて、次の表に示します。なお、「param-name 指定値」に対応する param-value の指定内容については、「[3.2.1 usrconf.cfg \(バッチサーバ用オプション定義ファイル\)](#)」を参照してください。「省略値」とは、パラメタの指定がない場合に仮定される値です。

バッチサーバ用オプション定義を設定するパラメタを指定する場合の指定形式を次に示します。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
```

値を複数指定する場合は、複数の<param-value>を指定します。

値を複数指定する場合の指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
<param-value>値</param-value>
```

usrconf.cfg（J2EE サーバ用オプション定義ファイル）に指定できるパラメタのうち、「[4.11.9 バッチサーバの拡張パラメタ](#)」の表と、この項の表の両方に記載していないパラメタは、次の形式で指定してください。

指定形式

```
<param-name>ex. param</param-name>
<param-value>パラメタ=値</param-value>
```

値を複数指定する場合は、複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>ex. param</param-name>
<param-value>パラメタ=値</param-value>
<param-value>パラメタ=値</param-value>
```

この項に記載があるパラメタをこの形式で指定した場合、動作が保証されません。

表 4-82 バッチサーバ用オプション定義を設定するパラメタ

param-name 指定値	省略値
add.jvm.arg※	なし
ejb.public.directory	【Windows の場合】 &{cosminexus.home}¥CC¥server¥public 【UNIX の場合】 &{cosminexus.home}/CC/server/public
ejb.server.log.directory	【Windows の場合】 <製品の作業ディレクトリ>¥ejb¥<サーバ名称>¥logs 【UNIX の場合】 <製品の作業ディレクトリ>/ejb/<サーバ名称>/logs

注※

値の指定形式について、次に示します。
(例)
<param-name>add.jvm.arg</param-name>


```
<param-value>-Xms256m</param-value>
<param-value>-Xmx512m</param-value>
:
```

4.11.8 バッチサーバの JavaVM のシステムプロパティに設定できるパラメタ

バッチサーバの JavaVM のシステムプロパティを設定するパラメタについて、次の表に示します。なお、「param-value 指定値」に指定する値の詳細については、「[3.2.2 usrconf.properties \(バッチサーバ用ユーザプロパティファイル\)](#)」を参照してください。「省略値」とは、パラメタの指定がない場合に仮定される値です。

バッチサーバの JavaVM のシステムプロパティに設定するパラメタを指定する場合の指定形式を次に示します。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
```

値を複数指定する場合は、複数の<param-value>を指定します。

値を複数指定する場合の指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
<param-value>値</param-value>
```

usrconf.properties (バッチサーバ用ユーザプロパティファイル) に指定できるパラメタのうち、「[4.11.6 バッチサーバ用ユーザプロパティを設定するパラメタ](#)」の表と、この項の表の両方に記載していないパラメタは、次の形式で指定してください。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>パラメタ=値</param-value>
```

値を複数指定する場合は、複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>ex. properties</param-name>
<param-value>パラメタ=値</param-value>
<param-value>パラメタ=値</param-value>
```

なお、この項の表で示すパラメタは、この形式で指定することもできます。

表 4-83 バッチサーバの JavaVM のシステムプロパティに設定できるパラメタ

param-name 指定値	省略値
ejbserver.connectionpool.applicationAuthentication.disabled	false
ejbserver.connectionpool.sharingOutsideTransactionScope.enabled	false
ejbserver.deploy.exclusive.lockAliveInterval	60
ejbserver.jndi.cache.reference	off
ejbserver.manager.agent.Agent.conf※ ¹	<Application Server のインストールディレクトリ>manager/config/mnagent.<実サーバ名>.properties
ejbserver.manager.agent.Agent.enabled※ ¹	true
ejbserver.manager.jpl.event.event_server_name	localhost
ejbserver.naming.nameroot	なし
vbroker.agent.enableLocator※ ²	false
vbroker.orb.htc.comt.entryCount	120000
vbroker.orb.htc.comt.fileCount	3
vbroker.orb.htc.tracePath	【Windows の場合】 <製品の作業ディレクトリ>¥ejb¥<サーバ名称>¥logs¥TPB¥logj 【UNIX の場合】 <製品の作業ディレクトリ>/ejb/<サーバ名称>/logs/TPB/logj

注※1

Management Server が自動で設定するため、指定は非推奨です。

注※2

通常は設定不要です。

4.11.9 バッチサーバの拡張パラメタ

バッチサーバの拡張パラメタを設定するパラメタについて、次の表に示します。バッチサーバの場合、batch.service.enabled パラメタは必ず指定してください。パラメタの値には true を指定します。

なお、「param-value 指定値」に指定する値の詳細については、「[3.2.1 usrconf.cfg \(バッチサーバ用オプション定義ファイル\)](#)」を参照してください。「省略値」とは、パラメタの指定がない場合に仮定される値です。

バッチサーバの拡張パラメタを設定するパラメタを指定する場合の指定形式を次に示します。

指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
```

値を複数指定する場合は、複数の<param-value>を指定します。

値を複数指定する場合の指定形式

```
<param-name>パラメタ</param-name>
<param-value>値</param-value>
<param-value>値</param-value>
```

usrconf.cfg (J2EE サーバ用オプション定義ファイル) に指定できるパラメタのうち、[「4.11.7 バッチサーバ用オプション定義を設定するパラメタ」](#)の表と、この項の表の両方に記載していないパラメタは、次の形式で指定してください。

指定形式

```
<param-name>ex. param</param-name>
<param-value>パラメタ=値</param-value>
```

値を複数指定する場合は、複数の param-value を指定します。

値を複数指定する場合の指定形式

```
<param-name>ex. param</param-name>
<param-value>パラメタ=値</param-value>
<param-value>パラメタ=値</param-value>
```

なお、この項の表で示すパラメタは、この形式で指定することもできます。

表 4-84 バッチサーバの拡張パラメタ

param-name 指定値	省略値
add.class.path	なし
add.library.path	なし
batch.service.enabled※ ¹	false
cpp.library.version	6
ejb.server.corefilenum※ ²	1
ejb.server.log.stderr.filesize	1048576
ejb.server.log.stdout.filesize	1048576
jvm.type	server

注※1

batch.service.enabled パラメタは必ず指定してください。設定値には true を指定します。パラメタの指定を省略すると false が設定されます。また、このパラメタは Web システムの初期構築時だけ指定できます。Web システムの構成変更では値を変更しないでください。

注※2

UNIX 用のパラメタです。

4.11.10 運用監視エージェントプロパティを設定するパラメタ

運用監視エージェントプロパティを設定するパラメタについて次の表に示します。これらのパラメタは J2EE サーバおよびバッチサーバで指定できます。

「param-name 指定値」に対応する param-value の指定内容については、「[8.2.5 mngagent.<実サーバ名>.properties \(運用監視エージェントプロパティファイル\)](#)」を参照してください。また、参照する場合はキーをパラメタに読み替えてください。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-85 運用監視エージェントプロパティを設定するパラメタ

param-name 指定値	指定可能値	省略値	VR
mngagent.connector.host	指定できる値を次に示します。 • ホスト名※ • IPv4 アドレス※ • @myhost	利用可能な IP アドレスが自動的に割り当てられます。	07-50
mngagent.connector.port	1～65535 の整数で指定します。	空いているポート番号から自動的に割り当てられます。	07-50
mngagent.log.filesize	4096～2147483647 の整数で指定します。	65536	06-50 07-00 07-50

注※

ホスト名または IPv4 アドレスを指定する場合は、ホストの定義の<host-name>タグの値を指定してください。異なる値を指定した場合、警告メッセージ (KEOS24186-W) が出力され、意図しない設定となることがあります。
なお、ホスト名には、英数字、アンダースコア「_」、ピリオド「.」、およびハイフン「-」で指定した 255 文字以内の文字列を指定してください。

4.11.11 Management イベント発行用プロパティを設定するパラメタ

Management イベント発行用プロパティを設定するパラメタについて次の表に示します。これらのパラメタは J2EE サーバおよびバッチサーバで指定できます。

「param-name 指定値」に対応する param-value の指定内容については、「[8.2.11 Management イベント発行用プロパティファイル](#)」を参照してください。また、参照する場合は、キーをパラメタに読み替

えてください。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-86 Management イベント発行用プロパティ設定するパラメタ

param-name 指定値	指定可能値	省略値	VR
manager.mevent.message_id.list	任意の文字列を指定します。	なし	06-50
manager.mevent.retry.interval	1～86400 の整数で指定します。	10	06-50
manager.mevent.retry.limit	0～86400 の整数で指定します。	0	06-50
manager.mevent.send.max	1～1000 の整数で指定します。	10	06-50
manager.mevent.send.timeout	10～600 の整数で指定します。	90	06-50
manager.mevent.sender.bind.host	次のどれかを指定します。 • ホスト名※ • IPv4 アドレス※ • @myhost	なし	07-50

注※

ホスト名または IPv4 アドレスを指定する場合は、ホストの定義の<host-name>タグの値を指定してください。異なる値を指定した場合、警告メッセージ（KEOS24186-W）が出力され、意図しない設定となることがあります。

なお、ホスト名には、英数字、アンダースコア「_」、ピリオド「.」、およびハイフン「-」で指定した 255 文字以内の文字列を指定してください。

4.11.12 JP1 連携を設定するパラメタ

JP1 連携を設定するパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

これらのパラメタは J2EE サーバおよびバッチサーバで指定できます。

表 4-87 JP1 連携を設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
manager.jp1event.system.filtering.severity.alert	システムが発行する JP1 イベントのフィルタリングを設定する項目です。 JP1 イベント重大度「Alert」を発行するかどうかを指定します。次のどちらかを指定します。 true : 発行します。 false : 発行しません。	指定できる文字列を次に示します。 • true • false	false	06-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
manager.jp1 event.system.filtering.severity.critical	システムが発行する JP1 イベントのフィルタリングを設定する項目です。 JP1 イベント重大度「Critical」を発行するかどうかを指定します。次のどちらかを指定します。 true : 発行します。 false : 発行しません。	指定できる文字列を次に示します。 • true • false	false	06-50
manager.jp1 event.system.filtering.severity.emergency	システムが発行する JP1 イベントのフィルタリングを設定する項目です。 JP1 イベント重大度「Emergency」を発行するかどうかを指定します。次のどちらかを指定します。 true : 発行します。 false : 発行しません。	指定できる文字列を次に示します。 • true • false	false	06-50
manager.jp1 event.system.filtering.severity.error	システムが発行する JP1 イベントのフィルタリングを設定する項目です。 JP1 イベント重大度「Error」を発行するかどうかを指定します。次のどちらかを指定します。 true : 発行します。 false : 発行しません。	指定できる文字列を次に示します。 • true • false	false	06-50
manager.jp1 event.system.filtering.severity.information	システムが発行する JP1 イベントのフィルタリングを設定する項目です。 JP1 イベント重大度「Information」を発行するかどうかを指定します。次のどちらかを指定します。 true : 発行します。 false : 発行しません。	指定できる文字列を次に示します。 • true • false	false	06-50
manager.jp1 event.system.filtering.severity.notice	システムが発行する JP1 イベントのフィルタリングを設定する項目です。 JP1 イベント重大度「Notice」を発行するかどうかを指定します。次のどちらかを指定します。 true : 発行します。 false : 発行しません。	指定できる文字列を次に示します。 • true • false	false	06-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
manager.jp1 event.system.filtering.severity.warning	システムが発行する JP1 イベントのフィルタリングを設定する項目です。 JP1 イベント重大度「Warning」を発行するかどうかを指定します。次のどちらかを指定します。 true : 発行します。 false : 発行しません。	指定できる文字列を次に示します。 • true • false	false	06-50
manager.jp1 event.user.filtering.enabled	ユーザ JP1 イベントをフィルタリングするかどうかを指定します。次のどちらかを指定します。 true : フィルタリングします。 false : フィルタリングしません。	指定できる文字列を次に示します。 • true • false	false	06-50
manager.jp1 event.user.filtering.filter	ユーザ JP1 イベントのフィルタを指定します。指定した文字列中の¥n は改行に変換されます。	任意の文字列を指定します。	なし	06-50
manager.jp1 event.user.mapping.level.config	ユーザ JP1 イベントのマッピングのログレベル「CONFIG」に対する JP1 イベント重大度を指定します。次のどれかを指定します。 • Emergency : 緊急 • Alert : 警戒 • Critical : 致命的 • Error : エラー • Warning : 警告 • Notice : 通知 • Information : 情報 指定しない場合、JP1 イベントは発行されません。	指定できる文字列を次に示します。 • Emergency • Alert • Critical • Error • Warning • Notice • Information	未指定	06-50
manager.jp1 event.user.mapping.level.fine	ユーザ JP1 イベントのマッピングのログレベル「FINE」に対する JP1 イベント重大度を指定します。次のどれかを指定します。 • Emergency : 緊急 • Alert : 警戒 • Critical : 致命的 • Error : エラー • Warning : 警告 • Notice : 通知 • Information : 情報 指定しない場合、JP1 イベントは発行されません。	指定できる文字列を次に示します。 • Emergency • Alert • Critical • Error • Warning • Notice • Information	未指定	06-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
manager.jp1 event.user. mapping.level.finer	ユーザ JP1 イベントのマッピングのログレベル「FINER」に対する JP1 イベント重大度を指定します。次のどれかを指定します。 • Emergency：緊急 • Alert：警戒 • Critical：致命的 • Error：エラー • Warning：警告 • Notice：通知 • Information：情報 指定しない場合、JP1 イベントは発行されません。	指定できる文字列を次に示します。 • Emergency • Alert • Critical • Error • Warning • Notice • Information	未指定	06-50
manager.jp1 event.user. mapping.level.finest	ユーザ JP1 イベントのマッピングのログレベル「FINEST」に対する JP1 イベント重大度を指定します。次のどれかを指定します。 • Emergency：緊急 • Alert：警戒 • Critical：致命的 • Error：エラー • Warning：警告 • Notice：通知 • Information：情報 指定しない場合、JP1 イベントは発行されません。	指定できる文字列を次に示します。 • Emergency • Alert • Critical • Error • Warning • Notice • Information	未指定	06-50
manager.jp1 event.user. mapping.level.info	ユーザ JP1 イベントのマッピングのログレベル「INFO」に対する JP1 イベント重大度を指定します。次のどれかを指定します。 • Emergency：緊急 • Alert：警戒 • Critical：致命的 • Error：エラー • Warning：警告 • Notice：通知 • Information：情報 指定しない場合、JP1 イベントは発行されません。	指定できる文字列を次に示します。 • Emergency • Alert • Critical • Error • Warning • Notice • Information	未指定	06-50
manager.jp1 event.user. mapping.level.severe	ユーザ JP1 イベントのマッピングのログレベル「SEVERE」に対する JP1 イベント重大度を指定します。次のどれかを指定します。 • Emergency：緊急 • Alert：警戒 • Critical：致命的 • Error：エラー • Warning：警告	指定できる文字列を次に示します。 • Emergency • Alert • Critical • Error • Warning • Notice • Information	未指定	06-50

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	• Notice：通知 • Information：情報 指定しない場合、JP1 イベントは発行されません。			
manager.jp1 event.user. mapping.level.warning	ユーザ JP1 イベントのマッピングのログレベル 「WARNING」に対する JP1 イベント重大度を指定 します。次のどれかを指定します。 • Emergency：緊急 • Alert：警戒 • Critical：致命的 • Error：エラー • Warning：警告 • Notice：通知 • Information：情報 指定しない場合、JP1 イベントは発行されません。	指定できる文字列を次に示します。 • Emergency • Alert • Critical • Error • Warning • Notice • Information	未指定	06-50

4.11.13 SecurityManager の使用を設定するパラメタ

SecurityManager の使用を設定するパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

このパラメタは J2EE サーバおよびバッチサーバで指定できます。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

表 4-88 SecurityManager の使用を設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR	関連情報
use.security	J2EE サーバの起動オプションに SecurityManager を使用するか どうかを指定します。次のどちら かを指定します。 true： 使用します。 false： 使用しません。	指定できる文字列を次に 示します。 • true • false	true	06-50	「機能解説 セキュリティ管理機能編」の「2.2.5 不正な処理の実行を防止する」

4.11.14 統合ユーザ管理の使用を設定するパラメタ

統合ユーザ管理の使用を設定するパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

なお、統合ユーザ管理の使用を設定するパラメタは、jaas.ua.enabled パラメタに true が指定された場合にだけ有効です。

表 4-89 統合ユーザ管理の使用を設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
jaas.ua.enabled	JavaVM の JAAS を有効にするかどうかを指定します。 有効にすると統合ユーザ管理機能を使用したアプリケーションが利用できるようになります。 次のどちらかを指定します。 true : 有効にします。 false : 無効にします。	指定できる文字列を次に示します。 • true • false	true	08-70
java.security.auth.login.config	JAAS のコンフィグレーションファイルを指定します。 相対パスで指定した場合は、<Application Server のインストールディレクトリ>/manager/config ディレクトリの下でのコンフィグレーションファイルが適用されます。	任意の文字列を指定します。	【Windows の場合】 &{cosminexus.home}\manager\config\jaas.conf 【UNIX の場合】 &{cosminexus.home}/manager/config/jaas.conf	08-70
com.cosminexus.admin.auth.config	統合ユーザ管理のコンフィグレーションファイルを指定します。 相対パスで指定した場合は、<Application Server のインストールディレクトリ>/manager/config ディレクトリの下でのコンフィグレーションファイルが適用されます。	任意の文字列を指定します。	【Windows の場合】 &{cosminexus.home}	08-70

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
			¥manager¥config¥ua.conf 【UNIXの場合】 &{cosminexus.home}/manager/config/ua.conf	
com.cosminexus.admin.auth.passwordScramble.enable	統合ユーザ管理のコンフィグレーションファイルのスクランブル化されたパスワードを複合化する機能を有効にするかどうかを指定します。次のどちらかを指定します。 true : 有効にします。 false : 無効にします。	指定できる文字列を次に示します。 • true • false	false	08-70
jaas.config.load_exclusively※	JAAS のコンフィグレーションファイルに指定したファイル以外で指定されたログイン構成を無視するかどうかを選択します。 次のどちらかを指定します。 true : 指定されたログイン構成を無視し、JAAS コンフィグレーションファイルの構成が有効となります。 false : 指定されたログイン構成を有効とします。	指定できる文字列を次に示します。 • true • false	true	08-70

注※

このパラメタに true を指定した場合、配布先の usrconf.properties ファイルの java.security.auth.login.config パラメタの値に、次のように「¥=」が付加されます。

(例) : java.security.auth.login.config=¥=<コンフィグファイル>

4.11.15 JavaVM の起動パラメタで使用するファイルを設定するパラメタ

JavaVM の起動パラメタで使用するファイルを設定するパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-90 JavaVM の起動パラメタで使用するファイルを設定するパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
AutoExplicitMemoryText	明示管理ヒープ機能の自動配置機能で使用する自動配置設定ファイルの内容を指定します。ここで指定した内容は、<Application Serverのインストールディレクトリ>/CC/server/usrconf/ejb/<実サーバ名>/auto_explicit_memory.cfg に反映されます。※ 注意事項 JDK17 以降の場合、明示管理ヒープ機能は非サポートのため、このパラメタは指定できません。指定できない値が指定された場合、エラーメッセージ KEOS24015-E を出力し、リターンコード 64 でコマンドを終了します。 明示管理ヒープ機能のシステムプロパティを指定した場合は、エラーメッセージ KEOS24060-E を出力し、リターンコード 64 でコマンドを終了します。	1 文字以上の文字列	なし	11-40
UserPrfText	ユーザ拡張性能解析トレースで使用するユーザ拡張性能解析トレース設定ファイルの内容を指定します。ここで指定した内容は、<Application Serverのインストールディレクトリ>/CC/server/usrconf/ejb/<実サーバ名>/userprf.cfg に反映されます。※	1 文字以上の文字列	なし	09-00

注※

param-value の値は CDATA セクションで指定してください。

(例)

```
<param-name>UserPrfText</param-name>
<param-value>
  <![CDATA[
    :
  ]]>
</param-value>
```

CDATA セクション内に指定する内容全体の前後の半角スペースおよび改行は無視されます。

4.12 論理パフォーマンストレーサで指定できるパラメタ

ここでは、論理パフォーマンストレーサで指定できるパラメタについて説明します。

論理パフォーマンストレーサで指定できるパラメタについて次の表に示します。なお、「省略値」とは、パラメタの指定がない場合に仮定される値です。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 4-91 論理パフォーマンストレーサに設定できるパラメタ

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
PRFID	PRF 識別子を指定します。	「TSC」, 「tsc」または「CTM」や「ctm」で始まる文字列を指定した場合、エラーとなります。	PRF_ID	06-50 07-00
prfspool	PRF トレース出力ディレクトリを指定します。 Windows の場合、ここで指定する値は、システム環境変数 PRFSPOOL で指定された値と一致している必要があります。	任意の文字列を 256 文字以内で指定します。	& ; {cosminexus .home }/PRF /spool	06-50 07-00
PrfTraceBufferSize	パフォーマンストレーサのバッファメモリサイズを指定します。 PrfTraceFileSize ≥ PrfTraceBufferSize の関係が成り立つ値を設定してください。	512～102400 の整数で指定します。	8192	06-50
PrfTraceCount	パフォーマンストレーサのファイル面数を指定します。	4, 16, 32, 64, 128, 256 のどれかを指定します。	4	06-50
PrfTraceFileSize	パフォーマンストレーサのファイルサイズをキロバイト単位で指定します。 PrfTraceFileSize ≥ PrfTraceBufferSize の関係が成り立つ値を設定してください。	指定できる文字列を次に示します。 1024 4096 8192 16384 32768 65536 131072 262144 524288 1048576	32768	06-50 07-00
PrfTraceLevel	パフォーマンストレーサのトレース取得レベルを指定します。次のどれかを設定してください。 - STANDARD : 標準 - DETAIL :	指定できる文字列を次に示します。 - STANDARD - DETAIL - 任意のトレースレベル値	STANDARD	06-50 07-00

param-name 指定値	param-value の指定内容	指定可能値	省略値	VR
	詳細 任意のトレース取得レベル値： 指定できる値については、マニュアル「アプリケーションサーバ リファレンス コマンド編」の「cprfstart (PRF デーモンの開始)」の引数-PrfTraceLevel を参照してください。			

4.13 論理 CTM ドメインマネジャで指定できるパラメタ

ここでは、論理 CTM ドメインマネジャで指定できるパラメタについて説明します。

論理 CTM ドメインマネジャで指定できるパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

表 4-92 論理 CTM ドメインマネジャに設定できるパラメタ

param-name 指定値	param-value の指定内容	省略値	関連情報
cdm.agent.port	スマートエージェントのポート番号を指定します。 このパラメタは、free-tier の場合にだけ使用します。	14000	
cdm.AliveCheckCount	CTM ドメインマネジャ間の稼働状態の監視間隔の係数を、2～255 の範囲の整数で指定します。	2	
cdm.port	CTM ドメインマネジャが CTM ドメイン構成情報を交換するために使用するポート番号を 5001～65535 の範囲の整数で指定します。 同じ CTM ドメイン名称を持つ CTM ドメインマネジャを複数のホストで開始する場合、-CTMPort オプションに同じ値を指定してください。省略した場合はサービス名ファイルが参照されます。サービス名ファイルに指定がないときは"20137"が設定されます。 CTM ドメインマネジャは一つのポート番号で tcp および udp の両方のプロトコルを使用します。そのため、CTM ドメインマネジャが使用するポート番号をほかの udp のポート番号として使用しないでください。	20137	
cdm.prf.PRFID	PRF 識別子を指定します。 PRF デーモン起動時に PRF 識別子を指定した場合に、同じ PRF 識別子を指定します。 PRF デーモン起動時に PRF 識別子を省略した場合は、指定しないでください。PRF 識別子が一致しないと、性能解析トレースが取得できません。 このパラメタは、free-tier の場合にだけ使用します。	なし	

param-name 指定値	param-value の指定内容	省略値	関連情報
cdm.SendHost [※]	CTM ドメイン構成情報の配布先となる、異なるネットワークセグメントにある CTM ドメインマネージャが動作するホストの IP アドレスを指定します。このとき指定する IP アドレスは、ホストの定義の <host-name> タグで指定する IP アドレスです。 この指定によって、複数のネットワークセグメントで構成した CTM ドメインで、CTM 構成情報を配布できるようになります。	なし	
cdm.SendHostInterval	cdm.SendHost で指定したホストに対して CTM ドメイン構成情報を送信する間隔を 0~65535 の範囲の整数（単位：秒）で指定します。 なお、指定しない場合は、ブロードキャスト間隔と同じであることを示します。0 を指定した場合は送信しないことを示します。	なし	
cdm.SendInterval	CTM ドメインマネージャが CTM ドメイン構成情報をネットワークセグメント内にブロードキャストする間隔を 0~65535 の範囲の整数（単位：秒）で指定します。0 は送信しないことを表します。	60	
cdm.SubnetMask	ネットワークのサブネットマスクを指定します。	255.255.255.0	
ctmdomname	CTM ドメインマネージャが属する CTM ドメイン名称を 31 文字以内で指定します。使用できる文字は英数字、アンダースコア「_」です。「CTMDOMAIN」以外の、「CTM」または「ctm」で始まる文字列は指定しないでください。 同一ホスト内でユニークな名称にする必要があります。	CTMDOMAIN	「機能解説 拡張編」の「3.3.5 CTM ドメインと CTM ドメインマネージャ」
ctmsspool	CTM 実行環境ディレクトリを指定します。ファイルセパレータには「/」を使用します。	&{cosminexus.home}/CTM/spool	

注※

param-value に値を複数指定する際は、コンマ「,」で区切って指定します。

(例)

<param-value>値 1,値 2,値 3</param-value>

4.14 論理 CTM で指定できるパラメタ

ここでは、論理 CTM で指定できるパラメタについて説明します。

論理 CTM で指定できるパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

表 4-93 論理 CTM に設定できるパラメタ

param-name 指定値	param-value の指定内容	省略値	関連情報
ctm.Agent	CTM レギュレータの ORB ゲートウェイ機能を使用するかどうかを指定します。次のどちらかを指定します。 バッチアプリケーションを実行するシステムの場合は、必ず「1」を指定してください。 1： 使用します。 0： 使用しません。	0	
ctm.cdm	利用する CTM ドメインマネージャの CTM ドメイン名を指定します。同一ホスト内の CTM ドメインマネージャの CTM ドメイン名を指定してください。	CTMDOMAIN	
ctm.CreatePolicy	create 要求の CTM ノード選択ポリシーを選択します。次のどちらかを指定します。 normalDispatch： スケジュールキューの負荷状況に応じて CTM デーモンを選択します。 priorDispatch： create 要求を受け付けた CTM デーモンを、優先的に選択します。ただし、その CTM デーモンに登録されているスケジュールキューが高負荷状態、および閉塞状態の場合は、ほかの CTM デーモンを選択します。	normalDispatch	「機能解説 拡張編」の「3.8 リクエストの負荷分散」
ctm.DCSendTimeOut	CTM 間通信のリクエスト転送時のタイムアウト時間を 0～2147483647 の範囲の整数（単位：秒）で指定します。 0 は設定しないことを表します。	180	
ctm.DispatchParallelCount	自 CTM 内でクライアントからの要求メッセージを J2EE サーバに振り分けるスレッ	255	

param-name 指定値	param-value の指定内容	省略値	関連情報
	ドの最大値を 0～32767 の範囲の整数で指定します。		
ctm.DispatchPolicy	リクエストのスケジュールポリシーを選択します。 normalDispatch : スケジュールキューの負荷状況に応じてリクエストをスケジュールします。 priorDispatch : リクエストを受け付けた CTM デーモンに登録されているスケジュールキューに、優先的にリクエストをスケジュールします。ただし、その CTM デーモンに登録されているスケジュールキューが高負荷状態、および閉塞状態の場合は、ほかの CTM デーモンに登録しているスケジュールキューにリクエストをスケジュールします。	priorDispatch	「機能解説 拡張編」の「3.8 リクエストの負荷分散」
ctm.EjbPort	EJB クライアントが CTM デーモンの新しいコネクションを探すときに使用するポート番号を 5001～65535 の範囲の整数で指定します。指定しない場合は OS が自動的に割り当てたポート番号を使用します。	自動	
ctm.LoadCheckInterval	スケジュールキューの負荷状況を監視する時間間隔を 0 から 32767 の範囲の整数（単位：秒）で指定します。0 は一定時間間隔での負荷状況監視を行わないことを表します。	10	
ctm.LogFileCount	CTM ログファイルの面数を指定します。指定できる値は、2, 4, 8, 16, 24, 32 です。	2	
ctm.LogFileSize	CTM ログファイルの 1 ファイルの最大サイズを 1～3 の範囲の整数（単位：メガバイト）で指定します。	1	
ctm.MaxRequestCount	CTM に登録する CTM キューが、同時に登録できるリクエスト数を 1～32767 の範囲の整数で指定します。	50	
ctm.ns.port	CTM と対となるネーミングサービスのポート番号を 1～65535 の範囲の整数で指定します。	20348	
ctm.NsBindHost	一つのマシンに複数の IP アドレスが割り当てられている場合に、特定の IP アドレスに割り当てかどうかを指定します。次のどちらかを指定します。	false	

param-name 指定値	param-value の指定内容	省略値	関連情報
	true : 特定の IP アドレスに割り当てます。 false : 特定の IP アドレスに割り当てません。		
ctm.port	CTM が使用するポート番号を 5001～65535 の範囲の整数で指定します。	20138	
ctm.QueueCount	CTM に登録できる CTM キュー数を 1～32767 の範囲の整数で指定します。	128	
ctm.QueueDeleteWait	異常終了した J2EE サーバが登録していた CTM キューを削除しないで保留し、J2EE サーバの再開始を待つ場合の待ち時間を、0～2147483647 の範囲の整数（単位：秒）で指定します。	0	
ctm.QueueInterval	システム停止しきい値の処理率を求める時間間隔を 1～32767 の範囲の整数（単位：秒）で指定します。	なし	
ctm.QueueRate	キュー滞留監視状態へ移行する滞留率のしきい値を 0～99 の範囲の整数（単位：%）で指定します。	なし	
ctm.QueueRegistCount	同じ CTM キューを共有できる J2EE アプリケーション数を 1～32767 の範囲の整数で指定します。	64	
ctm.RegOption	CTM レギュレータの設定ファイルを、CTM 稼働ホストの絶対パス（256 文字以内の文字列）で指定します。	なし	
ctm.RegStart	CTM 開始時に自動開始する CTM レギュレータのプロセス数を 0～32767 の範囲の整数で指定します。	1	
ctm.RequestCount	何回タイムアウトが発生したら自動閉塞するかを 1～32767 の範囲の整数で指定します。	なし	
ctm.RequestInterval	タイムアウト発生回数を求める時間間隔を 1～32767 の範囲の整数（単位：秒）で指定します。	なし	
ctm.ServerCacheSize	CTM ドメインの構成情報を格納するキャッシュテーブルのサイズを 1～32767 の範囲の整数（単位：キロバイト）で指定します。	1024	
ctm.ServerConnectCount	CTM に接続する J2EE サーバの最大数を 0～32767 の範囲の整数で指定します。	64	
ctm.StatsFileCount	CTM 稼働統計情報の世代数を指定します。	3	

param-name 指定値	param-value の指定内容	省略値	関連情報
	指定できる値は、3, 4, 16, 32, 64, 128, 256 です。		
ctm.StatsFileSize	CTM 稼働統計情報のファイルのサイズをメガバイト単位で指定します。 指定できる値は、1, 2, 3, 4, 8, 16, 32, 64, 128, 256, 512, 1024 です。	3	
ctm.StatsInterval	CTM 稼働統計情報をファイルに出力する時間間隔を 1～1440 の範囲の整数（単位：分）で指定します。	1	
ctm.StatsUse	CTM 稼働統計情報を取得するかどうかを設定します。次のどちらかを指定します。 Y： 取得します。 N： 取得しません。	Y	
ctm.SystemDown	システムを停止するかどうかを指定します。次のどちらかを指定します。 1： 停止します。 0： 停止しません。	1	
ctm.SystemDownRate	システム停止へ移行する処理率のしきい値を 1～100 の範囲の整数（単位：%）で指定します。	なし	
ctm.TSCGwOption	OTM ゲートウェイの設定ファイルを、CTM 稼働ホストの絶対パス（256 文字以内の文字列）で指定します。	なし	
ctm.TSCGwStart	CTM 開始時に自動開始する OTM ゲートウェイのプロセス数を 0～32767 の範囲の整数で指定します。 0 は OTM ゲートウェイを起動しないことを表します。	0	
ctm.WatchQueue	キューの停留監視をするかどうかを指定します。次のどちらかを指定します。 true： キューの停止監視をします。 false： キューの停止監視をしません。	false	
ctm.WatchRequest	J2EE サーバへのリクエストの送信でタイムアウトが発生したときにキューを閉塞す	false	

param-name 指定値	param-value の指定内容	省略値	関連情報
	るかどうかを指定します。次のどちらかを指定します。 true : キューを閉塞します。 false : キューを閉塞しません。		
ctmid	CTM の識別子を 31 文字以内の文字列で指定します。使用できる文字は英数字、アンダースコア「_」、およびピリオド「.」です（ピリオドは IP アドレスの場合だけ指定できます）。「CTM」または「ctm」で始まる文字列は指定しないでください。 このパラメタを指定する場合、運用管理ドメイン内のすべての論理 CTM サーバで一意の値を指定する必要があります。 なお、IP アドレスを指定した場合、ピリオド「.」は、アンダースコア「_」に置き換えられます。	<ホストの定義の <agent-host>タグに指定された値>※_<CTM のポート番号>	

注※

<agent-host>タグにホスト名を指定している場合、IP アドレスに変換した値が指定されます。また、<agent-host>タグに IP アドレスを指定している場合、ピリオド「.」はアンダースコア「_」に置き換えられます。

4.15 論理スマートエージェントで指定できるパラメタ

ここでは、論理スマートエージェントで指定できるパラメタについて説明します。

論理スマートエージェントで指定できるパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。

表 4-94 論理スマートエージェントに設定できるパラメタ

param-name 指定値	param-value の指定内容	省略値
smartagent.port	スマートエージェントが監視するポート番号を 5001～65535 の範囲の整数で指定します。	14000

4.16 論理ユーザサーバで指定できるパラメタ

ここでは、論理ユーザサーバで指定できるパラメタについて説明します。

「param-name 指定値」に対応する param-value の指定内容については、[\[8.2.19 論理ユーザサーバ定義ファイル\]](#)を参照してください。また、参照する場合は要素名をパラメタに読み替えてください。

論理ユーザサーバで指定できるパラメタについて次の表に示します。「省略値」とは、パラメタの指定がない場合に仮定される値です。

表 4-95 論理ユーザサーバに設定できるパラメタ

param-name 指定値	省略値
forceStop.arg ^{※1}	なし
getProcessID.arg ^{※1}	なし
getProcessID.timeout ^{※2}	180
group-id	なし
isAlive.arg ^{※1}	なし
isAlive.timeout ^{※2}	180
start.arg ^{※1}	なし
start-time-watch-interval	1
stop.arg ^{※1}	なし
type	direct
user-id	なし
watch-interval ^{※3}	運用管理エージェントプロパティファイルの adminagent.userserver.watch.interval キーの 値 (adminagent.userserver.watch.interval キーも省略した場合は 10)
working-dir	<Application Server のインストールディレクトリ>/manager/bin

注※1

[\[8.2.19 論理ユーザサーバ定義ファイル\]](#)の要素名<arg>についての説明を参照してください。

注※2

[\[8.2.19 論理ユーザサーバ定義ファイル\]](#)の要素名<timeout>についての説明を参照してください。

注※3

運用管理エージェントプロパティファイルの adminagent.userserver.watch.interval キーと同時に指定した場合は、このパラメタに指定した値が有効になります。

4.17 論理ネーミングサービスで指定できるパラメタ

ここでは、論理ネーミングサービスで指定できるパラメタについて説明します。

論理ネーミングサービスで指定できるパラメタについて次の表に示します。なお、表中の「省略値」とは、パラメタの指定がない場合に仮定される値です。

表 4-96 論理ネーミングサービスに設定できるパラメタ

param-name 指定値	param-value の指定内容	省略値
ns.agenthost	ホスト固定時に運用管理エージェントのホストで固定するかどうかを指定します。 true : 特定の IP アドレスに割り当てます。 false : 特定の IP アドレスに割り当てません。	false
ns.agent.port	スマートエージェントのポート番号を指定します。	14000
ns.bind.host	特定の IP アドレスに割り当てるかどうかを指定します。次のどちらかを指定します。 true : 特定の IP アドレスに割り当てます。 false : 特定の IP アドレスに割り当てません。	false
port	ネーミングサービスが稼働する時に使用するポート番号を指定します。	900

5

サーバ管理コマンドで使用するファイル

この章では、サーバ管理コマンドで使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

5.1 サーバ管理コマンドで使用するファイルの一覧

サーバ管理コマンドで使用するファイルの一覧を、次の表に示します。

表 5-1 サーバ管理コマンドで使用するファイルの一覧

ファイル名	分類	概要	参照先
usrconf (UNIX 用)	サーバ管理コマンド用オプション定義ファイル	サーバ管理コマンドを実行する JavaVM の起動オプションを指定します。 このファイルは、UNIX 用です。	5.2.1
usrconf.bat (Windows 用)	サーバ管理コマンド用オプション定義ファイル	サーバ管理コマンドを実行する JavaVM の起動オプションを指定します。 このファイルは、Windows 用です。	5.2.2
usrconf.properties	サーバ管理コマンド用システムプロパティファイル	サーバ管理コマンドを実行する JavaVM のシステムプロパティを指定します。	5.2.3

5.2 サーバ管理コマンドで使用するファイルの詳細

5.2.1 usrconf (サーバ管理コマンド用オプション定義ファイル)

(1) 形式

シェルスクリプトファイル形式のファイルです。

次のようにキーを指定します。

```
set <キー名称>="<値>"
```

(2) ファイルの格納先

/opt/Cosminexus/CC/admin/usrconf/

(3) 機能

サーバ管理コマンドを実行する JavaVM の起動オプションを指定します。なお、このファイルは、UNIX 用です。

(4) 指定できるキー

指定できるキーとデフォルト値を次の表に示します。

キー名称	内容	デフォルト
USRCONF_JVM_ARGS	指定されたオプションを使って JavaVM を起動します。 次の JavaVM のオプションを指定できます。 • -Xms<size> • -Xmx<size> • -Xss<size> • -Xdebug • -Xrun<libraryName> • -Dejbserver.log.directory※ • -XX:MetaspaceSize • -XX:MaxMetaspaceSize	デフォルト値については 「5.2.1(6) サーバ管理コマンドでの JavaVM オプションのデフォルト値」を参照してください。
USRCONF_JVM_CLPATH	設定されている値をクラスパスに追加します。複数指定したい場合は、コロン (:) で区切ってください。	なし
USRCONF_JVM_LIBPATH	設定されている値をライブラリ検索パス環境変数に追加します。複数指定したい場合は、コロン (:) で区切ってください。	なし

注※ -Dejbsvr.log.directory を指定しても、次のどれかの操作を行なう場合、サーバ管理コマンドのデフォルトのログ出力先へログが出力されます。

- mngsvrutil コマンドを使用した J2EE アプリケーションのリスト一覧表示
mngsvrutil list appStatus | resStatus | resJavaBeansStatus
- 運用管理ポータル「論理サーバのアプリケーション管理」の操作
- Management Server リモート管理機能
- HCSC サーバの運用操作

(5) USRCONF_JVM_ARGS に指定できる JavaVM のオプション (usrconf の場合)

ここでは、USRCONF_JVM_ARGS に指定できる Java HotSpot VM のオプション、およびオプションを指定する場合の注意事項について説明します。

USRCONF_JVM_ARGS に指定できる Java HotSpot VM のオプションを次の表に示します。なお、「VR」とは、パラメタが導入または変更されたアプリケーションサーバのバージョンです。

オプション名	内容	指定可能値	VR
-Xms	Java ヒープの初期サイズを設定します。	入力は制限されません。	06-00
-Xmx	Java ヒープの最大サイズを設定します。 • コマンドの実行でメモリ不足が発生しコマンドが正常に実行できない場合は、このオプションを指定して、メモリ割り当てプールの最大サイズを拡張することで対応します。 • 操作対象ファイルのファイルサイズが大きいと、サーバ管理コマンドを起動したコンソールに、「java.lang.OutOfMemoryError」メッセージが表示されることがあります。この場合は、このオプションおよび-Xms オプションを使用して、JavaVM のメモリ割り当てサイズを妥当な値に変更してください。 記述例 set USRCONF_JVM_ARGS="-Xms64m -Xmx256m"	入力は制限されません。	06-00
-Xss	1 スタック領域の最大サイズを設定します。	入力は制限されません。	06-00
-Xdebug	デバッガが有効な状態で起動します。	入力は制限されません。	06-00
-Xrun	ライブラリをロードします。	入力は制限されません。	06-00
-Dejbsvr.log.directory	ログファイルの出力先のパスを指定します。ディレクトリのパスの区切り記号には、"/"を使用してください。	入力は制限されません。	06-00

オプション名	内容	指定可能値	VR
-XX:MetaspaceSize	Metaspace 領域の FullGC のしきい値を変更する計算に使用する値であり、しきい値の最低値となる値を指定します。	入力は制限されません。	09-70
-XX:MaxMetaspaceSize	Metaspace 領域の最大サイズを指定します。	入力は制限されません。	09-70

(6) サーバ管理コマンドでの JavaVM オプションのデフォルト値

サーバ管理コマンドでの JavaVM オプションのデフォルト値を示します。

- -Xmx512m

(7) 記述例

```
#!/bin/csh -f

set USRCONF_JVM_ARGS="-Xms64m -Xmx512m"
set USRCONF_JVM_CLPATH=/usr/home/lib/mylib.jar
set USRCONF_JVM_LIBPATH=/usr/home/bin
```

(8) 注意事項

- シェルで設定されている CLASSPATH 環境変数は、サーバ管理コマンドには無効ですが、そのほかの環境変数（PATH など）は有効です。
- 指定する値に次の文字を使用しないでください。
", &, |, <, >
- USRCONF_JVM_ARGS で指定を省略した JavaVM のオプションは、JavaVM のデフォルト値で適用されます。JavaVM のデフォルト値については、「[14.4 Application Server で指定できる Java HotSpot VM のオプションのデフォルト値](#)」を参照してください。ただし、-Dejbsserver.log.directory を指定しない場合は、/opt/Cosminexus/CC/admin/logs が適用されます。
- Management Server を使用する Eclipse プラグインを用いて操作を行う場合は、このファイルの指定は無効です。Management Server を使用する Eclipse プラグインを用いて操作を行う場合は、運用管理エージェント用オプション定義ファイル（adminagentuser.cfg）に指定してください。

5.2.2 usrconf.bat（サーバ管理コマンド用オプション定義ファイル）

(1) 形式

Win32 のバッチファイル形式のファイルです。

次のようにキーを指定します。

```
set <キー名称>=<値>
```

(2) ファイルの格納先

<Application Serverのインストールディレクトリ>%CC¥admin¥usrconf¥

(3) 機能

サーバ管理コマンドを実行する JavaVM の起動オプションを指定します。なお、このファイルは、Windows 用です。

(4) 指定できるキー

指定できるキーとデフォルト値を次の表に示します。

キー名称	内容	デフォルト
USRCONF_JVM_ARGS	指定されたオプションを使って JavaVM を起動します。次の JavaVM のオプションを指定できます。 • -Xms<size> • -Xmx<size> • -Xss<size> • -Xdebug • -Xrun<libraryName> • -Dejbserver.log.directory • -XX:MetaspaceSize • -XX:MaxMetaspaceSize	デフォルト値については「 5.2.2(6) サーバ管理コマンドでの JavaVM オプションのデフォルト値 」を参照してください。
USRCONF_JVM_CLASSPATH	設定されている値をクラスパスに追加します。複数指定したい場合は、セミコロン (;) で区切ってください。	なし
USRCONF_JVM_LIBPATH	設定されている値をライブラリ検索パス環境変数に追加します。複数指定したい場合は、セミコロン (;) で区切ってください。	なし

(5) USRCONF_JVM_ARGS に指定できる JavaVM のオプション (usrconf.bat の場合)

ここでは、USRCONF_JVM_ARGS に指定できる Java HotSpot VM のオプション、およびオプションを指定する場合の注意事項について説明します。

USRCONF_JVM_ARGS に指定できる Java HotSpot VM のオプションを次の表に示します。なお、「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

オプション名	内容	指定可能値	VR
-Xms	Java ヒープの初期サイズを設定します。	入力は制限されません。	06-00
-Xmx	Java ヒープの最大サイズを設定します。 • コマンドの実行でメモリ不足が発生しコマンドが正常に実行できない場合は、このオプションを指定して、メモリ割り当てプールの最大サイズを拡張することに対応します。 • 操作対象ファイルのファイルサイズが大きいと、サーバ管理コマンドを起動したコンソールに、「java.lang.OutOfMemoryError」メッセージが表示されることがあります。この場合は、このオプションおよび-Xms オプションを使用して、JavaVM のメモリ割り当てサイズを妥当な値に変更してください。 記述例 <pre>set USRCONF_JVM_ARGS="-Xms64m -Xmx256m"</pre>	入力は制限されません。	06-00
-Xss	1 スタック領域の最大サイズを設定します。	入力は制限されません。	06-00
-Xdebug	デバッガが有効な状態で起動します。	入力は制限されません。	06-00
-Xrun	ライブラリをロードします。	入力は制限されません。	06-00
-Dejbsserver.log.directory	ログファイルの出力先のパスを指定します。ディレクトリのパスの区切り記号には、「¥」を使用してください。	入力は制限されません。	06-00
-XX:MetaspaceSize	Metaspace 領域の FullGC のしきい値を変更する計算に使用する値であり、しきい値の最低値となる値を指定します。	入力は制限されません。	09-70
-XX:MaxMetaspaceSize	Metaspace 領域の最大サイズを指定します。	入力は制限されません。	09-70

(6) サーバ管理コマンドでの JavaVM オプションのデフォルト値

サーバ管理コマンドでの JavaVM オプションのデフォルト値を示します。

- -Xmx512m

(7) 記述例

```
set USRCONF_JVM_ARGS=-Xms64m -Xmx512m
set USRCONF_JVM_CLASSPATH=C:¥home¥lib¥mylib.jar
set USRCONF_JVM_LIBPATH=C:¥home¥bin
```

(8) 注意事項

- コマンドプロンプトで設定されている CLASSPATH 環境変数は、サーバ管理コマンドには無効ですが、そのほかの環境変数 (PATH など) は有効です。
- 指定する値に次の文字を使用しないでください。
", &, |, <, >
- USRCONF_JVM_ARGS で指定を省略した JavaVM のオプションは、JavaVM のデフォルト値で適用されます。JavaVM のデフォルト値については、「[14.4 Application Server で指定できる Java HotSpot VM のオプションのデフォルト値](#)」を参照してください。ただし、-Dejbsserver.log.directory を指定しない場合は、<Cosminexus インストールディレクトリ>/CC/admin/logs/が適用されます。
- Management Server を使用する Eclipse プラグインを用いて操作を行う場合は、このファイルの指定は無効です。Management Server を使用する Eclipse プラグインを用いて操作を行う場合は、運用管理エージェント用オプション定義ファイル (adminagentuser.cfg) に指定してください。

5.2.3 usrconf.properties (サーバ管理コマンド用システムプロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

次のようにキーを指定します。

`<キー名称> = <値>`

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。
(例) <キー名称> = <値> #<コメント>
- 記載する文字は Java の仕様に従って、ISO 8859-1 文字エンコーディングを使用してください。

(2) ファイルの格納先

- Windows の場合
<Application Server のインストールディレクトリ>¥CC¥admin¥usrconf¥

- UNIX の場合
/opt/Cosminexus/CC/admin/usrconf/

(3) 機能

サーバ管理コマンドを実行する JavaVM のシステムプロパティを指定します。

(4) サーバ管理コマンドの予約済みキー

サーバ管理コマンドでは、次に示す接頭子で始まるキーを内部的に利用します。このため、アプリケーションでこれらの接頭子で始まるキーを利用してはいけません。

- ejbserver.*
- webserver.*
- appclient.*
- j2eeserver.*

(5) サーバ管理コマンドのカスタマイズ用キー

サーバ管理コマンドでは、次に示すシステムプロパティのキーに値を設定することで、J2EE サーバの動作をカスタマイズできます。

ここでは、キーを次の分類に分けて説明します。

- [ejbserver.cui](#) から始まるキー
- [ejbserver.deploy](#) から始まるキー
- [ejbserver.logger](#) から始まるキー
- [ejbserver.naming](#) から始まるキー
- [ejbserver.rmi](#) から始まるキー
- [vbroker](#) から始まるキー

(a) [ejbserver.cui](#) から始まるキー

キー名称	内容	指定可能値	デフォルト	VR
ejbserver.cui.checkmethod.compatible	cjsetappprop または cjsetresprop コマンド実行時に引数で指定された属性ファイル中のメソッドのチェック方法を指定します。チェックの引数になるメソッドは、属性ファイルの次のタグで指定されたメソッドです。 • <method-permission> • <container-transaction>	次のどちらかを指定します。 • true • false	false	07-50

キー名称	内容	指定可能値	デフォルト	VR
	• <exclude-list> • <ejb-method-observation-timeout> • <ejb-transaction-timeout>※			
ejbserver.cui.exit code.compatible	サーバ管理コマンド実行時に排他エラーで実行できなくなった場合に、終了コード「1」を返却するかどうかを指定します。 true を指定した場合： 排他エラーの場合に終了コード「1」を返却します。 false を指定した場合： 排他エラーの場合に終了コード「2」を返却します。 タイムアウトエラーの場合に終了コード「3」を返却します。	次のどちらかを指定します。 • true • false	false	06-00
ejbserver.cui.logfile.compatible	07-00 以降のログ出力形態にするか 06-70 以前のログ出力形態にするかを指定します。指定できる値と動作を次に示します。 false を指定した場合： 07-00 以降のログ出力形態にします。 true を指定した場合： 06-70 以前のログ出力形態にします。 不正な値（すべて小文字の true または false 以外）を指定した場合は、メッセージを出力し、デフォルト値を設定します。 Management Server を使用する Eclipse プラグインから操作した場合、この指定は無効となり、false が仮定されます。 なお、サーバ管理コマンドのログ出力ディレクトリに必要なディスク容量は、次の式で算出できます。 false を指定した場合： 18,624KB + TPBroker のトレース情報 true を指定した場合： 51,655KB + TPBroker のトレース情報	次のどちらかを指定します。 • true • false	false	07-00

キー名称	内容	指定可能値	デフォルト	VR
	TPBroker のトレース情報を出力するために必要なディスク容量については、マニュアル「TPBroker 運用ガイド」のディスク占有量に関する説明を参照してください。			
ejbserver.cui.optionalname.enabled	JNDI のユーザ指定名前空間管理機能を使用する場合に、サーバ管理コマンドから EJBHome オブジェクトに別名 (Optional Name) を指定するときに指定します。 true を指定した場合： サーバ管理コマンドから別名を指定できます。 false を指定した場合： サーバ管理コマンドから別名を指定できません。	次のどちらかを指定します。 • true • false	true	05-05

注※ このキーの設定値によるチェック内容は次のとおりです。

表 5-2 キーの設定値によるチェック内容一覧

属性ファイル中のメソッド指定箇所	プロパティ指定値	
	ture	false
<method-permission>	○	○
<container-transaction>	×	○
<exclude-list>	○	○
<ejb-method-observation-timeout>	×	○
<ejb-transaction-timeout>	×	○

(凡例)

○：指定されたメソッドが Enterprise Bean 中にある場合エラーとする。

×

(b) ejbserver.deploy から始まるキー

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	内容	指定可能値	デフォルト	VR	関連情報
ejbserver.deploy.annotations.load_check.enabled	アノテーション情報取得のためのクラスロード時に例外が発生した場合に	指定できる文字列を次に示します。 • true • false	false	08-00	「機能解説 基本・開発編(コンテナ共通機能)」の「17.3 ロー

キー名称	内容	指定可能値	デフォルト	VR	関連情報
	無視するかどうかを指定します。 true を指定した場合： エラーとなり処理を中断します。 false を指定した場合： 例外ログを取得し処理を続行します。				ド対象のクラスとロード時に必要なクラスパス」

(c) ejbserver.logger から始まるキー

キー名称	内容	指定可能値	デフォルト	VR
ejbserver.logger.enabled.*	サーバ管理コマンドが出力するログレベルを指定します。一つだけ指定した場合は、該当するログレベルのログだけが出力されます。複数指定した場合は、レベル名の文字列の間をコンマ (,) で区切ります。 Error, Warning, Information, Debug のうち、一つまたは複数を指定します。通常はデフォルトのまま利用してください。	次に示す文字列をコンマ (,) で区切って指定します。 - Error - Warning - Information - Debug	Error, Warning	06-00

(d) ejbserver.naming から始まるキー

キー名称	内容	指定可能値	デフォルト	VR
ejbserver.naming.host	サーバ管理コマンド実行時に、ネーミングサービスとして利用する CORBA ネーミングサービスが起動しているホスト名称または IP アドレスを指定します。 ネーミングの切り替え機能を使用する場合、ホスト名称として"localhost"は使用しないでください。CORBA ネーミングサービスが起動しているホスト名称または IP アドレスを指定してください。 CORBA ネーミングサービスを自動起動モード (ejbserver.naming.startupMode=automatic または inprocess) で使用する場合、デフォルト値 ("localhost"), または J2EE サーバを起動させるホスト名称もしくは IP アドレスを指定してください。	次のどれかを指定します。 - IPv4 アドレス - ホスト名 (半角英数字または記号 (._-) で指定された 255 文字以内の文字列) @myhost	localhost	06-00
ejbserver.naming.port	サーバ管理コマンド実行時に、J2EE サーバがネーミングサービスとして利用する CORBA ネーミングサービスのポート番号を指定します。	1~65535 の整数で指定します。	900	06-00

キー名称	内容	指定可能値	デフォルト	VR
ejbserver.naming.protocol	サーバ管理コマンド実行時に、ネーミングサービスとして利用する CORBA ネーミングサービスへのアクセスプロトコルを指定します。現状、corbaname だけをサポートしています。ただし、旧バージョンで使用されていたプロトコル (iioploc または iopname) でも動作できます。	入力は制限されません。	corbaname	06-00

(e) ejbserver.rmi から始まるキー

キー名称	内容	指定可能値	デフォルト	VR
ejbserver.rmi.request.timeout	サーバ管理コマンドが通信障害によって応答が返らなくなった場合の通信タイムアウト時間（単位：秒）を 0～86400 の整数で指定します。 0 を指定した場合はタイムアウトしません。86400 を超える値を設定した場合、警告メッセージが出力されて、デフォルト値が適用されます。 J2EE サーバ用ユーザプロパティの <code>ejbserver.deploy.exclusive.lockAliveInterval</code> キーの値よりも小さい値を指定した場合、サーバ管理コマンドでタイムアウトが発生しやすくなりますので、J2EE サーバ用ユーザプロパティの <code>ejbserver.deploy.exclusive.lockAliveInterval</code> キーの値よりも大きい値を設定することを推奨します。 なお、次のコマンドでは、アプリケーションの停止処理が <code>-t</code> オプションで指定した時間内に終了しなかった場合に、強制終了処理が終了するまでのタイムアウト時間となります。 • <code>cjstopapp</code> • <code>cjreplaceapp</code> • <code>cjreloadapp</code>	0～86400 の整数で指定します。	180（秒）	06-00

(f) vbroker から始まるキー

キー名称	内容	指定可能値	デフォルト	VR
vbroker.orb.htc.tracePath	TPBroker のトレースファイル出力先のパスを 1～210 バイトの範囲で指定します。指定したパスのサブディレクトリとして <code>comtrc</code> と <code>mdltrc</code> をあらかじめ作成しておく必要があります。デフォルトの出力先の場合、サブディレクトリ <code>comtrc</code> と <code>mdltrc</code>	入力は制限されません。	<Application Server のインストールディレクトリ >%CC%admin%logs%TPB%logj	06-00

キー名称	内容	指定可能値	デフォルト	VR
	はサーバ初回起動時に自動作成されます。ディレクトリのパスの区切り記号には、"/"を使用してください。 例えば、work ディレクトリとして C:*temp*work を設定する場合、次のように指定します。 (指定例) vbroker.orb.htc.tracePath=c:/temp/work		または /opt/Cosminexus/CC/admin/logs/TPB/logj	
vbroker.se.iiop_tp.scm.iiop_tp.listener.port	任意の値を指定して、J2EE サーバからの受信を受け付けるポートを固定できます。ポート番号はほかのプログラムなどと重複しないようにしてください。また、J2EE サーバ用ユーザプロパティファイルにも同じキー名称が存在しますので、そのポート番号と重複しないようにしてください。 このプロパティを設定しない場合は、TPBroker によってランダムな値が設定されます。	入力制限されません。	0	06-50

(6) 記述例

```
ejbserver.naming.host=localhost
ejbserver.naming.port=900
```

6

CJMS プロバイダで使用するファイル

この章では、CJMS プロバイダで使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

6.1 CJMS プロバイダで使用するファイルの一覧

CJMS プロバイダで使用するファイルの一覧を、次の表に示します。

表 6-1 CJMS プロバイダで使用するファイルの一覧

ファイル名	分類	概要	参照先
admin.properties	管理コマンドプロパティファイル	管理コマンドのログ出力について設定します。	6.2.1
commonconfig.properties	CJMSP ブローカー共通プロパティファイル	すべての CJMSP ブローカーのシステムプロパティを一括設定します。	6.2.2
config.properties	CJMSP ブローカー個別プロパティファイル	CJMSP ブローカーのシステムプロパティを個別に設定します。	6.2.3

6.2 CJMS プロバイダで使用するファイルの詳細

6.2.1 admin.properties (管理コマンドプロパティファイル)

(1) 形式

次のようにキーを指定します。

`<キー名称> = <値>`

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 指定範囲外の値を指定した場合は範囲外の値指定時の値が適用されます。値を指定しなかった場合は値未指定時の値が適用されます。これらの場合、コンソール上に警告メッセージが表示されます。

(2) ファイルの格納先

- Windows の場合
 <Application Server のインストールディレクトリ>%CC%cjmsp%var%admin%config%
- UNIX の場合
 /opt/Cosminexus/CC/cjmsp/var/admin/config/

なお、以降、<Application Server のインストールディレクトリ>%CC%cjmsp (Windows の場合) または /opt/Cosminexus/CC/cjmsp (UNIX の場合) のことを、<CJMSP_HOME>と表記します。

(3) 機能

管理コマンドのログ出力について設定します。

なお、プロパティの値はログの初期化よりも前に検証されます。このファイルにプロパティとして不正な値を指定した場合、ログの内容はコンソールだけに出力され、ログファイルには出力されません。

また、次に示すプロパティを変更する場合、変更した内容は<CJMSP_HOME>%var%admin%log ディレクトリ下 (Windows の場合) または<CJMSP_HOME>/var/admin/log ディレクトリ下 (UNIX の場合) のすべてのログファイルを削除したときに有効になります。

- admin.logger.ExceptionLogFile.filename
- admin.logger.ExceptionLogFile.filesize

- admin.logger.MessageLogFile.fileenum
- admin.logger.MessageLogFile.filesize

(4) 設定できるキー

設定できるキーについて次に示します。なお「値未指定時」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
admin.logger.ExceptionLogFile.fileenum	作成するログファイル数の最大値を指定します。	1～16	2	2	08-50
admin.logger.ExceptionLogFile.filepath	ログファイルの格納先を絶対パスで指定します。 ファイルは自動的にcjmsadmin_errN.logという名前で作成されます。Nはログファイルの面数です。 指定するディレクトリが存在しない場合は、CJMS プロバイダが警告メッセージを表示し、デフォルトの内容が適用されます。また、デフォルトのディレクトリが存在しない場合、CJMS プロバイダがデフォルトディレクトリを作成し、使用できるようになります。 読み取り権限と書き込み権限があるパスを指定してください。読み取り権限だけしかないパスを指定した場合、デフォルトの内容が適用されます。また、デフォルトのパスに読み取り権限しかない場合は、操作が失敗します。 ディレクトリ名として指定できる文字およびパスの長さについては、ご使用の OS の制限に従ってください。 なお、パスに「¥」を指定する場合は、「¥¥」と指定して、エスケープしてください。	<パス名>	• Windows の場合 <CJMS P_HOME>¥var¥admin¥log • UNIX の場合 <CJMS P_HOME>/var/admin/log	• Windows の場合 <CJMS P_HOME>¥var¥admin¥log • UNIX の場合 <CJMS P_HOME>/var/admin/log	08-50
admin.logger.ExceptionLogFile.filesize	ログファイルサイズの最大値をバイトで指定します。	8192～4194304	1048576	1048576	08-50
admin.logger.MessageLogFile.fileenum	作成するログファイル数の最大値を指定します。	1～16	2	2	08-50

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
admin.logger.MessageLogFile.filepath	ログファイルの格納先を絶対パスで指定します。 ファイルは自動的にcjmsadmin_msgN.log という名前で作成されます。N はログファイルの面数です。 指定するディレクトリが存在しない場合は、CJMS プロバイダが警告メッセージを表示し、デフォルトの内容が適用されます。また、デフォルトのディレクトリが存在しない場合、CJMS プロバイダがデフォルトディレクトリを作成し、使用できるようになります。 読み取り権限と書き込み権限があるパスを指定してください。読み取り権限だけしかないパスを指定した場合、デフォルトの内容が適用されます。また、デフォルトのパスに読み取り権限しかない場合は、操作が失敗します。 ディレクトリ名として指定できる文字およびパスの長さについては、ご使用の OS の制限に従ってください。 なお、パスに「¥」を指定する場合は、「¥¥」と指定して、エスケープしてください。	<パス名>	- Windows の場合 <CJMS P_HOME>¥var¥ admin¥l og - UNIX の場合 <CJMS P_HOME>/var/ admin/l og	- Windows の場合 <CJMS P_HOME>¥var¥ admin¥l og - UNIX の場合 <CJMS P_HOME>/var/ admin/l og	08-50
admin.logger.MessageLogFile.filesize	ログファイルサイズの最大値を指定します。	8192～4194304	1048576	1048576	08-50
admin.logger.MessageLogFile.trace.level	ログのレベルを指定します。 指定したレベルに基づいて情報がログファイルに記録されます。	指定できる文字列を次に示します。 - ERROR - WARNING - INFO	ERROR	ERROR	08-50

6.2.2 commonconfig.properties (CJMSP ブローカー共通プロパティファイル)

(1) 形式

次のようにキーを指定します。

`<キー名称> = <値>`

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 指定範囲外の値を指定した場合は範囲外の値指定時の値が適用されます。値を指定しなかった場合は値未指定時の値が適用されます。これらの場合、コンソール上に警告メッセージが表示されます。

(2) ファイルの格納先

- Windows の場合
 <Application Server のインストールディレクトリ>%CC%cjmsp%lib%props%broker%
- UNIX の場合
 /opt/Cosminexus/CC/cjmsp/lib/props/broker/

(3) 機能

すべての CJMSP ブローカーのプロパティを一括設定するプロパティファイルについて説明します。

CJMSP ブローカーの起動中にこのファイルの内容を変更した場合、変更した内容は CJMSP ブローカーを再起動したときに反映され、有効になります。

(4) 設定できるキー

設定できるキーを次に示します。なお、詳細は、「[6.2.3 config.properties \(CJMSP ブローカー個別プロパティファイル\)](#)」を参照してください。

(a) コネクションサービス

コネクションサービスについて設定できるキーを次に示します。詳細は、「[6.2.3\(4\)\(a\) コネクションサービス](#)」を参照してください。

- imq.hostname キー

(b) 永続化サービス

永続化サービスについて設定できるキーを次に示します。詳細は、「[6.2.3\(4\)\(b\) 永続化サービス](#)」を参照してください。

- imq.persist.file.sync.enabled キー

(c) 送信先自動生成の設定

送信先自動生成について設定できるキーを次に示します。詳細は、「[6.2.3\(4\)\(c\) 送信先自動生成の設定](#)」を参照してください。

- imq.autocreate.destination.maxNumMsgs キー
- imq.autocreate.destination.maxTotalMsgBytes キー
- imq.autocreate.queue.consumerFlowLimit キー
- imq.autocreate.topic.consumerFlowLimit キー

(d) モニタリングプロパティの設定

モニタリングプロパティについて設定できるキーを次に示します。詳細は、「[6.2.3\(4\)\(d\) モニタリングプロパティの設定](#)」を参照してください。

- imq.metrics.interval キー

(e) CJMSP ブローカーログプロパティ

CJMSP ブローカーログプロパティについて設定できるキーを次に示します。詳細は、「[6.2.3\(4\)\(e\) CJMSP ブローカーログプロパティ](#)」を参照してください。

- broker.logger.ExceptionLogFile.filenum キー
- broker.logger.ExceptionLogFile.filesize キー
- broker.logger.MessageLogFile.filenum キー
- broker.logger.MessageLogFile.filesize キー
- broker.logger.MessageLogFile.trace.level キー

6.2.3 config.properties (CJMSP ブローカー個別プロパティファイル)

(1) 形式

次のようにキーを指定します。

<キー名称> = <値>

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 指定範囲外の値を指定した場合は範囲外の値指定時の値が適用されます。値を指定しなかった場合は値未指定時の値が適用されます。これらの場合、コンソール上に警告メッセージが表示されます。

(2) ファイルの格納先

デフォルトの格納先を次に示します。なお、var ディレクトリの格納先は、cjmsbroker コマンドの-varhome オプションで変更できます。

- Windows の場合
 <Application Server のインストールディレクトリ>%CC%cjmsp*var*instances*< CJMSP ブローカーのインスタンス名称>*props*
- UNIX の場合
 /opt/Cosminexus/CC/cjmsp/var/instances/< CJMSP ブローカーのインスタンス名称>/props/

以降、<Application Server のインストールディレクトリ>%CC%cjmsp (Windows の場合) または/opt/Cosminexus/CC/cjmsp (UNIX の場合) のことを、<CJMSP_HOME>と表記します。

(3) 機能

個別の CJMSP ブローカーを実行するシステムプロパティを指定します。

config.properties と commonconfig.properties とで異なる値が指定されている場合、config.properties の値が優先されます。

CJMSP ブローカーの起動中にこのファイルの内容を変更した場合、変更した内容は CJMSP ブローカーを再起動したときに反映され、有効になります。

(4) 設定できるキー

設定できるキーについて次に示します。なお「値未指定時」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

(a) コネクションサービス

コネクションサービスに関するプロパティについて次に示します。

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
imq.admin.tcp.port	admin サービス用のポート番号を指定します。 ファイアウォールを使用して CJMSP ブローカーに接続する場合は、このプロパティを使用して静的にポート番号を指定する必要があります。 次の場合、ポートマッパーによってポート番号が動的に割り当てられます。 • 0 を指定した場合 • ユーザが指定可能値範囲外の値（65535 以上）を指定した場合 -1 および 1～1024 の値を指定した場合、動作は保証されません。	0, 1025～65535	0	0	08-50
imq.hostname	すべてのコネクションサービス用デフォルトホスト名または IP アドレスを指定します。 複数のサーバに CJMS プロバイダをインストールする場合、このプロパティを指定することをお勧めします。 ホスト名に無効な文字列を指定した場合、CJMSP ブローカーを開始できません。また、localhost またはループバックアドレス（127.*.*) を指定した場合、CJMSP ブローカーへリモート接続はできません。 なお、このプロパティに値を指定しない場合 ([imq.hostname=] までを指定した場合)、CJMSP ブローカー起動時には、ローカルホストの CJMSP ブローカーに接続します。このとき、ホスト名は出力されません。例えば、このプロパティに値を指定しないで imq.portmapper.port に 7676 を指定した場合、CJMSP ブローカーを起動時には「:7676」と出力されます。	—	—	<ホスト名>	08-50
imq.jms.tcp.port	jms サービス用のポート番号を指定します。	0, 1025～65535	0	0	08-50

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
	ファイアウォールを使用して CJMSP ブローカーに接続する場合は、このプロパティを使用して静的にポート番号を指定する必要があります。 次の場合、ポートマッパーによってポート番号が動的に割り当てられます。 • 0 を指定した場合 • ユーザが指定可能値範囲外の値（65535 以上）を指定した場合 -1 および 1～1024 の値を指定した場合、動作は保証されません。 すでに使用されているポート番号を指定した場合、ブローカーの起動には成功しますが、jms サービスは使用できません。この状態で cjmsicmd list svc コマンドを実行すると、jms サービスの状態は「UNKNOWN」と表示されます。 「UNKNOWN」はサービスが起動していない、または使用できない状態を示します。				
imq.portmapper.port	CJMSP クライアントが CJMSP ブローカーに接続するためのポートマッパーのポート番号を指定します。 複数の CJMSP ブローカーインスタンスが同じホスト上で起動している場合、それぞれ異なるポートマッパーのポート番号を指定する必要があります。	1025～65535	7676	7676	08-50

(凡例)

－：該当なし。

(b) 永続化サービス

永続化サービスに関するプロパティについて次に示します。

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
imq.persist.file.sync.enabled	メモリ上のデータのファイルへの書き込み処理を同期と非同期のどちらで実行するかを指定します。 true を指定した場合、書き込み処理は同期で実行されます。この場合、システム障害が発生したときにデータが紛失することを防げますが、処理性能に影響があります。false を指定した場合、書き込み処理は非同期で実行されます。 どちらを指定するかは、性能テストを実行した上で決定することをお勧めします。 また、アプリケーションで配信モードとして「javax.jms.DeliveryMode.PERSISTENT」と指定している場合、このプロパティは true にすることをお勧めします。	指定できる値を次に示します。 • true • false	false	false	08-50

(c) 送信先自動生成の設定

送信先自動生成の設定に関するプロパティについて次に示します。

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
imq.autocreate.destination.maxNumMsgs	未受信のメッセージ数の最大値を指定します。 -1 または 0 を指定した場合、メッセージ数は無制限になります。	-1 ~ 2147483647	100000	100000	08-50
imq.autocreate.destination.maxTotalMsgBytes	未受信メッセージ用の総合メモリサイズの最大値をバイトで指定します。単位を指定する場合は、値の末尾に次の文字列を追加してください。単位を指定しない場合は、バイトとして扱われます。 • バイト：b • キロバイト：k • メガバイト：m	-1 ~ 2147483647 (単位: バイト)	10m	10m	08-50

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
	-1 または 0 を指定した場合、 メッセージ数は無制限になります。				
imq.autocreate.queue.consumerFlowLimit	同一バッチでキューコンシューマーに配信されるメッセージ数の最大値を指定します。 -1 を指定した場合、メッセージ数は無制限になります。	指定できる 値を次に示 します。 • -1 • 1～ 214748 3647 の 整数	100	100	08-50
imq.autocreate.topic.consumerFlowLimit	同一バッチでトピックコンシューマーに配信されるメッセージ数の最大値を指定します。 -1 を指定した場合、メッセージ数は無制限になります。	指定できる 値を次に示 します。 • -1 • 1～ 214748 3647 の 整数	1000	1000	08-50

(d) モニタリングプロパティの設定

モニタリングの設定に関するプロパティについて次に示します。

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
imq.metrics.interval	メトリクス情報をログおよび コンソールに書き込む時間を秒 単位で指定します。 0 を指定した場合、メトリクス を出力しません。	0～ 214748364 7	0	0	08-50

(e) C JMSP ブローカーログプロパティ

C JMSP ブローカーログに関するプロパティについて次に示します。

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
broker.logger.ExceptionLog File.filenum	作成するログファイル数の最大 値を指定します。	1～16	2	2	08-50
broker.logger.ExceptionLog File.filesize	ログファイルサイズの最大値を 指定します。	8192～ 4194304	1048576	1048576	08-50

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
	最大値を超える場合、新しい ファイルが作成され、ログファ イルの内容が新しいファイルに 出力されます。				
broker.logger.MessageLogF ile.fileenum	作成するログファイル数の最大 値を指定します。	1～16	2	2	08-50
broker.logger.MessageLogF ile.filesize	ログファイルサイズの最大値を 指定します。 最大値を超える場合、ログファ イルの内容が新しいファイルに 出力されます。	8192～ 4194304	1048576	1048576	08-50
broker.logger.MessageLogF ile.trace.level	ログのレベルを指定します。 指定したレベルに基づいて情報 がログファイルに出力されます。	指定できる 文字列を次 に示します。 • ERROR • WARNI NG • INFO	ERROR	ERROR	08-50

(f) そのほかのプロパティ

そのほかに関するプロパティについて次に示します。

キー名称	内容	指定可能値	範囲外の値 指定時	値未指定時	VR
imq.instanceconfig.version	内部処理で使用するプロパティ です。設定されている内容を変 更しないでください。	—	—	—	08-50

(凡例)

—：該当なし。

7

CTM で使用するファイル

この章では、CTM で使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

7.1 CTM で使用するファイルの一覧

CTM で使用するファイルの一覧を、次の表に示します。

表 7-1 CTM で使用するファイルの一覧

ファイル名	分類	概要	参照先
(任意) ※	CTM ユーザ環境変数定義ファイル	ctmstart コマンド (CTM デーモンの開始) を実行するときに必要な環境変数を指定します。	7.2.1
(任意) ※	CTM コマンドオプションファイル	ctmstart コマンド (CTM デーモンの開始) を実行するときに、CTM レギュレータを自動開始するために指定する引数 (コマンドオプション) をあらかじめファイルに指定します。	7.2.2

注※

使用できる文字は、Windows の場合、英数字、ピリオド (.), 円符号 (¥), コロン (:), プラス (+), またはハイフン (-) です。UNIX の場合、英数字、ピリオド (.), スラッシュ (/), プラス (+), またはハイフン (-) です。

なお、CTM を使用してリクエストのスケジューリングや負荷分散をするためには、[表 7-1](#) に示したファイルのほかに、J2EE サーバ用 `usrconf.properties` ファイルで、CTM の使用の有無、CTM ドメイン名、CTM デーモンの CTM 識別子、CTM が使用するホスト名などを指定する必要があります。J2EE サーバ用 `usrconf.properties` ファイルの CTM のキーについては、「[2.2.3\(5\)\(i\) ejbserver.ctm から始まるキー](#)」を参照してください。

7.2 CTM で使用するファイルの詳細

7.2.1 CTM ユーザ環境変数定義ファイル

(1) 形式

次の形式で指定します。

<環境変数名称>=<値>

- 環境変数名称、イコール (=), および値のどれも省略できません。
- 行の先頭から最初のイコール (=) の直前までが、環境変数名称とみなされます。
- 最初のイコール (=) の直後からその行またはファイルの終端の直前までが値とみなされます。
- シャープ (#) で始まる行はコメントとみなされます。行の途中にシャープ (#) がある場合は、シャープ (#) 以降から行の最後までがコメントとみなされます。シャープ (#) の手前の空白およびタブは無視されます。
- シャープ (#) は値には指定できません。
- 一つの定義を複数行に分けて記述する場合は、改行の直前に継続符号 (¥) を記述します。継続符号 (¥) の直前の空白およびタブは無視されます。行の途中に継続符号 (¥) がある場合は、継続記号 (¥) 以降から行の最後までがコメントとみなされます。このため、継続記号 (¥) は行の最後に指定してください。
- 指定された環境変数と同一の環境変数名称がすでに存在する場合は、指定した値で上書きされます。同一の環境変数名称がない場合は追加されます。

(2) ファイルの格納先

格納先およびファイル名称は任意です。ファイル名には次の文字が使用できます。

- Windows の場合
英数字, ピリオド (.), 円符号 (¥), コロン (:), プラス (+), またはハイフン (-)
- UNIX の場合
英数字, ピリオド (.), スラッシュ (/), プラス (+), またはハイフン (-)

(3) 機能

ctmstart コマンド (CTM デーモンの開始) を実行するときに必要な環境変数を指定します。

(4) 指定できるキー

任意の環境変数名称を指定できます。ただし、次の環境変数名称は指定できません。

- 先頭が"CTM"または"ctm"で始まる環境変数名称

(5) 記述例

- Windows の場合

```
# コメント行
LANG=ja_JP.SJIS                                # 以降コメント
C:¥Program Files¥HITACHI¥Cosminexus¥CTM¥Lib¥ctmj2sv.jar:    ¥
C:¥Program Files¥HITACHI¥Cosminexus¥java¥classes.zip        # 継続行
```

- UNIX の場合

```
# コメント行
LANG=ja_JP.SJIS                                # 以降コメント
CLPATH=/opt/Cosminexus/CTM/lib/ctmj2sv.jar:    ¥
/opt/java/lib/classes.zip                        # 継続行
```

(6) 注意事項

- 先頭が"CTM"または"ctm"で始まる環境変数名称は使用できません。
- 環境変数名称には、英数字またはアンダースコア (_) で構成される文字列が指定できます。
- 行の先頭から環境変数名称までは空白およびタブは記述できません。行の先頭から環境変数名称を記述する必要があります。
- 環境変数名称とイコール (=) の間、およびイコール (=) と環境変数定義の間に空白およびタブを記述できません。
- 環境変数定義ファイル中に、%AAA% (Windows の場合)、または\$AAA (UNIX の場合) の記述がある場合、これらはそのままの文字列として認識されます。環境変数 AAA としては展開されません。
- 空行 (環境変数定義もコメントもない行) を記述する場合は、空白およびタブを含めないで改行コードだけを指定してください。
- 次に示す場合は定義内容が不正とみなされます。
 - 環境変数定義内にイコール (=) がない場合
 - 行の先頭または最後にイコール (=) がある場合
 - 環境変数名称の先頭が"CTM"または"ctm"で始まる場合
 - 空白またはタブだけの行がある場合
- Windows の場合、継続符号 (¥) の扱いについては、次の点に注意して環境変数を設定してください。
 - パス名の記述に使用する「¥」と継続符号 (¥) が混同しない
 - 必要なスペースが無効にならない

設定例を次に示します。なお、設定例の<改行>は改行コードを指します。

(例 1) 環境変数 ROOT_PATH に c:¥を設定する場合

項目	誤った例	正しい例
CTM ユーザ環境変数定義 ファイルの記述	ROOT_PATH=c:¥<改行> ROOT_PORT=18000<改行>	ROOT_PATH=c:¥#<改行> ROOT_PORT=18000<改行>
設定される環境変数	ROOT_PATH=c:ROOT_PORT=18000	ROOT_PATH=c:¥ ROOT_PORT=18000

誤った例では、c:¥の¥が継続符号と解釈されています。この場合、¥の直後にコメント符号（#）を入れることで、¥が継続の意味を持たなくなります。

(例 2) 環境変数 CTMDIR に C:¥Program Files¥Cosminexus¥CTM を設定する場合

項目	誤った例	正しい例
CTM ユーザ環境変数定義 ファイルの記述	CTMDIR=C:¥Program ¥<改行> Files¥Cosminexus¥CTM<改行>	CTMDIR=C:¥Program Files¥<改行> ¥Cosminexus¥CTM<改行>
設定される環境変数	CTMDIR=C:¥ProgramFiles¥Cosminexus¥CTM	CTMDIR=C:¥Program Files¥Cosminexus¥CTM

誤った例では、「Program Files」に含まれるスペースが継続符号の直前にあるため、無効になっています。継続符号の直前がスペースにならないように記述してください。

7.2.2 CTM コマンドオプションファイル

(1) 形式

次の形式で指定します。

<引数> <値>

(2) ファイルの格納先

格納先およびファイル名称は任意です。ファイル名には次の文字が使用できます。

- Windows の場合
英数字, ピリオド (.), 円符号 (¥), コロン (:), プラス (+), またはハイフン (-)
- UNIX の場合
英数字, ピリオド (.), スラッシュ (/), プラス (+), またはハイフン (-)

(3) 機能

ctmstart コマンド (CTM デーモンの開始) を実行するときに、CTM レギュレータを自動開始するために指定する引数 (コマンドオプション) をあらかじめファイルに指定しておきます。ctmstart コマンド

の-CTMRegOption オプションにファイル名を指定すると、このファイルに指定した内容を基に CTM レギュレータが自動開始されます。

なお、-CTMRegStart オプションが指定されていない場合、-CTMRegOption オプションの指定およびこのファイルの指定内容は無視されます。

(4) 指定できるキー

ctmregltd コマンドの引数を指定できます。ctmregltd コマンドについては、マニュアル「アプリケーションサーバ リファレンス コマンド編」の「ctmregltd (CTM レギュレータの開始)」を参照してください。なお、このファイルの指定を省略した場合、CTM レギュレータは、ctmstart コマンドに指定した引数のうち、-CTMPort と-CTMEjbPort 以外の引数をデフォルト値として引き継ぎます。

(5) 記述例

```
-CTMClientConnectCount 64
```

(6) 注意事項

- コマンドオプションファイルには英数字、空白、ハイフン (-)、ピリオド (.), またはスラッシュ (/) を記述できます。
- コマンドオプション以外は記述しないでください。

8

Manager で使用するファイル

この章では、Manager で使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

8.1 Manager で使用するファイルの一覧

Manager で使用するファイルの一覧を、次の表に示します。

表 8-1 Manager で使用するファイルの一覧

ファイル名	分類	概要	参照先
adminagent.properties	運用管理エージェントプロパティファイル	運用管理エージェント本体に関する設定をします。	8.2.1
AdminAgentrc	運用管理エージェント自動起動用設定ファイル	運用管理エージェントを自動起動する場合の設定をします。 対象 OS は Windows と Linux だけです。	8.2.2
adminagentuser.cfg	運用管理エージェント用オプション定義ファイル	運用管理エージェントを実行する JavaVM の起動オプションを指定します。	8.2.3
adminagent.xml	運用管理エージェント設定ファイル	システム管理者の代わりに、論理サーバの起動、または設定ファイルの更新をします。	8.2.4
mngagent.<実サーバ名>.properties	運用監視エージェントプロパティファイル	運用監視エージェントの動作を定義します。	8.2.5
mserver.properties	Management Server 環境設定ファイル	Management Server の使用するポート番号を設定します。	8.2.6
mserver.cfg	Management Server 用オプション定義ファイル	Management Server 全体でのシステムプロパティを指定します。	8.2.7
mserverenv.cfg	Management Server 用環境変数定義ファイル	Management Server を操作するときの環境変数を設定します。	8.2.8
manager.cfg	Manager 設定ファイル	Manager 共通の設定をします。	8.2.9
maction.properties	Management アクション実行用プロパティファイル	Management イベントで通知されるメッセージ ID に対応する Management アクションを定義します。	8.2.10
任意	Management イベント発行用プロパティファイル	Management イベント発行時の動作を定義します。	8.2.11
任意	Management イベント発行用メッセージ ID リストファイル	Management イベントとして通知するメッセージ ID を定義します。	8.2.12
任意	Management Server 管理ファイル用退避対象定義ファイル	mstrexport コマンドで退避する対象ファイルを定義します。	8.2.13

ファイル名	分類	概要	参照先
.mnngsvrutilrc	mnngsvrutil コマンドのクライアント側定義ファイル	mnngsvrutil コマンドのオプションのデフォルト値を設定できます。	8.2.14
mnngsvrutil.properties	mnngsvrutil コマンドのサーバ側定義ファイル	mnngsvrutil コマンドの実行環境を設定します。	8.2.15
mnngsvrutilcl.properties	mnngsvrutil コマンドのクライアント側共通定義ファイル	mnngsvrutil コマンドのオプションのデフォルト値を共通定義として設定できます。	8.2.16
.mnngsvrmonitorrc	JP1/IM 連携用モニタ起動コマンドの設定ファイル	JP1/IM 連携をする場合に、JP1/IM の画面から Management Server の運用ポータルを起動するために必要な設定をします。 このファイルは、Windows 用です。	8.2.17
setup.cfg	セットアップウィザード用設定ファイル	セットアップウィザードに関する設定をします。	8.2.18
任意	mnngsvrutil コマンドの論理ユーザ定義サーバファイル	追加する論理ユーザサーバの内容を定義します。	8.2.19
• mserver.jp1event.system.mapping.properties • manager.jp1event.system.mapping.properties • manager.<論理サーバ名>.jp1event.system.mapping.properties	JP1/IM 連携用システムログメッセージマッピングファイル	アプリケーションサーバシステムで出力されるメッセージと JP1 イベントの重大度のマッピングを定義します。	8.3

8.2 Manager で使用するファイルの詳細

Windows 環境で JDK21 以降の場合、Manager で使用するファイルにマルチバイト文字は使用できません。

8.2.1 adminagent.properties（運用管理エージェントプロパティファイル）

(1) 形式

J2SE のプロパティファイル形式です。

運用管理エージェントのプロパティファイルです。

Java の仕様に従わない形式の場合、Management Server の起動に失敗することがあります。

(例)
不正な Unicode エスケープシーケンスを含む場合（"%u"に続く文字列が適切な Unicode 16 進数の値ではない）。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config%
- UNIX の場合
 /opt/Cosminexus/manager/config/

(3) 機能

運用管理エージェント本体に関する設定をします。

(4) 設定できるキー

指定できるキーとデフォルト値を次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
adminagent.adapter.allowedHosts	サーバへのアクセスを許可するホスト名または IP アドレスを指定します。 複数のアドレスを指定する場合はコンマ (,) で空白を入れずに区切ります。IP アドレスはメタキャラクタを用いた正規表現で指定できます。	指定できる値を次に示します。 - ホスト名 - IPv4 アドレス	なし	05-05

キー名称	内容	指定可能値	省略値	VR
	IP アドレスを正規表現で指定した場合、利用可能なローカル IP アドレスのうちマッチする IP アドレスが使用されます。複数の IP アドレスがマッチした場合は、マッチしたすべての IP アドレスが許可されます。 なお、IP アドレスを正規表現で指定する場合は、コンマ(,) で複数の値を指定することはできません。			
adminagent.adapter.bind_host	運用管理エージェントとの通信で使用するホスト名または IP アドレスを指定します。複数の物理ネットワークインタフェースを持つホスト、または一つの物理ネットワークインタフェースに対して複数の論理 IP アドレスを割り当てているホストで運用管理エージェントを利用するとき、任意の IP アドレスを選択できます。 不正な値を指定した場合は、運用管理エージェントがエラー終了します。 IP アドレスはメタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で指定された場合、利用可能なローカル IP アドレスのうちマッチする IP アドレスを採用します。※1	指定できる値を次に示します。 • ホスト名 • IPv4 アドレス	なし	06-00
adminagent.adapter.port	サーバ機能の使用するポート番号を指定します。	1～65535 の整数で指定します。	20295	05-05
adminagent.cluster.localaddress.check	アプリケーションサーバの系切り替え実行時に、待機系を停止するかどうかを設定します。 アプリケーションサーバの系切り替え実行時、待機系の論理サーバや運用管理エージェントが停止しなかった場合に、論理サーバと運用管理エージェントを停止します。	次のどちらかを指定します。 • true • false	false	07-10

キー名称	内容	指定可能値	省略値	VR
	true を指定した場合： アプリケーションサーバの系切り替え実行時に、停止しなかった待機系の論理サーバおよび運用管理エージェントを停止します。 ローカルアドレスをチェックし、クラスタ IP アドレスが設定されていなければ論理サーバを停止して、運用管理エージェントを停止します。 false を指定した場合： アプリケーションサーバの系切り替え実行時に、停止しなかった待機系の論理サーバおよび運用管理エージェントを停止しません。			
adminagent.finalization.stop_servers	運用管理エージェントの終了処理で論理サーバを停止するかどうかを指定します。 true を指定した場合： 運用管理エージェントの終了処理で、論理サーバをすべて停止します。 false を指定した場合： 運用管理エージェントの終了処理で論理サーバを停止しません。	次のどちらかを指定します。 • true • false	false	06-50
adminagent.forcestop.thread_dump	J2EE サーバの強制停止時にスレッドダンプを出力するかどうかを指定します。この設定が有効な論理サーバは J2EE サーバだけです。 true を指定した場合： スレッドダンプを出力します。この場合、cjstopsv コマンドに-fd オプションを指定して実行した場合と同じ動作で強制停止します。※2	次のどちらかを指定します。 • true • false	false	06-50

キー名称	内容	指定可能値	省略値	VR
	false を指定した場合： スレッドダンプを出力しません。			
adminagent.forcestop.threaddump.interval	J2EE サーバの強制停止時にスレッドダンプを出力する設定にした場合に、スレッドダンプ出力の完了を確認する間隔を、1～2147483647 の整数（単位：秒）で指定します。 1～2147483647 以外が指定された場合、デフォルト値が設定されます。	1～2147483647 の整数で指定します。	1	06-50
adminagent.forcestop.threaddump.timeout	J2EE サーバの起動／停止の設定で、「強制停止監視時間」を「監視しない」設定にしている場合の、スレッドダンプ/クラッシュダンプ/core の出力待ち時間を-1～2147483647 の整数（単位：秒）で指定します。 -1 を指定した場合、スレッドダンプ/クラッシュダンプ/core の出力が終了するまで待機します。 -1～2147483647 以外を指定した場合、省略値が設定されます。 指定した時間が経過してもスレッドダンプ/クラッシュダンプ/core の出力が終了しない場合は、J2EE サーバおよび SFO サーバは強制停止します。※2	-1～2147483647 の整数で指定します。	45	06-50
adminagent.hws.group	HTTP Server を使用する場合に、HTTP Server を起動するユーザが所属するグループ名を指定します。 省略した場合は、運用管理エージェントのグループ名で起動します。 このキーは UNIX 用です。 ※3	グループ名	なし	08-00
adminagent.hws.owner	HTTP Server を使用する場合に、HTTP Server を起動するユーザ名を指定します。	ユーザ名	なし	08-00

キー名称	内容	指定可能値	省略値	VR
	省略した場合は、運用管理エージェントのユーザ名で起動します。 なお、このキーは UNIX 用です。※3			
adminagent.hws.sys_cmd.abnormal_end.traceinfo	障害検知時コマンドの実行の際に Web サーバの内部トレースを採取するかどうかを指定します。 true を指定した場合： 内部トレースを採取します。 false を指定した場合： 内部トレースを採取しません。	次のどちらかを指定します。 • true • false	true	07-00
adminagent.hws.watch.method	adminagent.hws.watch.level で 2 を指定したときの HTTP Server の動作確認用 HTTP メソッドを指定します。指定可能なメソッドは"HEAD"または"OPTIONS"のどちらかです。※4 "HEAD"を指定した場合： adminagent.hws.watch.url で指定された URL にアクセスして応答が返ることを確認します。 "OPTIONS"を指定した場合： Web サーバが HTTP メソッドを受け付け可能な状態であることを確認します。 ProxyPass ディレクティブのパス名に"/"を指定した場合、すべての URL は J2EE サーバの Web コンテナにリダイレクトされてしまいます。したがって、ProxyPass ディレクティブのパス名に"/"を指定する場合は、"OPTIONS"を指定し、URL アクセスを行わないことを推奨します。	次のどちらかを指定します。 • HEAD • OPTIONS	HEAD※5	06-50

キー名称	内容	指定可能値	省略値	VR
adminagent.hws.watch.url	adminagent.hws.watch.method で"HEAD"を指定したときの、HTTP Server の動作確認用 URL を指定します。指定された URL にアクセスし、レスポンスが返るかどうかで判定するため、実際にアクセスできる URL を指定してください。なお、URL は"http"で始まる絶対パス、またはルートコンテキストからの相対パスで指定します。同一のホストに複数の Web サーバを構築する場合は相対パスで指定してください。 (指定例) http://HostA:80/index.html を指定する場合 <pre>adminagent.hws.watch.url=http://HostA:80/index.html</pre> http://localhost:<ポート番号>/index.htmlを指定する場合（複数Webサーバ対応） <pre>adminagent.hws.watch.url=index.html</pre>	URL	ルートコンテキスト (http://localhost:<ポート番号>/)	06-00
adminagent.j2ee.process.console_event.enabled	J2EE サーバのコンソール出力情報を ManagementServer を使用する Eclipse プラグインで表示するかどうかを指定します。 true を指定した場合： コンソール出力情報を ManagementServer を使用する Eclipse プラグインに出力します。 false を指定した場合： コンソール出力情報を ManagementServer を使用する Eclipse プラグインに出力しません。 adminagent.process.consolelog.enabled を true に設定する必要があります。	次のどちらかを指定します。 • true • false	false	07-10

キー名称	内容	指定可能値	省略値	VR
	リソースを消費するため、ManagementServer を使用する Eclipse プラグインを使用しない運用では false に設定することを推奨します。			
adminagent.j2ee.process.console_log.enabled	J2EE サーバのコンソール出力情報をコンソールログに出力するかどうかを指定します。 true を指定した場合： コンソール出力情報をコンソールログに出力します。 false を指定した場合： コンソール出力情報をコンソールログに出力しません。 adminagent.process.console_log.enabled を true に設定する必要があります。	次のどちらかを指定します。 • true • false	false	07-10
adminagent.j2ee.sys_cmd.abnormal_end.javatrace	障害検知時コマンドの実行の際に J2EE サーバのスタックトレースを取得するかどうかを指定します。 true を指定した場合： スタックトレースを取得します。 false を指定した場合： スタックトレースを取得しません。	次のどちらかを指定します。 • true • false	true	07-00
adminagent.j2ee.sys_cmd.abnormal_end.threaddump	障害検知時コマンド実行の際に J2EE サーバのスレッドダンプを収集するかどうかを指定します。 true を指定した場合： J2EE サーバのスレッドダンプを収集します。 false を指定した場合： J2EE サーバのスレッドダンプを収集しません。	次のどちらかを指定します。 • true • false	true	06-50
adminagent.jp1event.event_server_name	使用する JP1/Base のイベントサービスのイベントサーバ設定ファイル (conf) に指定した、ports キーのアド	ホスト名または IPv4 ドット記法	localhost	06-70

キー名称	内容	指定可能値	省略値	VR
	レスと同じ値を指定します。 ports キーに複数のアドレスを指定している場合は、指定したアドレスのうちのどれか一つを指定します。なお、ports キーのアドレスに「0.0.0.0」（デフォルト値）を指定したイベントサービスを使用する場合は、このキーを省略するか、自マシンのホスト名または localhost を指定します。			
adminagent.jplevent_enabled	運用管理エージェントから JP1 イベントを発行するかどうかを指定します。 true を指定した場合： JP1 イベントを発行します。 false を指定した場合： JP1 イベントを発行しません。	次のどちらかを指定します。 • true • false	false	06-70
adminagent.log.filenum	ログファイル面数を 1～16 の整数で指定します。	1～16 の整数で指定します。	4	05-05
adminagent.log.filesize	ログファイル一つ当たりの上限サイズ（単位：バイト）を、65536～2147483647 の数値で指定します。	65536～2147483647 の整数で指定します。	65536	05-05
adminagent.log.level	ログ出力レベルを-1～1000 の整数で指定します。 • -1：出力しません。 • 0：通常運用：通常運用。 • 10：通常運用 (verbose)：通常運用に近い形態での再現待ち。 • 20：再現テスト：システム環境構築時やテストフェーズでの、デバッグレベルの情報の取得。 • 30：障害調査：原因究明困難な障害発生時、より詳細な障害情報の取得。	-1～1000 の整数で指定します。	0	05-05
adminagent.lsininfo_dir	論理サーバ情報ファイルの出力先ディレクトリを指定します。指定したディレクトリが	ディレクトリ名(絶対パス)	<Manager インストールディレクトリ	06-50

キー名称	内容	指定可能値	省略値	VR
	存在しない場合は作成します。 運用管理エージェント再起動後に、停止前に管理していた論理サーバを再び管理下に置き、操作できるようにするための情報です。論理サーバの起動中に作成され、論理サーバの停止後に削除されます。		>/spool/ lsinfo	
adminagent.maintenance.log.fileenum	保守ログのファイル面数を1～16の整数で指定します。	1～16の整数で指定します。	4	06-70
adminagent.maintenance.log.filesize	保守ログのファイル一つ当たりの上限サイズ（単位：バイト）を、65536～2147483647の整数で指定します。	65536～2147483647の整数で指定します。	16777216	06-70
adminagent.prftrace_dir	性能解析トレースファイルが一時的に出力されるディレクトリパス名を、文字列で指定します。指定したディレクトリがない場合、指定したディレクトリが作成されます。 注意事項 性能解析トレースファイルは次の場合、一時的に出力されます。 • 論理サーバの異常停止によってシステム障害検知コマンドが実行される場合 • Management Serverの運用管理コマンド（mnngsvrutil）のサブコマンド「collect」を指定してsnapshotログを収集する場合 • 運用管理ポータルでsnapshotログを収集する場合 そのため、次の算出式を基に出力先の空き容量を確保してください。指定先に出力されるファイル容量（最大値）＝PRFトレースのサイズ	ディレクトリ名(絶対パス)	<Application Serverのインストールディレクトリ> >/manager/tmp/	09-50

キー名称	内容	指定可能値	省略値	VR
	(PrfTraceFileSize 指定値) × PRF トレースの面数 (PrfTraceCount 指定値) なお、同一ホスト上に PRF が複数存在する場合は、すべての PRF トレースファイルサイズの総和となります。			
adminagent.process.consolelog.enabled	運用管理エージェントが起動したプロセスのコンソール出力情報を取得するかどうかを指定します。ただし、ManagementServer を使用する Eclipse プラグインを使用する場合は true を指定する必要があります。 true を指定した場合： コンソール出力情報を取得します。 false を指定した場合： コンソール出力情報を取得しません。	次のどちらかを指定します。 • true • false	true	07-00
adminagent.process.consolelog.event.queue_size	運用管理エージェントが起動したプロセスのコンソール出力情報を、Management Server にイベント通知する際の、イベントキューのサイズを設定します。 キューがいっぱいになった場合は、古いイベントから破棄されます。 このプロパティは、デフォルト値が推奨のため、設定を変更しないでください。	1024～2147483647 の整数で指定します。	1024	07-00
adminagent.process.consolelog.filenum	運用管理エージェントが起動したプロセスのコンソール出力情報を出力するログファイル面数を、1～16 の整数で指定します。	1～16 の整数で指定します。	4	07-00
adminagent.process.consolelog.filesize	運用管理エージェントが起動したプロセスのコンソール出力情報を出力するログファイル一つ当たりの上限サイズ (単位：バイト) を、	65536～2147483647 の整数で指定します。	65536	07-00

キー名称	内容	指定可能値	省略値	VR
	65536～2147483647 の整数で指定します。			
adminagent.process.consolelog.time	運用管理エージェントが起動したプロセスのコンソール出力情報を出力するログファイルをローテーションする時刻を、000000～235959 の HHMMSS 形式で指定します。ただし、このプロパティで設定した時刻に達する前に、ログファイルのサイズが上限に達した場合、その時点でログファイルをローテーションします。 指定がない場合は、ログサイズだけでローテーションします。	000000～235959 の HHMMSS 形式で指定します。	なし	09-00
adminagent.process.consolelog.style	運用管理エージェントが起動したプロセスのコンソール出力情報を出力するログファイルのローテーション方式を指定します。 SHIFT を指定した場合： シフトモードでローテーションします。 WRAP を指定した場合： ラップアラウンドモードでローテーションします。	次のどちらかを指定します。 • SHIFT • WRAP	WRAP	09-00
adminagent.rmi.log.filename	運用管理エージェントが行う RMI 処理での保守ログのファイル面数を数値で指定します。	1～16 の整数で指定します。	8	07-60
adminagent.rmi.log.filesize	運用管理エージェントが行う RMI 処理での保守ログのファイル一つ当たりの上限サイズを数値で指定します。	65536～2147483647 の整数で指定します。	16777216	07-60
adminagent.rmi.log.level	運用管理エージェントが行う RMI 処理での保守ログの出力レベルを文字列で指定します。 < OFF > 出力しません。 < SEVERE >, < WARNING >, < INFO >, < CONFIG >, <	次のどれかを指定します。 • OFF • SEVERE • WARNING • INFO • CONFIG • FINE • FINER • FINEST	FINER	07-60

キー名称	内容	指定可能値	省略値	VR
	FINE >, < FINER >, < FINEST > SEVERE は深刻な問題だけを出力します。ここに示した順番でより詳細なログが出力されるようになります。			
adminagent.snapshotlog.logfile.2.num_snapshots	二次資料として収集する、論理サーバごとの snapshot ログファイル数を、-2147483648～2147483647 の数値で指定します。指定されたファイル数を超えた場合、作成時刻の古い順にログファイルは削除されます。また、0 以下の値を指定した場合、ログファイルは削除されません。	-2147483648～2147483647 の整数で指定します。	10	06-50
adminagent.snapshotlog.log_dir	snapshot ログの出力先ディレクトリを、文字列で指定します。指定したディレクトリがない場合は作成します。	ディレクトリ名(絶対パス)	<manager.cfg の com.cosminexus.manager.log.dir>/snapshot/	06-00
adminagent.snapshotlog.num_snapshots	一次資料として収集する、論理サーバごとの snapshot ログファイル数を、-2147483648～2147483647 の数値で指定します。指定されたファイル数を超えた場合、作成時刻の古い順にログファイルは削除されます。また、0 以下の値を指定した場合、ログファイルは削除されません。	-2147483648～2147483647 の整数で指定します。	10	06-00
adminagent.sys_cmd.abnormal_end.prfttrace	障害検知時コマンド実行の際に性能解析トレースファイルを収集するかどうかを指定します。 true を指定した場合： 性能解析トレースファイルを収集します。 false を指定した場合： 性能解析トレースファイルを収集しません。	次のどちらかを指定します。 • true • false	true	06-50

キー名称	内容	指定可能値	省略値	VR
adminagent.userserver.process.console_log.enabled	ユーザ定義サーバのコンソール出力情報をコンソールログに出力するかどうかを指定します。 true を指定した場合： コンソール出力情報をコンソールログに出力します。 false を指定した場合： コンソール出力情報をコンソールログに出力しません。 adminagent.process.consolelog.enabled を true に設定する必要があります。	次のどちらかを指定します。 • true • false	false	07-10
adminagent.watch.retry_timeout.enabled	論理サーバの動作確認で、タイムアウトの発生をリトライ対象とするかどうかを指定します。 true を指定した場合： リトライ対象とします。 false を指定した場合： リトライ対象としません。	次のどちらかを指定します。 • true • false	• デフォルト：false • 初期値：true	06-70
adminagent.<サーバ種別> ※ ⁶ .usr_cmd.abnormal_end	障害検知時コマンドを絶対パスで指定します。ファイルセパレータは「/」を使用します。障害が発生した論理サーバの種別に応じてコマンドを変えることができます。<サーバ種別>※⁶に指定できる値以外の値を指定した場合は、無効になります。 Windows の場合： 障害検知時コマンドをバッチファイルで作成する場合、コマンド名の前に “%{ComSpec} /C ” を追加する必要があります。“%{ComSpec}” は、cmd.exe の絶対パスに置換されます。“%{ComSpec} /C” が指定されていない場合、コマンドが正常に動作しないことがあります。また、	コマンド名(絶対パス)	なし	06-50

キー名称	内容	指定可能値	省略値	VR
	コマンドのパスに半角スペースを含む場合は、コマンドの絶対パスを必ずダブルクォーテーション (") で囲みます。 (例) J2EE サーバ障害発生時に、d:/my bat/j2eeabnormalend.bat を実行する場合 <pre>adminagent.j2ee.usr_cmd.abnormal_end=%{ComSpec} /c "d:/my bat/j2eeabnormalend.bat"</pre> UNIX の場合： 障害検知時コマンドをシェルスクリプトで作成する場合、コマンド名の前に "/bin/sh" を追加する必要があります。 "/bin/sh" が指定されていない場合、コマンドが正常に動作しないことがあります。 (例) J2EE サーバ障害発生時に、/home/user1/j2eeabnormalend.sh を実行する場合 <pre>adminagent.j2ee.usr_cmd.abnormal_end=/bin/sh /home/user1/j2eeabnormalend.sh</pre>			
adminagent.<サーバ種別> ※6.watch.interval	論理サーバの動作確認間隔 (単位：秒) を 1～86400 の整数で指定します。 不正な値が指定された場合は、デフォルト値が設定されます。 なお、サーバ種別がユーザーサーバの場合、論理ユーザーサーバ定義ファイルまたは簡易構築定義ファイルで指定した watch-interval の値が有効になります。論理サーバの停止機能で設定する停止・強制停止監視時間は、このプロパティに指定した値よりも大きい値を指定する必要があります。	1～86400 の整数で指定します。	10	06-00

キー名称	内容	指定可能値	省略値	VR
	ます。指定した値よりも小さい値を指定した場合は停止・強制停止に失敗するおそれがあります。			
adminagent.<サーバ種別> ※6.watch.level	論理サーバの動作確認レベルを指定します。指定できる値は、1 または 2 です。 1 を指定した場合： プロセスの存在確認で論理サーバの動作を確認します。 2 を指定した場合： プロセスの存在確認および論理サーバへのアクセスで論理サーバの動作を確認します。動作を確認できる論理サーバは、ネーミングサービス、J2EE サーバおよび Web サーバです。論理ユーザサーバの指定はできません。そのほかの論理サーバについては 2 を指定しても、プロセスの存在確認だけになります。	1 または 2 を指定します。	- ネーミングサービス、J2EE サーバおよび Web サーバの場合：2 - 上記以外の論理サーバの場合：1	06-00
adminagent.<サーバ種別> ※6.watch.retry_count	論理サーバの動作確認で、動作確認処理に失敗した場合にリトライする回数を 0～86400 の整数で指定します。リトライ回数を設定することで、一時的な動作確認の失敗を許容できます。 リトライする間隔は、adminagent.<サーバ種別>.watch.interval で指定します。0 を指定した場合、動作確認処理に失敗すると、リトライしないで異常発生とみなされます。 動作確認処理に失敗し、リトライする場合は KEOS21033-W、または KEOS21034-W が運用管理エージェントのログに出力されます。 動作確認処理に失敗し、リトライしない場合、またはリト	0～86400 の整数で指定します。	1	06-70

キー名称	内容	指定可能値	省略値	VR
	ライ回数が指定回数を越えた場合は KEOS20511-E, または KEOS21035-E が運用管理エージェントのログに出力され, Management Server に異常が通知されます。			
adminagent.<サーバ種別> ※6.watch.start_time	論理サーバの起動で, 起動コマンドを実行してから動作確認を開始するまでの時間 (単位: 秒) を 0~86400 の整数で指定します。 論理サーバを起動した際の論理サーバのログ (J2EE サーバの場合, cjmessage?.log の KDJE30028-I) から実際に起動に掛かる所要時間を求め, その時間から数秒短い時間をこのプロパティに指定しておくことで, 起動前に出力される不要なログ出力を減らし, 論理サーバの起動完了を Management Server に迅速に通知できます。 論理サーバの起動/停止機能で設定する起動監視時間は, このプロパティに指定した値よりも大きい値を指定する必要があります。指定した値よりも小さい値を指定した場合は起動に失敗します。	0~86400 の整数で指定します。	<サーバ種別>※6 によって次のように異なります。 • prf : 0 • smartagent : 3 • naming : 10 • ctmdm : 0 • ctm : 0 • ots : 0 • tcs : 3 • sfo : 45 • j2ee : 45 • hws : 0 • userserver : 0	06-70
adminagent.<サーバ種別> ※6.watch.timeout	論理サーバの動作確認でタイムアウトするまでの時間を, 1~9223372036854775 の整数 (単位: 秒) で指定します。	1~9223372036854775 の整数で指定します。	60	06-50

注※1

運用管理ドメイン内で運用管理エージェントとの通信に使用される IP アドレスのサブネットが一つに決められている場合, 指定する IP アドレスを"192.168.0.32"のように記述しておけば, 前方が"192.168.0."である IP アドレス ("192.168.0.32"や"192.168.0.128"など) にマッチするため, 設定ファイルを修正しないで, すべてのホストに配布して利用できます。正規表現についての詳細は, Java の java.util.regex.Pattern クラスの仕様を確認してください。

ただし, "¥"は 1 文字の"¥"に置き換えられてしまうため, "¥"を指定する場合は 2 文字連続して指定してください。指定した正規表現にマッチする IP アドレスが複数個検出された場合は, 最も小さい値の IP アドレスを採用します。例えば, "192.168.0.32"と"192.168.0.128"が検出された場合

は"192.168.0.32"を採用します。この場合、採用された IP アドレスが意図する IP アドレスになるとは限らないため、"192.168.0.1.."のように必ず 1 個だけマッチする正規表現を記述してください。

注※2

強制停止時にスレッドダンプ/クラッシュダンプ/core を出力する場合は、「強制停止監視時間」または、`adminagent.forcestop.threaddump.timeout` キーに、スレッドダンプ/クラッシュダンプ/core が出力されるのに十分な時間を設定してください。スレッドダンプ/クラッシュダンプ/core の出力が終わる時間よりも短い時間が設定されていると、スレッドダンプ/クラッシュダンプ/core の出力内容が不完全になることがあります。

注※3

このプロパティを設定する場合は、一般ユーザアカウントによる運用の設定をしてください。一般ユーザアカウントによる運用の設定については、マニュアル「HTTP Server」を参照してください。

なお、HTTP Server のリソースの所有者・グループの変更は、`cmx_build_system` コマンドの実行後、または運用管理ポータルからのセットアップの実行後に変更してください。

注※4

ProxyPass ディレクティブのパス名に"/"を指定した場合、すべての URL は J2EE サーバの Web コンテナにリダイレクトされてしまいます。したがって、ProxyPass ディレクティブのパス名に"/"を指定する場合は、"OPTIONS"を指定し、URL アクセスを行わないことを推奨します。

注※5

省略値は HEAD ですが、初期値として `adminagent.hws.watch.method=OPTIONS` の行が定義されています。

注※6

<サーバ種別>は、論理サーバの種別を表す、次のどれかになります。

- `smartagent` : スマートエージェント
- `j2ee` : J2EE サーバ
- `naming` : ネーミングサービスおよび論理 CTM のグローバル CORBA ネーミングサービス
- `hws` : Web サーバ (HTTP Server)
- `ctm` : CTM
- `ctmdm` : CTM ドメインマネージャ
- `prf` : パフォーマンストレーサ
- `userserver` : ユーザサーバ

8.2.2 AdminAgentrc (運用管理エージェント自動起動用設定ファイル)

(1) 形式

sh 形式 (Linux の場合は bash 形式) のシェルスクリプトを記述します。

(2) ファイルの格納先

/opt/Cosminexus/manager/config/

(3) 機能

運用管理エージェントを自動起動する場合の設定をします。

このファイルでの設定が運用管理エージェントから起動する論理サーバに引き継がれます。

(4) 記述できる設定

sh 形式（Linux の場合は bash 形式）のシェルスクリプトのうち、次の設定を記述することができます。

- リソースの制御設定（ulimit）
- ファイル作成時のパーミッション（umask）
- 環境変数

これらの設定以外を記述した場合の動作は保証しません。

(5) 記述例

```
ulimit -c unlimited
```

(6) 注意事項

- このファイルはシェルスクリプトとして実行されますので、記載内容を変更する場合は十分な動作確認を実施してください。
- ulimit -u の値が limits.conf の nproc の値を超える場合、論理サーバが使用するプロセス数・スレッド数が limits.conf の nproc を超えてしまい、論理サーバを起動したユーザで別のプロセスの実行に問題が発生するおそれがあります。ulimit -u のサイズを拡大する場合は、limits.conf の nproc も拡大する必要があります。ulimit および limits.conf の設定に関する詳細については、OS 付属のマニュアルで ulimit または limits.conf の説明を参照してください。

8.2.3 adminagentuser.cfg（運用管理エージェント用オプション定義ファイル）

(1) 形式

次のようにキーを指定します。

<キー名称>=<値>

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 1 行に 65535 バイトまで指定できます。
- 値が存在しない行を定義した場合、その行は無視されます。
- ASCII 文字で指定してください。

(2) ファイルの格納先

- Windows の場合

<Application Serverのインストールディレクトリ>%manager%config%

- UNIX の場合

/opt/Cosminexus/manager/config/

(3) 機能

運用管理エージェントを実行する JavaVM の起動オプションを指定します。

このファイルが存在し、かつ運用管理エージェントからこのファイルの読み込みができる場合に、ファイルの内容が反映されます。このファイルが存在する場合に、運用管理エージェントからこのファイルの読み込みができないときには、運用管理エージェントの起動に失敗します。

このファイルが存在しない場合は、オプションの設定は行わないで、運用管理エージェントが起動されます。

(4) 指定できるキー

指定できるキーとデフォルト値を次に示します。不正なキーを指定した場合は、運用管理エージェントの起動に失敗します。

キー名称	内容	デフォルト値
add.class.path	クラスパスに追加する値を指定します。 複数の値を指定する場合、同じキー名称を使用して、クラスパスを指定してください。指定内容には、Application Server のインストールディレクトリを「\${cosminexus.home}」で指定できます。 (指定例) add.class.path=C:/home/lib/mylib.jar	なし
add.jvm.arg	JavaVM を起動するときのオプションを指定します。 次の JavaVM のオプションを指定できます。 • -Xms<size>	なし

キー名称	内容	デフォルト値
	• -Xmx<size> • -XX:MetaspaceSize=<value> • -XX:MaxMetaspaceSize=<value> • -XX:[+ -]HitachiOutOfMemoryAbort (08-50 以降) 複数のオプションを指定する場合、同じキー名称を使用して、次のように指定してください。 (指定例) <pre>add.jvm.arg=-Xms128m add.jvm.arg=-Xmx256m</pre> JavaVM が認識できない値を指定した場合、運用管理エージェントの起動は失敗します。	
add.library.path	ライブラリ検索パスに追加する値を指定します。 複数の値を指定する場合、同じキー名称を使用して、ライブラリ検索パスを指定してください。指定内容には、Application Server のインストールディレクトリを「\${cosminexus.home}」で指定できます。 (指定例) <pre>add.library.path=C:/home/bin</pre>	なし
add.network.drive	ネットワークドライブとして割り当てるドライブの名前および割り当て先のディレクトリパスを指定します。 (指定例) <pre>X=¥¥host¥dir</pre> なお、パスの最後が"¥"で終わっていると、ネットワークドライブに接続できない場合があります。 UNIX で設定した場合、/opt/Cosminexus/manager/adminagent.err に KEOS21401-E メッセージが出力されて、終了コード 1 で運用管理エージェントが終了します。	なし

(5) 注意事項

運用管理エージェントで java.lang.OutOfMemoryError が発生した場合に、このファイルで、次のように Java ヒープの最大サイズを拡張します。

```
add.jvm.arg=-Xmx<Java ヒープの最大サイズ>
```

8.2.4 adminagent.xml (運用管理エージェント設定ファイル)

(1) 形式

ファシリティ名（論理サーバの種類と起動／停止の動作種別を示す）の設定のあとに、環境変数名および環境変数の値を設定します。

ファシリティ名の設定

```
<facility>
  <name>「ファシリティ名」</name>
  <class>com.cosminexus.mngsvr.adminagent.impl.CmdExecutor</class>
```

環境変数名および環境変数名の設定

上記ファシリティ名の設定のあとに、次の記述を追加します。

```
<init-param>
  <param-name>adminagent.facility.CmdExecutor.env.「環境変数名」</param-name>
  <param-value>「環境変数の値」</param-value>
</init-param>
```

(2) ファイルの格納先

- Windows の場合

<Application Serverのインストールディレクトリ>%manager%config%

- UNIX の場合

/opt/Cosminexus/manager/config/

(3) 機能

システム管理者の代わりに、論理サーバの起動、または設定ファイルの更新をします。

このファイルで初期設定されている環境変数の値を変更する必要がある場合、または環境変数を追加または削除する必要がある場合は、この設定ファイルを編集してください。

(4) ファシリティ名で指定できる論理サーバ種類

ファシリティ名で指定できる論理サーバ種類と、起動／停止の動作の種別を次の表に示します。

論理サーバ種類	起動／停止種別	ファシリティ名	備考
パフォーマンストレーサ	起動	prf-start	—
	停止	prf-stop	—
スマートエージェント (Windows の場合)	起動	osagent-start	—
	停止	osagent-stop	—
スマートエージェント (UNIX の場合)	起動	osagent-start	—
ネーミングサービス	起動	naming-service-start	—
CTM ドメインマネージャ	起動	ctm-dommng-start	—
	停止	ctm-dommng-stop	—

論理サーバ種類	起動／停止種別	ファシリティ名	備考
CTM	起動	ctm-start	—
	停止	ctm-stop	—
J2EE サーバ	起動	j2ee-server-start	J2EE サーバでデータベースを利用する場合には、J2EE サーバ起動時の環境変数にデータベース固有の設定をする必要があります。
	停止	j2ee-server-stop	
WEB サーバ (Windows の場合)	起動	web-server-start	—
	停止	web-server-stop	—
WEB サーバ (UNIX の場合)	起動	web-server-start	—

(凡例) —：ありません。

(5) 注意事項

- adminagent.xml を変更した場合は、運用管理エージェントを再起動してください。
- 各論理サーバの起動／停止時には、運用管理エージェントの起動時の環境変数を引き継ぎます。環境変数を引き継ぎたくない場合は、該当環境変数を空に設定してください。
- adminagent.xml で設定した環境変数は、該当ホストの運用管理エージェントから起動される論理サーバの種類ごとに有効になります。論理サーバのインスタンス単位に環境変数の設定はできません。

(例 1)

J2EE サーバ起動時に、環境変数 NLS_LANG に Japanese_japan.ja16sjis を設定する場合

```
<facility>
<name>j2ee-server-start</name>
<class>com.cosminexus.mngsvr.adminagent.impl.CmdExecutor</class>
<init-param>
<param-name>adminagent.facility.CmdExecutor.env.NLS_LANG</param-name>
<param-value>Japanese_japan.ja16sjis</param-value>
</init-param>
...
...
```

(例 2)

運用管理エージェント起動時に設定された環境変数 AIXTHREAD_SCOPE を引き継がない場合

```
<facility>
...
...
```

```

<init-param>
<param-name>adminagent.facility.CmdExecutor.env.AIXTHREAD_SCOPE</param-name>
<param-value></param-value>
</init-param>
...

```

- 「環境変数名」および「環境変数の値」に設定した値は、コマンドシェルのような環境変数の置換や演算子による処理がされません。このため、「環境変数名」および「環境変数の値」にシステム環境変数は使用できません。

8.2.5 mngagent.<実サーバ名>.properties（運用監視エージェントプロパティファイル）

（1）形式

J2SE のプロパティファイル形式です。

運用監視エージェントのプロパティファイルです。

（2）ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config%
- UNIX の場合
 /opt/Cosminexus/manager/config/

（3）運用管理ドメインおよび運用監視エージェントの名称設定で利用できるキー

指定できるキーとデフォルト値を次に示します。これらのキーに設定したドメイン名およびエージェント名は、ログファイル名のデフォルト値として使用されます。

キー名称	内容	デフォルト値
mngagent.domain_name	運用管理ドメイン名を、文字列で指定します。使用できる文字は ASCII 文字の英数字（A～Z, a～z, 0～9）、アンダースコア（_）、およびハイフン（-）です。	なし
mngagent.agent_name	該当ノードの論理サーバ名を、文字列で指定します。使用できる文字は ASCII 文字の英数字（A～Z, a～z, 0～9）、アンダースコア（_）、およびハイフン（-）です。	なし

(4) ログ・トレースの設定で利用できるキー

指定できるキーとデフォルト値を次に示します。

キー名称	内容	デフォルト値
mngagent.log.filename	ログファイル名を指定します。空文字列や空白文字を指定した場合、ログは出力されません。実際の出力先ファイル名は、指定したファイル名のあとに「.log」が付きます（n は 1～4）。	- Windows の場合 <manager.cfg の com.cosminexus.manager.log.dir>*mngagent-ドメイン名-Agent 名 - UNIX の場合 <manager.cfg の com.cosminexus.manager.log.dir>/mngagent-ドメイン名-Agent 名
mngagent.log.filesize	ログファイル一つ当たりの上限サイズ（バイト）を、4096～2147483647 で指定します。範囲外の値を指定した場合は、65536 を仮定します。	65536

(5) 通信ポートおよび IP アドレス固定による運用の設定で利用できるキー

指定できるキーとデフォルト値を次に示します。

キー名称	内容	デフォルト値
mngagent.connector.port	運用監視エージェントの通信ポートを 1～65535 までの整数で固定します。ポート番号は、ほかのプログラムなどと重複しないようにしてください。	空いているポート番号から自動的に割り当てられます。
mngagent.connector.host	任意の値を指定して、運用監視エージェントの IP アドレスを固定します。	利用可能な IP アドレスが自動的に割り当てられます。

(6) 注意事項

- mngagent.<実サーバ名>.properties ファイルは、次の場合に作成されます。

- Smart Composer 機能で Web システムを構築した場合
- 運用管理ポータルで J2EE サーバの設定情報を配布した場合

これらの機能を使用しないで構築した J2EE サーバの運用監視を行う場合は、設定ファイル mngagent.<実サーバ名>.properties を作成し、運用監視エージェント本体に関する設定をしてください。

- ファイル名には、サーバインスタンスごとに異なるものを設定してください。
- mngagent.domain_name, mngagent.agent_name の値が同一の運用監視エージェントを複数同時に起動した場合、デフォルトでは同一のログファイルに出力するため、ログファイルの内容が破壊され

る可能性があります。同一マシン上で複数の J2EE サーバを起動する場合、mngagent.agent_name の値が重複しないように定義してください。

8.2.6 mserver.properties (Management Server 環境設定ファイル)

(1) 形式

J2SE のプロパティファイル形式です。

Management Server のプロパティファイルです。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config%
- UNIX の場合
 /opt/Cosminexus/manager/config/

(3) 機能

Management Server が使用するポート番号の設定や、障害検知時コマンドの動作の設定をします。Management Server を起動しないでポート番号を変更する場合は、このファイルを編集します。

(4) 指定できるキー

指定できるキーとその内容、および指定を省略した場合のデフォルト値を次に示します。

キー名称	内容	指定可能値	デフォルト	VR
adminagent.connector.commm.state.cache_max_time	運用管理エージェントとの通信状態をキャッシュする最大時間を、0～2147483 の数値（単位：秒）で指定します。0 未満を指定した場合はキャッシュしません。	0～2147483 の整数で指定します。	60	06-50
com.cosminexus.mngsvr.agent.read_timeout	運用監視エージェント接続時のタイムアウト時間を、1～2147483 の整数値（単位：秒）で指定します。	1～2147483 の整数で指定します。	180	07-00
com.cosminexus.mngsvr.jp1event.alert	Alert レベルのイベントを発行するかどうかを指定します。 ^{※3} true を指定した場合： Alert レベルのイベントを発行します。 false を指定した場合： Alert レベルのイベントを発行しません。	次のどちらかを指定します。 • true • false	false	07-00

キー名称	内容	指定可能値	デフォルト	VR
com.cosminexus.mngsvr.jplevent.emergency	Emergency レベルのイベントを発行するかどうかを指定します。※3 true を指定した場合： Emergency レベルのイベントを発行します。 false を指定した場合： Emergency レベルのイベントを発行しません。	次のどちらかを指定します。 • true • false	false	07-00
com.cosminexus.mngsvr.jplevent.critical	Critical レベルのイベントを発行するかどうかを指定します。※3 true を指定した場合： Critical レベルのイベントを発行します。 false を指定した場合： Critical レベルのイベントを発行しません。	次のどちらかを指定します。 • true • false	false	07-00
com.cosminexus.mngsvr.jplevent.enabled	システム JP1 イベント発行機能を有効にするかどうかを指定します。※3 true を指定した場合： システム JP1 イベント発行機能を有効にします。 false を指定した場合： システム JP1 イベント発行機能を無効にします。	次のどちらかを指定します。 • true • false	false	07-00
com.cosminexus.mngsvr.jplevent.error	Error レベルのイベントを発行するかどうかを指定します。※3 true を指定した場合： Error レベルのイベントを発行します。 false を指定した場合： Error レベルのイベントを発行しません。	次のどちらかを指定します。 • true • false	false	07-00
com.cosminexus.mngsvr.jplevent.information	Information レベルのイベントを発行するかどうかを指定します。※3 true を指定した場合： Information レベルのイベントを発行します。 false を指定した場合： Information レベルのイベントを発行しません。	次のどちらかを指定します。 • true • false	false	07-00
com.cosminexus.mngsvr.jplevent.notice	Notice レベルのイベントを発行するかどうかを指定します。※3 true を指定した場合： Notice レベルのイベントを発行します。 false を指定した場合： Notice レベルのイベントを発行しません。	次のどちらかを指定します。 • true • false	false	07-00

キー名称	内容	指定可能値	デフォルト	VR
com.cosminexus.mnngsvr.jplevent.warning	Warning レベルのイベントを発行するかどうかを指定します。※3 true を指定した場合： Warning レベルのイベントを発行します。 false を指定した場合： Warning レベルのイベントを発行しません。	次のどちらかを指定します。 • true • false	false	07-00
com.cosminexus.mnngsvr.log.display_number	mnngsvr 構成機能がログの表示画面でログ情報を表示する最大表示件数を、次の数値で指定します。※3 20, 60, 100, 200, 400	20, 60, 100, 200, 400 のどれかを指定します。	100	07-00
com.cosminexus.mnngsvr.log.level	Management Server のログの出力レベルを指定します。※3 • 0：通常運用。 • 10：通常運用に近い形態での再現待ち。 • 20：システム環境構築時やテストフェーズでの、デバッグレベルの情報の取得。 • 30：原因究明困難な障害発生時、より詳細な障害情報の取得。	0, 10, 20, 30 のどれかを指定します。	0	07-00
com.cosminexus.mnngsvr.log.rotate	Management Server のログのファイル面数を次の数値で指定します。※3 1, 2, 4, 8, 16	1, 2, 4, 8, 16 のどれかを指定します。	4	07-00
com.cosminexus.mnngsvr.log.size	Management Server のログのファイルサイズを次の数値で指定します。※3 4096 (4 キロバイト), 65536 (64 キロバイト), 262144 (256 キロバイト), 524288 (512 キロバイト), 1048576 (1 メガバイト), 2097152 (2 メガバイト), 4194304 (4 メガバイト), 16777216 (16 メガバイト), 67108864 (64 メガバイト)	次のどれかを指定します。 • 4096 (4 キロバイト) • 65536 (64 キロバイト) • 262144 (256 キロバイト) • 524288 (512 キロバイト) • 1048576 (1 メガバイト) • 2097152 (2 メガバイト) • 4194304 (4 メガバイト)	65536	07-00

キー名称	内容	指定可能値	デフォルト	VR
		16777216 (16 メガバイト) 67108864 (64 メガバイト)		
com.cosminexus.mnngsvr.logical_server_abnormal_stop.exit	Management Server で管理している論理サーバの稼働状況ステータスが異常停止状態（自動再起動回数がオーバーした状態または自動再起動回数の設定が 0 の状態）になった場合の動作を指定します。 true を指定した場合： KEOS10038-I を出力して、Management Server を停止します。 false を指定した場合： Management Server の処理を続行します。	次のどちらかを指定します。 - true - false	false	08-50
com.cosminexus.mnngsvr.maintenance.log.filenum	Management Server の保守ログファイルの面数を 1～16 の整数で指定します。	1～16 の整数で指定します。	2	06-70
com.cosminexus.mnngsvr.maintenance.log.filesize	Management Server の保守ログファイルの最大サイズ（単位：バイト）を 65536～2147483647 の整数で指定します。	65536～2147483647 の整数で指定します。	16777216	06-70
com.cosminexus.mnngsvr.management.connector.enabled	Management Server リモート管理機能への外部接続を有効にするかどうかを指定します。 true を指定した場合： Management Server リモート管理機能への外部接続を有効にします。 false を指定した場合： Management Server リモート管理機能への外部接続を有効にしません。	次のどちらかを指定します。 - true - false	false	07-60
com.cosminexus.mnngsvr.management.enabled	Management Server リモート管理機能を有効にするかどうかを指定します。 true を指定した場合： Management Server リモート管理機能を有効にします。 false を指定した場合： Management Server リモート管理機能を有効にしません。	次のどちらかを指定します。 - true - false	false	07-60
com.cosminexus.mnngsvr.management.host	Management Server リモート管理機能への外部接続のホスト固定を行う場合に、ホスト名、または IP アドレスを指定します。	ホスト名または IPv4 ドット記法	なし	07-60

キー名称	内容	指定可能値	デフォルト	VR
	IP アドレスはメタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で指定された場合、利用可能なローカル IP アドレスのうちマッチする IP アドレスが使用されます。※¹ ホストを固定する場合には、mserver.properties ファイルに java.rmi.server.hostname キーを追加して、このキーと同じ値を指定してください。ただし、java.rmi.server.hostname には、メタキャラクタを用いた正規表現は使用できません。 webserver.connector.http.bind_host に値を指定している場合は、このプロパティにも同じ値を指定してください。			
com.cosminexus.mnngsvr.management.listen.port	Management Server リモート管理機能への外部接続時に作成する、クライアント接続用のポート番号を、0～65535 の整数で指定します。0 が指定された場合は空いているポート番号を自動的に割り当てます。	0～65535 の整数で指定します。	0	07-60
com.cosminexus.mnngsvr.management.port	Management Server リモート管理機能への外部接続ポート番号を 1～65535 の整数で指定します。	1～65535 の整数で指定します。	28099	07-60
com.cosminexus.mnngsvr.management.read_time_out	Management Server リモート管理機能への外部接続時の読み込みタイムアウト時間(単位：秒)を、1～2147483 の整数で指定します。	1～2147483 の整数で指定します。	180	07-60
com.cosminexus.mnngsvr.management_user_account.enabled	Management Server のユーザアカウントを有効にするかどうかを指定します。 true を指定した場合： Management Server のログイン認証処理を実行します。 false を指定した場合： Management Server のログイン認証処理を実行しません。運用管理ポータルログイン画面で管理ユーザアカウントを入力しないでログインできます。また、Manager で使用するコマンドのうち、共通引数である -u, -p の指定を省略できます。	次のどちらかを指定します。 • true • false	true	08-50
com.cosminexus.mnngsvr.on_start	Management Server の開始時に論理サーバの一括起動を行うかどうかを指定します。※³ true を指定した場合： 論理サーバを一括起動します。 false を指定した場合： 論理サーバを一括起動しません。	次のどちらかを指定します。 • true • false	false	07-00

キー名称	内容	指定可能値	デフォルト	VR
com.cosminexus.mnngsvr.snapshot.auto_collect.enabled	障害発生時または自動再起動時に snapshot ログを収集するかどうかを指定します。 true を指定した場合： snapshot ログを収集します。 false を指定した場合： snapshot ログを収集しません。	次のどちらかを指定します。 • true • false	true	06-50
com.cosminexus.mnngsvr.snapshot.collect.point	snapshot ログ収集のタイミングを指定します。 before_stop を指定した場合： 論理サーバ停止前に snapshot ログを収集します。 j2ee_restart を指定した場合： J2EE サーバ再起動前に snapshot ログを収集します。	次のどちらかを指定します。 なお、大文字と小文字は区別されません。 • before_stop • j2ee_restart	before_stop	06-50
com.cosminexus.mnngsvr.sys_cmd.abnormal_end.enabled	システムによる障害検知時コマンド実行機能を利用するかどうかを指定します。 true を指定した場合： 障害検知時コマンド実行機能を利用します。 false を指定した場合： 障害検知時コマンド実行機能を利用しません。	次のどちらかを指定します。 • true • false	true	06-50
com.cosminexus.mnngsvr.sys_cmd.abnormal_end.timeout	次に示す処理それぞれの終了を待つ時間を、-1～2147483647 の整数値（単位：秒）で指定します。 • システム提供の障害検知時に実行したコマンド • 性能解析トレースの収集 指定した時間を経過してもコマンド、または性能解析トレースの収集が終了しない場合は、実行したコマンド、または性能解析トレースの収集を無視して処理を続行します。 -1 が指定された場合は、コマンド、または性能解析トレースの収集が終了するまで待機します。 -1～2147483647 以外の値が指定された場合、デフォルト値が設定されます。	-1～2147483647 の整数で指定します。	600	06-50
com.cosminexus.mnngsvr.trace	運用管理コマンド（mnngsvrutil）の性能解析トレースまたは CTM の稼働統計情報の収集時、Management Server で保持する性能解析トレース、または CTM の稼働統計情報のファイルの最大数を指定します。 最大数は運用管理ドメイン、ホスト、論理サーバごとに管理されます。※3 1, 2, 4, 8, 16	1, 2, 4, 8, 16 のどれかを指定します。	4	07-00

キー名称	内容	指定可能値	デフォルト	VR
com.cosminexus.mngsvr. usr_cmd.abnormal_end. enabled	ユーザ作成の障害検知時コマンド実行機能を利用するかどうかを指定します。 true を指定した場合： 障害検知時コマンド実行機能を利用します。 false を指定した場合： 障害検知時コマンド実行機能を利用しません。	次のどちらかを指定します。 • true • false	false	06-50
com.cosminexus.mngsvr. usr_cmd.abnormal_end. timeout	ユーザ作成の障害検知時に実行したコマンドの終了を待つ時間を、-1～2147483647 の整数値（単位：秒）で指定します。 指定した時間を経過してもコマンドが終了しない場合は、実行したコマンドを無視して処理を続行します。 0 を指定した場合は、コマンドの終了を待たずに処理を続行します。 -1 を指定した場合は、コマンドが終了するまで待機します。 -1～2147483647 以外の値が指定された場合、デフォルト値が設定されます。	-1～2147483647 の整数で指定します。	60	06-50
com.cosminexus.mngsvr. vmi.enabled	Management Server を仮想サーバマネージャとして動作させるかどうかを指定します。 true を指定した場合： Management Server が仮想サーバマネージャとして動作します。なお、Management Server を使用した論理サーバの構築・運用はしないでください。true を指定した状態で仮想サーバマネージャ以外の機能を使用した場合、動作保証の対象外となります。 false を指定した場合： Management Server として動作します。	次のどちらかを指定します。 • true • false	false	08-53
com.cosminexus.mngsvr. upload_app.enabled	運用管理ポータルから Management Server が稼働するホストへアプリケーションファイルをアップロードする機能を有効にするかどうかを指定します。 true を指定した場合： 運用管理ポータルから Management Server 稼働ホストへアプリケーションファイルをアップロードする機能を有効にします。 false を指定した場合： 運用管理ポータルから Management Server 稼働ホストへアプリケーションファイルをアップロードする機能を無効にします。	次のどちらかを指定します。 • true • false	true	08-70

キー名称	内容	指定可能値	デフォルト	VR
com.cosminexus.mngsvr.upload_app.directory	運用管理ポータルから Management Server が稼働するホストへアプリケーションファイルをアップロードするディレクトリを指定します。	ディレクトリ名を絶対パスで指定します。	<Application Server のインストールディレクトリ>/manager/apps	08-70
com.cosminexus.mngsvr.upload_app.maxsize	運用管理ポータルから Management Server が稼働するホストへアプリケーションファイルをアップロードできる最大サイズ（単位：バイト）を指定します。 1024 バイト（1KB）以上のサイズを指定してください。 0 を指定した場合、アップロードできるファイルサイズを制限しません。	0～ 9223372036 854775807	314572800 (300MB)	08-70
com.cosminexus.mngsvr.snapshot.auto_collect.timeout	論理サーバの障害検知時の snapshot ログの一次送付資料と二次送付資料の収集をすべて完了するまでのタイムアウト時間（単位：秒）を指定します。 0 を指定した場合 タイムアウトは発生しません。 1～2147483 を指定した場合 指定された値の秒数が経過しても、snapshot ログの収集が完了しない場合、Management Server は snapshot ログ収集を中断します。	0～2147483	- デフォルト：0 - 初期値：3600	08-70
ejbserver.naming.port	HCSC-Manager から Management Server を利用する場合の Management Server の内部通信ポート番号を指定します。 HCSC-Manager から利用されるときだけ有効になります。	1～65535	28900	09-00
ejbserver.rmi.logger.file num	Management Server の RMI 通信ログの面数を、2～16 の整数で指定します。 数値以外の文字列や範囲外の数値が指定された場合、または文字列が何も指定されていない場合は、メッセージが出力され、デフォルト値が設定されます。	2～16	4	09-00
ejbserver.rmi.logger.file size	Management Server の RMI 通信ログのサイズ（単位：バイト）を、8192～2147483647 の整数で指定します。	8192～ 2147483647	1048576	09-00
ejbserver.server.edition.settingforce	HCSC-Manager から Management Server を利用するかどうかを指定します。 HCSC-Manager から Management Server を利用しない場合は、このキーに Smart を指定するか、このキーを省略してください。	Smart	Smart	09-00

キー名称	内容	指定可能値	デフォルト	VR
	このキーを省略した場合は、Smart が指定されたものとして解釈されます。 - Smart を指定した場合： Management Server を HCSC-Manager から利用しません。なお、Smart を指定すると、ejbserver.naming.port キーの設定は無視されます。 - 指定なし： Management Server を HCSC-Manager から利用します。 注意事項 指定なしとはキーだけを指定し、値を指定しないことです。 (指定例) ejbserver.server.edition.settingforce=			
java.rmi.server.hostname	Management Server リモート管理機能への外部接続のホスト固定を行う場合に、ホスト名、または IP アドレスを指定します。 このプロパティの設定値については com.cosminexus.mngsvr.management.host を参照してください。 また、NAT (IP マスカレード) のような、グローバル IP アドレスとローカル IP アドレスの変換を行っている環境では、com.cosminexus.mngsvr.management.host を指定したかどうかに関係なく、ホスト名 (IP アドレス不可) を指定してください。この場合、com.cosminexus.mngsvr.management.host に IP アドレスを指定しているときには、その IP アドレスを示すホスト名を指定しなければなりません。	ホスト名または IPv4 ドット記法	なし	07-60
mngsvr.jplevent.event_server_name	使用する JP1/Base のイベントサービスのイベントサーバ設定ファイル (conf) に指定した、ports キーのアドレスと同じ値を指定します。ports キーに複数のアドレスを指定している場合は、指定したアドレスのうちのどれか一つを指定します。 なお、ports キーのアドレスに「0.0.0.0」(デフォルト値) を指定したイベントサービスを使用する場合は、このキーを省略するか、自マシンのホスト名または localhost を指定します。	ホスト名または IPv4 ドット記法	localhost	06-70
mngsvr.myhost.name	論理サーバを配置するホスト名に、論理サーバを配置するホストからアクセス可能な IP アドレス、またはその IP アドレスを示すホスト名を指定します。 Management Server が稼働するホストが複数個の IP アドレスを持っていて、ホスト名が意図す	ホスト名または IPv4 ドット記法	なし	06-50

キー名称	内容	指定可能値	デフォルト	VR
	る IP アドレスに変換されないおそれがある場合に、このキーを指定します。 省略した場合、および空文字を指定した場合、ループバックアドレスは取得可能な自ホスト名に変換され使用されます。自ホスト名が取得できない場合は、ループバックアドレスが使用されます。 IP アドレスはメタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で指定された場合、利用可能なローカル IP アドレスのうちマッチする IP アドレスが使用されます。※1 注意事項 webserver.connector.http.bind_host に値を指定している場合は、このプロパティにも同じ値を指定してください。			
webserver.connector.ajp13.port	Management Server 内部通信用ポート番号を指定します。 インストール時は 28009 が設定されます。 指定されていない場合、Management Server の起動に失敗します。	1～65535 の整数で指定します。	なし	05-00
webserver.connector.http.bind_host	複数の物理ネットワークインタフェースを持つホスト、または一つの物理ネットワークインタフェースに対して複数の論理 IP アドレスを割り当てているホストで Management Server を利用するとき、任意の IP アドレスを選択できます。ただし、ループバックアドレスは指定できません。 値を指定しなかった場合、すべてのローカルアドレスに対する接続が受け付けられます。 IP アドレスはメタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で指定された場合、利用可能なローカル IP アドレスのうちマッチする IP アドレスが使用されます。※1	次のどれかを指定します。 - IPv4 アドレス - ホスト名 (半角英数字または記号 (._-) で指定された 255 文字以内の文字列)	なし	07-10
webserver.connector.http.permitted.hosts※2	Management Server へのアクセスを許可するホストおよび運用管理エージェント稼働ホストの IP アドレスまたはホスト名を指定します。運用管理ポータル (Web ブラウザ)、Management Server リモート管理機能または運用管理コマンド (mngsvrutil) を使用したアクセスが対象です。 また、Management Server で管理するホストの IP アドレス、またはホスト名もすべて指定しなければなりません。指定されていないホストの論理サーバを操作した場合、操作が完了しなかったりタイムアウトが発生したりします。 なお、キーを省略した場合、アクセス制限をしません。値を省略した場合または、入力した値が不正な場合は、アクセスできるのはローカルホストだけです。	次のどちらかを指定します。 IPv4 アドレスまたはホスト名 複数指定する場合は、コンマ(,)で区切って指定します。 ホスト名は英数字または記号 (._-) で指定された	- デフォルト：* - 初期値：localhost	06-00

キー名称	内容	指定可能値	デフォルト	VR
	複数指定する場合には IP アドレスまたはホスト名の間をコンマ (,) で区切ります。アクセス制限をしない場合はアスタリスク (*) だけを指定します。IP アドレスまたはホスト名の前後の半角スペースは無視されます。	255 文字以内の文字列を指定します。 • *(アスタリスク)		
webserver.connector.http.port	Management Server 接続 HTTP ポート番号を指定します。 インストール時は 28080 が設定されます。 指定されていない場合、Management Server の起動に失敗します。 変更したポート番号を反映させるための手順については、マニュアル「アプリケーションサーバ 運用管理ポータル操作ガイド」の「4.1.2(2)(a) 起動前の確認事項」の注意事項を参照してください。	1～65535 の整数で指定します。	なし	05-00
webserver.shutdown.port	Management Server 終了要求受信ポート番号を指定します。 インストール時は 28005 が設定されます。 指定されていない場合、Management Server の起動に失敗します。	1～65535 の整数で指定します。	なし	05-00
com.cosminexus.mnngsvr.compat.operation_app	アプリケーション管理方法を指定します。なお、この項目は互換用の項目です。 true を指定した場合： 動作保証の対象外となります。 false を指定した場合： V8 モード 注意事項 このキーは、手動で編集しないでください。	false を指定します。	- デフォルト：true - 初期値：false	08-00

注※1

運用管理ドメイン内で運用管理エージェントとの通信に使用される IP アドレスのサブネットが一つに決められている場合、指定する IP アドレスを"192.168.0.+"のように記述しておけば、前方が"192.168.0."である IP アドレス ("192.168.0.32"や"192.168.0.128"など) にマッチするため、設定ファイルを修正しないで、すべてのホストに配布して利用できます。正規表現についての詳細は、Java の java.util.regex.Pattern クラスの仕様を確認してください。

ただし、"¥¥"は 1 文字の"¥"に置き換えられてしまうため、"¥"を指定する場合は 2 文字連続して指定してください。指定した正規表現にマッチする IP アドレスが複数個検出された場合は、最も小さい値の IP アドレスを採用します。例えば、"192.168.0.32"と"192.168.0.128"が検出された場合は"192.168.0.32"を採用します。この場合、採用された IP アドレスが意図する IP アドレスになるとは限らないため、"192.168.0.1.."のように必ず 1 個だけマッチする正規表現を記述してください。

注※2

キーを省略した場合は*、キーはあるが値を省略した場合は localhost が指定されたものとして動作します。

注※3

設定範囲外の値を入力したときは前回設定したときの値が有効になります。

未設定や未定義である場合は初回の起動を除き、前回設定したときの値が有効になります。

(5) 注意事項

Management Server 起動後は、運用管理ポータル「Cosminexus Management Server の設定」の「ネットワークの設定」からもポート番号を変更できます。ただし、mserver.properties ファイルにコメントを書いていた場合、変更時に削除されます。変更したポート番号を反映させるための手順については、マニュアル「アプリケーションサーバ 運用管理ポータル操作ガイド」の「4.1.2(2)(a) 起動前の確認事項」の注意事項を参照してください。

8.2.7 mserver.cfg (Management Server 用オプション定義ファイル)

(1) 形式

次のようにキーを指定します。

`<キー名称> = <値>`

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) `<キー名称> = <値> #<コメント>`

(2) ファイルの格納先

- Windows の場合
`<Application Serverのインストールディレクトリ>%manager%config%`
- UNIX の場合
`/opt/Cosminexus/manager/config/`

(3) 機能

Management Server を実行する JavaVM の起動オプションを指定します。

Management Server の稼働中に、このファイルの内容を変更しないでください。変更した場合の動作は保証されません。

(4) 指定できるキー

指定できるキーは、「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」の「(4) 指定できるキー」にあるキー（ただし `ejb.public.directory` キーは除く）および `web.add.class.path` キーです。
`web.add.class.path` キーを指定した場合は、J2EE サーバ用オプション `add.class.path` キーとして動作します。

指定できるキーの `mserver.cfg` のデフォルト値を次の表に示します。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	デフォルト	VR
<code>add.jvm.arg</code>	指定されたオプションを使って JavaVM を起動します。 複数のオプションを指定する場合、同じキー名称を使用して、次のように複数回指定してください。 (指定例) <pre>add.jvm.arg= -Xms256m add.jvm.arg= -Xmx512m</pre> なお、次のように一つのキー名称に複数のオプションは指定できません。指定した場合、JavaVM の初期化に失敗します。 (指定例) <pre>add.jvm.arg=-Xms256m -Xmx512m</pre> <code>add.jvm.arg</code> で指定できる JavaVM のオプションについては、「14.1 JavaVM 拡張オプションの一覧」および「14.5 Application Server で指定できる Java HotSpot VM のオプション」を参照してください。	デフォルト値については「 8.2.7(5) Management Server での JavaVM オプションのデフォルト値 」を参照してください。	—
<code>ejb.server.corefilenum</code> (UNIX 用)	Management Server の再起動時に残す core ファイルの数を 0~16 の整数で指定します。なお、このキーは、UNIX 用です。 次の対象 core ファイルのうち、作成日時が新しいものから指定した数のファイルを残し、それ以外のファイルは削除されます。 対象 core ファイル <pre><Application Server のインストールディレクトリ>/manager/containers/m/ejb/ <Management Server のサーバ名>/core*</pre> - 再起動時にすべての core ファイルを削除する場合は、0 を指定します。 - 手動による運用または Management Server を使用する場合は、1 以上を指定します。 対象 core ファイルのうち、上書き、または削除される可能性のある core ファイルについては再起動時にリネームされます。	1	—

キー名称	内容	デフォルト	VR
	Linux（プロセス ID 付きの設定）の場合 core.<core 生成プロセス ID> → core.<core 生成プロセス ID>.<core ファイル作成日時> 上記以外の場合 core → core.<core ファイル作成日時> <core ファイル作成日時>の形式は、 yymmddHHMMSS です。 なお、core ファイルのリネーム、または削除に失敗した場合はメッセージ KDJE40047-E を出力し、Management Server 起動処理を中止して異常終了します。 また、次のことに注意してください。 - 対象 core ファイルと同じ名称のファイルを作成した場合はユーザが作成したファイルが削除されます。 - core ファイルの削除は Management Server 再起動時に処理されるため、再起動するまでは定義数以上の core ファイルが出力されている場合があります。 - 定義数+n 分の core ファイルが残るためディスク容量に注意してください（n:再起動されるまでに Management Server が出力する core ファイル数）。		

(凡例)

－：08-00 より前のバージョンを示します。

(5) Management Server での JavaVM オプションのデフォルト値

Management Server での Java VM オプションのデフォルト値を示します。

- -Xms128m
- -Xmx256m
- -XX:MetaspaceSize=64m
- -XX:MaxMetaspaceSize=64m
- -XX:SurvivorRatio=8
- -XX:HitachiJavaLog:<ejb.server.log.directory>
- -XX:HitachiJavaLogFileSize=4m
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:-HitachiThreadDumpToStdout
- -XX:+HitachiOutOfMemoryAbort

- -XX:+HitachiJavaClassLibTrace
- -XX:HitachiJavaClassLibTraceLineSize=1024
- -XX:+HitachiLocalsSimpleFormat
- -XX:+HitachiTrueTypeInLocals
- -XX:+HitachiLocalsInStackTrace
- -XX:+HitachiVerboseGC
- -XX:+HitachiVerboseGCPrintCause
- -XX:+HitachiOutputMilliTime
- -XX:+HitachiUseExplicitMemory
- -XX:HitachiExplicitHeapMaxSize=64m
- -XX:HitachiExplicitMemoryLogLevel:normal
- -XX:HitachiExplicitMemoryJavaLog:<ejb.server.log.directory>
- -XX:HitachiExplicitMemoryJavaLogFileSize=4m
- -XX:-HitachiAutoExplicitMemory
- -XX:+HitachiExplicitMemoryAutoReclaim
- -XX:-HitachiExplicitMemoryCompatibleToV8
- -XX:+HitachiOutOfMemoryHandling
- -Djava.util.Arrays.useLegacyMergeSort=true
- -XX:-CompactStrings

(6) 注意事項

- add.class.path キーに指定されている値の中に、<cosminexus.home>タグを使用してコンテナ拡張ライブラリ用の JAR を指定しているものがあります。これは Application Server のインストール先を意味しているため、add.class.path キーの記述内容を変更する必要はありません。なお、add.class.path キーにコンテナ拡張ライブラリを追加する場合は、インストールおよびアンインストール時の動作を保証できなくなるおそれがあるので、<cosminexus.home>タグを使用しないでください。

8.2.8 mserverenv.cfg (Management Server 用環境変数定義ファイル)

(1) 形式

次のようにキーを指定します。

<キー名称>=<値>

指定方法

- キーと値の間のセパレータには「=」を使用します。「=」前後の空白は無視されます。
- 改行までが値になります。改行文字は OS に従います。
- 1 行の最大サイズは 65536 バイトとします。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。

(2) ファイルの格納先

- Windows の場合

<Application Serverのインストールディレクトリ>%manager%config%

- UNIX の場合

/opt/Cosminexus/manager/config/

(3) 機能

Management Server を操作（起動、停止およびセットアップ）するときの環境変数を指定します。

(4) 指定できるキー

指定できるキーとデフォルト値を次に示します。

キー名称	内容	デフォルト値
add.env	環境変数を次の形式で設定します。 <環境変数名>=<環境変数値> 複数のキーを指定できます。 環境変数値には特殊変数※も指定できます。 (例) add.env=AAAHOME=C:%aaa add.env=BBBHOME=C:%bbb 環境変数 AAAHOME の値を C:%aaa に、BBBHOME の値を C: %bbb に設定します。	なし

注※

指定できる特殊変数を次に示します。

特殊変数	内容
\${cosminexus.home}	Application Server のインストールディレクトリ
%{XXX}	環境変数 XXX の値

(5) 注意事項

- 次の環境変数は、Management Server を起動したときに J2EE サーバが上書きするため、設定した値は有効となりません。
 - PATH
 - 共有ライブラリ検索パス（UNIX の場合だけ）
 - CLASSPATH
 - HVI_TRACEPATH
 - HVI_COMTFILECOUNT
 - HVI_COMTENTRYCOUNT
 - TPJDIR
- LANG の値をデフォルト（LANG=ja_JP.utf8）から変更した場合、動作は保障されません。変更した場合、ログの出力で文字化けなどの問題が発生するおそれがあります。

8.2.9 manager.cfg (Manager 設定ファイル)

(1) 形式

次のようにキーを指定します。

`<キー名称>=<値>`

指定方法

- キーと値の間のセパレータには「=」を使用します。「=」前後の空白は無視されます。
- 改行までが値になります。改行文字は OS に従います。
- 1 行は 1KB（1024 バイト）未満とします。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 行頭行末の空白文字や空行は無視されます。
- 同じキーを複数指定した場合は、最後のキーの値を有効とします。
- ASCII 文字だけ使用できます。
- ファイルセパレータには「/」を使用します。
- キーや値の一部に「¥」は使用できません。
- パスセパレータは OS に従います。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config%
- UNIX の場合
 /opt/Cosminexus/manager/config/

(3) 機能

Management Server, 運用管理エージェント, それぞれの上で動作するコンポーネントやコマンドに共通の設定を行います。

統合ログとして Manager の統合メッセージログ, 統合トレースログ, コマンド保守ログがあります。

(4) 指定できるキー

指定できるキーとデフォルト値を次に示します。

キー名称	内容	指定可能値	デフォルト	VR
com.cosminexus.manager.log.dir	Manager のログ出力ディレクトリを指定します。 ファイルセパレータには「/」を使用してください。 半角英数字, アンダースコア (_), 空白 (), ハイフン (-) 以外の文字を含むパスを指定しないでください。 なお, 指定したログ出力ディレクトリが不正な場合は, Manager は起動されません。J2EE サーバやバッチサーバ, Web コンテナサーバ (互換機能) とログ出力先が, 同じディレクトリにならないように注意してください。同じディレクトリを指定した場合, 動作は保証されません。	ディレクトリ名 (絶対パス)	<Application Server のインストールディレクトリ>/manager/log	07-00
com.cosminexus.manager.messagelog.size	統合メッセージログファイル一つ当たりの上限サイズ (バイト) を, 4096~16777216 で指定します。範囲外の値を指定した場合は, 262144 を仮定します。 なお, 次の操作をした後に, 設定値を変更しても変更は有効になりません。 • Management Server の起動 • 運用管理エージェントの起動 • Manager が提供するコマンドの実行 値を変更する場合は, バックアップ後に次のディレクトリを削除してください。 • <Manager のログ出力ディレクトリ>/message	4096~16777216 の整数で指定します。	262144	07-00
com.cosminexus.manager.messagelog.fnum	統合メッセージログファイルの面数を, 1~64 で指定します。	1~64 の整数で指定します。	4	07-00

キー名称	内容	指定可能値	デフォルト	VR
	なお、次の操作をした後に、設定値を変更しても変更は有効になりません。 • Management Server の起動 • 運用管理エージェントの起動 • Manager が提供するコマンドの実行 値を変更する場合は、バックアップ後に次のディレクトリを削除してください。 • <Manager のログ出力ディレクトリ>/message			
com.cosminexus.manager.messagelog.time	Manager 統合メッセージログファイルをローテーションする時刻を、000000～235959 の HHMMSS 形式で指定します。ただし、このプロパティで設定した時刻に達する前に、ログファイルのサイズが上限に達した場合、その時点でログファイルをローテーションします。 指定がない場合は、ログサイズだけでローテーションします。	000000～235959 の HHMMSS 形式で指定します。	なし	09-00
com.cosminexus.manager.messagelog.style	Manager 統合メッセージログファイルのローテーション方式を指定します。 SHIFT を指定した場合： シフトモードでローテーションします。 WRAP を指定した場合： ラップアラウンドモードでローテーションします。	次のどちらかを指定します。 • SHIFT • WRAP	WRAP	09-00
com.cosminexus.manager.tracelog.size	統合トレースログファイル一つ当たりの上限サイズ（バイト）を、4096～16777216 で指定します。範囲外の値を指定した場合は、1048576 を仮定します。 なお、次の操作をした後に、設定値を変更しても変更は有効になりません。 • Management Server の起動 • 運用管理エージェントの起動 • Manager が提供するコマンドの実行 値を変更する場合は、バックアップ後に次のディレクトリを削除してください。 • <Manager のログ出力ディレクトリ>/trace	4096～16777216 の整数で指定します。	1048576	07-00
com.cosminexus.manager.tracelog.fnum	統合トレースログファイルの面数を、1～64 で指定します。 なお、次の操作をした後に、設定値を変更しても変更は有効になりません。 • Management Server の起動 • 運用管理エージェントの起動 • Manager が提供するコマンドの実行	1～64 の整数で指定します。	4	07-00

キー名称	内容	指定可能値	デフォルト	VR
	値を変更する場合は、バックアップ後に次のディレクトリを削除してください。 ・ <Manager のログ出力ディレクトリ>/trace			
com.cosminexus.manager.tracelog.time	統合トレースログファイルをローテーションする時刻を、HHMMSS の形式で指定します。ただし、このプロパティで設定した時刻に達する前に、ログファイルのサイズが上限に達した場合は、その時点でログファイルをローテーションします。 指定がない場合は、ログサイズだけでローションします。	000000～235959 の HHMMSS 形式で指定します。	なし	09-00
com.cosminexus.manager.tracelog.style	統合トレースログファイルのローテーション方式を指定します。 SHIFT を指定した場合： シフトモードでローテーションします。 WRAP を指定した場合： ラップアラウンドモードでローテーションします。	次のどちらかを指定します。 ・ SHIFT ・ WRAP	WRAP	09-00
com.cosminexus.manager.cmdtracelog.size	コマンド保守ログファイル一つ当たりの上限サイズ（バイト）を、4096～16777216 で指定します。範囲外の値を指定した場合は、16777216 を仮定します。 なお、次の操作をした後に、設定値を変更しても変更は有効になりません。 ・ Manager が提供するコマンドを実行 値を変更する場合は、バックアップ後に次のディレクトリとログファイルを削除してください。 ・ <Manager のログ出力ディレクトリ>/maintenance/mmap ・ <Manager のログ出力ディレクトリ>/maintenance/mngcmd[n].log	4096～16777216 の整数で指定します。	16777216	07-00
com.cosminexus.manager.cmdtracelog.fnum	コマンド保守ログファイルの面数を、1～64 で指定します。 なお、次の操作をした後に、設定値を変更しても変更は有効になりません。 ・ Manager が提供するコマンドを実行 値を変更する場合は、バックアップ後に次のディレクトリとログファイルを削除してください。 ・ <Manager のログ出力ディレクトリ>/maintenance/mmap ・ <Manager のログ出力ディレクトリ>/maintenance/mngcmd[n].log	1～64 の整数で指定します。	4	07-00
com.cosminexus.manager.log.compatible	Manager のログの上位互換性を指定します。	次のどちらかを指定します。	true	07-00

キー名称	内容	指定可能値	デフォルト	VR
	true を指定した場合： Manager の統合ログによって不要になった ファイルを出力します。 false を指定した場合： Manager の統合ログによって不要になった ファイルを出力しません。	• true • false		

8.2.10 maction.properties (Management アクション実行用プロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

Management Server のプロパティファイルです。

(2) ファイルの格納先

- Windows の場合
<Application Serverのインストールディレクトリ>%manager%config%
- UNIX の場合
/opt/Cosminexus/manager/config/

(3) 機能

Management イベントで通知されるメッセージ ID に対応する Management アクションを定義します。

(4) 指定できるキー

指定できるキーとその内容および指定を省略した場合のデフォルト値を次に示します。

キー名称	内容	分類	デフォルト値
maction.<Management アクション ID>.command	Management アクションとして実行するコマンドを絶対パスで指定します。＊1	Management アクションの定義	なし
maction.<Management アクション ID>.timeout	実行するコマンドプロセスに対するタイムアウトを、0～86400 の整数（単位：秒）で指定します。 0 を指定した場合は、タイムアウトしません。	Management アクションの定義	10
maction.<Management アクション ID>.suppress	タイムアウト発生時の、実行したコマンドプロセスに対する操作を指定します。	Management アクションの定義	false

キー名称	内容	分類	デフォルト値
ID>.timeout.forced_stop	true を指定した場合： コマンドプロセスを強制終了します。 false を指定した場合： 何もしません。 強制終了する場合、コマンドプロセスが作成したサブプロセスについては保証されません。		
maction.<Management アクション ID>.exclusive_time	Management アクション実行開始後から、同一 Management アクション ID の Management アクションの実行を抑止する時間（単位：秒）を 0～86400 の整数で指定します。 0 を指定した場合、抑止しません。	Management アクションの定義	0
maction.<Management アクション ID>.max_executable_actions	同一 Management アクション ID の Management アクションの同時実行数の最大値を 0～1000 の整数で指定します。 0 を指定した場合、同時実行数を制限しません。	Management アクションの定義	0
maction.message.<メッセージ ID>.mactions	Management イベントで通知するメッセージ ID に対する Management アクション ID を指定します。英数字と区切り文字としてのコンマ (,) が使用できます。 ※2 複数指定する場合はコンマ (,) で区切ります。 指定した Management アクションに対してコマンドが設定されていない場合、Management アクションは実行されません。	メッセージ ID と Management アクションのマッピング	なし
maction.server.<論理サーバ名>.mactions	論理サーバに対応する Management アクション ID を指定します。英数字と区切り文字としてのコンマ (,) が使用できます。 ※2 複数指定する場合はコンマ (,) で区切ります。 キー名称の<論理サーバ名>には、J2EE サーバ、J2EE サーバクラスタ、および J2EE サーバクラスタのメンバーが指定できます。指定した Management アクションに対してコマンドが設定されていない場合、Management アクションは実行されません。	論理サーバと Management アクションのマッピング	なし
maction.tier.<Web システム名>.<物理ティア種別名>.mactions	物理ティアに対応する Management アクション ID を指定します。英数字と区切り文字としてのコンマ (,) が使用できます。 ※2 複数指定する場合はコンマ (,) で区切ります。 指定した Management アクションに対してコマンドが設定されていない場合、Management アクションは実行されません。	論理サーバと Management アクションのマッピング	なし
maction.unit.<Web システム名>.<サービスユニット名>.mactions	サービスユニットに対応する Management アクション ID を指定します。英数字と区切り文字としてのコンマ (,) が使用できます。 ※2 複数指定する場合はコンマ (,) で区切ります。	論理サーバと Management アクションのマッピング	なし

キー名称	内容	分類	デフォルト値
	指定した Management アクションに対してコマンドが設定されていない場合、Management アクションは実行されません。		
maction.mevent.receive_info.keep_size.max	Management イベントの受信情報を保持する最大数を、0～2147483647 の整数で指定します。0 を指定した場合、Management イベント受信情報を保持しません。Management イベント受信情報を保持することで、ネットワークの高負荷や障害による Management アクションの二重実行を防止することができます。Management イベント受信情報保持数が最大数を超えた場合、保持している受信情報のうち最も古い受信情報を削除します。Management イベント受信情報は、次の場合に破棄されます。 • 任意の Management イベント受信時に、Management イベントの再送期限が過ぎた受信情報を破棄 • Management Server 停止時にすべての受信情報を破棄	Management イベント受信時の設定	100

注※1

Management アクションとして実行するコマンドは Management Server が実行します。また、Management Server で設定された環境変数を引き継ぎます。

コマンドファイルのサンプルは、マニュアル「アプリケーションサーバ 機能解説 運用／監視／連携編」の「9.4.6 Management アクション実行コマンドの設定」を参照してください。

注※2

コンマ (,) 前後の空白文字は無視されます。

(5) 設定例

maction.properties ファイルの内容を次に示します。

• Windows の場合

```
# Managementアクションの定義
maction.restart.command=c%:%tmp%command1.bat
maction.restart.timeout=12
maction.restart.timeout.forced_stop=true
maction.restart.exclusive_time=60
maction.restart.max_executable_actions=1

# メッセージIDとManagementアクションのマッピング
maction.message.KDJE11111-E.mactions=restart
maction.message.KDJE22222-E.mactions=restart

# 論理サーバとManagementアクションのマッピング
maction.server.j2ee1.mactions=restart
maction.server.j2ee2.mactions=restart
maction.server.j2eeClstr1.mactions=restart
```

• UNIX の場合

```
# Managementアクションの定義
maction.restart.command=/tmp/command1.sh
maction.restart.timeout=12
maction.restart.timeout.forced_stop=true
maction.restart.exclusive_time=60
maction.restart.max_executable_actions=1

# メッセージIDとManagementアクションのマッピング
maction.message.KDJE11111-E.mactions=restart
maction.message.KDJE22222-E.mactions=restart

# 論理サーバとManagementアクションのマッピング
maction.server.j2ee1.mactions=restart
maction.server.j2ee2.mactions=restart
maction.server.j2eeClstr1.mactions=restart
```

(6) 注意事項

- Management アクション指定重複時の優先

次のプロパティキーについて、指定する論理サーバ、サービスユニットおよび物理ティアには J2EE サーバ < J2EE サーバクラスタ < サービスユニット < 物理ティア という包含関係が存在します。

- maction.server.<論理サーバ名>.mactions
- maction.unit.<Web システム名>.<サービスユニット名>.mactions
- maction.tier.<Web システム名>.<物理ティア種別名>.mactions

そのため、包含関係を持つ論理サーバ（例えば、J2EE クラスタと、その J2EE クラスタの要素となる J2EE サーバ）それぞれに対して別々の Management アクションが指定された場合、次の優先順でどれか一つの Management アクションが実行されます。

1. Management イベントを発行した J2EE サーバ
2. Management イベントを発行した J2EE サーバを含む J2EE サーバクラスタ
3. Management イベントを発行した J2EE サーバを含むサービスユニット
4. Management イベントを発行した J2EE サーバを含む物理ティア

また、メッセージ ID と Management アクションのマッピング、論理サーバと Management アクションのマッピングで Management アクションの指定順序が異なる場合、メッセージ ID と Management アクションのマッピング（maction.message.<メッセージ ID>.mactions キー）に指定された Management アクション ID の順番を優先します。

（例）"act1"を優先する例

```
maction.message.KDJE99999-E.mactions=act1, act2
maction.server.J2EE01.mactions=act3, act2, act1
```

なお、この例では、「KDJE99999-E」がメッセージ ID、「J2EE01」が論理サーバです。

- 標準出力／標準エラー出力について

Management アクションとして実行するコマンドから出力される標準出力／標準エラー出力については Management Server では取得しません。そのため、コマンドの標準出力／標準エラー出力を取得する場合は、コマンドの中でファイルに出力する必要があります。

8.2.11 Management イベント発行用プロパティファイル

(1) 形式

J2SE のプロパティファイル形式です。

Management Server のプロパティファイルです。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config%mevent.<論理サーバ名>.properties
- UNIX の場合
 /opt/Cosminexus/manager/config/mevent.<論理サーバ名>.properties

(3) 機能

Management イベント送信元の論理サーバ、送信先の Management Server、通知するメッセージ ID リストファイルなどを定義します。

(4) 指定できるキー

指定できるキーとその内容および指定を省略した場合のデフォルト値を次に示します。

キー名称	内容	デフォルト値
manager.mevent.send.host	送信先ホストを指定します。 Management イベントの送信先である Management Server のホスト名または IP アドレスを指定します。英数字および記号が使用できます。 指定がない場合、Management イベントを発行しません。	なし
manager.mevent.send.port	送信先ポート指定します。 Management イベントの送信先である Management Server の HTTP ポート番号を、1～65535 の整数で指定します。 指定がない、または不正である場合は Management イベントを発行しません。	なし
manager.mevent.logical_server_name	Management イベントの発行元となる論理サーバ名を指定します。英数字および記号が使用できます。 指定がない場合、Management イベントを発行しません。	なし

キー名称	内容	デフォルト値
	存在しない論理サーバ名を指定した場合、Management イベントは無視されます。	
manager.mevent.send.timeout	送信タイムアウトを指定します。 Management イベントの発行時に、Management Server からの応答を待つ時間を、10～600 の整数（単位：秒）で指定します。 指定時間内に応答がない場合、Management イベント発行失敗と見なします。	90
manager.mevent.retry.limit	送信期限を指定します。 Management イベントの発行に失敗した場合の再送を繰り返す期限を、0～86400 の整数（単位：秒）で指定します。 Management イベントの発行は、発行に成功するか、このキーに指定された時間が過ぎるまで、manager.mevent.retry.interval キーで指定された再送間隔で繰り返します。 0 が指定された場合は、Management イベントを再送しません。	0
manager.mevent.retry.interval	Management イベントを再送する場合の再送間隔を、1～86400 の整数（単位：秒）で指定します。	10
manager.mevent.message_id.list	Management イベント発行用メッセージ ID リストファイルを絶対パスで指定します。 指定されたファイルが不正である場合は Management イベントを発行しません。 省略された場合は、デフォルトの Management イベント発行用メッセージ ID リストファイルに登録されているメッセージ ID を Management イベントとして通知します。 Management イベント発行用メッセージ ID リストファイルについては、 [8.2.12 Management イベント発行用メッセージ ID リストファイル] を参照してください。	なし
manager.mevent.send.max	Management イベントの同時発行最大数を、1～1000 の整数で指定します。	10
manager.mevent.sender.bind.host	Management イベント送信時に使用するローカルアドレスのホスト名または IP アドレスを指定します。英数字および記号が使用できます。 省略した場合、任意のローカル IP アドレスが割り当てられます。	なし

なお、ファイルを作成する場合は、次の Management イベント発行用プロパティファイルの雛型を参照してください。

- Windows の場合
`<Application Serverのインストールディレクトリ>%manager%config%templates%mevent.properties`
- UNIX の場合
`/opt/Cosminexus/manager/config/templates/mevent.properties`

(5) 注意事項

- Management イベント発行機能の設定ファイルは、J2EE サーバインスタンスごとに指定する必要があります。同時に動作する複数の J2EE サーバインスタンスの間で同じファイルを指定しないでください。

- Management イベントは、Management イベント発生順に発行されます。ただし、発行に失敗した Management イベントは、予定されている Management イベントの発行後に再発行されます。

8.2.12 Management イベント発行用メッセージ ID リストファイル

(1) 形式

次の形式で指定します。

[<インジケータ>]<メッセージ ID>

- 1 行に一つのメッセージ ID を記述します。
- インジケータにシャープ (#) を指定した場合、その行はコメントとみなされます。
- インジケータにプラス (+) を指定、またはインジケータを省略した場合、該当メッセージ ID を Management イベントとして通知します。
- インジケータにハイフン (-) を指定した場合、該当メッセージ ID を Management イベントとして通知しません。デフォルトの Management イベント発行用メッセージ ID リストに記載されたメッセージ ID を Management イベントとして通知しない場合に使用します。
- 行頭行末の空白文字や空行は無視されます。
- 同一のメッセージ ID を複数指定した場合、後方に記述されたメッセージ ID が有効になります。
- メッセージ ID がプラス (+) またはハイフン (-) で始まる場合、インジケータは必ず指定してください。

(2) ファイルの格納先

格納先およびファイル名称は任意です。ファイル名には次の文字が使用できます。

英数字、ピリオド (.), 円符号 (¥), コロン (:), プラス (+), またはハイフン (-)

(3) 機能

Management イベントとして通知するメッセージ ID を定義します。

(4) デフォルト

Management イベントとして通知するメッセージ ID のデフォルトを次に示します。

表 8-2 Management 発行用メッセージ ID (デフォルト)

機能	監視対象	メッセージ ID
リソース枯渇監視	メモリ使用状況	KDJE34500-W

機能	監視対象	メッセージ ID
	ファイルディスクリプタ数	KDJE34520-W
	スレッド数	KDJE34540-W
	スレッドダンプファイル数	KDJE34580-W KDJE34581-E
	HTTP リクエスト実行待ちキュー（Web アプリケーション単位およびデフォルトの実行待ちキュー）	KDJE34621-W
	HTTP セッション数	KDJE34640-W
	コネクションプールの使用状況	KDJE34660-W KDJE34661-W
J2EE アプリケーション実行時間監視	J2EE アプリケーション実行時間	KDJE52702-W KDJE52703-W KDJE52705-W KDJE52713-E
コネクション障害検知	コネクション障害検知のタイムアウト	KDJE48602-W
	コネクション管理スレッド	KDJE48603-W
FullGC 回数の監視	FullGC 回数	KDJE53850-W
URL グループ単位の実行待ちリクエスト数の監視	URL グループ単位の実行待ちリクエスト数の上限しきい値	KDJE53860-W
	URL グループ単位の実行待ちリクエスト数の下限しきい値	KDJE53861-I
クラスタコネクションプール（互換機能）	コネクションプールの状態	KDJE49650-I KDJE49653-I KDJE49655-E KDJE49657-E KDJE49660-I KDJE49663-E KDJE49664-E KDJE49669-E
	コネクション管理スレッド	KDJE49671-I
Web コンテナ単位の全体実行待ちリクエスト数の監視	Web コンテナ単位の全体実行待ちリクエスト数	KDJE53862-W KDJE53863-I KDJE53864-W KDJE53865-I KDJE53866-W KDJE53867-I KDJE53868-W KDJE53869-I

(5) サンプル

サンプルを次に示します。

KDJE34580-W と KDJE34660-W を Management イベントとして通知しない場合

```
# Monitoring of resources
# : Status of memory(Java Heap)
KDJE34500-W

# : Number of file descriptors
KDJE34520-W

# : Number of threads
KDJE34540-W

# : Number of thread dump files
-KDJE34580-W
KDJE34581-E

# : Number of HTTP requests in queue
KDJE34621-W

# : Number of HTTP sessions
KDJE34640-W

# : Status of connection pool
-KDJE34660-W
KDJE34661-W

# Monitoring of execution time of user program
KDJE52702-W
KDJE52703-W
KDJE52705-W
KDJE52713-E
```

なお、ファイルを作成する場合は、次の Management イベント発行用メッセージ ID リストファイルの雛型を参照してください。

- Windows の場合
〈Application Serverのインストールディレクトリ〉\manager\config\templates\mevent.midlist.conf
- UNIX の場合
/opt/Cosminexus/manager/config/templates/mevent.midlist.conf

(6) 注意事項

- 次のメッセージ ID は Management イベントの発行対象外です。これらのメッセージ ID を指定した場合、その指定は無視されます。
 - KDJE90001-E
 - KDJE90002-E

- KDJE90003-E
 - KDJE90005-W
 - KDJE90006-W
 - KDJE90009-W
- 指定可能なメッセージ ID は、Management イベント発行機能と連携する J2EE サーバやバッチサーバが出力する特定のメッセージ ID だけです。それ以外のシステムメッセージ ID、J2EE アプリケーション、またはバッチアプリケーションから出力されるメッセージ ID を指定しても Management イベントは発行されません。

8.2.13 Management Server 管理ファイル用退避対象定義ファイル

(1) 形式

次の形式で指定します。

<ファイル名>

- 絶対パスで指定します。
- シャープ (#) で始まる行はコメントとみなされます。
- ファイルセパレータはスラッシュ (/) を使用します。
- 指定されたパスにファイルが存在しない場合、該当する行を無視して処理を継続します。
- 使用できる変数は次の変数です。\$ はシステムプロパティを示します。指定する値には \$ を含めないでください。

`${cosminexus.home}` : Application Server のインストールディレクトリ

`${user.home}` : ユーザのホームディレクトリ

(2) ファイルの格納先

格納先およびファイル名称は任意です。

(3) 機能

運用管理サーバの実行系と待機系を 1:1 にして系切り替えをする場合、実行系の運用管理サーバの Management Server 管理ファイルとして収集、退避させるファイルを定義します。デフォルトで収集対象となっているファイル以外に収集したいファイルがある場合に指定します。

(4) デフォルト

Management Server 管理ファイルとしてデフォルトで収集されるファイルの対象ディレクトリを次に示します。

種類	対象ディレクトリ
定義ファイル格納ディレクトリ	- Windows の場合 <Application Server のインストールディレクトリ>/manager/config/ - UNIX の場合 /opt/Cosminexus/manager/config/
アプリケーション登録ディレクトリ	- Windows の場合 <Application Server のインストールディレクトリ>/manager/containers/m/j2eeapps/mngsvr/mportal/mapps/regapps/ - UNIX の場合 /opt/Cosminexus/manager/containers/m/j2eeapps/mngsvr/mportal/mapps/regapps/
リソースアダプタ登録ディレクトリ	- Windows の場合 <Application Server のインストールディレクトリ>/manager/containers/m/j2eeapps/mngsvr/mportal/mwebsystems/regras/ - UNIX の場合 /opt/Cosminexus/manager/containers/m/j2eeapps/mngsvr/mportal/mwebsystems/regras/
リポジトリ管理用構成情報定義結果格納ファイル	- Windows の場合 <Application Server のインストールディレクトリ>/manager/containers/m/j2eeapps/mngsvr/mportal/useradmin/config/data/editusersConfig.xml - UNIX の場合 /opt/Cosminexus/manager/containers/m/j2eeapps/mngsvr/mportal/useradmin/config/data/editusersConfig.xml

(5) 記述例

- /home/confdir/message1.conf を収集する場合

```
/home/confdir/message1.conf
```

- <Application Server のインストールディレクトリ>/manager/apps/MyApp.ear (Windows の場合), または /opt/Cosminexus/manager/apps/MyApp.ear (UNIX の場合) を収集する場合

```
${cosminexus.home}/manager/apps/MyApp.ear
```

8.2.14 .mngsvrutilrc (mngsvrutil コマンドのクライアント側定義ファイル)

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

各 OS ユーザのホームディレクトリ

(3) 機能

mngsvrutil コマンドのオプションのデフォルト値を設定できます。

(4) 反映のタイミング

設定ファイルの変更後，次回 mngsvrutil コマンドの発行時に更新情報が反映されます。

(5) 指定できるキー

クライアント側定義ファイルで指定できるキーとその内容および指定を省略した場合のデフォルト値を次に示します。

キー名称	内容	指定可能値	デフォルト	VR
mngsvrutil.connect.host	-m オプションに指定する Management Server のホスト名，およびポート番号を設定します。 (設定例) <pre>mngsvrutil.connect.host=localhost:28080</pre> - ホスト名 ドット記法での IP アドレス指定，または名前解決できるホスト名で指定します。 IP アドレスはメタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で指定された場合，利用可能なローカル IP アドレスのうちマッチする IP アドレスが使用されます。※1 - ポート番号 Management Server の設定に従います。デフォルト値は 28080 です。 注意事項 - ホスト単位管理モデルで運用する場合に，mserver.properties の webserver.connector.http.bind_host に値を指定しているときは，同じ値を指定してください。 - IP アドレスを表す正規表現にコロン(:)を指定しないでください。コロン(:)が指定された場合，コロン以降はポート番号とみなされます。	次のどちらかを指定します。 - ホスト名 - IPv4 アドレス	なし	06-00

キー名称	内容	指定可能値	デフォルト	VR
mngsvrutil.connect.password	-p オプションに指定するパスワードを設定します。なお、パスワードを設定していない場合、このキーは必要ありません。 (設定例) <pre>mngsvrutil.connect.password=admin</pre> 管理ユーザアカウントの省略機能が有効の場合、このプロパティの指定は無視されます。	Management Server の管理ユーザアカウントのパスワード	なし	06-00
mngsvrutil.connect.userid	-u オプションに指定するユーザ ID を設定します。 (設定例) <pre>mngsvrutil.connect.userid=admin</pre> 管理ユーザアカウントの省略機能が有効の場合、このプロパティの指定は無視されます。	Management Server の管理ユーザアカウントのユーザ ID	なし	06-00
mngsvrutil.output.file	-o オプションに指定する出力ファイル名を設定します。 (設定例) Windows の場合 <pre>mngsvrutil.output.file=C:/tmp/output.txt</pre> UNIX の場合 <pre>mngsvrutil.output.file=/tmp/output.txt</pre>	ファイル名 (絶対パス)	なし	06-00
mngsvrutil.output.format	-f オプションに指定する出力形式を設定します。 CSV 形式で出力する場合は csv, SNMP 連携形式で出力する場合は snmp を指定します。 (設定例) <pre>mngsvrutil.output.format=csv</pre>	指定できる文字列を次に示します。 • csv • snmp	csv	06-00
mngsvrutil.output.suppress_header	-h オプションを指定する場合は true を、指定しない場合は false を設定します。なお、大文字と小文字は区別されません。 (設定例) <pre>mngsvrutil.output.suppress_header=true</pre>	指定できる文字列を次に示します。 • true • false	false	06-00
mngsvrutil.target_name※2	-t オプションに指定する論理サーバ名、またはホスト名を指定します。 (設定例) <pre>mngsvrutil.target_name=myServer</pre> サブコマンドの対象となるターゲット名称 (論理サーバ名またはホスト名) を指定します。 運用管理ポータル「論理サーバの構成定義」の定義に従います。 IP アドレスはメタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で	次のどれかを指定します。 • IPv4 アドレス • ホスト名 • 論理サーバ名	なし	06-00

キー名称	内容	指定可能値	デフォルト	VR
	指定された場合、利用可能なローカル IP アドレスのうちマッチする IP アドレスが使用されます。※1 注意事項 -k オプションまたは mngsvrutil.target_kind に host を指定する場合に、mserver.properties の webserver.connector.http.bind_host に値を指定しているときは、同じ値を指定してください。			
mngsvrutil.target_kind	-k オプションに指定するターゲット種別を指定します。 (設定例) mngsvrutil.target_kind=host	指定できる文字列を次に示します。 - logicalServer - host	logicalServer	06-00
mngsvrutil.target_server_name※2	-t オプションに指定する論理サーバ名を設定します。このキーは互換用です。 mngsvrutil.target_name を使用することをお勧めします。mngsvrutil.target_name と同時に指定した場合は、mngsvrutil.target_name の指定が有効になります。 (設定例) mngsvrutil.target_server_name=myServer	英数字および記号で表記します。	なし	06-00

注 1

各キーが存在しない、または誤っている場合はデフォルト値を仮定します。

注 2

パスワードなどの情報が含まれるため、適切にファイルパーミッションを設定してください。

注 3

ASCII 文字以外の文字を記述する場合は、設定ファイルの文字エンコーディングと mngsvrutil コマンド実行時の OS ロケールの文字エンコーディングを合わせてください。また、日本語を使用する場合にエスケープシーケンスを含めた文字列で指定しないでください。

注※1

運用管理ドメイン内で運用管理エージェントとの通信に使用される IP アドレスのサブネットが一つに決められている場合、指定する IP アドレスを"192.168.0.128"のように記述しておけば、前方が"192.168.0."である IP アドレス("192.168.0.32"や"192.168.0.128"など)にマッチするため、設定ファイルを修正しないで、すべてのホストに配布して利用できます。正規表現についての詳細は、Java の java.util.regex.Pattern クラスの仕様を確認してください。

ただし、"1"は 1 文字の"1"に置き換えられてしまうため、"1"を指定する場合は 2 文字連続して指定してください。指定した正規表現にマッチする IP アドレスが複数個検出された場合は、最も小さい値の IP アドレスを採用します。例えば、"192.168.0.32"と"192.168.0.128"が検出された場合は"192.168.0.32"を採用します。この場合、採用された IP アドレスが意図する IP アドレスになるとは限らないため、"192.168.0.128"のように必ず 1 個だけマッチする正規表現を記述してください。

注※2

mngsvrutil.target_name キーと mngsvrutil.target_server_name キーの指定と、mngsvrutil コマンドの-k オプションおよび-t オプションの指定の関係については、マニュアル「アプリケーションサーバ リファレンス コマンド編」の「mngsvrutil (Management Server の運用管理コマンド)」を参照してください。

8.2.15 mngsvrutil.properties (mngsvrutil コマンドのサーバ側定義ファイル)

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

- Windows の場合
＜Application Serverのインストールディレクトリ＞¥manager¥config¥
- UNIX の場合
/opt/Cosminexus/manager/config/

(3) 機能

mngsvrutil コマンドの実行環境を設定します。

(4) 反映のタイミング

Management Server 起動中に設定ファイルを更新した場合、Management Server の再起動後に更新情報が反映されます。

(5) 指定できるキー

サーバ側定義ファイルで指定できるキーとその内容および指定を省略した場合のデフォルト値を次に示します。

キー名称	内容	デフォルト値
mngsvrutil.auth.failed_wait	管理ユーザの認証に失敗した場合に待機する秒数を指定します（指定できる値：0～2147483647）。 (設定例) mngsvrutil.auth.failed_wait=10	5
mngsvrutil.auth.log_challenged	HTTP によるアクセスがあったことをログに出力するかどうかを指定します。※1	true

キー名称	内容	デフォルト値
	true を指定した場合： 出力します。 false を指定した場合： 出力しません。 (設定例) <code>mngsvrutil.auth.log_challenged=false</code>	
<code>mngsvrutil.auth.log_denied</code>	HTTP によるアクセスを拒否したことをログに出力するかどうかを指定します。※1 true を指定した場合： 出力します。 false を指定した場合： 出力しません。 (設定例) <code>mngsvrutil.auth.log_denied=false</code>	true
<code>mngsvrutil.auth.log_failed</code>	管理ユーザの認証に失敗したことをログに出力するかどうかを指定します。※1 true を指定した場合： 出力します。 false を指定した場合： 出力しません。 (設定例) <code>mngsvrutil.auth.log_failed=false</code>	true
<code>mngsvrutil.auth.log_succeeded</code>	管理ユーザの認証に成功したことをログに出力するかどうかを指定します。※1 true を指定した場合： 出力します。 false を指定した場合： 出力しません。 (設定例) <code>mngsvrutil.auth.log_succeeded=false</code>	true
<code>mngsvrutil.compat.monitoring_tree</code>	アプリケーション管理方法が V8 モードの場合に、JP1/IM の監視ツリーの自動生成で取得する内容にアプリケーション情報を含めるかどうかを指定します。※1 true を指定した場合： J2EE アプリケーション情報、および J2EE サーバと J2EE アプリケーションとの関連情報を出力します。 false を指定した場合： J2EE アプリケーション情報、および J2EE サーバと J2EE アプリケーションとの関連情報を出力しません。 (設定例) <code>mngsvrutil.compat.monitoring_tree=true</code>	false

キー名称	内容	デフォルト値
mngsvrutil.session.max_inactive_interval	サーバがクライアントにセッションを生成して、クライアントから最後にリクエストを受けてからセッションを無効化するまでの最大の秒数を指定します（指定できる値：-2147483648～2147483647）※2。 （設定例） <pre>mngsvrutil.session.max_inactive_interval=10</pre>	5

注 1

各キーが存在しない、または誤っている場合はデフォルト値を仮定します。

注 2

Management Server 起動中に設定ファイルを更新した場合、Management Server の再起動後に更新情報が反映されます。

注※1

設定値（true, false）は、大文字、小文字の区別はしません。

注※2

負の値を指定した場合、セッションを無効化しません。

8.2.16 mngsvrutilcl.properties（mngsvrutil コマンドのクライアント側共通定義ファイル）

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config%
- UNIX の場合
 /opt/Cosminexus/manager/config/

(3) 機能

mngsvrutil コマンドのオプションのデフォルト値を共通定義として設定できます。

また、この定義ファイルを適用するユーザを制限できます。

なお、クライアント側定義ファイル（.mngsvrutilrc）と mngsvrutilcl.properties（mngsvrutil コマンドのクライアント側共通定義ファイル）の両方が設定されている場合、クライアント側定義ファイル（.mngsvrutilrc）が適用されます。

(4) 反映のタイミング

設定ファイルの変更後、次回 mngsvrutil コマンドの発行時に更新情報が反映されます。

(5) 指定できるキー

クライアント側共通定義ファイルで指定できるキーとその内容および指定を省略した場合のデフォルト値を次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
mngsvrutil.apply_user	mngsvrutil コマンドまたはアダプタコマンドを実行する時の OS ユーザで、クライアント側共通定義ファイルを適用するユーザを指定します。このキーで指定した OS ユーザが、mngsvrutil コマンドまたはアダプタコマンドを実行した場合だけ、これ以外に設定したキーが適用されます。 複数のユーザを指定する場合は、コンマ (,) で区切って指定します。 - Windows の場合 ドメイン名を含まないユーザ名を指定してください。ユーザ名が一致している場合、ログイン先がローカル、ドメインのどちらでも、このパラメタは有効になります。 大文字と小文字は区別されません。 また、Windows のサービスから mngsvrutil コマンドが呼び出される場合に、この定義ファイルを適用するには、「<ホスト名>\$」(<ホスト名>は、Management Server が稼働するホストのホスト名)を指定してください。 - UNIX の場合 大文字と小文字は区別されます。 SNMP 連携および JP1/IM 連携（監視ツリーの自動生成や Web システムの構成情報の収集）で、この定義ファイルを適用するには、「root」を指定してください。 (設定例) Windows の場合 mngsvrutil.apply_user=SYSTEM, Administrator UNIX の場合 mngsvrutil.apply_user=root, user01	OS に登録されているユーザ名	なし	06-70
.mngsvrutilrc (mngsvrutil コマンドのクライアント側定義ファイル) のキー	指定できるキーについては、 「 8.2.14 .mngsvrutilrc (mngsvrutil コマンドのクライアント側定義ファイル) 」を参照してください。	—	—	—

(凡例)
-：該当なし。

(6) 注意事項

クライアント側定義ファイル（.mngsvrutilrc）が存在しない場合，無条件にクライアント側共通定義ファイル（mngsvrutilcl.properties）を読み込むため，このファイルにアクセス権限がないと Warning が発生します。

mngsvrutil コマンドおよびアダプタコマンドを実行する OS ユーザには，クライアント側共通定義ファイルへのアクセス権限を付与し，mngsvrutil.apply_user キーでクライアント側共通定義ファイルを適用するユーザを選別することを推奨します。

8.2.17 .mngsvrmonitorrc（JP1/IM 連携用モニタ起動コマンドの設定ファイル）

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

JP1/IM-View を操作する OS ユーザのホームディレクトリ

(3) 機能

JP1/IM 連携をする場合に，JP1/IM の画面から Management Server の運用ポータルを起動するために必要な設定をします。なお，このファイルは，Windows 用です。

(4) 反映のタイミング

設定ファイルの変更後，次回 mngsvrmonitor コマンド実行時に更新情報が反映されます。

(5) 指定できるキー

モニタ起動コマンドの設定ファイルで指定できるキーとその内容および指定を省略した場合のデフォルト値を次に示します。

キー名称	内容	デフォルト値
mngsvrmonitor.browser	起動する Web ブラウザのコマンドをフルパスで指定します。必ず Microsoft Edge のコマンドをフルパスで指定してください。空白を含むパスを指定する場合はダブルクォーテーション（"）で囲んで指定してください。このキーを省略した場合は	レジストリに設定されている

キー名称	内容	デフォルト値
	Windows のレジストリに設定されているデフォルトブラウザが起動されますが、動作は保証されません（デフォルトブラウザが設定されていない場合はエラーになります）。 (設定例) <code>mngsvrmonitor.browser ="C:/Program Files (x86)/Microsof/Edge/Application/msedge.exe"</code>	るブラウザ
<code>mngsvrmonitor.connect.host</code>	Management Server のホスト名およびポート番号を次の形式で指定します。 ホスト名[:ポート番号] (設定例) <code>mngsvrmonitor.connect.host=h001:28080</code>	ポート番号: 28080
<code>mngsvrmonitor.connect.password</code>	運用管理ポータル「管理ユーザアカウントの設定」で設定した管理ユーザ ID に対応する管理ユーザパスワードを指定します。なお、パスワードを設定していない場合、このキーは必要ありません。 (設定例) <code>mngsvrmonitor.connect.password=p43269</code> 管理ユーザアカウントの省略機能が有効の場合、このプロパティの指定は無視されます。	なし
<code>mngsvrmonitor.connect.userid</code>	運用管理ポータル「管理ユーザアカウントの設定」で設定した管理ユーザ ID を指定します。このキーを省略した場合は運用管理ポータルのログイン画面を表示します。 (設定例) <code>mngsvrmonitor.connect.userid=admin</code> 管理ユーザアカウントの省略機能が有効の場合、このプロパティの指定は無視されます。	なし

8.2.18 setup.cfg (セットアップウィザード用設定ファイル)

(1) 形式

次のようにキーを指定します。

<キー名称>=<値>

指定方法

- 行頭に「#」を記述した場合、その行はコメントと見なされ、無視されます。
- 値を指定しない場合、その行は無視されます。
- 行頭および行末の空白文字や空行は無視されます。
- キーと値の間のセパレータには「=」を使用します。「=」前後の空白は無視されます。
- 改行までが値と見なされます。
- 同じキーを複数指定した場合は、最後のキーの値を有効とします。
- 1 行は 64KB 未満とします。

- ASCII 文字だけ使用できます。

(2) ファイルの格納先

- Windows の場合

<Application Serverのインストールディレクトリ>%manager%setup%config

- UNIX の場合

/opt/Cosminexus/manager/setup/config

(3) 機能

セットアップウィザードのログファイルの出力先, 最大ファイルサイズ, 最大ファイル面数などを指定します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
setup.log.dir	ログファイルの出力先ディレクトリを指定します。ファイルセパレータには「/」を使用してください。範囲外の値を指定した場合は、デフォルト値が仮定されます。 UNIX の場合、指定した出力先ディレクトリの上位のディレクトリが存在している必要があります。	ディレクトリ名（絶対パス）	• Windows の場合 <Application Serverのインストールディレクトリ>%manager%setup%log • UNIX の場合 /opt/Cosminexus/manager/setup/log	08-00
setup.log.filenum	ログファイル面数を指定します。範囲外の値を指定した場合は、デフォルト値が仮定されます。	1～16 の整数で指定します。	4	08-00
setup.log.filesize	ログファイル一つ当たりの上限サイズ（単位：バイト）を指定します。範囲外の値を指定した場合は、デフォルト値が仮定されます。	4096～2147483647の数値で指定します。	262144	08-00

キー名称	内容	指定可能値	省略値	VR
setup.maintenance.log.file lenum	保守ログのファイル面数を指定します。範囲外の値を指定した場合は、デフォルト値が仮定されます。	1～16 の整数 で指定します。	4	08-00
setup.maintenance.log.file lesize	保守ログのファイル一つ当たりの上限サイズ（単位：バイト）を指定します。範囲外の値を指定した場合は、デフォルト値が仮定されます。	4096～ 2147483647 の整数で指定 します。	16777216	08-00

(5) 注意事項

セットアップウィザード実行時に、setup.cfg ファイルの読み込みに失敗した場合は、メッセージ KEOS28056-E が出力され、セットアップウィザードは終了します。

8.2.19 論理ユーザーバ定義ファイル

(1) 形式

xml ファイル形式です。

(2) ファイルの格納先

格納先およびファイル名称は任意です。

(3) 機能

mnngsvrutil コマンドで論理ユーザーバを追加するときに作成します。

(4) 定義できる内容

論理ユーザーバ定義ファイルに記述できる内容を次の表に示します。

要素名	説明	デフォルト値
<user-server-definition>	名前空間として次のパスを指定します。http://www.cosminexus.com/mnngsvr/schema/UserServerDefinition-1.1	—
<service>	サービスの定義。 必ず指定してください。	—
<type>	起動方法を直接起動にするか間接起動にするかを指定します。 direct を指定した場合： 直接起動します。	direct

要素名		説明	デフォルト値
		indirect を指定した場合： 間接起動します。	
	<startup-order>	起動順序を-1～999の整数で指定します。-1を指定した場合は、指定なしとなります。	900
	<start-watch-time>	起動監視時間を次の数値（単位：秒）で指定します。0を指定した場合は、監視しません。 0, 60, 300, 600, 3600	60
	<stop-watch-time>	停止監視時間を次の数値（単位：秒）で指定します。0を指定した場合は、監視しません。 0, 60, 300, 600, 1800	60
	<force-watch-time>	強制停止監視時間を次の数値（単位：秒）で指定します。0を指定した場合は、監視しません。 0, 60, 300, 600, 1800	60
	<retry-count>	自動再起動回数を次の数値で指定します。 0, 1, 5, 10, 1000	1
	<retry-interval>	自動再起動リトライ間隔を次の数値（単位：秒）で指定します。0を指定した場合は、即時リトライします。 0, 60, 300, 600, 1800	60
	<watch-interval>	監視間隔を、1～86400の整数（単位：秒）で指定します。 なお、運用管理エージェントプロパティファイルの adminagent.userserver.watch.interval キーと同時に指定した場合は、このタグに指定した値が有効になります。	1
	<start-time-watch-interval>	起動時監視間隔を、1～86400の整数（単位：秒）で指定します。	1
	<env>	環境変数の定義。 複数指定できます。	—
	name	環境変数名を指定します。 必ず指定してください。	—
	value	環境変数値を指定します。 必ず指定してください。 環境変数の指定値に次の文字列「\${cosminexus.home}」が含まれている場合は、<Application Server のインストールディレクトリ>に置換されます。	—
	<working-dir>	ワーキングディレクトリを絶対パスで指定します。	<Application Server のインストールディレクトリ>/manager/bin
	<user-id>	コマンド実行時のユーザ ID を指定します。	—

要素名		説明	デフォルト値
		この要素は、UNIX 用です。	
	<group-id>	コマンド実行時のグループ ID を指定します。 この要素は、UNIX 用です。	—
	<command>	コマンドの定義 必ず指定してください。また、複数指定できます。	—
	type	コマンド種別※ ¹ を指定します。指定できる値は、次のとおりです。 • start：サーバ起動用コマンド • stop：サーバ停止用コマンド • forceStop：サーバ強制停止用コマンド • getProcessID：プロセス ID 取得用コマンド • isAlive：サービス監視用コマンド 必ず指定してください。	—
	timeout	タイムアウト時間を、1～86400 の整数（単位：秒）で指定します。 ただし、コマンド種別が getProcessID または isAlive のときだけ有効となります。	180
	<arg>	コマンド引数を指定します。※ ² 必ず指定してください。 コマンドに引数および引数値を付加する場合は、<arg>タグを複数指定してください。 <arg>タグの指定例を次に示します。 <arg>コマンド</arg> <arg>引数 1</arg> <arg>引数 1 の指定値</arg> <arg>引数 2（値の指定なし）</arg> <arg>引数 3</arg> <arg>引数 3 の指定値</arg>	—
	<server>	論理ユーザサーバ定義。 必ず指定してください。また、複数指定できます。	—
	<logical-server-name>	論理サーバ名を 128 文字以下で指定します。使用できる文字は半角文字の英数字（A～Z, a～z, 0～9）、アンダースコア（_）、およびハイフン（-）です。 必ず指定してください。	—
	<display-name>	論理サーバの表示名を 128 文字以下で指定します。	—
	<description>	論理サーバの説明を 1024 文字以下で指定します。	—
	<host-ref>	あらかじめ定義されたホスト名を 255 文字以下で指定します。使用できる文字は半角文字の英数字（A～Z, a～z, 0～9）、アンダースコア（_）、ハイフン（-）およびピリオド（.）です。	—

要素名	説明	デフォルト値
	必ず指定してください。	

注※1

起動種別によってコマンド種別の指定条件が異なります。起動種別によるコマンド種別の指定条件と要件を次に示します。

起動種別	コマンド種別	指定条件	要件	備考
直接起動	start	必須	サーバ起動用コマンドが監視対象となります。	—
	stop	任意	サーバ停止用コマンドの正常終了時には、監視対象プロセスが終了しています。	指定していない場合は次の動作をします。 - Windows の場合 該当プロセスに対し Ctrl-C イベントを発行します。プロセスが停止しない場合は、TerminateProcess() を実行します。 - UNIX の場合 該当プロセスに対し SIGTERM を送信します。その 1 秒後にプロセスが停止しない場合は、SIGKILL を送信します。
	forceStop	任意	- サーバ強制停止用コマンドの終了時には、監視対象プロセスが終了しています。 - サーバ強制停止用コマンド発行後は、サーバ起動用コマンドが実行可能状態となります。	指定していない場合は次の動作をします。 - Windows の場合 該当プロセスに対し Ctrl-C イベントを発行します。プロセスが停止しない場合は、TerminateProcess() を実行します。 - UNIX の場合 該当プロセスに対し SIGTERM を送信します。その 1 秒後にプロセスが停止しない場合は、SIGKILL を送信します。
	getProcessID	指定不可	—	指定した場合はエラーとなります。
	isAlive	任意	サービスが稼働中の場合は終了コードが 0、稼働中ではない場合は終了コードが 0 以外となります。	指定した場合はサービス監視を行います。
間接起動	start	必須	- プロセス起動用コマンドの正常終了時には、すべての監視対象プロセスが起動完了します。 - プロセス起動用コマンドの正常終了時には、プロセス停止用コマンドが実行可能状態となります。	—

起動種別	コマンド種別	指定条件	要件	備考
			正常終了の場合は終了コードが 0, 異常終了の場合は終了コードが 1 となります。	
	stop	任意	サーバ停止用コマンドの正常終了時には、監視対象プロセスが終了しています。	指定していない場合は次の動作をします。 - Windows の場合 該当プロセスに対し Ctrl-C イベントを発行します。プロセスが停止しない場合は、TerminateProcess() を実行します。 - UNIX の場合 該当プロセスに対し SIGTERM を送信します。その 1 秒後にプロセスが停止しない場合は、SIGKILL を送信します。
	forceStop	任意	- サーバ強制停止用コマンドの終了時には、監視対象プロセスが終了しています。 - サーバ強制停止用コマンド発行後は、サーバ起動用コマンドが実行可能状態となります。	指定していない場合は次の動作をします。 - Windows の場合 該当プロセスに対し Ctrl-C イベントを発行します。プロセスが停止しない場合は、TerminateProcess() を実行します。 - UNIX の場合 該当プロセスに対し SIGTERM を送信します。その 1 秒後にプロセスが停止しない場合は、SIGKILL を送信します。
	getProcessID	必須	- サーバ起動用コマンドで起動したプロセスのプロセス ID を 10 進表現で標準出力に出力します。 - 複数のプロセスが起動される場合はすべてのプロセスのプロセス ID を出力します。 - 正常終了の場合は終了コードが 0, 異常終了の場合は終了コードが 1 となります。 - 出力フォーマットは次のようになります。 9 9 9 6 ¥n 1 0 0 2 3 ¥n 1 0 2 5 7 ¥n	—
	isAlive	任意	サービスが稼働中の場合は終了コードが 0, 稼働中ではない場合は終了コードが 0 以外となります。	指定した場合はサービス監視を行います。

コマンド引数の指定値に次の文字列が含まれている場合は、対応する値に置換されます。

置換前の文字列	置換後の文字列	備考
\${cosminexus.home}	Application Server のインストールディレクトリ	—
%{XXX}	環境変数 XXX の値	環境変数 XXX が存在しない場合は空白文字列に置換されます。
\${server.name}	論理サーバ名	—

(5) 定義例

```
<?xml version="1.0" encoding="UTF-8" ?>
<user-server-definition xmlns="http://www.cosminexus.com/mngsvr/schema/UserServerDefinition-1.1">

  <!-- サービス定義 -->
  <service>
    <!-- 起動種別 -->
    <type>direct</type>
    <!-- 起動順序 -->
    <startup-order>900</startup-order>
    <!-- 起動監視時間 -->
    <start-watch-time>60</start-watch-time>
    <!-- 停止監視時間 -->
    <stop-watch-time>60</stop-watch-time>
    <!-- 強制停止監視時間 -->
    <force-watch-time>60</force-watch-time>
    <!-- 自動再起動回数 -->
    <retry-count>1</retry-count>
    <!-- 自動再起動リトライ間隔 -->
    <retry-interval>60</retry-interval>
    <!-- 監視間隔 -->
    <watch-interval>1</watch-interval>
    <!-- 起動時監視間隔 -->
    <start-time-watch-interval>1</start-time-watch-interval>
    <!-- 環境変数 -->
    <env name="USERSERVER_HOME" value="C:¥UserServer"/>
    <!-- ワーキングディレクトリ -->
    <working-dir>C:¥UserServer</working-dir>
    <!-- ユーザID -->
    <!--
    <user-id>userserver</user-id>
    -->
    <!-- グループID -->
    <!--
    <group-id>userserver</group-id>
    -->
    <!-- サーバ起動用コマンド -->
    <command type="start">
      <arg>C:¥UserServer¥start.exe</arg>
    </command>
    <!-- サーバ停止用コマンド -->
```

```

    <command type="stop">
      <arg>C:\UserServer\stop.exe</arg>
    </command>
    <!-- サーバ強制停止用コマンド -->
    <command type="forceStop">
      <arg>C:\UserServer\forceStop.exe</arg>
    </command>
    <!-- プロセスID取得用コマンド -->
  <!--
    <command type="getProcessID" timeout="180">
      <arg>C:\UserServer\getProcessID.exe</arg>
    </command>
  -->
  <!-- サービス監視用コマンド -->
  <command type="isAlive" timeout="180">
    <arg>C:\UserServer\isAlive.exe</arg>
  </command>
</service>

<!-- 論理ユーザサーバ定義 -->
<server>
  <logical-server-name>UserServer01</logical-server-name>
  <display-name>論理ユーザサーバ</display-name>
  <description>これは論理ユーザサーバです</description>
  <host-ref>192.168.1.1</host-ref>
</server>
</user-server-definition>

```

(6) 注意事項

- stop コマンドおよび forceStop コマンドを指定していない場合に、start コマンドに別のプロセスを起動するようなバッチファイル、またはシェルスクリプトを指定しないでください。指定すると、運用管理エージェントを再起動した場合に、バッチファイルまたはシェルスクリプトから起動したプロセスを、運用管理エージェントの停止操作または強制停止操作の延長で終了できなくなります。
- 直接起動で start コマンドに別のプロセスを起動するようなバッチファイル、またはシェルスクリプトを指定する場合は、stop コマンドおよび forceStop コマンドを指定してください。
- 間接起動で start コマンドに別のプロセスを起動するようなバッチファイル、またはシェルスクリプトを指定する場合は次のようにしてください。
 - stop コマンドおよび forceStop コマンドを指定する。
 - バッチファイルまたはシェルスクリプトから起動したプロセスのプロセス ID を取得するような getProcessID コマンドを指定する。
- 論理ユーザサーバの起動処理で使用する isAlive コマンドは、論理ユーザサーバの起動確認ができる状態になるまで待つのではなく、確認に失敗したらすぐに 0 以外の終了コードで終了するように設定してください。

isAlive コマンドがすぐに終了しないと、論理ユーザサーバが起動処理中にハングアップしてしまった場合に、論理ユーザサーバの起動監視時間オーバー検知時に論理ユーザサーバの自動停止処理が実行されないことがあります。

また、isAlive コマンドと getProcessID コマンドは、マシン負荷の高い状態でもすぐに応答が返るように、できるだけ単純な確認方法で実装してください。

8.3 JP1/IM 連携用システムログメッセージマッピングファイル

JP1/IM 連携用システムログメッセージマッピングファイルには、次のファイルがあります。これらのファイルでは、アプリケーションサーバシステムで出力されるメッセージと JP1 イベントの重大度のマッピングを定義します。

- mserver.jp1event.system.mapping.properties (Management Server 用メッセージマッピングファイル)
- manager.jp1event.system.mapping.properties (J2EE サーバ共通用メッセージマッピングファイル)
- manager.<論理サーバ名>.jp1event.system.mapping.properties (J2EE サーバ個別用メッセージマッピングファイル)

なお、Web サービスおよび Web リソースに関するメッセージは、JP1/IM 連携用システムログメッセージマッピングファイルには定義できません。

8.3.1 mserver.jp1event.system.mapping.properties (Management Server 用メッセージマッピングファイル)

(1) 形式

J2SE のプロパティファイル形式です。

```
mserver.jp1event.system.mapping.severity.<メッセージID>=<JP1 イベントの重大度>
```

<メッセージ ID>

次のマニュアルに記載されているメッセージ ID を指定します。

- マニュアル「アプリケーションサーバ メッセージ(構築／運用／開発用)」

<JP1 イベントの重大度>

Information, Notice, Warning, Error, Critical, Alert, Emergency のどれかを指定します。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config
- UNIX の場合
 /opt/Cosminexus/manager/config

(3) 機能

Management Server が出力するログメッセージを JP1 イベントに変換する際に使用するメッセージマッピングファイルです。

(4) 記述例

```
mserver.jp1event.system.mapping.severity.KEOS10001-E=Error  
mserver.jp1event.system.mapping.severity.KEOS10010-I=Information
```

8.3.2 manager.jp1event.system.mapping.properties (J2EE サーバ共通メッセージマッピングファイル)

(1) 形式

J2SE のプロパティファイル形式です。

```
manager.jp1event.system.mapping.severity.<メッセージID>=<JP1 イベントの重大度>
```

<メッセージ ID>

次のマニュアルに記載されているメッセージ ID を指定します。

- マニュアル「アプリケーションサーバ メッセージ(構築／運用／開発用)」

<JP1 イベントの重大度>

Information, Notice, Warning, Error, Critical, Alert, Emergency のどれかを指定します。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config
- UNIX の場合
 /opt/Cosminexus/manager/config

(3) 機能

サーバマシン中の J2EE サーバ共通で J2EE サーバが出力するログメッセージを JP1 イベントに変換する際に使用するメッセージマッピングファイルです。

(4) 記述例

```
manager.jp1event.system.mapping.severity.KDJE42019-E=Error  
manager.jp1event.system.mapping.severity.KDJE42040-I=Information
```

8.3.3 manager.<論理サーバ名>.jp1event.system.mapping.properties (J2EE サーバ個別用メッセージマッピングファイル)

(1) 形式

J2SE のプロパティファイル形式です。

```
manager.jp1event.system.mapping.severity.<メッセージID>=<JP1 イベントの重大度>
```

<メッセージ ID>

次のマニュアルに記載されているメッセージ ID を指定します。

- マニュアル「アプリケーションサーバ メッセージ(構築／運用／開発用)」

<JP1 イベントの重大度>

Information, Notice, Warning, Error, Critical, Alert, Emergency のどれかを指定します。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%config
- UNIX の場合
 /opt/Cosminexus/manager/config

(3) 機能

J2EE サーバごとにマッピングルールを定義したい場合に作成します。このファイルが定義されている場合、Application Server は該当する J2EE サーバに対して個別用ファイルだけを使用し、J2EE サーバが出力するログメッセージを JP1 イベントに変換します。

(4) 記述例

```
manager.jp1event.system.mapping.severity.KDJE42020-E=Error  
manager.jp1event.system.mapping.severity.KDJE42041-I=Information
```


8.3.4 JP1 イベントへの変換

Management Server, J2EE サーバおよびユーザログ機能で出力するログメッセージは、メッセージマッピングファイルを使用して JP1 イベントに変換されます。

(1) Management Server 用 JP1 イベント

Management Server 用 JP1 イベントには、次の二つの種類があります。

- Application Server で定義されている JP1 イベント
- Management Server 用メッセージマッピングファイルで指定する JP1 イベント

(a) Application Server で定義されている JP1 イベント

Application Server は、次のメッセージ ID を持つメッセージが出力されたときに、JP1 イベントを発行します。

表 8-3 Application Server で定義されている Management Server 用 JP1 イベント

イベント ID	発行契機	メッセージ ID
0x00012050	Management Server の起動	KEOS10101-I
0x00012051	Management Server の終了	KEOS10102-I
0x00012060	ドメインへの Web システム追加	KEOS23301-I
0x00012061	ドメインからの Web システム削除	KEOS23302-I
0x0001206A	Web システムへのユニット追加・削除や J2EE アプリケーションのデプロイ・アンデプロイなど (Web システムの変更)	KEOS23303-I
0x00012070	運用管理エージェントの起動	KEOS21100-I
0x00012071	運用管理エージェントの終了	KEOS21101-I

(b) Management Server 用メッセージマッピングファイルで指定する JP1 イベント

Management Server 用メッセージマッピングファイルのテンプレートファイル (mserver.jp1event.system.mapping.properties) で指定されているメッセージが JP1 イベント発行の対象となります。

テンプレートファイルの格納先は、次のとおりです。

- Windows の場合
 <Application Server のインストールディレクトリ>%manager%config%templates
- UNIX の場合
 /opt/Cosminexus/manager/config/templates

メッセージマッピングファイルの指定によって発行される Management Server 用 JP1 イベントを次の表に示します。

表 8-4 メッセージマッピングファイルの指定による Management Server 用 JP1 イベント

イベント ID	発行契機	マッピングで指定した JP1 イベントの重大度
0x00012000	Management Server で検出した論理サーバの Emergency レベルの障害	Emergency
0x00012001	Management Server で検出した論理サーバの Alert レベルの障害	Alert
0x00012002	Management Server で検出した論理サーバの Critical レベルの障害	Critical
0x00012003	Management Server で検出した論理サーバの Error レベルの障害	Error
0x00012004	Management Server で検出した論理サーバの Warning レベルの障害	Warning
0x00012005	Management Server で検出した論理サーバの Notice レベルの障害	Notice
0x00012006	Management Server で検出した論理サーバの Information レベルの障害	Information
0x00012020	Management Server の Emergency レベルの障害	Emergency
0x00012021	Management Server の Alert レベルの障害	Alert
0x00012022	Management Server の Critical レベルの障害	Critical
0x00012023	Management Server の Error レベルの障害	Error
0x00012024	Management Server の Warning レベルの障害	Warning
0x00012025	Management Server の Notice レベルの障害	Notice
0x00012026	Management Server の Information レベルの障害	Information

(2) J2EE サーバ用 JP1 イベント

次のテンプレートファイルで指定されているメッセージが J2EE サーバ用 JP1 イベント発行の対象となります。

- J2EE サーバ共通用メッセージマッピングファイル (manager.jp1event.system.mapping.properties)
- J2EE サーバ個別用メッセージマッピングファイル (manager.<論理サーバ名>.jp1event.system.mapping.properties)

テンプレートファイルの格納先は、次のとおりです。

- Windows の場合
<Application Server のインストールディレクトリ>%manager%config%templates
- UNIX の場合
/opt/Cosminexus/manager/config/templates

メッセージマッピングファイルの指定による J2EE サーバ用 JP1 イベントを次の表に示します。

表 8-5 J2EE サーバ用 JP1 イベント

イベント ID	発行契機	マッピングで指定した JP1 イベントの重大度
0x00012080	J2EE サーバの Emergency レベルの障害	Emergency
0x00012081	J2EE サーバの Alert レベルの障害	Alert
0x00012082	J2EE サーバの Critical レベルの障害	Critical
0x00012083	J2EE サーバの Error レベルの障害	Error
0x00012084	J2EE サーバの Warning レベルの障害	Warning
0x00012085	J2EE サーバの Notice レベルの障害	Notice
0x00012086	J2EE サーバの Information レベルの障害	Information
0x00012090	J2EE サーバで検出したアプリケーションの Emergency レベルの障害	Emergency
0x00012091	J2EE サーバで検出したアプリケーションの Alert レベルの障害	Alert
0x00012092	J2EE サーバで検出したアプリケーションの Critical レベルの障害	Critical
0x00012093	J2EE サーバで検出したアプリケーションの Error レベルの障害	Error
0x00012094	J2EE サーバで検出したアプリケーションの Warning レベルの障害	Warning
0x00012095	J2EE サーバで検出したアプリケーションの Notice レベルの障害	Notice
0x00012096	J2EE サーバで検出したアプリケーションの Information レベルの障害	Information

(3) J2EE ユーザ用 JP1 イベント

ユーザログ機能で出力するメッセージが J2EE ユーザ用 JP1 イベント発行の対象となります。ユーザログ用 JP1 イベントの設定については、マニュアル「アプリケーションサーバ 機能解説 運用／監視／連携編」の「13.4.2 障害監視の設定」を参照してください。

表 8-6 J2EE ユーザ用 JP1 イベント

イベント ID	発行契機	JP1 イベントの重大度
0x000120D0	アプリケーションの Emergency レベルの障害	Emergency
0x000120D1	アプリケーションの Alert レベルの障害	Alert
0x000120D2	アプリケーションの Critical レベルの障害	Critical
0x000120D3	アプリケーションの Error レベルの障害	Error
0x000120D4	アプリケーションの Warning レベルの障害	Warning
0x000120D5	アプリケーションの Notice レベルの通知	Notice
0x000120D6	アプリケーションの Information レベルの通知	Information

9

仮想化システムで使用するファイル

この章では、仮想化システムで使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

9.1 仮想化システムで使用するファイルの一覧

仮想化システムで使用するファイルの一覧を次の表に示します。

表 9-1 仮想化システムで使用するファイルの一覧

ファイル名	分類	概要	参照先
<LB 接続情報の識別名>.properties	仮想サーバマネージャ側の負荷分散機接続設定プロパティファイル	仮想サーバマネージャに、負荷分散機へのアクセスに必要な接続情報を設定します。	9.2.1
rasetup.cfg	rasetup コマンド用設定ファイル	rasetup コマンドの実行に関する設定をします。	9.2.2
sinaviagent.cfg	サーバ通信エージェント用オプション定義ファイル	サーバ通信エージェント、およびサーバ通信エージェントによって起動されるコマンドやプロセスのログファイルの情報を設定します。	9.2.3
sinaviagent.properties	サーバ通信エージェント用プロパティファイル	サーバ通信エージェントへの接続情報を設定します。	9.2.4
tier.properties	ティア別プロパティファイル	使用する負荷分散機の接続情報の種類や、VirtualServer の情報など、管理ユニットに関する情報をティアごとに設定します。	9.2.5
tierlb.properties	ティア側の負荷分散機接続設定プロパティファイル	ティアに、負荷分散機へのアクセスに必要な接続情報を設定します。	9.2.6
version.properties	インタフェースバージョン定義ファイル	定義ディレクトリに登録するファイル群のフォーマットのバージョンを管理するための情報を設定します。	9.2.7
vmi.properties	仮想サーバマネージャプロパティファイル	仮想サーバマネージャの処理データ格納ディレクトリや、サーバ通信エージェントのタイムアウトなどを設定します。	9.2.8
vmiclient.properties	仮想サーバマネージャのクライアント共通設定プロパティファイル	仮想化システムで使用するコマンドの、共通引数のデフォルト値を設定します。 すべてのクライアントに、共通のデフォルト値を設定したい場合に使用します。	9.2.9
.vmirc	仮想サーバマネージャのクライアント設定プロパティファイル	仮想化システムで使用するコマンドの、共通引数のデフォルト値を設定します。 クライアントごとに、異なるデフォルト値を設定したい場合に使用します。	9.2.10
vmsetup.properties	管理者設定用プロパティファイル	Component Container 管理者のユーザとグループ、および Management Server 管理者のユーザとグループを設定します。	9.2.11
<任意の名称>	仮想サーバグループ情報ファイル	管理ユニットに仮想サーバを登録するための情報（ティア、仮想サーバグループおよび仮想サーバの情報）を設定します。	9.2.12

9.2 仮想化システムで使用するファイルの詳細

9.2.1 <LB 接続情報の識別名>.properties（仮想サーバマネージャ側の負荷分散機接続設定プロパティファイル）

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%vmi%config%lb%
- UNIX の場合
 /opt/Cosminexus/manager/vmi/config/lb/

(3) 機能

仮想サーバマネージャに、負荷分散機へのアクセスに必要な接続情報を設定します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。API を使用した直接接続を使用する負荷分散機で指定できるキーについては、マニュアル「アプリケーションサーバ 機能解説 セキュリティ管理機能編」の「18.2.2 <LB 接続情報の識別名>.properties（仮想サーバマネージャ側の負荷分散機接続設定プロパティファイル）」を参照してください。

キー名称	内容	指定可能値	省略値	VR
lb.type	負荷分散機の種類を指定します。	指定できる値を次に示します。 • BIG-IPv9 • BIG-IPv10.1 • BIG-IPv10.2 • BIG-IPv11 • ACOS	なし	08-53
lb.host	接続する負荷分散機の管理用 IP アドレスを指定します。	IPv4 ドット記法	なし	08-53

キー名称	内容	指定可能値	省略値	VR
lb.protocol	負荷分散機への接続方式を指定します。 telnet : telnet プロトコルを使用して接続します。 ssh : ssh プロトコルを使用して接続します。	指定できる値を次に示します。 • telnet • ssh	なし	08-53
lb.port	負荷分散機で使用するポート番号を指定します。	指定できる値を次に示します。 • ssh 1 ~ 65535 • telnet 1 ~ 65535	• ssh : 22 • telnet : 23	08-53
lb.user	負荷分散機に接続するときのユーザ名を指定します。	任意の文字列を指定します。	なし	08-53
lb.password	負荷分散機に接続するときのユーザのパスワードを指定します。	任意の文字列を指定します。	空文字	08-53
lb.timeout	負荷分散機へのログイン処理時、またはコマンド送信時のタイムアウト時間（単位：秒）を指定します。 このキーで設定した時間内に、負荷分散機へのログイン処理、または負荷分散機に対して発行した CLI コマンドが完了しない場合は、vmiunit コマンドがタイムアウトエラーで異常終了します。	1 ~ 2147483	10	08-53
lb.ACOS.privilegedexec.password	ACOS 上に設定した Privileged EXEC レベルに必要なパスワードを指定します。 このキーの指定は、負荷分散機の種類が ACOS の場合だけ有効になります。	任意の文字列を指定します。	なし	08-53

9.2.2 rasetup.cfg (rasetup コマンド用設定ファイル)

(1) 形式

次のようにキーを指定します。

<キー名称>=<値>

指定方法

- 行頭に「#」を記述した場合、その行はコメントと見なされ、無視されます。
- 値を指定しない場合、その行は無視されます。
- 行頭および行末の空白文字や空行は無視されます。

- キーと値の間のセパレータには「=」を使用します。「=」前後の空白は無視されます。
- 改行までが値と見なされます。
- 同じキーを複数指定した場合は、最後のキーの値を有効とします。
- 1行は 64KB 未満とします。
- ISO 8859-1 の範囲の文字だけが使用できます。

(2) ファイルの格納先

- Windows の場合

<Application Serverのインストールディレクトリ>%manager%setup%config%

- UNIX の場合

/opt/Cosminexus/manager/setup/config/

(3) 機能

rasetup コマンド実行時のログファイルの出力先、最大ファイルサイズ、最大ファイル面数などを指定します。rasetup コマンドについては、マニュアル「アプリケーションサーバリファレンス コマンド編」の「rasetup（仮想サーバへのセットアップ・アンセットアップ）」を参照してください。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
setup.log.dir	ログファイルの出力先ディレクトリを指定します。ファイルセパレータには「/」を使用してください。 Linux の場合、指定した出力先ディレクトリの上位のディレクトリが存在している必要があります。	ディレクトリ名（絶対パス）	• Windows の場合 <Application Serverのインストールディレクトリ>%manager%setup%log • Linux の場合 /opt/Cosminexus/manager/setup/log	09-00

キー名称	内容	指定可能値	省略値	VR
setup.log.filenum	ログファイル面数を指定します。範囲外の値を指定した場合は、デフォルト値が仮定されます。	1～16 の整数で指定します。	4	09-00
setup.log.filesize	ログファイル一つ当たりの上限サイズ（単位：バイト）を指定します。範囲外の値を指定した場合は、デフォルト値が仮定されます。	4096～2147483647 の数値で指定します。	262144	09-00
setup.log.time	ログファイルをローテーションする時刻を 000000～235959 の HHMMSS 形式で指定します。ただし、このプロパティで設定した時刻に達する前に、ログファイルのサイズが上限に達した場合、その時点でログファイルをローテーションします。 指定が無い場合は、ログサイズだけでローションします。	000000～235959 の HHMMSS 形式で指定します。	なし	09-00
setup.log.style	ログファイルのローテーション方式を指定します。 SHIFT： シフトモードでローテーションする場合に指定します。 WRAP を指定した場合： ラップアラウンドモードでローテーションする場合に指定します。	次のどちらかを指定します。 • SHIFT • WRAP	WRAP	09-00
setup.maintenance.log.filenum	保守ログのファイル面数を指定します。範囲外の値を指定した場合は、デフォルト値が仮定されます。	1～16 の整数で指定します。	4	09-00
setup.maintenance.log.filesize	保守ログのファイル一つ当たりの上限サイズ（単位：バイト）を指定します。範囲外の値を指定した場合は、デフォルト値が仮定されます。	4096～2147483647 の整数で指定します。	16777216	09-00

9.2.3 sinaviagent.cfg（サーバ通信エージェント用オプション定義ファイル）

(1) 形式

次のようにキーを指定します。

<キー名称>=<値>

指定方法

- 行頭に「#」を記述した場合、その行はコメントと見なされ、無視されます。
- 値を指定しない場合、その行は無視されます。

- キーと値の間のセパレータには「=」を使用します。「=」前後の空白は無視されます。
- 改行までが値と見なされます。
- 1行は 64KB 未満とします。
- ISO 8859-1 の範囲の文字だけが使用できます。

(2) ファイルの格納先

- Windows の場合
`<Application Serverのインストールディレクトリ>%sinagent%config%`
- UNIX の場合
`/opt/Cosminexus/sinagent/config/`

(3) 機能

サーバ通信エージェント，およびサーバ通信エージェントによって起動されるコマンドやプロセスのログファイルの情報を設定します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは，キーの指定がない場合に仮定される値です。「VR」とは，キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
sinaviagent.log.dir ^{*1}	サーバ通信エージェントのログ出力ディレクトリを指定します。 ファイルセパレータには「/」を使用してください。 指定したログ出力ディレクトリが不正な場合は，サーバ通信エージェントが起動されません。	ディレクトリ名（絶対パス）	• Windows の場合 <code><Application Serverのインストールディレクトリ>%sinagent%log</code> • UNIX の場合 <code>/opt/Cosminexus/sinagent/log</code>	09-00
sinaviagent.spool.dir ^{*1}	サーバ通信エージェントの処理データを格納するディレクトリを指定します。 ファイルセパレータには「/」を使用してください。 指定したログ出力ディレクトリが不正な場合は，サーバ通信エージェントが起動されません。	ディレクトリ名（絶対パス）	• Windows の場合 <code><Application Serverのインス</code>	09-00

キー名称	内容	指定可能値	省略値	VR
			トールディレクトリ >¥sinagent¥spool • UNIX の場合 /opt/Cosminexus/sinagent/spool	
sinaviagent.work.dir※2	サーバ通信エージェントによって起動されるコマンド用の作業ディレクトリを指定します。 ファイルセパレータには「/」を使用してください。 指定したログ出力ディレクトリが不正な場合は、サーバ通信エージェントが起動されません。	ディレクトリ名（絶対パス）	• Windows の場合 <Application Server のインストールディレクトリ>¥sinagent¥work • UNIX の場合 /opt/Cosminexus/sinagent/work	09-00
sinaviagent.log.filenum	サーバ通信エージェント用ログのログファイル面数を指定します。 sinaviagent.log.style キーでシフトモードを設定した場合は、バックアップファイルの面数になります。	1～16 の整数で指定します。	4	09-00
sinaviagent.log.filesize	サーバ通信エージェント用ログのログファイル一つ当たりの上限サイズ（単位：バイト）を指定します。	65536～2147483647 の数値で指定します。	524288	09-00
sinaviagent.log.level	サーバ通信エージェント用ログの出力レベルを指定します。指定したレベル以下の情報がログに出力されます。 -1： サーバ通信エージェント用ログを出力しない場合に指定します。 0： 通常運用でサーバ通信エージェント用ログを出力する場合に指定します。	-1～1000 の数値で指定します。	0	09-00

キー名称	内容	指定可能値	省略値	VR
	10 : 通常運用に近い形態での再現待ちのサーバ通信エージェント用ログを出力する場合に指定します。 20 : システム環境構築時やテストフェーズでの、デバッグレベルの情報の取得する場合に指定します。 30 : 原因究明困難な障害発生したときの、より詳細な障害情報の取得する場合に指定します。			
sinaviagent.log.time	サーバ通信エージェント用ログのログファイルをローテーションする時刻を HHMMSS 形式で指定します。ただし、このプロパティで設定した時刻に達する前に、ログファイルのサイズが上限に達した場合、その時点でログファイルをローテーションします。 指定が無い場合は、ログサイズだけでローションします。	000000～235959 の HHMMSS 形式で指定します。	なし	09-00
sinaviagent.log.style	サーバ通信エージェント用ログのログファイルのローテーション方式を指定します。 SHIFT : シフトモードでローテーションする場合に指定します。 WRAP : ラップアラウンドモードでローテーションする場合に指定します。	次のどちらかを指定します。 • SHIFT • WRAP	WRAP	09-00
sinaviagent.service.log.filename	Windows の場合に、サーバ通信エージェントサービス用ログのログファイル面数を指定します。	1～16 の整数で指定します。	4	09-00
sinaviagent.service.log.filesize	Windows の場合に、サーバ通信エージェントサービス用ログのログファイル一つ当たりの上限サイズ（単位：バイト）を指定します。	65536～2147483647 の数値で指定します。	65536	09-00
sinaviagent.service.log.level	Windows の場合に、サーバ通信エージェントサービス用ログのログ出力レベルを指定します。指定したレベル以下の情報がログに出力されます。 -1 : サーバ通信エージェントサービス用ログを出力しない場合に指定します。 0 : 通常運用でサーバ通信エージェントサービス用ログを出力する場合に指定します。	-1～1000 の数値で指定します。	0	09-00

キー名称	内容	指定可能値	省略値	VR
	10 : システム環境構築時やテストフェーズでの、デバッグレベルの情報の取得する場合に指定します。 20 : システム環境構築時やテストフェーズでの、デバッグレベルの情報の取得する場合に指定します。 30 : 原因究明困難な障害発生したときの、より詳細な障害情報の取得する場合に指定します。			
sinaviagent.snactl.log.fileenum	snactl コマンド用ログのログファイル面数を指定します。	1～16 の整数で指定します。	4	09-00
sinaviagent.snactl.log.filesize	snactl コマンド用ログのログファイル一つ当たりの上限サイズ（単位：バイト）を指定します。	65536～2147483647 の数値で指定します。	65536	09-00
sinaviagent.snactl.log.level	snactl コマンド用ログの出力レベルを指定します。指定したレベル以下の情報がログに出力されます。 -1 : snactl コマンド用ログを出力しない場合に指定します。 0 : 通常運用で snactl コマンド用ログを出力する場合に指定します。 10 : 通常運用に近い形態での再現待ちの snactl コマンド用ログを出力する場合に指定します。 20 : システム環境構築時やテストフェーズでの、デバッグレベルの情報の取得する場合に指定します。 30 : 原因究明困難な障害発生したときの、より詳細な障害情報の取得する場合に指定します。	-1～1000 の数値で指定します。	0	09-00
sinaviagent.maintenance.log.fileenum	サーバ通信エージェント用保守ログのファイル面数を指定します。	1～16 の整数で指定します。	4	09-00
sinaviagent.maintenance.log.filesize	サーバ通信エージェント用保守ログのファイル一つ当たりの上限サイズ（単位：バイト）を指定します。	65536～2147483647 の数値で指定します。	65536	09-00

キー名称	内容	指定可能値	省略値	VR
sinaviagent.service.maintenance.log.filenum	Windows の場合に、サーバ通信エージェントサービス用保守ログのファイル面数を指定します。	1～16 の整数で指定します。	4	09-00
sinaviagent.service.maintenance.log.filesize	Windows の場合に、サーバ通信エージェントサービス用保守ログのファイル一つ当たりの上限サイズ（単位：バイト）を指定します。	65536～2147483647 の数値で指定します。	65536	09-00
sinaviagent.snactl.maintenance.log.filenum	snactl コマンド用保守ログのファイル面数を指定します。	1～16 の整数で指定します。	4	09-00
sinaviagent.snactl.maintenance.log.filesize	snactl コマンド用保守ログのファイル一つ当たりの上限サイズ（単位：バイト）を指定します。	65536～2147483647 の数値で指定します。	65536	09-00
sinaviagent.process.consolelog.enabled	サーバ通信エージェントが起動したプロセスのコンソール出力情報を取得するかどうかを設定します。 true : コンソール出力情報を取得する場合に指定します。 false : コンソール出力情報を取得しない場合に指定します。	次のどちらかを指定します。 • true • false	true	09-00
sinaviagent.process.consolelog.filenum	サーバ通信エージェントが起動したプロセスのコンソール出力情報を出力するログファイル面数を指定します。 sinaviagent.process.consolelog.style キーでシフトモードを指定した場合は、バックアップファイルの面数になります。	1～16 の整数で指定します。	4	09-00
sinaviagent.process.consolelog.filesize	サーバ通信エージェントが起動したプロセスのコンソール出力情報を出力するログファイル一つ当たりの上限サイズ（単位：バイト）を指定します。	65536～2147483647 の数値で指定します。	65536	09-00
sinaviagent.process.consolelog.time	サーバ通信エージェントが起動したプロセスのコンソール出力情報を出力するログファイルをローテーションする時刻を指定します。 指定が無い場合は、ログサイズによるローテーションをします	000000～235959 の HHMMSS 形式で指定します。	なし	09-00
sinaviagent.process.consolelog.style	サーバ通信エージェントが起動したプロセスのコンソール出力情報を出力するログファイルのローテーション方式を指定します。 SHIFT : シフトモードでローテーションする場合に指定します。	次のどちらかを指定します。 • SHIFT • WRAP	WRAP	09-00

キー名称	内容	指定可能値	省略値	VR
	WRAP を指定した場合： ラップアラウンドモードでローテーションする 場合に指定します。			

注※1 キーの変更はサーバ通信エージェントが停止しているときに実行してください。

注※2 キーの変更は、環境の構築・変更・削除、または構築対象サーバ情報の取得などを実行していないとき、かつ、サーバ通信エージェントが停止しているときに実行してください。また、このキーの指定を変更した場合で、かつ、変更前のコマンド用作業ディレクトリの下にディレクトリやファイルがあるときは、サーバ通信エージェントを起動する前に、変更前のコマンド用作業ディレクトリの下にあるディレクトリやファイルを、そのままの形式で、変更後のコマンド用作業ディレクトリに移動した後、サーバ通信エージェントを起動してください。

9.2.4 sinaviagent.properties（サーバ通信エージェント用プロパティファイル）

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

- Windows の場合
`<Application Serverのインストールディレクトリ>%sinagent%config%`
- UNIX の場合
`/opt/Cosminexus/sinagent/config/`

(3) 機能

サーバ通信エージェントへの接続情報を設定します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値または動作です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
sinaviagent.port	サーバ通信エージェントで使用するポート番号を指定します。	1～65535	20580	08-00
sinaviagent.bind_host	サーバ通信エージェントで使用する IP アドレス、または解決できるローカルホスト名称を指定します※1。	ホスト名または IPv4 ドット記法	なし	08-00

キー名称	内容	指定可能値	省略値	VR
	値を指定しなかった場合、ローカルのすべての IP アドレスに対する接続が受け付けられます。IP アドレスはメタキャラクタを用いた正規表現で指定できます。IP アドレスが正規表現で指定された場合、ローカルの利用可能な IP アドレスのうちマッチする IP アドレスを採用します※2。 不正な値を指定した場合は、サーバ通信エージェントはエラー終了します。			
sinaviagent.permitted.hosts	サーバ通信エージェントへのアクセスを許可するホスト名または IP アドレスを指定します。複数のアドレスを指定する場合はコンマ (,) で空白を入れずに区切ります。 IP アドレスはメタキャラクタを用いた正規表現で指定できます。IP アドレスを正規表現で指定した場合、IP アドレスのうちマッチする IP アドレスが使用されます。複数の IP アドレスがマッチした場合は、マッチしたすべての IP アドレスが許可されます。 なお、IP アドレスを正規表現で指定する場合は、コンマ (,) で複数の値を指定することはできません。 値を指定しなかった場合、サーバ通信エージェントはエラー終了します。	ホスト名または IPv4 ドット記法	なし	08-00

注※1 複数の物理ネットワークインタフェースを持つホスト、または一つの物理ネットワークインタフェースに対して複数の論理 IP アドレスを割り当てているホストでサーバ通信エージェントを利用するとき、特定の物理ネットワークインタフェースに対する接続だけを受け付けるよう制限したい場合に、接続を受け付ける IP アドレスまたはそのホスト名を指定します。

注※2 サーバ通信エージェントとの通信に使用される IP アドレスのサブネットが一つに決められている場合、指定する IP アドレスを "192.168.0.0/24" のように記述しておけば、前方が "192.168.0." である IP アドレス ("192.168.0.32" や "192.168.0.128" など) にマッチするため、設定ファイルを修正しないで、すべてのホストに配布して利用できます。正規表現についての詳細は、Java の `java.util.regex.Pattern` クラスの仕様を確認してください。ただし、"`¥`" は 1 文字の "`¥`" に置き換えられてしまうため、"`¥`" を指定する場合は 2 文字連続して指定してください。指定した正規表現にマッチする IP アドレスが複数個検出された場合は、最も小さい値の IP アドレスを採用します。例えば、"192.168.0.32" と "192.168.0.128" が検出された場合は "192.168.0.32" を採用します。この場合、採用された IP アドレスが意図する IP アドレスになるとは限らないため、"192.168.0.0/24" のように必ず 1 個だけマッチする正規表現を記述してください。

9.2.5 tier.properties (ティア別プロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

- Windows の場合

<定義ディレクトリ>%<ティア別定義ディレクトリ>%vmi%

- UNIX の場合

<定義ディレクトリ>/<ティア別定義ディレクトリ>/vmi/

次のテンプレートファイルをコピーして利用してください。

- Windows の場合

<Application Serverのインストールディレクトリ>%manager%vmi%templates%tier.properties

- UNIX の場合

/opt/Cosminexus/manager/vmi/templates/tier.properties

(3) 機能

使用する負荷分散機の接続情報の種類や、VirtualServer の情報、仮想サーバの起動順序や OS 種別など、管理ユニットに関する情報をティアごとに設定します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
lb.use	使用する負荷分散機の接続情報を指定します。 ＜LB 接続情報の識別名＞： 仮想サーバマネージャ側の接続情報（＜LB 接続情報の識別名＞.properties）を使用する場合に指定します。 :unit: : 管理ユニットに登録するティア別定義ディレクトリにある接続情報（tierlb.properties）を使用する場合に指定します。 値を設定していない、またはキーを指定していない場合、負荷分散機連携機能は有効になりません。	指定できる値を次に示します。 • <LB 接続情報の識別名> （先頭 1 文字は英字から始まる、英数字、アンダースコア「_」、ピリオド「.」、またはハイフン「-」で指定した 31 文字以内の文字列） • :unit:	空文字	08-53
lb.virtualserver.name	VirtualServer 名を指定します。 負荷分散機連携機能を使用する場合に指定してください。	先頭 1 文字は英字から始まる、英数字、	管理ユニット名	08-53

キー名称	内容	指定可能値	省略値	VR
		アンダースコア「_」、ピリオド「.」、またはハイフン「-」で指定した 31 文字以内の文字列		
lb.virtualserver.host	VirtualServer のホスト名、または IP アドレスを指定します。 負荷分散機連携機能を使用する場合に指定してください。	ホスト名または IPv4 ドット記法	なし	08-53
lb.virtualserver.port	VirtualServer のポート番号を指定します。 負荷分散機連携機能を使用する場合に指定してください。	1～65535	なし	08-53
lb.persistence.method	cookie パーシステンスを維持する場合、「cookie-insert」を指定します。 次の場合は、パーシステンスを維持しません。 - 値を設定していない。 - キーを指定していない。	cookie-insert	空文字	08-53
lb.persistence.cookie-insert.cookie-name	負荷分散機が設定する cookie の名称を指定します。 このキーの指定は、「lb.persistence.method=cookie-insert」が指定されている場合に有効になります。 また、< LB 接続情報の識別名 >.properties または tierlb.properties で、次に示す値が両方とも指定されている場合、このキーの指定は無効になります。 - lb.type=ACOS - lb.protocol=API	先頭 1 文字は英字から始まる、英数字、アンダースコア「_」で指定した 63 文字以内の文字列	VML_COOKIE_NAME	08-53
tier.startup.no	複数のティアを管理する管理ユニットを一括起動、または一括停止する場合に、ティアに割り当てられている仮想サーバを起動または停止する順序を指定します。 一括起動時には番号の小さい順に、一括停止時には番号の大きい順に処理が実行されます。なお、同じ番号の場合は同時に処理が実行されます。 管理ユニットを一括起動または一括停止しない場合は、-1 を指定してください。	-1～999	- combined-tier の場合：70 - ctm-tier の場合：40 - http-tier の場合：80 - j2ee-tier の場合：60	09-50
tier.target.os	ティアに割り当てられている仮想サーバの OS 種別（ゲスト OS）を指定します。	指定できる値を次に示します。 - windows - linux	なし	09-00

キー名称	内容	指定可能値	省略値	VR
	windows : Windows の OS で仮想サーバを構築する場合に指定します。 linux : Linux の OS で仮想サーバを構築する場合に指定します。 このキーで指定した OS 種別と、vmiunit コマンドのサブコマンド update を実行する仮想サーバの OS 種別が異なる場合、コマンドはエラーとなり、仮想サーバに定義情報は反映されません。			

9.2.6 tierlb.properties (ティア側の負荷分散機接続設定プロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

- Windows の場合
<定義ディレクトリ>%<ティア別定義ディレクトリ>%vmi%
- UNIX の場合
<定義ディレクトリ>/<ティア別定義ディレクトリ>/vmi/

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
<Application Serverのインストールディレクトリ>%manager%vmi%templates%tierlb.properties
- UNIX の場合
/opt/Cosminexus/manager/vmi/templates/tierlb.properties

(3) 機能

負荷分散機へのアクセスに必要な接続情報をティアに設定します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値または動作です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。API を使用した

直接接続を使用する負荷分散機で指定できるキーについては、マニュアル「アプリケーションサーバ 機能解説 セキュリティ管理機能編」の「18.2.3 tierlb.properties（ティア側の負荷分散機接続設定プロパティファイル）」を参照してください。

キー名称	内容	指定可能値	省略値	VR
lb.type	負荷分散機の種類を指定します。	指定できる値を次に示します。 • BIG-IPv9 • BIG-IPv10.1 • BIG-IPv10.2 • BIG-IPv11 • ACOS	なし	08-53
lb.host	接続する負荷分散機の管理用 IP アドレスを指定します。	IPv4 ドット記法	なし	08-53
lb.protocol	負荷分散機への接続方式を指定します。 telnet : telnet プロトコルを使用して接続します。 ssh : ssh プロトコルを使用して接続します。	指定できる値を次に示します。 • telnet • ssh	なし	08-53
lb.port	負荷分散機で使用するポート番号を指定します。	指定できる値を次に示します。 • ssh 1～65535 • telnet 1～65535	• ssh : 22 • telnet : 23	08-53
lb.user	負荷分散機に接続するときのユーザ名を指定します。	任意の文字列を指定します。	なし	08-53
lb.password	負荷分散機に接続するときのユーザのパスワードを指定します。	任意の文字列を指定します。	空文字	08-53
lb.timeout	負荷分散機へのログイン処理時、またはコマンド送信時のタイムアウト時間（単位：秒）を指定します。 このキーで設定した時間内に、負荷分散機へのログイン処理、または負荷分散機に対して発行した CLI コマンドが完了しない場合は、vmiunit コマンドがタイムアウトエラーで異常終了します。	1～2147483	10	08-53
lb.ACOS.privilegedexec.password	ACOS 上に設定した Privileged EXEC レベルに必要なパスワードを指定します。	任意の文字列を指定します。	なし	08-53

キー名称	内容	指定可能値	省略値	VR
	このキーの指定は、負荷分散機の種類が ACOS の場合だけ有効になります。			

9.2.7 version.properties（インタフェースバージョン定義ファイル）

（1）形式

次のようにキーを指定します。

<キー名称>=<値>

指定方法

- 行頭に「#」を記述した場合、その行はコメントと見なされ、無視されます。
- 値を指定しない場合、その行は無視されます。
- 行頭および行末の空白文字や空行は無視されます。
- キーと値の間のセパレータには「=」を使用します。「=」前後の空白は無視されます。
- 改行までが値と見なされます。
- 同じキーを複数指定した場合は、最後のキーの値を有効とします。
- ISO 8859-1 の範囲の文字だけが使用できます。

（2）ファイルの格納先

- Windows の場合
 <定義ディレクトリ>%<ティア別定義ディレクトリ>%config%Cosminexus
- UNIX の場合
 <定義ディレクトリ>/<ティア別定義ディレクトリ>/config/Cosminexus

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%vmi%templates%version.properties
- UNIX の場合
 /opt/Cosminexus/manager/vmi/templates/version.properties

（3）機能

定義ディレクトリに登録するファイル群のフォーマットのバージョンを管理するための情報を設定します。意図しないバージョンのアプリケーションサーバが、仮想サーバ上に構築されることを防ぎます。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
vm.interface.version	定義ディレクトリに配置する定義ファイル群の構成や、定義ファイル内の情報の形式に対応するバージョン情報です。	090000 の固定値を指定します。	なし	09-00
vm.target.edition	仮想サーバに構築するアプリケーションサーバのエディションです。値は「standard」で固定です。	指定できる値を次に示します。 • standard	なし	09-00

9.2.8 vmi.properties (仮想サーバマネージャプロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

- Windows の場合
＜Application Serverのインストールディレクトリ＞¥manager¥vmi¥config¥
- UNIX の場合
/opt/Cosminexus/manager/vmi/config/

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
＜Application Serverのインストールディレクトリ＞¥manager¥vmi¥templates¥vmi.properties
- UNIX の場合
/opt/Cosminexus/manager/vmi/templates/vmi.properties

(3) 機能

仮想サーバマネージャの処理データ格納ディレクトリや、サーバ通信エージェントのタイムアウトなどを設定します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
vmi.spool.dir	仮想サーバマネージャの処理データ格納ディレクトリを絶対パスで指定します。 ファイルセパレータにはスラッシュ (/) を使用します。 環境を構築済みの状態でこのキーを変更した場合は、仮想サーバマネージャ再起動時に、管理ユニットの情報（定義ディレクトリ、仮想サーバの情報も含まれます）、およびアカウントの情報などの仮想サーバに登録した情報がすべて削除されます。 このキーを変更する際に既存の情報を継続して使用する場合は、変更前のディレクトリパス以下に存在するすべてのディレクトリおよびファイルを変更先のディレクトリパスの下にコピーしてください。	ディレクトリ名（絶対パス）	<Application Server のインストールディレクトリ>/manager/vmi/spool	08-53
vmi.serverconnectionagent.connect.timeout	仮想サーバマネージャと仮想サーバ上のサーバ通信エージェントとの接続で、タイムアウトが発生するまでの時間（単位：秒）を指定します。0 を指定するとタイムアウトしません。	0～3600	2	08-53
vmi.serverconnectionagent.socket.timeout	仮想サーバマネージャと仮想サーバ上のサーバ通信エージェントとの間の情報の送受信で、タイムアウトが発生するまでの時間（単位：秒）を指定します。0 を指定するとタイムアウトしません。	0～3600	2	08-53
vmi.serverconnectionagent.exec.timeout	仮想サーバ上のサーバ通信エージェントでコマンドを実行する場合に、コマンドの実行で、タイムアウトが発生するまでの時間（単位：秒）を指定します。0 を指定するとタイムアウトしません。	0～3600	1800	08-53
vmi.jp1.base.auth.enabled	JP1/Base と連携してユーザのアカウントを管理するための機能（JP1 ユーザ認証連携機能）を有効にするかどうかを指定します。 true : JP1 ユーザ認証連携機能を有効にします。仮想サーバマネージャ管理者アカウント、およびシステム構築者アカウントは、JP1/Base（認証サーバ）によって管理されます。 false : JP1 ユーザ認証連携機能を無効にします。仮想サーバマネージャ管理者アカウント、およびシステム構築者アカウントは、仮想サーバマネージャによって管理されます。	指定できる値を次に示します。 • true • false	false	08-70

(5) 注意事項

仮想サーバマネージャを起動した状態でパラメタを変更した場合は、仮想サーバマネージャを再起動する必要があります。

9.2.9 vmiclient.properties (仮想サーバマネージャのクライアント共通設定プロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%vmi%config%
- UNIX の場合
 /opt/Cosminexus/manager/vmi/config/

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%vmi%templates%vmiclient.properties
- UNIX の場合
 /opt/Cosminexus/manager/vmi/templates/vmiclient.properties

(3) 機能

仮想サーバマネージャで提供するコマンドの共通引数のデフォルト値を設定します。すべてのクライアントに、共通のデフォルト値を設定したい場合に使用します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
vmi.apply_user	このファイルを適用するクライアントの OS のユーザ ID を指定します。ユーザ ID はコンマ「,」で区切って指定します。	OS に登録されているユーザ名	なし	08-53

キー名称	内容	指定可能値	省略値	VR
	また、Windows のサービスから Smart Composer 機能のコマンドが呼び出される場合に、このプロパティファイルを適用するには、「<ホスト名>\$」（<ホスト名>は、Management Server が稼働するホストのホスト名）を指定してください。			
vmi.connect.m	共通引数の-m オプション省略時に仮定するホスト名とポート番号を指定します。 ポート番号は省略できます。	指定できる値を次に示します。 • ホスト名 [: ポート番号] • IPv4 ドット記法[: ポート番号]	なし	08-53
vmi.connect.u	共通引数の-u オプション省略時に仮定するユーザ ID を指定します。	ユーザアカウントのユーザ ID	なし	08-53
vmi.connect.p	共通引数の-p オプション省略時に仮定するパスワードを指定します。	ユーザアカウントのパスワード	なし	08-53
vmi.connect.wait	共通引数の-wait オプション省略時に仮定する実行待ち時間（単位：秒）を指定します。	0～2147483647	なし	08-53

9.2.10 .vmirc（仮想サーバマネージャのクライアント設定プロパティファイル）

(1) 形式

J2SE のプロパティファイル形式です。

(2) ファイルの格納先

コマンドを実行する OS ユーザのホームディレクトリに格納します。

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
<Application Serverのインストールディレクトリ>%manager%vmi%templates%.vmirc
- UNIX の場合
/opt/Cosminexus/manager/vmi/templates/.vmirc

(3) 機能

仮想サーバマネージャで提供するコマンドの共通引数のデフォルト値を設定します。クライアントごとに、異なるデフォルト値を設定したい場合に使用します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
vmi.connect.m	共通引数の-m オプション省略時に仮定するホスト名とポート番号を指定します。 ポート番号は省略できます。	指定できる値を次に示します。 • ホスト名 [:ポート番号] • IPv4 ドット記法[:ポート番号]	なし	08-53
vmi.connect.u	共通引数の-u オプション省略時に仮定するユーザID を指定します。	ユーザアカウントのユーザID	なし	08-53
vmi.connect.p	共通引数の-p オプション省略時に仮定するパスワードを指定します。	ユーザアカウントのパスワード	なし	08-53
vmi.connect.wait	共通引数の-wait オプション省略時に仮定する実行待ち時間（単位：秒）を指定します。	0～2147483647	なし	08-53

9.2.11 vmsetup.properties（管理者設定用プロパティファイル）

(1) 形式

次のようにキーを指定します。

<キー名称>=<値>

指定方法

- 行頭に「#」を記述した場合、その行はコメントと見なされ、無視されます。
- 値を指定しない場合、その行は無視されます。
- 行頭および行末の空白文字や空行は無視されます。
- キーと値の間のセパレータには「=」を使用します。「=」前後の空白は無視されます。

- 改行までが値と見なされます。
- 同じキーを複数指定した場合は、最後のキーの値を有効とします。
- ISO 8859-1 の範囲の文字だけが使用できます。

(2) ファイルの格納先

- Windows の場合
`<定義ディレクトリ>%<ティア別定義ディレクトリ>%config%Cosminexus`
- UNIX の場合
`<定義ディレクトリ>/<ティア別定義ディレクトリ>/config/Cosminexus`

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
`<Application Serverのインストールディレクトリ>%manager%vmi%templates%vmsetup.properties`
- UNIX の場合
`/opt/Cosminexus/manager/vmi/templates/vmsetup.properties`

(3) 機能

Component Container 管理者のユーザとグループ、および Management Server 管理者のユーザ (Management Server の管理ユーザアカウント) とグループを設定します。

(4) 指定できるキー

指定できるキーを次に示します。なお「省略値」とは、キーの指定がない場合に仮定される値です。「VR」とは、キーが導入・変更されたアプリケーションサーバのバージョンです。

キー名称	内容	指定可能値	省略値	VR
setup.mng.administrator .userid	Management Server の管理ユーザアカウントのユーザ ID を指定します。	半角英数字, 「!」, 「#」, 「\$」, 「&」, 「'」, 「(」, 「)」, 「*」, 「+」, 「,」, 「-」, 「.」, 「/」, 「:」, 「;」, 「<」, 「=」, 「>」, 「?」, 「@」, 「[」, 「]」, 「_」, 「`」, 「{」, 「 」, 「}」, 「~」	admin	09-00
setup.mng.administrator .password	Management Server の管理ユーザアカウントのパスワードを指定します。	半角英数字, 「!」, 「#」,	なし	09-00

キー名称	内容	指定可能値	省略値	VR
		「\$」, 「&」, 「'」, 「(」, 「)」, 「*」, 「+」, 「,」, 「-」, 「.」, 「/」, 「:」, 「;」, 「<」, 「=」, 「>」, 「?」, 「@」, 「[」, 「]」, 「_」, 「`」, 「{」, 「 」, 「}」, 「~」		
setup.CC.user	Component Container 管理者として設定するユーザを設定します。	1 文字以上の文字列を指定します。	root	09-00
setup.CC.group	Component Container 管理者として設定するユーザが所属するグループを設定します。	1 文字以上の文字列を指定します。	Linux :root	09-00
setup.manager.group	Management Server 管理者として設定するユーザが所属するグループを設定します。	1 文字以上の文字列を指定します。	Linux :root	09-00

9.2.12 仮想サーバグループ情報ファイル

(1) 形式

xml 形式です。

(2) ファイルの格納先

任意の作業場所に格納してください。

次のテンプレートファイルをコピーして利用してください。

- Windows の場合
 <Application Serverのインストールディレクトリ>%manager%vmi%templates%virtualserver-info.xml
- UNIX の場合
 /opt/Cosminexus/manager/vmi/templates/virtualserver-info.xml

(3) 機能

管理ユニットに仮想サーバを格納する場合に使用する定義ファイルです。仮想サーバグループ情報ファイルには、仮想サーバグループの名称や、仮想サーバの IP アドレス、仮想サーバを割り当てるティアなどの情報を定義します。

(4) ファイルの指定内容

仮想サーバグループ情報ファイルの指定内容の説明で使用する項目の意味を次に示します。

項目名	説明
タグ名	設定するタグを示します。
属性名	設定する属性を示します。
出現パターン	タグまたは属性が出現する回数を示します。
内容	タグまたは属性に定義する内容を示します。
指定可能値	タグまたは属性に指定できる値を示します。
省略値	タグまたは属性に指定する内容が省略されている場合に有効になる指定値を示します。デフォルト値がない場合は「なし」を示します。
VR	定義できるアプリケーションサーバのバージョンを示します。

仮想サーバグループ情報ファイルでは、XML 宣言と xmlns 属性を次のように設定します。

```
<?xml version="1.0" encoding="utf-8"?>
<virtualserver-info xmlns="http://www.cosminexus.com/mngsvr/schema/VirtualServerInfo-1.0">
```

仮想サーバグループ情報ファイルの構成および指定内容を次の表に示します。

タグ名		属性名	出現パターン	内容	指定可能値	省略値	VR
virtualserver-info		—	1 回	仮想サーバ情報を定義します。	—	なし	09-00
L	group	—	1 回以上	仮想サーバグループを定義します。	—	なし	09-00
		name	1 回	仮想サーバグループの名称を指定します。	先頭 1 文字は英字から始まる、英数字、アンダースコア「_」またはハイフン「-」で指定した 31 文字以内の文字列	なし	09-00
	L	virtualserver	—	仮想サーバグループに割り当てる仮想サーバを定義します。	—	なし	09-00

タグ名			属性名	出現パターン	内容	指定可能値	省略値	VR
			tier	1 回	仮想サーバを割り当てるティア（ティア識別子）を指定します。 仮想サーバを登録する管理ユニットに取り込んだ、ティア別定義ディレクトリのディレクトリ名を指定します。	先頭 1 文字は英字から始まる、英数字、アンダースコア「_」またはハイフン「-」で指定した 31 文字以内の文字列	なし	09-00
			ip	1 回	仮想サーバ識別子を指定します。管理 LAN に接続されている NIC の IP アドレスを指定します。	IPv4 ドット記法	なし	09-00
			serviceip	0 または 1 回	仮想サーバの業務用 IP アドレスを指定します。業務 LAN に接続されている NIC の IP アドレスを指定します。 負荷分散機連携機能を有効にする場合で、業務 LAN と管理 LAN が分かれているときに指定してください。	IPv4 ドット記法	ip 属性に指定した IP アドレス	09-00
			port	0 または 1 回	仮想サーバのサーバ通信エージェントのポート番号を指定します。	1～65535 の数値で指定します。	20580	09-00

（凡例）－：該当なし

10

ログの運用で使用するファイル

この章では、ログの運用で使用するファイルの形式、格納先、使用できる変数などについて説明します。

10.1 ログの運用で使用するファイルの一覧

ログの運用で使用するファイルの一覧を，次の表に示します。

表 10-1 ログの運用で使用するファイルの一覧

ファイル名	分類	概要	参照先
(任意)	snapshot ログ収集対象定義 ファイル	snapshot ログとして収集する，一次送付資 料，二次送付資料，および定義送付資料の対 象ファイルを定義します。	10.2.1

10.2 ログの運用で使用するファイルの詳細

Windows 環境で JDK21 以降の場合、ログの運用で使用するファイルにマルチバイト文字は使用できません。

10.2.1 snapshot ログ収集対象定義ファイル

snapshot ログは、システム保守に必要な情報およびアプリケーション保守に必要な情報を収集します。

snapshot ログ収集対象定義ファイルでは、snapshot ログとして収集する、一次送付資料、二次送付資料、および定義送付資料の対象ファイルを設定します。このファイルを編集することで、snapshot ログ収集対象のパスを変更することができます。

(1) 形式

次の形式で指定します。

- 収集対象のパスはフルパスで指定します。
- #で始まる行はコメントとみなされます。
- ファイルセパレータは「/」を使用します。
- ファイルセパレータで分類されたパス要素に対して、正規表現[※]が使用できます。
- 不正な正規表現[※]を含む場合、その指定は無視されます。
- 指定したパスにファイルが存在しない場合、その指定は無視されます。

注※

正規表現については、`java.util.regex.Pattern` の定義を参照してください。

(2) 格納先およびファイル名

インストール時のデフォルト設定ファイルを編集する場合

次のファイルを編集してください。

- Windows の場合
 <Application Server のインストールディレクトリ>%manager%config%snapshotlog.conf
 <Application Server のインストールディレクトリ>%manager%config%snapshotlog.2.conf
 <Application Server のインストールディレクトリ>%manager%config%snapshotlog.param.conf
- UNIX の場合
 /opt/Cosminexus/manager/config/snapshotlog.conf
 /opt/Cosminexus/manager/config/snapshotlog.2.conf

/opt/Cosminexus/manager/config/snapshotlog.param.conf

snapshotlog.conf には、一次送付資料として収集する対象を設定します。

snapshotlog.2.conf には、二次送付資料として収集する対象を設定します。

snapshotlog.param.conf には、定義送付資料として収集する対象を設定します。

snapshot ログの収集対象ファイルについては、マニュアル「アプリケーションサーバ 機能解説 保守／移行編」の「2.3.3 snapshot ログの収集」を参照してください。

任意のファイルを指定する場合

格納先およびファイル名は任意です。

次の場所にあるテンプレートをコピーして使用してください。

- Windows の場合

<Application Server のインストールディレクトリ>%manager%config%templates%

- UNIX の場合

/opt/Cosminexus/manager/config/templates/

(3) 使用できる変数

パスを指定する場合、次の変数が使用できます。

表 10-2 snapshot ログ収集対象定義ファイルで使用できる変数

変数	内容
\${cosminexus.home}	Application Server のインストールディレクトリ
\${user.home}	ユーザのホームディレクトリ
\${hws.home}	HTTP Server のインストールディレクトリ
\${com.cosminexus.manager.log.dir}	Manager のログ出力ディレクトリ
&{prfspool}	論理サーバの環境設定で、パフォーマンスストレサの[設定]タブを選択し、[パフォーマンスストレサの設定]のスパールディレクトリで設定された値
&{ctmspool}	論理サーバの環境設定で、CTM ドメインマネジャの[基本設定]タブを選択し、[CTM ドメインマネジャの基本設定]のスパールディレクトリで設定された値
&{ejb.public.directory}	論理サーバの環境設定で、J2EE サーバの[コンテナ]-[J2EE]タブを選択し、[J2EE コンテナの設定]の作業ディレクトリの定義で設定された値
&{webserver.work.directory}	論理サーバの環境設定で、J2EE サーバの[コンテナ]-[Web]タブを選択し、JSP 用一時ディレクトリで設定された値
&{ejb.server.log.directory}	論理サーバの環境設定で、J2EE サーバの[サービス]-[ログ]タブを選択し、ログ出力ディレクトリで設定された値

変数	内容
<code>&{webserver.logger.access_log.inprocess_http.filename}</code>	論理サーバの環境設定で、J2EE サーバの[サービス]-[トランザクション]タブを選択し、[アクセスログの設定]の出力ファイル名で設定された値
<code>&{ejbserver.distributedtx.ots.status.directory1}</code>	論理サーバの環境設定で、J2EE サーバの[HTTP サーバ]-[ログ]タブを選択し、インプロセス OTS のステータスファイル格納先で設定された値
<code>&{ejbserver.distributedtx.ots.status.directory2}</code>	論理サーバの環境設定で、J2EE サーバの[HTTP サーバ]-[ログ]タブを選択し、インプロセス OTS のステータスファイル格納先（予備）で設定された値
<code>&{core.dump.directory}</code>	論理サーバの環境設定で、Web サーバの[設定]タブを選択し、[Web サーバの基本的な設定]のコアダンプ出力先ディレクトリで設定された値
<code>&{hws.logfile.dir}</code>	論理サーバの環境設定で、Web サーバの[設定]タブを選択し、[Web サーバの出力するログの設定]のログ出力先ディレクトリで設定された値
<code>&{jklogfiledir}</code>	論理サーバの環境設定で、Web サーバの[リダイレクタ]タブを選択し、ログ出力先ディレクトリで設定された値
<code>&{jktracelogfiledir}</code>	論理サーバの環境設定で、Web サーバの[リダイレクタ]タブを選択し、保守用のトレースログのログ出力先ディレクトリで設定された値
<code>&{ctm.RegOption}</code>	CTM レギュレータの設定ファイル
<code>&{ctm.TSCGwOption}</code>	OTM ゲートウェイの設定ファイル
<code>\${adminagent.prftrace_dir}</code>	一時 PRF トレースファイルの出力先ディレクトリ

注

- `$`はシステムプロパティ、`&`は Management Server の内部変数を表します。
- 値に"`$`"を含む変数は使用できません。
- snapshot ログ収集コマンドを使って収集する場合は、`&`を使用した Management Server の内部変数は使用できません。ただし、「`&{ejb.public.directory}`」は snapshot ログ収集コマンドでも使用できます。

(4) 記述例

/home/logdir/message1.log を収集する場合

```
/home/logdir/message1*.log
```

/opt/Cosminexus/manager/log ディレクトリ直下のファイルをすべて収集する場合

```
${cosminexus.home}/manager/log/.+*
```

J2EE サーバの JavaVM のスレッドダンプを収集する場合（作業ディレクトリ=/home/workdir, J2EE サーバ名=Server1）

```
/home/workdir/ejb/Server1/javacore.+¥※.txt
```

注※

"¥"は"."（ピリオド）の打ち消し文字です。

"."（ピリオド）は、任意の文字を表します。

"+"は、1 回以上を表します。

11

監査ログの設定で使用するファイル

この章では、監査ログの設定で使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

11.1 監査ログの設定で使用するファイルの一覧

監査ログの設定で使用するファイルの一覧を，次の表に示します。

表 11-1 監査ログの設定で使用するファイルの一覧

ファイル名	分類	概要	参照先
(任意)	監査ログ定義ファイル	監査ログに関する設定をします。	11.2.1

11.2 監査ログの設定で使用するファイルの詳細

11.2.1 監査ログ定義ファイル

(1) 形式

J2SE のプロパティファイル形式です。

次のようにキーを指定します。

`<キー名称> = <値>`

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。また、行頭や行末の空白は無視されます。
- 同じ<キー名称>を複数指定した場合、最後に指定した<キー名称>の<値>が有効になります。
- 値の後ろには、コメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。
(誤りの例) <キー名称> = <値> #<コメント>
- 記載する文字は Java の仕様に従って、ISO 8859-1 文字エンコーディングを使用してください。
- <キー名称>や<値>には、「¥」を使用しないでください。また、<値>に指定するディレクトリの区切り文字は「/」を使用してください。

(2) 機能

監査ログ定義ファイルでは、監査ログ出力時のログファイルの設定、監査ログの有効無効の設定、出力を抑止するメッセージの指定など監査ログに関する設定をします。

(3) 格納先およびファイル名

インストール時のデフォルト設定ファイルを編集する場合

次のファイルを編集してください。

- Windows の場合
 <Application Server のインストールディレクトリ>%common%conf%auditlog.properties
- UNIX の場合
 /opt/Cosminexus/common/conf/auditlog.properties

任意のファイルを指定する場合

格納先名は任意です。

環境変数 COSMINEXUS_AUDITLOG_CONF に、監査ログ定義ファイルをフルパスで指定してください。

例

```
COSMINEXUS_AUDITLOG_CONF=D:/auditlog/auditlog.properties
```

(4) 設定できるキー

指定できるキーとデフォルト値を次に示します。このキーに不正な値を指定した場合、動作は保証されません。

キー名称	内容	デフォルト値
auditlog.enabled	監査ログを出力するかどうかを指定します。 true を指定した場合： 監査ログを出力します。 false を指定した場合： 監査ログを出力しません。	false
auditlog.user	監査ログファイルの所有者（ユーザ）を指定します。	<auditsetup コマンドを実行したユーザ>
auditlog.group	監査ログファイルの所有者（グループ）を指定します。	<auditsetup コマンドを実行したユーザのプライマリグループ>
auditlog.mode	監査ログファイルのアクセス権限を、777 までの 3 けたの数値で指定します※1。 UNIX の場合だけ指定します。	666
auditlog.filtered.message.list	出力抑止対象のメッセージ ID をコンマ (,) で区切って指定します。512 個まで指定できます。	なし
auditlog.filesize	監査ログファイルの 1 面当たりのファイルサイズ（単位：バイト）を、4096～33554432 で指定します。	1048576
auditlog.filenum	監査ログファイルの面数を、2～32 で指定します。	4
auditlog.directory	監査ログファイルの出力ディレクトリの絶対パスを指定します。	Windows の場合 <Application Server のインストールディレクトリ> >*auditlog UNIX の場合 /opt/Cosminexus/auditlog これらのディレクトリが存在しない場合は、新規に作成されます※2。

キー名称	内容	デフォルト値
auditlog.raslog.message.filesize	監査ログのメッセージログを出力するファイルの 1 面当たりのファイルサイズ（単位：バイト）を、4096～16777216 で指定します。	1048576
auditlog.raslog.message.fileenum	監査ログのメッセージログを出力するファイルの面数を、2～64 で指定します。	4
auditlog.raslog.message.directory	監査ログのメッセージログを出力するファイルの出力ディレクトリを絶対パスで指定します。	監査ログファイルの出力ディレクトリ（auditlog.directory）の値
auditlog.raslog.exception.filesize	監査ログの例外情報を出力するファイルの 1 面当たりのファイルサイズ（単位：バイト）を、4096～16777216 で指定します。	1048576
auditlog.raslog.exception.fileenum	監査ログの例外情報を出力するファイルの面数を、2～64 で指定します。	8
auditlog.raslog.exception.directory	監査ログの例外情報を出力するファイルの出力ディレクトリを絶対パスで指定します。	監査ログファイルの出力ディレクトリ（auditlog.directory）の値

注※1 指定値は chmod コマンドで指定する数値モードに従います。ただし、実際に設定される値は、umask に影響されます。例えば、umask=0222 の場合、auditlog.mode に 777 を指定しても、755 が設定されます。

注※2 指定したディレクトリが存在しない場合、新規に作成されるディレクトリの所有者およびアクセス権は次のように設定されます。

- Windows の場合
ディレクトリの所有者やアクセス権限には、上位ディレクトリの設定が引き継がれます。
- UNIX の場合
ディレクトリの所有者は監査ログを出力したユーザとそのプライマリグループになります。アクセス権限に 777 が設定されます。

(5) 監査ログのアクセス権限の設定

Windows と UNIX で、監査ログ定義ファイルの指定値と監査ログの所有者およびアクセス権限の対応が異なります。

- Windows の場合
監査ログ定義ファイルの指定値と監査ログファイルの所有者およびアクセス権限の対応を、次の表に示します。

表 11-2 監査ログ定義ファイルの指定値と監査ログファイルの所有者およびアクセス権限の対応（Windows の場合）

監査ログ定義ファイル		監査ログファイル	
auditlog.user	auditlog.group	所有者	アクセス権限※
—	—	上位ディレクトリの設定を引き継ぐ	Everyone:RW Administrators:F SYSTEM:F

監査ログ定義ファイル		監査ログファイル	
auditlog.user	auditlog.group	所有者	アクセス権限※
<ユーザ名>	—		<ユーザ名>:RW Administrators:F SYSTEM:F
—	<グループ名>		<グループ名>:RW Administrators:F SYSTEM:F
<ユーザ名>	<グループ名>		<ユーザ名>:RW <グループ名>:RW Administrators:F SYSTEM:F

(凡例) —：指定しない。

注※

Windows のアクセス権限を、次の形式で記述します。

<ユーザ名またはグループ名>:<許可する権限>

<許可する権限>は次のとおりです。

F：フルコントロール

RW：読み取りと書き込みを許可

• UNIX の場合

監査ログ定義ファイルの指定値と監査ログファイルの所有者およびアクセス権限の対応を、次の表に示します。

表 11-3 監査ログ定義ファイルの指定値と監査ログファイルの所有者およびアクセス権限の対応 (UNIX の場合)

監査ログ定義ファイル			監査ログファイル		
auditlog.user	auditlog.group	auditlog.mode※ ¹	所有者 (ユーザ)	所有者 (グループ)	アクセス権限※ ²
—	—	—	root※ ³	root のプライマリグループ※ ³	666※ ³
—	—	777	root※ ³	root のプライマリグループ※ ³	777
<ユーザ名>	—	—	<ユーザ名>	<ユーザ名>のプライマリグループ※ ³	666※ ³
—	<グループ名>	777	root※ ³	<グループ名>	777
<ユーザ名>	<グループ名>	777	<ユーザ名>	<グループ名>	777

(凡例) - : 指定しない。

注※1

「auditlog.mode」に指定する値を、「777」として説明しています。

注※2

実際に設定される値は umask によるマスクが実行されます。例えば、umask=0222 が設定されている場合、auditlog.mode=777 と指定しても、755 が設定されます。

注※3

デフォルト値です。

参考

auditsetup コマンドを使用しないで、監査ログファイルを出力した場合、出力ディレクトリおよびファイルのアクセス権限は、次のように設定されます。

- Windows の場合
ディレクトリおよびファイルの所有者やアクセス権限には、上位ディレクトリの設定が引き継がれます。
- UNIX の場合
ディレクトリおよびファイルの所有者やアクセス権限は、次のように設定されます。

表 11-4 ディレクトリおよびファイルの所有者とアクセス権限 (UNIX の場合)

ディレクトリ／ファイル	ログ種別	所有者	アクセス権限
出力ディレクトリ	監査ログ	実行ユーザおよびそのプライマリグループ	777※
	メッセージログ		777
	例外情報		777
出力ファイル	監査ログ		666※
	メッセージログ		666
	例外情報		666

注※ umask によるマスクが実行されます。

(例) umask=0022 が設定されている場合、777 と指定しても、755 が設定されます。ただし、サーバ管理コマンドを実行したことによって監査ログファイルが作成されたときは、umask の値として 0 が使用されます。

(6) 記述例

監査ログ定義ファイル (auditlog.properties) の記述内容を次に示します。

```
auditlog.enabled=true
#auditlog.user=
#auditlog.group=
#auditlog.mode=666
auditlog.filtered.message.list=KDJExxxxxx-E,KDJEyyyyyy-I
```

```
auditlog.directory=D:/auditlog/logs
#auditlog.filesize=1048576
#auditlog.filenum=4
#auditlog.raslog.message.directory=${auditlog.directory}
#auditlog.raslog.message.filesize=1048576
#auditlog.raslog.message.filenum=4
#auditlog.raslog.exception.directory=${auditlog.directory}
#auditlog.raslog.exception.filesize=1048576
#auditlog.raslog.exception.filenum=8
```

(7) 注意事項

監査ログ定義ファイルは監査ログを出力するすべてのプロセスから参照されるため、次の注意が必要です。

- 監査ログ出力中に監査ログ定義ファイルの格納場所や記述内容を変更しないでください。また、変更した場合には監査ログを出力するすべてのプロセスを停止したあとで、監査ログ出力機能を再セットアップしてください。
- 監査ログ定義ファイルを参照するすべてのプロセスに対して、読み込み権限を付与してください。付与していない場合、そのプロセスは監査ログ定義ファイルの読み込みに失敗して、監査ログを出力できません。
- 製品のインストール単位に一つだけ監査ログ定義ファイルを設定してください。
- 監査ログ出力ライブラリで、COSMINEXUS_AUDITLOG_CONF 環境変数を指定しないで、かつ、監査ログ定義ファイルが存在しない場合、標準エラー出力およびメッセージログおよび例外情報には、監査ログ定義ファイル読み込み失敗のメッセージは出力されません。

12

Java アプリケーションで使用するファイル

この章では、Java アプリケーションで使用するファイルの形式、格納先、機能、指定できるキーなどについて説明します。

12.1 Java アプリケーションで使用するファイルの一覧

Java アプリケーションで使用するファイルの一覧を次の表に示します。

表 12-1 Java アプリケーションで使用するファイルの一覧

ファイル名	分類	概要	参照先
usrconf.cfg	Java アプリケーション用オプション定義ファイル	Java アプリケーションを実行する JavaVM の起動オプションを指定します。 Java アプリケーションを cjclstartap コマンドで開始する場合に使用します。	12.2.1
usrconf.properties	Java アプリケーション用ユーザプロパティファイル	Java アプリケーションを実行する JavaVM のシステムプロパティを指定します。 Java アプリケーションを cjclstartap コマンドで開始する場合に使用します。	12.2.2
(任意) ※	Java アプリケーションに指定するシステムプロパティ	Java アプリケーションを実行する JavaVM のシステムプロパティを指定します。 vbj コマンドで Java アプリケーションを開始する場合に、コマンドの引数としてバッチファイルまたはシェルスクリプトに指定します。	12.2.3
(任意)	Java アプリケーションのユーザログ用プロパティ設定ファイル	Java アプリケーションのユーザログで使用します。 vbj コマンドで Java アプリケーションを開始する場合に使用します。	12.2.4

注※

Java アプリケーションの開始に使用するコマンド (vbj コマンド) ごとに、必要なシステムプロパティを指定します。システムプロパティの指定は、Java アプリケーションからの Enterprise Bean 呼び出しの場合に必要です。

Java アプリケーションを開始するには、次のどれかのコマンドを使用します。

- **cjclstartap コマンド**

通常使用するコマンドです。コマンドの格納場所を次に示します。

- Windows の場合

<Application Server のインストールディレクトリ>%CC%\client\bin\cjclstartap

- UNIX の場合

/opt/Cosminexus/CC/client/bin/cjclstartap

- **vbj コマンド**

すでにこのコマンドを使用しているユーザ向けの互換用コマンドです。コマンドの格納場所を次に示します。

- Windows の場合

<Application Server のインストールディレクトリ>%TPB%\bin\vbj

- UNIX の場合
/opt/Cosminexus/TPB/bin/vbj

12.2 Java アプリケーションで使用するファイルの詳細

12.2.1 usrconf.cfg (Java アプリケーション用オプション定義ファイル)

(1) 形式

次のようにキーを指定します。

`<キー名称> = <値>`

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) `<キー名称> = <値> #<コメント>`

- 値にスペースを含むパスを指定する場合、パスをダブルクォーテーション (") で囲む必要はありません。
- 実行時の文字エンコーディングと異なる文字エンコーディングでは記述できません。

(2) ファイルの格納先

- Windows の場合
`<ユーザ定義ファイル格納ディレクトリ>*\usrconf.cfg`
- UNIX の場合
`<ユーザ定義ファイル格納ディレクトリ>*/usrconf.cfg`

注※

ユーザ定義ファイル格納先環境変数 (CJCLUSRCONFDIR) で指定したディレクトリです。ユーザ定義ファイル格納先環境変数が設定されていない場合、カレントディレクトリが参照されます。EJB クライアントアプリケーションの実行に必要な環境変数の設定については、マニュアル「アプリケーションサーバ 機能解説 基本・開発編(EJB コンテナ)」の「3.3.4 EJB クライアントアプリケーションの実行に必要な環境変数の設定」を参照してください。

(3) 機能

cjclstartap コマンドを使用する場合に、Java アプリケーションを実行する JavaVM の起動オプションを指定します。

Java アプリケーションを実行中に、このファイルの内容を変更した場合、変更した内容は、次回 Java アプリケーションを起動したときに反映されます。

(4) 指定できるキー

指定できるキーとデフォルト値を次に示します。このキーに不正な値を指定した場合は、動作は保証されません。

ポイント

ここでは、次の内容について説明します。

- Java アプリケーション用 usrconf.cfg ファイルで指定できるキーの概要

指定できるキーのうち、J2EE サーバ用 usrconf.cfg ファイルと共通のキーについては、ここでは詳細を説明しません。

共通のキーの詳細については、「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」を参照してください。参照する場合には、「J2EE サーバ」を「Java アプリケーション」と読み替えてください。

なお、J2EE サーバ用のキーと指定内容が異なる場合には、ここでキーの詳細について説明します。

- J2EE サーバ用のキーとの差異 (J2EE サーバ用 usrconf.cfg ファイルのキーと指定内容が異なる場合)

Java アプリケーション用のキーと J2EE サーバ用のキーの指定内容が異なる場合には、キーの詳細について説明します。差異の有無は「差異」列に表記します。

表 12-2 usrconf.cfg に指定できるキーとデフォルト値 (Java アプリケーション)

キー名称	差異	内容	デフォルト値
add.jvm.arg	なし	指定されたオプションを使って JavaVM を起動します。 add.jvm.arg で指定できる JavaVM のオプションについては、 「 14.1 JavaVM 拡張オプションの一覧 」および「 14.5 Application Server で指定できる Java HotSpot VM のオプション 」 を参照してください。	デフォルト値については 「 12.2.1(6) Java アプリケーション用オプション定義ファイルに指定する JavaVM オプションのデフォルト値 」を

キー名称	差異	内容	デフォルト値
			参照してください。
add.class.path	あり	Java アプリケーションの実行に必要なクラスパスを指定します。 EJB アプリケーションごとの実行に必要なクラスパス (stubs.jar, <数字>.jar, <ユーザクラス>.jar) を指定します。 なお、スペースを含むパスを指定する場合、パスをダブルクォーテーション (") で囲む必要はありません。また、ダブルクォーテーションでの指定は有効となりません。	—
add.library.path※	なし	JNI 用の共有ライブラリを指定します。	—
cpp.library.version	なし	プロセス内で使用する libstdc++ ライブラリのバージョンを指定します。なお、このキーは、Linux 用です。 指定できる文字列を次に示します。 6 : libstdc++.so.6 ライブラリを使用します。Red Hat Enterprise Linux 以外のプラットフォームで設定しても無効です。 09-00 以降では、設定できるのは「6」だけです。	6
ejb.client.directory.shareable	独自	同じカレントディレクトリまたは、同じログ出力先ディレクトリを使用する複数の cjclstartap コマンドを同時に起動するかどうかを指定します。 true を指定した場合： 同時に起動します。 false を指定した場合： 同時に起動しません。	true
ejb.client.ejb.log	独自	ejb.client.log.directory キーに指定したディレクトリの下での、Java アプリケーションが出力するメッセージのログ出力先のディレクトリ名を、1～16 バイトで指定します。 指定できる文字は、半角英数字、アンダースコア (_), およびハイフン (-) です。 16 バイトを超える文字列を指定した場合、KDJE40051-W のメッセージが出力されます。 ディレクトリにアクセス権がない場合や、ファイル名を指定した場合は、KDJE40052-E のメッセージが出力され、Java アプリケーションが終了します。 ユーザプロパティファイル (usrconf.properties) に ejbserver.client.ejb.log を指定した場合、次のログファイルの出力先はユーザプロパティファイルの指定が優先されます。 - 稼働ログ - 障害発生時の例外情報 - 保守情報	system
ejb.client.log.appid	独自	Java アプリケーションが出力するメッセージのログ出力先のサブディレクトリ名を 1～16 バイトで指定します。	ejbcl

キー名称	差異	内容	デフォルト値
		指定できる文字は、半角英数字、アンダースコア (_), およびハイフン (-) です。 16 バイトを超える文字列を指定した場合、KDJE40051-W のメッセージが出力されます。 ディレクトリにアクセス権がない場合や、ファイル名を指定した場合は、KDJE40052-E のメッセージが出力され、Java アプリケーションが終了します。 ユーザプロパティファイル (usrconf.properties) に ejbserver.client.log.appid を指定した場合、次のログファイルの出力先はユーザプロパティファイルの指定が優先されます。 - 稼働ログ - 障害発生時の例外情報 - 保守情報	
ejb.client.log.directory	独自	Java アプリケーションのログ出力先のパスを 1~180 バイトの範囲で指定します。 パスは絶対パスまたはカレントディレクトリからの相対パスで指定します。 指定できる文字は、半角英数字、アンダースコア (_), ハイフン (-) およびパス区切り文字です。 180 バイトを超えるパス (相対パスを指定したときは、カレントディレクトリまでの絶対パスと相対パスの合計) を指定した場合、KDJE40059-W のメッセージが出力され、デフォルト値が使用されます。デフォルト値が 180 バイトを超える場合は、ログの初期化に失敗し、異常終了します。 Windows の場合、UNC 名を含むパスは指定できません。UNIX の場合、nfs マウントされたディスク上へのパスは指定できません。 ディレクトリにアクセス権がない場合や、ファイル名を指定した場合は、KDJE40052-E のメッセージが出力され、Java アプリケーションが終了します。 ユーザプロパティファイル (usrconf.properties) に ejbserver.client.log.directory を指定した場合、次のログファイルの出力先はユーザプロパティファイルの指定が優先されます。 - 稼働ログ - ログ稼働ログ - 障害発生時の例外情報 - 保守情報	カレントディレクトリ/logs
ejb.client.log.lockInterval	独自	次に示す対象ログファイルの排他処理のパラメタを指定します。 - cjclstartap コマンドの稼働ログ - 起動プロセス標準出力情報 - 起動プロセス標準エラー情報 排他処理には、UNIX と Windows で違いがあります。 UNIX の場合 ログファイルの排他取得に失敗したときのリトライ間隔 (単位: ミリ秒) を 1~2147483647 の整数値で指定します。	30

キー名称	差異	内容	デフォルト値
		Windows の場合 1～2147483647 の整数値を指定します。指定した ejb.client.log.lockRetryCount の値と ejb.client.log.lockInterval の値を掛けた値が排他取得のタイムアウト時間（単位:ミリ秒）になります。掛けた値が 2147483647 を超えた場合、ejb.client.log.lockRetryCount と ejb.client.log.lockInterval のデフォルト値を掛けた値が排他取得のタイムアウト時間になります。 UNIX/Windows 共通 範囲外の値や整数値以外（整数値に変換不能な値）を指定した場合は、デフォルト値を使用します。 また、複数の cjclstartap コマンドをサブディレクトリ共有モードで同時に起動した際、cjclstartap コマンドが、KDJE40016-E (errno が 22), KDJE30043-E, KDJE40052-E メッセージを標準エラー出力に出力した場合（ログファイルの排他処理に失敗した場合）などに使用します。	
ejb.client.log.lockRetryCount	独自	次に示す対象ログファイルの排他処理のパラメタを指定します。 • cjclstartap コマンドの稼働ログ • 起動プロセス標準出力情報 • 起動プロセス標準エラー情報 排他処理には、UNIX と Windows で違いがあります。 UNIX の場合 ログファイルの排他取得の総試行数（単位：回）を 1～2147483647 の整数値で指定します。 Windows の場合 1～2147483647 の整数値を指定します。指定した ejb.client.log.lockRetryCount の値と ejb.client.log.lockInterval の値を掛けた値が排他取得のタイムアウト時間（単位:ミリ秒）になります。掛けた値が 2147483647 を超えた場合、ejb.client.log.lockRetryCount と ejb.client.log.lockInterval のデフォルト値を掛けた値が排他取得のタイムアウト時間になります。 UNIX/Windows 共通 範囲外の値や整数値以外（整数値に変換不能な値）を指定した場合は、デフォルト値を使用します。 また、複数の cjclstartap コマンドをサブディレクトリ共有モードで同時に起動した際、cjclstartap コマンドが、KDJE40016-E (errno が 22), KDJE30043-E, KDJE40052-E メッセージを標準エラー出力に出力した場合（ログファイルの排他処理に失敗した場合）などに使用します。	100
ejb.client.log.stdout.enabled	独自	標準出力へメッセージを出力するかどうかを指定します。 true : 標準出力へ出力します。	true

キー名称	差異	内容	デフォルト値
		false : 標準出力へ出力しません。 標準出力へのメッセージ出力を制御できるログは次のとおりです。 - 稼働ログ - cjclstartap コマンドログ - 起動プロセス標準出力情報	
jvm.type	独自	使用する JavaVM のタイプを設定します。指定できる値は次のどちらかです。 - client Java HotSpot Client VM が使用されます。 - server Java HotSpot Server VM が使用されます。 指定値に誤りがあった場合は," client"," server" の順で JavaVM を検索します。なお、このとき KDJE40020-W メッセージが出力されます。	client

(凡例)

あり：

Java アプリケーション用 usrconf.properties ファイルのキーと J2EE サーバ用 usrconf.properties ファイルのキーに差異があります。詳細については、「内容」の列を参照してください。

なし：

Java アプリケーション用 usrconf.cfg ファイルと J2EE サーバ用 usrconf.cfg ファイルで共通のキー、またはデフォルト値です。キーの詳細については、「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」を参照してください。

独自：

Java アプリケーション用 usrconf.cfg ファイル固有のキーです。キーの詳細については、「内容」の列を参照してください。

ー：

Java アプリケーション用 usrconf.cfg ファイルと J2EE サーバ用 usrconf.cfg ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」を参照してください。

注※

cjclstartap コマンドの実行時に add.library.path と OS の環境変数 (LIBPATH など) が両方設定されていた場合、add.library.path の設定が優先されます。

(5) 記述例

```
# EJB client application classpath
# RMI-IIOP Stubs jar file
add.class.path=<Javaアプリケーション格納パス>%stubs.jar

# RMI-IIOP interface jar file
add.class.path=<Javaアプリケーション格納パス>%1.jar

# User class
add.class.path=<Javaアプリケーション格納パス>%<Javaアプリケーションjar>
#add.class.path=<User classpath>

# java vm options
```

```
add.jvm.arg=-Xms256m  
add.jvm.arg=-Xmx512m
```

(6) Java アプリケーション用オプション定義ファイルに指定する JavaVM オプションのデフォルト値

Java アプリケーション用オプション定義ファイルに指定する JavaVM オプションのデフォルト値を示します。

- -XX:HitachiJavaLog:<ejb.client.log.directory>¥<ejb.client.ejb.log>¥<ejb.client.log.appid>
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:-HitachiThreadDumpToStdout
- -XX:+HitachiOutOfMemoryAbort
- -XX:+HitachiJavaClassLibTrace
- -XX:+HitachiLocalsInStackTrace
- -XX:HitachiJavaClassLibTraceLineSize=1024
- -XX:+HitachiLocalsSimpleFormat
- -XX:+HitachiTrueTypeInLocals
- -XX:+HitachiLocalsInStackTrace
- -XX:+HitachiVerboseGC
- -XX:+HitachiVerboseGCPrintCause
- -XX:+HitachiOutputMilliTime

(7) 注意事項

- OS のロケールと異なる文字エンコーディングで記述しないでください。また、OS のロケールに UTF-8 を使用する場合、BOM 付き UTF-8 では記述しないでください。
- usrconf.properties と usrconf.cfg の add.jvm.arg の -D に同じキーが指定されている場合は、usrconf.properties の値が優先されます。
- add.class.path キーに指定されている値の中に、<cosminexus.home>タグを使用してコンテナ拡張ライブラリ用の JAR を指定しているものがあります。これは Application Server のインストール先を意味しているため、add.class.path キーの記述内容を変更する必要はありません。なお、add.class.path キーにコンテナ拡張ライブラリを追加する場合は、インストールおよびアンインストール時の動作を保証できなくなるおそれがあるので、<cosminexus.home>タグを使用しないでください。
- 次に示すプロパティは"-D"で指定できません。

java.endorsed.dirs

java.security.policy

java.class.path

java.library.path

- Java EE の機能を使用する Java アプリケーションの場合、add.jvm.arg キーで--add-modules および--module-path オプションを指定しないでください。指定した場合は、動作を保証しません。
- Java EE の機能を使用しない Java アプリケーションの場合、add.jvm.arg キーで--add-modules, および--module-path オプションを指定して、モジュール化したライブラリを使用できます。
- JDK17 以降のデフォルトの GC は、通常は G1GC が選択されます。デフォルトの GC の選択方式についてはマニュアル「アプリケーションサーバ 機能解説 保守／移行編」の「9.19.1 デフォルトで選択されるメモリ管理方式」を参照してください。JDK11 以前の SerialGC を使用したい場合は、-XX:+UseSerialGC を設定してください。
- Windows で使用する場合、JDK21 より前のバージョンでは Java のデフォルトの文字コードは Shift_JIS でしたが、JDK21 以降のバージョンからは UTF-8 に変更されました。ただし、usrconf.cfg ファイルについては、JDK のバージョンに関係なく Shift_JIS で記述してください。

12.2.2 usrconf.properties (Java アプリケーション用ユーザプロパティファイル)

(1) 形式

J2SE のプロパティファイル形式です。

次のようにキーを指定します。

<キー名称> = <値>

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) <キー名称> = <値> #<コメント>

- 記載する文字は Java の仕様に従って、ISO 8859-1 文字エンコーディングを使用してください。
- Java の仕様に従わない形式の場合、Java アプリケーションの起動に失敗することがあります。
(例) 不正な Unicode エスケープシーケンスを含む場合 ("¥u"に続く文字列が適切な Unicode 16 進数の値ではない)。

(2) ファイルの格納先

- Windows の場合
＜ユーザ定義ファイル格納ディレクトリ※＞%usrconf.properties
- UNIX の場合
＜ユーザ定義ファイル格納ディレクトリ※＞/usrconf.properties

注※

ユーザ定義ファイル格納先環境変数（CJCLUSRCONFDIR）で指定したディレクトリです。ユーザ定義ファイル格納先環境変数が設定されていない場合、カレントディレクトリが参照されます。EJB クライアントアプリケーションの実行に必要な環境変数の設定については、マニュアル「アプリケーションサーバ 機能解説 基本・開発編(EJB コンテナ)」の「3.3.4 EJB クライアントアプリケーションの実行に必要な環境変数の設定」を参照してください。

(3) 機能

cjclstartap コマンドを使用する場合に、Java アプリケーションを実行する JavaVM のシステムプロパティを指定します。

usrconf.properties と usrconf.cfg の add.jvm.arg の-D に同じキーが指定されている場合は、usrconf.properties の値が優先されます。

Java アプリケーションを実行中に、このファイルの内容を変更した場合、変更した内容は、次回 Java アプリケーションを起動したときに反映されます。

(4) Java アプリケーションのカスタマイズ用キー

Java アプリケーションの開始コマンドでは、次に示すシステムプロパティのキーに値を設定することで、Java アプリケーションの動作をカスタマイズできます。

ここでは、キーを次の分類に分けて説明します。

- ejbserver.application から始まるキー
- ejbserver.client から始まるキー
- ejbserver.container から始まるキー
- ejbserver.distributedtx から始まるキー
- ejbserver.jndi から始まるキー
- ejbserver.jta から始まるキー
- ejbserver.logger から始まるキー
- ejbserver.rmi から始まるキー
- ejbserver.security から始まるキー

- ejbserver.server から始まるキー
- ejbserver.stdoutlog から始まるキー
- java から始まるキー
- org から始まるキー
- vbroker から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できるキーを次の表に示します。

注意事項

ここでは、次の内容について説明します。

- **Java アプリケーション用 usrconf.properties ファイルで指定できるキーの概要**
指定できるキーのうち、J2EE サーバ用 usrconf.properties ファイルと共通のキーについては、ここでは詳細を説明しません。
共通のキーの詳細については、「[2.2.3 usrconf.properties \(J2EE サーバ用ユーザプロパティファイル\)](#)」を参照してください。Java アプリケーションの場合には、「J2EE サーバ」を「Java アプリケーション」と読み替えてください。
なお、J2EE サーバ用のキーと指定内容が異なる場合には、ここでキーの詳細についても説明します。
- **J2EE サーバ用のキーとの差異 (J2EE サーバ用 usrconf.properties ファイルのキーと指定内容が異なる場合)**
Java アプリケーション用のキーと J2EE サーバ用のキーの指定内容が異なる場合には、キーの詳細について説明します。差異に関する記述が特にない場合には、J2EE サーバ用 usrconf.properties ファイルのキーと同じ内容を指定してください。差異の有無は「差異」列に表記します。
- **Java アプリケーション用 usrconf.properties ファイル独自のキーの詳細**
Java アプリケーション用 usrconf.properties ファイル独自のキーについては、ここでキーの詳細を説明します。

(a) ejbserver.application から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.application から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(a\) ejbserver.application から始まるキー](#)」を参照してください。

キー名称	差異	内容	デフォルト値
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>* ¹ .appname	なし	AppName のフィールドに出力するデフォルトのアプリケーション名を指定します。	—

キー名称	差異	内容	デフォルト値
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>※ 1.autoFlush.enabled	独自	メッセージ出力後に flush をするかどうかを指定します。 true を指定した場合： メッセージごとに flush をします。 false を指定した場合： flush をしません。 CJMPMessageFileHandler を使用しない場合、指定は不要です。ま た、アプリケーションプログラムで明示的に java.util.logging.Handler.flush() を呼び出している場合には、true を指定しないでください。	false
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>※1.count	なし	ログファイルの面数を次の範囲の整数で指定します。 CJMessageFileHandler を使用した場合： 2～16 CJMPMessageFileHandler を使用した場合： 2～64 ここで指定した面数に到達すると、次の番号は初めの"1"に戻ります。 同名の旧ログファイル、または同名のユーザファイルがあった場合は 上書きされます。	—
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>※1.encoding	なし	出力する文字列のエンコーディングを指定します。	—
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>※1.filter	なし	使用するフィルタを、パッケージ名を含むフルネームで指定します。	—
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>※1.formatter	なし	使用するフォーマッタを、パッケージ名を含むフルネームで指定しま す。	—
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>※1.level	なし	ログの取得レベルの上限を指定します。	—
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>※1.limit	なし	ログファイルの容量（単位：バイト）を次の範囲の整数で指定します。 CJMessageFileHandler を使用した場合： 8192～2147483647 CJMPMessageFileHandler を使用した場合： 8192～16777216 ここで指定した容量に到達すると次のログファイルに切り替わりま す。トレース共通ライブラリの場合、容量チェックがメッセージの書 き込み前であるため、実際の最大容量が設定値より、最大で約 4200 バイト超えることがあります（1 バイトでも指定した容量に満たなけ れば、メッセージが書き込まれてしまうため）。例えば、このキーに 8192 を指定した場合で out1.log の現在使用中の領域が 8000 バイ ト、空き容量が 192 バイトのときに 4000 バイトのメッセージが出	—

キー名称	差異	内容	デフォルト値
		力されると、メッセージは out1.log に出力されます。このため、out1.log は上限値を超えたサイズになります。	
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>*1.msgid	なし	MsgID のフィールドに出力するデフォルトのメッセージ ID を指定 します。	—
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>*1.path	あり	ログファイル名のプリフィックスを 1～255 バイトの範囲で指定しま す。パスは、相対パスで指定します。なお、パス名の終端に、半角数 値を指定しないでください。 ここで指定したプリフィックスに"xx (1～16 の整数).log"が付加さ れた値が、トレース情報ファイルの名称になります。 パスに指定された半角英文字の大文字と小文字は区別されません。ま た、パス値の長さは、次の A+B+C 値の合計が値の範囲に収まるよ う指定してください。 • A (Windows の場合)：<ログ出力先ルート (ejbserver.client.log.directory の値)>%user% (デフォルトは <Application Server のインストールディレクトリ >%CC%client%logs%user%) • A (UNIX の場合)：<ログ出力先ルート (ejbserver.client.log.directory の値)>/user/ (デフォルト は/opt/Cosminexus/CC/client/logs/user/) • B：ユーザの指定したプリフィックスの長さ • C："xx.log"の 6 文字 このほか、パスを指定するときには次の点に注意してください。 • Properties ファイルには native の 2 バイトコードなどは使用で きないため、native2ascii などを利用して Unicode の文字列を 取得して指定してください。 • 絶対パスで指定しないでください。また、相対パスの"..%" (UNIX の場合, "../") は使用しないでください。 • パスに日本語文字などを含む場合は、その文字を処理できるロケ ール設定がされている環境下で JavaVM が動作している必要があり ます。 パスをほかのハンドラと同一に設定することはできません。そのよ うな設定にした場合、そのハンドラは作成されません。	—
ejbserver.application.us erlog.CJLogHandler.<ハ ンドラ名称>*1.separator	なし	CJSimpleFormatter を利用して、1 文でメッセージを出力するた めのセパレータを指定します。	—
ejbserver.application.us erlog.loggers	なし	使用するロガーの名称を宣言します。	—
ejbserver.application.us erlog.Logger.<ロガー名 称>*2.filter	なし	ロガーで使用するフィルタを、パッケージ名を含むフルネームで指定 します。	—

キー名称	差異	内容	デフォルト値
ejbserver.application.userlog.Logger.<ロガー名称> ^{※2} .handlers	なし	指定のロガー名称に対して、ハンドラクラスを接続 (addHandler(Handler)) するためのプロパティです。ハンドラクラスは、Reflection 機能で作成するため、0~65535 バイトのパッケージ名を含むフルネームで指定してください。","で区切ること複数のハンドラクラスを指定できます。指定できるハンドラクラスを次に示します。 - com.hitachi.software.ejb.application.userlog.CJMessageFileHandler - com.hitachi.software.ejb.application.userlog.CJMPMessageFileHandler - java.util.logging.ConsoleHandler - java.util.logging.FileHandler - java.util.logging.SocketHandler - java.util.logging.StreamHandler - java.util.logging.MemoryHandler - ユーザが独自に作成した Handler クラスのパッケージ名を含むフルネーム CJLogHandler 系の場合は、ハンドラクラス名の後ろに","で区切ってハンドラ名称を指定することで、指定のハンドラ名称の設定で初期化してハンドラクラスを作成し、ロガーに接続します。Java2-1.4 標準の Logging クラスについては、","で区切ってハンドラ名称を指定しても無効になります。Java2-1.4 標準の Logging クラスの設定方法については、該当するクラスの仕様に従ってください。 ハンドラ名称は、半角英数字で始まる 1~1024 バイトのクラス文字セットで指定してください。 複数のクラス名を定義しており、指定された値の長さが範囲を超えた場合は、長さの範囲内に収まり、値の規則を満たしたクラス名だけが設定されます。ejbserver.application.userlog.loggers でロガー名称が指定されていない場合、ロガーは作成されません。	—
ejbserver.application.userlog.Logger.<ロガー名称> ^{※2} .level	なし	ロガーのログ出力レベルを指定します。	—
ejbserver.application.userlog.Logger.<ロガー名称> ^{※2} .useParentHandlers	なし	ログレコードを受け取ったロガーから、親ロガーの接続しているハンドラへログレコードを伝播させるかどうかを指定します。	—

(凡例)

あり：

Java アプリケーション用 usrconf.properties ファイルのキーと J2EE サーバ用 usrconf.properties ファイルのキーに差異があります。詳細については、「内容」の列を参照してください。

なし：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のキーです。キーの詳細については、「[2.2.3\(5\)\(a\) ejbserver.application から始まるキー](#)」を参照してください。

独自：
Java アプリケーション用 usrconf.properties ファイル固有のキーです。キーの詳細については、「内容」の列を参照してください。

—：
Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「2.2.3(5)(a) ejbserver.application から始まるキー」を参照してください。

注※1
ハンドラ名称はプロパティの値を区別するために使用します。半角英数字で始まる 1～1024 バイトのクラス文字セットで指定してください。"default"を指定した場合は、すべての CJLogHandler が共通のデフォルト値として認識するプロパティを示します。

注※2
ロガー名称は、Logger.getLogger(<ロガー名称>) でロガーのインスタンスを取得するときに指定する"."付きの名称で、あらかじめ、ejbserver.application.userlog.loggers プロパティで宣言する必要があります。ロガー名称については、java.util.logging.Logger の仕様に従ってください。

半角英数字で始まる 1～1024 バイトのクラス文字セットで指定してください。"default"を指定した場合は、すべての CJLogHandler が共通のデフォルト値として認識するプロパティを示します。

なお、ロガー名称で識別されるプロパティは、ユーザログ機能のロガークラスの初期化で使用されます。

ロガー名称の最後を.handlers で終了させた場合、ejbserver.application.userlog.Logger.<ロガー名称>.handlers のプロパティ指定時に紛らわしくなるため、ロガー名称は".handlers"で終了させないことを推奨します。

(b) ejbserver.client から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.client から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「2.2.3(5)(c) ejbserver.client から始まるキー」を参照してください。

なお、Client を使用する場合は、格納ディレクトリの「<Application Server のインストールディレクトリ>¥CC」を、「<Application Server のインストールディレクトリ>¥CCL」と読み替えてください。

キー名称	差異	内容	デフォルト値
ejbserver.client.ctm.RequestPriority	なし	CTM 内のキューに滞留したリクエストを引き出す時のプライオリティ（優先順位）を指定します。	—
ejbserver.client.ejb.log	独自	ejbserver.client.log.directory キーに指定したディレクトリの下 の Java アプリケーションが出力するメッセージのログ出力先のディ レクトリ名を 1～16 バイトの範囲で指定します。指定できる文字は、 半角英数字、アンダースコア (_), およびハイフン (-) だけです。 ディレクトリのパス区切り文字には、スラッシュ (/) を使用してく ださい。複数の Java アプリケーションを使用する場合は、EJB アプ リケーションごとに、ログ出力先を指定してください。 アクセス権がないディレクトリまたはファイル名を指定した場合は、 KDJE51003-E のメッセージが標準出力と cjlogger.log ファイルに出 力され、Java アプリケーションが終了します。 このキーでは出力先を変更できないログがあります。ログの出力先を 変更するには、Java アプリケーション用オプション定義ファイル (usrconf.cfg) の ejb.client.ejb.log キーを指定してください。	なし

キー名称	差異	内容	デフォルト値
ejbserver.client.log.appid	独自	Java アプリケーションのログ出力先のサブディレクトリ名を 1～16 バイトで指定します。 指定できる文字は、半角英数字、アンダースコア (_), およびハイフン (-) だけです。このキーを指定した場合は、毎回同じサブディレクトリが使用されるため、サブディレクトリの削除はされません。一つの Java アプリケーションを複数起動しない場合に使用すると、専用のサブディレクトリでログファイルを管理できます。複数同時に起動する場合、このキーは指定しないでください。 16 バイトを超える文字列を指定した場合は、KDJE51002-W のメッセージが cjlogger.log ファイルに出力され、通常値が使用されます。このキーでは出力先を変更できないログがあります。ログの出力先を変更するには、Java アプリケーション用オプション定義ファイル (usrconf.cfg) の ejb.client.log.appid キーを指定してください。	なし
ejbserver.client.log.directory	独自	Java アプリケーションのログ出力先の絶対パスを 1～200 バイトで指定します。 指定できる文字は、半角英数字、アンダースコア (_), ハイフン (-), およびスペース () だけです。なお、スペースを指定できるのは Windows の場合だけです。空白文字*を含むディレクトリを指定する場合は、パス名を引用符で囲んでください。例えば、C:¥logs¥ejb client を設定する場合は、次のように指定します。 ejbserver.client.log.directory="C:¥logs¥ejb client" 200 バイトを超えるパスを指定した場合は KDJE51001-W のメッセージ、指定がない場合は KDJE51002-W のメッセージが cjlogger.log ファイルに出力され、通常値が使用されます。 ディレクトリにアクセス権がない場合や、ファイル名を指定した場合は、KDJE51003-E のメッセージが標準出力と cjlogger.log ファイルに出力され、Java アプリケーションが終了します。 通常値を使用する場合で、Developer's Kit for Java 以外の Java を使用して Java アプリケーションを起動した場合は、KDJE51004-W のメッセージが cjlogger.log ファイルに出力され、Java アプリケーション実行ディレクトリが使用されます。 Windows の場合、UNC 名を含むパスは指定できません。UNIX の場合、nfs マウントされたディスク上へのパスは指定できません。 このキーでは出力先を変更できないログがあります。ログの出力先を変更するには、Java アプリケーション用オプション定義ファイル (usrconf.cfg) の ejb.client.log.directory キーを指定してください。	なし
ejbserver.client.log.lockInterval	独自	ログファイルの排他処理に失敗したときのリトライ間隔 (単位: ミリ秒) を 0～2147483647 の整数値で指定します。範囲外の値や整数値以外を指定した場合は、KDJE51002-W のメッセージが cjlogger.log ファイルに出力されて、デフォルト値が使用されます。サブディレクトリ共有モードで動作する Java アプリケーションが cjlogger.log ファイルに KDJE90002-E メッセージを出力して終了する場合に使用します。	10
ejbserver.client.log.lockRetryCount	独自	ログファイルの排他処理に失敗したときのリトライ回数 (単位: 回) を 0～2147483647 の整数値で指定します。範囲外の値や整数値以外を指定した場合は、KDJE51002-W のメッセージを cjlogger.log	1000

キー名称	差異	内容	デフォルト値
		ファイルに出力して、デフォルト値を使用します。サブディレクトリ共有モードで動作する Java アプリケーションが <code>cjlogger.log</code> ファイルに <code>KDJE90002-E</code> メッセージを出力して終了する場合に使用します。	
<code>ejbserver.client.transaction.clientName</code>	独自	トランザクションサービスが使用するクライアント名を 1～128 文字の文字列で指定します。使用できる文字は、半角英数字 (0～9, A～Z, a～z)、アンダースコア (<code>_</code>) またはハイフン (<code>-</code>) です。 ここに設定するクライアント名は、Java アプリケーションのプロセスごとに別の名前を設定する必要があります。 また、同一マシン上で動作する J2EE サーバ名とは別の名前にする必要があります。	なし
<code>ejbserver.client.transaction.enabled</code>	独自	Java アプリケーションでトランザクションを使用するかどうかを指定します。 true を指定した場合： トランザクションを使用できます。 false を指定した場合： トランザクションを使用できません。	false

(凡例)

なし：

Java アプリケーション用 `usrconf.properties` ファイルと J2EE サーバ用 `usrconf.properties` ファイルで共通のキーです。キーの詳細については、「[2.2.3\(5\)\(c\) ejbserver.client から始まるキー](#)」を参照してください。

独自：

Java アプリケーション用 `usrconf.properties` ファイル固有のキーです。キーの詳細については、「内容」の列を参照してください。

－：

Java アプリケーション用 `usrconf.properties` ファイルと J2EE サーバ用 `usrconf.properties` ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(c\) ejbserver.client から始まるキー](#)」を参照してください。

注※

空白文字とは、半角スペース、タブ、LF (0x0a)、CR (0x0d) または FF (0x0c) のことを指します。

(c) ejbserver.container から始まるキー

Java アプリケーション用 `usrconf.properties` ファイルで指定できる `ejbserver.container` から始まるキーを次に示します。J2EE サーバ用 `usrconf.properties` ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(h\) ejbserver.container から始まるキー](#)」を参照してください。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	差異	内容	デフォルト値	関連情報
<code>ejbserver.container.rebindpolicy</code>	なし	EJB メソッドの呼び出し時に通信障害が発生した場合の、EJB クライアント側でのコネクションの再接続動作とリクエストの再送動作を指定します。	－	「機能解説 基本・開発編(EJB コンテナ)」の「2.13 EJB の

キー名称	差異	内容	デフォルト値	関連情報
				リモートインタフェースの呼び出し

(凡例)

なし：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のキーです。キーの詳細については、「[2.2.3\(5\)\(h\) ejbserver.container から始まるキー](#)」を参照してください。

－：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(h\) ejbserver.container から始まるキー](#)」を参照してください。

(d) ejbserver.distributedtx から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.distributedtx から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(k\) ejbserver.distributedtx から始まるキー](#)」を参照してください。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

キー名称	差異	内容	デフォルト値	関連情報
ejbserver.distributedtx.ots.status.directory1	あり	インプロセストランザクションサービスのステータスファイル、および保守資料としてのステータスファイルのバックアップを格納するディレクトリを指定します。パス区切り文字は” / ” を使用してください。 アプリケーションサーバで相対パスを指定した場合、次のディレクトリからのパスとなります。 - Windows の場合 <製品の作業ディレクトリ>¥ejb¥<サーバ名称> - UNIX の場合 <製品の作業ディレクトリ>/ejb/<サーバ名称> Java アプリケーションで相対パスを指定した場合、Java アプリケーションの実行ディレクトリからのパスとなります。 同一マシン内で稼働するほかの J2EE サーバや、ほかの Java アプリケーションプロセスと重ならないように、別のディレクトリを指定する必要があります。	なし	「機能解説 基本・開発編(コンテナ共通機能)」の「3.4 トランザクション管理」
ejbserver.distributedtx.ots.status.directory2	なし	インプロセストランザクションサービスのステータスファイルを二重化する場合、予	－	「機能解説 基本・開発編(コンテナ

キー名称	差異	内容	デフォルト値	関連情報
		備ステータスファイルおよび保守資料としての予備ステータスファイルのバックアップを格納するディレクトリを指定します。		共通機能)」の「3.4 トランザクション管理」
ejbserver.distributedtx.recovery.port	あり	トランザクションリカバリで使用する固定ポート番号を 1～65535 の整数値で指定します。無効な値を指定した場合や指定したポートがすでに使用されていた場合には、初期化処理がエラーとなります。	なし	

(凡例)

あり：

Java アプリケーション用 usrconf.properties ファイルのキーと J2EE サーバ用 usrconf.properties ファイルのキーに差異があります。詳細については、「内容」の列を参照してください。

なし：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のキーです。キーの詳細については、「[2.2.3\(5\)\(k\) ejbserver.distributedtx から始まるキー](#)」を参照してください。

—：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(k\) ejbserver.distributedtx から始まるキー](#)」を参照してください。

空欄：

関連情報はありません。

(e) ejbserver.jndi から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.jndi から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(s\) ejbserver.jndi から始まるキー](#)」を参照してください。

キー名称	差異	内容	デフォルト値
ejbserver.jndi.cache	あり	分散オブジェクトのキャッシングをするかどうかを指定します。 キャッシングをする場合： on を指定します。 キャッシングをしない場合： off を指定します。 また、javax.rmi.CORBA.PortableRemoteObjectClass キーに com.hitachi.software.ejb.rmi.PortableRemoteObject を、 javax.rmi.CORBA.UtilClass キーに com.hitachi.software.ejb.rmi.Util を指定する必要があります。	off
ejbserver.jndi.cache.interval	あり	分散オブジェクトのキャッシングをする場合、キャッシュをクリアする間隔（単位：秒）を 0～2147483647 の整数値で指定します。ただし、1～2147483647 以外の値を指定した場合、クリアしません。また、javax.rmi.CORBA.PortableRemoteObjectClass キーに com.hitachi.software.ejb.rmi.PortableRemoteObject を、	—

キー名称	差異	内容	デフォルト値
		javax.rmi.CORBA.UtilClass キーに com.hitachi.software.ejb.rmi.Util を指定する必要があります。 ネーミングサービスの障害検知機能を使用している場合は、短い値 (60 秒) を指定することを推奨します。	
ejbserver.jndi.cache.interval.clear.option	なし	インターバル経過後のネーミングでのキャッシュ領域に対する動作を決定します。	—
ejbserver.jndi.namingservice.group.list	なし	ラウンドロビン検索をする場合に、検索対象になる論理ネーミングサービスのグループを定義します。	—
ejbserver.jndi.namingservice.group.<Specify group name>.providerurls	なし	各グループに属するネーミングサービスのルート位置を指定します。	—
ejbserver.jndi.request.timeout	なし	ネーミングサービスとの通信タイムアウト時間を指定します。	—

(凡例)

あり：

Java アプリケーション用 usrconf.properties ファイルのキーと J2EE サーバ用 usrconf.properties ファイルのキーに差異があります。詳細については、「内容」の列を参照してください。

なし：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のキーです。キーの詳細については、「[2.2.3\(5\)\(s\) ejbserver.jndi から始まるキー](#)」を参照してください。

—：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(s\) ejbserver.jndi から始まるキー](#)」を参照してください。

(f) ejbserver.jta から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.jta から始まるキーを次に示します。

キー名称	差異	内容	デフォルト値
ejbserver.jta.TransactionManager.defaultTimeout	あり	Java アプリケーションで開始されるトランザクションのトランザクションタイムアウトのデフォルト値（単位：秒）を次の範囲で指定します。 1～2147483647	—

(凡例)

あり：

Java アプリケーション用 usrconf.properties ファイルのキーと J2EE サーバ用 usrconf.properties ファイルのキーに差異があります。詳細については、「内容」の列を参照してください。

—：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(u\) ejbserver.jta から始まるキー](#)」を参照してください。

(g) ejbserver.logger から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.logger から始まるキーを次に示します。

キー名称	差異	内容	デフォルト値
ejbserver.logger.channel s.define.<チャンネル名> ※.filenum	あり	Java アプリケーションのログファイルの面数を指定します。 1～64 の整数値で指定します。 面数を変更する場合は、該当するログファイルに出力しているプロセスをすべて停止し、ログファイルと mmap ディレクトリ以下のログ管理ファイルを別のディレクトリへ移動、または削除してください。	2
ejbserver.logger.channel s.define.<チャンネル名> ※.filesize	あり	4096～16777216 の整数値で指定します。 サイズを変更する場合には、該当するログファイルに出力しているプロセスをすべて停止し、ログファイルと mmap ディレクトリ以下のログ管理ファイルを別のディレクトリへ移動、または削除してください。	1048576
ejbserver.logger.enabled .*	あり	ログレベルを指定します。Error, Warning, Information, Debug のうち、一つまたは複数指定します。 一つだけ指定した場合は、該当するログレベルのログだけが出力されます。複数指定する場合は、レベル名の文字列の間をコンマ (,) で区切ります。 なお、シェルスクリプトを使用してシステムプロパティを設定する場合、このキーは指定できません。	Error

(凡例)

あり：

Java アプリケーション用 usrconf.properties ファイルのキーと J2EE サーバ用 usrconf.properties ファイルのキーに差異があります。詳細については、「内容」の列を参照してください。

注※

チャンネル名称として次に示す名称を設定できます。

ClientMessageLogFile, ClientExceptionLogFile, ClientMaintenanceLogFile, EJBContainerLogFile, UserOutLogFile, UserErrLogFile

資料の取得については、マニュアル「アプリケーションサーバ 機能解説 保守／移行編」の「2.3 資料の取得」を参照してください。

(h) ejbserver.rmi から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.rmi から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(z\) ejbserver.rmi から始まるキー](#)」を参照してください。

キー名称	差異	内容	デフォルト値
ejbserver.rmi.request.timeout	なし	クライアントとサーバ間の通信タイムアウト時間（単位：秒）を指定します。	—

(凡例)

なし：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のキーです。キーの詳細については、「[2.2.3\(5\)\(z\) ejbserver.rmi から始まるキー](#)」を参照してください。

ー：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(z\) ejbserver.rmi から始まるキー](#)」を参照してください。

(i) ejbserver.security から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.security から始まるキーを次に示します。

キー名称	差異	内容	デフォルト値
ejbserver.security.service.url	独自	セキュリティ認証処理をする J2EE サーバが接続されている CORBA ネーミングサービスを指定します。java.naming.provider.url で指定されたネーミングサービス以外のネーミングサービスに接続している J2EE サーバで認証処理をする場合に指定します。なお、接続する J2EE サーバ名は、ejbserver.serverName キーに指定しておく必要があります。 corbaname 形式で指定してください。省略した場合は、java.naming.provider.url で指定されたネーミングサービスが使用されます。 (指定形式) corbaname::<ネーミングサービスのホスト名>:<ネーミングサービスのポート番号>	なし

(凡例)

独自：

Java アプリケーション用 usrconf.properties ファイル固有のキーです。キーの詳細については、「内容」の列を参照してください。

(j) ejbserver.server から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.server から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(ab\) ejbserver.server から始まるキー](#)」を参照してください。

キー名称	差異	内容	デフォルト値
ejbserver.serverName	独自	Java アプリケーションが接続する J2EE サーバのサーバ名称を指定します。	なし
ejbserver.server.prf.PRFD	なし	PRF 識別子を指定します。	ー

(凡例)

なし：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のキーです。キーの詳細については、「[2.2.3\(5\)\(ab\) ejbserver.server から始まるキー](#)」を参照してください。

独自：

Java アプリケーション用 usrconf.properties ファイル固有のキーです。キーの詳細については、「内容」の列を参照してください。

－：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(ab\) ejbserver.server から始まるキー](#)」を参照してください。

(k) ejbserver.stdoutlog から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる ejbserver.stdoutlog から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(ad\) ejbserver.stdoutlog から始まるキー](#)」を参照してください。

キー名称	差異	内容	デフォルト値
ejbserver.stdoutlog.auto flush	なし	ユーザ出力ログおよびユーザエラーログの自動フラッシュ機能を有効にするかどうかを指定します。	－

(凡例)

なし：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のキーです。キーの詳細については、「[2.2.3\(5\)\(ad\) ejbserver.stdoutlog から始まるキー](#)」を参照してください。

－：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(ad\) ejbserver.stdoutlog から始まるキー](#)」を参照してください。

(l) java から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる java から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(ag\) java から始まるキー](#)」を参照してください。

キー名称	差異	内容	デフォルト値
java.naming.factory.initial	あり	JNDI の InitialContext がデレゲートするコンテキストの実装クラスのファクトリクラスを指定します。各設定値での動作を次に示します。 com.hitachi.software.ejb.jndi.InsContextFactory : JNDI ラウンドロビン検索機能を使用しないで、通常の検索機能を使用する場合に指定するコンテキストです。 com.hitachi.software.ejb.jndi.GroupContextFactory : J2EE サーバ内アプリケーション (EJB) の実行時に、JNDI ラウンドロビン検索機能を使用する場合に指定するコンテキストです。 なお、旧バージョンで使用されていた値 (com.hitachi.software.ejb.jndi.spi.cosnaming.CNCtxFactory) と互換性があります。	－
java.naming.provider.url	独自	Java アプリケーションが利用する CORBA ネーミングサービスのホスト名称とポート番号を指定します。この CORBA ネーミングサー	corbaname::localhost:900

キー名称	差異	内容	デフォルト値
		ビスは、該当する Java アプリケーションから呼び出す J2EE サーバが利用しているものです。 なお、旧バージョンで使用されていたプロトコル (iioploc または iiopname) と互換性があります。 (指定形式) corbaname::[ホスト名称]:[ポート番号]	
javax.xml.datatype.DatatypeFactory	独自	javax.xml.datatype.DatatypeFactory の実装クラス名を指定します。	com.cosminexus.jaxp.impl.parsers.jaxp.datatype.DatatypeFactoryImpl
javax.xml.parsers.DocumentBuilderFactory	独自	javax.xml.parsers.DocumentBuilderFactory の実装クラス名を指定します。	com.cosminexus.jaxp.impl.parsers.jaxp.DocumentBuilderFactoryImpl
javax.xml.parsers.SAXParserFactory	独自	javax.xml.parsers.SAXParserFactory の実装クラス名を指定します。	com.cosminexus.jaxp.impl.parsers.jaxp.SAXParserFactoryImpl
javax.xml.transform.TransformerFactory	独自	javax.xml.transform.TransformerFactory の実装クラス名を指定します。	com.cosminexus.jaxp.impl.transform.processor.TransformerFactoryImpl
javax.xml.xpath.XPathFactory<http://java.sun.com/jaxp/xpath/dom>	独自	javax.xml.xpath.XPathFactory の実装クラス名を指定します。	com.cosminexus.jaxp.impl.xpath.jaxp.XPathFactoryImpl
javax.xml.validation.SchemaFactory<http://www.w3.org/2001/XMLSchema>	独自	javax.xml.validation.SchemaFactory の実装クラス名を指定します。	com.cosminexus.jaxp.impl.parsers.jaxp.validation.XMLSchemaFactory

(凡例)

あり：

Java アプリケーション用 usrconf.properties ファイルのキーと J2EE サーバ用 usrconf.properties ファイルのキーに差異があります。詳細については、「内容」の列を参照してください。

独自：

Java アプリケーション用 usrconf.properties ファイル固有のキーです。キーの詳細については、「内容」の列を参照してください。

ー：

Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「[2.2.3\(5\)\(ag\) java から始まるキー](#)」を参照してください。

(m) org から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる org から始まるキーを次に示します。

キー名称	差異	内容	デフォルト値
org.w3c.dom.DOMImplementationSourceList	独自	org.w3c.dom.DOMImplementationSourceList の実装クラス名を指定します。	com.cosminexus.jaxp.impl.parsers.dom.DOMXSIImplementationSourceImpl
org.xml.sax.driver	独自	org.xml.sax.driver の実装クラス名を指定します。	com.cosminexus.jaxp.impl.parsers.parsers.SAXParser
org.xml.sax.parser	独自	org.xml.sax.parser の実装クラス名を指定します。	org.xml.sax.helpers.XMLReaderAdapter

(凡例)

独自：

Java アプリケーション用 usrconf.properties ファイル固有のキーです。キーの詳細については、「内容」の列を参照してください。

(n) vbroker から始まるキー

Java アプリケーション用 usrconf.properties ファイルで指定できる vbroker から始まるキーを次に示します。J2EE サーバ用 usrconf.properties ファイルと共通のキーの詳細については、「[2.2.3\(5\)\(ai\) vbroker から始まるキー](#)」を参照してください。

キー名称	差異	内容	デフォルト値
vbroker.agent.enableLocator	あり	Java アプリケーションが J2EE サーバを呼び出すだけの場合は、false を指定します。J2EE サーバの通信処理を実現している TPBroker のネーミングサービスには、CORBA ネーミングサービスとスマートエージェントがあります。Java アプリケーションと J2EE サーバ間の通信は、ネーミングサービスとして CORBA ネーミングサービスだけを利用します。この場合、J2EE サーバ、Java アプリケーション間に CORBA ネーミングサービスだけを開始する必要があり、該当キーは false を指定します。	true

キー名称	差異	内容	デフォルト値
		一方、Java アプリケーションが J2EE サーバを呼び出すことに加え、スマートエージェントを利用し TPBroker のサーバアプリケーションを呼び出す場合は、CORBA ネーミングサービス、およびスマートエージェントを開始する必要があるため、該当キーに true を指定します。	
vbroker.ce.iiop.ccm.htc.readerPerConnection	なし	リモートインタフェースとして定義された EJB メソッドの呼び出しでタイムアウトが発生した場合に、コネクションのクローズを抑止するかどうかを設定します。 true を指定した場合 タイムアウト発生時のコネクションのクローズを抑止します。 false を指定した場合 タイムアウト発生時のコネクションのクローズを抑止しません。	false
vbroker.orb.htc.comt.entryCount	あり	TPBroker 通信トレースファイル 1 個当たりのエントリ数の上限値を指定します。	20000
vbroker.orb.htc.comt.fileCount	なし	TPBroker 通信トレースファイルの個数の上限値を指定します。	15
vbroker.orb.htc.mtr.trace	あり	モジュールトレースを取得するかどうかを設定します。 true を指定した場合： モジュールトレースを取得します。 false を指定した場合： モジュールトレースを取得しません。	true
vbroker.orb.htc.tracePath	あり	TPBroker のトレースファイル出力先のパスを 1~210 バイトの範囲で指定します。指定するパスは、Java アプリケーションごとにユニークにしてください。指定したパスのサブディレクトリとして comtrc と mdltrc をあらかじめ作成しておく必要があります。デフォルトの出力先の場合、サブディレクトリ comtrc と mdltrc は Java アプリケーション初回起動時に自動作成されます。 ディレクトリのパスの区切り記号には、"/"を使用してください。例えば、work ディレクトリとして/temp/work を設定する場合、次のように指定します。 vbroker.orb.htc.tracePath=/temp/work	- Windows の場合 <ejbserver.client.ejb.log.directory の値 >¥TPB¥logj - UNIX の場合 <ejbserver.client.ejb.log.directory の値 >/TPB/logj

(凡例)

あり：

Java アプリケーション用 usrconf.properties ファイルのキーと J2EE サーバ用 usrconf.properties ファイルのキーに差異があります。詳細については、「内容」の列を参照してください。

- なし：
- Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のキーです。キーの詳細については、「2.2.3(5)(ai) vbroker から始まるキー」を参照してください。
- ー：
- Java アプリケーション用 usrconf.properties ファイルと J2EE サーバ用 usrconf.properties ファイルで共通のデフォルト値です。デフォルト値の詳細については、「2.2.3(5)(ai) vbroker から始まるキー」を参照してください。

(5) 記述例

```
java.naming.provider.url=corbaname::localhost:900
ejbserver.serverName=MyServer
```

12.2.3 Java アプリケーションに指定するシステムプロパティ

(1) 形式

Win32 のバッチファイル形式（Windows の場合）、またはシェルスクリプトファイル形式（UNIX の場合）のファイルに指定します。

(2) 機能

vbj コマンドを使用する場合に、Java アプリケーションを実行する JavaVM のシステムプロパティを指定します。

また、java.lang.System.setProperty() で定義する場合は、該当機能を最初に使う前までにプロパティの定義が必要になります。

Java アプリケーションを実行中に、このファイルの内容を変更した場合、変更した内容は、次回 Java アプリケーションを起動したときに反映されます。

(3) 指定できるキー

vbj コマンドで指定できるキーを次の表に示します。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

表 12-3 Java アプリケーションに指定するシステムプロパティ

分類	キー名称	種別	指定可否	関連情報
共通のキー	ejbserver.client.ctm.RequestPriority	選択 可変	○	
	ejbserver.client.ejb.log	選択	○	

分類	キー名称	種別	指定可否	関連情報
		可変		
	ejbserver.client.log.appid	選択 可変	○	
	ejbserver.client.log.directory	選択 可変	○	
	ejbserver.client.log.lockInterval	選択 可変	○	
	ejbserver.client.log.lockRetryCount	選択 可変	○	
	ejbserver.container.rebindpolicy	選択 可変	○	「機能解説 基本・開発編 (EJB コンテナ)」の「2.13 EJB のリモートインタフェースの呼び出し」
	ejbserver.jndi.cache	選択 可変	○	
	ejbserver.jndi.cache.interval	選択 可変	○	
	ejbserver.jndi.cache.interval.clear.option	選択 可変	○	
	ejbserver.jndi.namingservice.group.list	選択 可変	○	
	ejbserver.jndi.namingservice.group.<Specify group name>.providerurls	選択 可変	○	
	ejbserver.jndi.request.timeout	選択 可変	○	
	ejbserver.logger.channels.define.<チャンネル名> ※.filenum	選択 可変	○	
	ejbserver.logger.channels.define.<チャンネル名> ※.filesize	選択 可変	○	
	ejbserver.logger.enabled.*	選択 可変	○	
	ejbserver.rmi.request.timeout	選択 可変	○	

分類	キー名称	種別	指定可否	関連情報
	ejbserver.security.service.url	選択 可変	○	
	ejbserver.serverName	可変	○	
	ejbserver.server.prf.PRfid	選択 可変	○	
	java.naming.factory.initial	可変	○	
	java.naming.provider.url	可変	○	
	vbroker.agent.enableLocator	選択 可変	○	
	vbroker.orb.htc.comt.entryCount	選択 可変	○	
	vbroker.orb.htc.comt.fileCount	選択 可変	○	
	vbroker.orb.htc.tracePath	選択 可変	○	
独自のキー	ejbserver.jndi.log.message.verboseMode	選択 可変	○	
	ejbserver.server.prf.processName	選択 可変	○	
	java.endorsed.dirs	可変	×	
	java.naming.factory.object	固定	○	
	java.naming.factory.state	固定	○	
	javax.rmi.CORBA.UtilClass	固定	○	
	javax.rmi.CORBA.StubClass	固定	×	
	javax.rmi.CORBA.PortableRemoteObjectClass	固定	○	
	org.omg.PortableInterceptor.ORBInitializerClass.com.hitachi.software.ejb.security.std.interceptor.SecurityClientInterceptorInit	固定	○	
	java.security.policy	固定	○	
	org.omg.CORBA.ORBClass	固定	×	
	org.omg.CORBA.ORBSingletonClass	固定	×	

(凡例)

○：コマンドにキーを指定する必要があります。

×：コマンドにキーを指定する必要はありません。

空欄：関連情報ははありません。

種別

固定の場合、該当するキーに対する値は固定で、必ず指定する必要があります。可変の場合、システムの実行環境に従って値を指定する必要があります。選択可変の場合は、システムの実行環境に従って値を指定するか、または指定を省略します。

注※

チャンネル名称として次に示す名称を設定できます。

ClientMessageLogFile, ClientExceptionLogFile, ClientMaintenanceLogFile

資料の取得については、マニュアル「アプリケーションサーバ 機能解説 保守／移行編」の「2.3 資料の取得」を参照してください。

(a) 共通のキー

Java アプリケーション用 usrconf.properties ファイルのキーと共通のシステムプロパティのキーです。

共通のキーの詳細については、「[12.2.2 usrconf.properties \(Java アプリケーション用ユーザプロパティファイル\)](#)」を参照してください。また、必要に応じて、「[2.2.3 usrconf.properties \(J2EE サーバ用ユーザプロパティファイル\)](#)」も参照してください。

(b) 独自のキー

システムプロパティ独自のキーです。

独自のキーの詳細を次に示します。

ejbserver.jndi.log.message.verbosemode

種別：選択可変

値：on または off

通常値：off

内容：

標準出力にメッセージログ出力を指定します。

ejbserver.server.prf.processName

種別：選択可変

値：任意のクライアント識別子

通常値：EJBClient

内容：

性能解析トレースファイルの編集結果で、プロセス名称として表示させたい名前を指定します。null を指定した場合またはこのキーを指定しない場合、通常値が使用されます。

java.endorsed.dirs

種別：可変

値：ディレクトリパス

通常値：<Application Server のインストールディレクトリ>%TPB%lib%endorsed (Windows の場合)、または/opt/Cosminexus/TPB/lib/endorsed (UNIX の場合)

内容：

Endorsed Standards Override Mechanism を使って置き換えるクラスがあるディレクトリパスを指定します。

java.naming.factory.object

種別：固定

値：com.hitachi.software.ejb.jndi.InsNamingFactory

内容：

JNDI の SPI である ObjectFactory の実装クラスを指定します。

ObjectFactory はオブジェクト検索時にオブジェクト変換をします。

なお、旧バージョンで使用されていた値

(com.hitachi.software.ejb.jndi.nameserver.spi.cos.wrappers.vbj.WrapperObjectFactory) と互換性があります。

java.naming.factory.state

種別：固定

値：com.hitachi.software.ejb.jndi.InsNamingFactory

内容：

JNDI の SPI である StateFactory の実装クラスを指定します。

StateFactory はオブジェクト登録時のオブジェクト変換をします。

なお、旧バージョンで使用されていた値

(com.hitachi.software.ejb.jndi.nameserver.spi.cos.wrappers.vbj.WrapperStateFactory) と互換性があります。

javax.rmi.CORBA.UtilClass

種別：固定

値：com.hitachi.software.ejb.rmi.Util

内容：

javax.rmi.CORBA.Util の実装クラス名を指定します。

なお、旧バージョンで使用されていた値 (com.inprise.vbroker.rmi.CORBA.UtilImpl) と互換性があります。

javax.rmi.CORBA.StubClass

種別：固定

値：com.inprise.vbroker.rmi.CORBA.StubImpl

内容：

javax.rmi.CORBA.Stub の実装クラス名を指定します。

javax.rmi.CORBA.PortableRemoteObjectClass

種別：固定

値：com.hitachi.software.ejb.rmi.PortableRemoteObject

内容：

javax.rmi.PortableRemoteObject の実装クラス名を指定します。

なお、旧バージョンで使用されていた値

(com.inprise.vbroker.rmi.CORBA.PortableRemoteObjectImpl) と互換性があります。

org.omg.PortableInterceptor.ORBInitializerClass.com.hitachi.software.ejb.security.std.interceptor.SecurityClientInterceptorInit

種別：固定

値：1

内容：

セキュリティ機能用のクライアント側インターセプタの初期化クラスを指定します。

Java アプリケーションが EJB のセキュリティインタフェースを利用し、J2EE サーバにログインをして EJB を呼び出す場合だけ指定します。

java.security.policy

種別：固定

値：<セキュリティポリシーファイルへの絶対パス>

内容：

<Application Server のインストールディレクトリ>%CC%client%cli.policy (Windows の場合), または/opt/Cosminexus/CC/client/cli.policy (UNIX の場合) をクライアント側の任意のディレクトリにコピーし、そのファイル名を絶対パスで指定します。ファイルの内容は変更しないでください。

org.omg.CORBA.ORBClass

種別：固定

値：com.inprise.vbroker.orb.ORB

内容：

ORB の実装クラス名を指定します。

org.omg.CORBA.ORBSingletonClass

種別：固定

値：com.inprise.vbroker.orb.ORBSingleton

内容：

ORB のシングルトン用の実装クラス名を指定します。

(4) 記述例

システムプロパティの記述例を OS ごとに示します。

(a) Windows の場合

Java アプリケーションを開始する bat ファイルの記述例を次に示します。これは vbj コマンドを使って Java アプリケーションを開始する場合の例です。なお、CLASS_PATH に指定する JAR ファイルのクラスパスについては、マニュアル「アプリケーションサーバ 機能解説 基本・開発編(EJB コンテナ)」の「3.7.4 EJB クライアントアプリケーションのクラスパスへの JAR ファイルの設定」を参照してください。

環境変数については、マニュアル「アプリケーションサーバ リファレンス コマンド編」の「付録 H システムの環境変数」を参照してください。

```
setlocal

rem --- set environment ---
set COSMI_HOME=C:\Program Files\Hitachi\Cosminexus
set SERVERNAME=MyServer
set PRF_HOME=%COSMI_HOME%\PRF
set TPB_HOME=%COSMI_HOME%\TPB
set JAVA_HOME=%COSMI_HOME%\jdk
set VBJ="%TPB_HOME%\bin\vbj"
set VBROKER_ADM=%TPB_HOME%\adm
set PATH=%PRF_HOME%\bin;%TPB_HOME%\bin;%JAVA_HOME%\bin;%PATH%

rem --- set class path ---
set CLASS_PATH=%COSMI_HOME%\TPB\lib\vbjorb.jar
set CLASS_PATH=%CLASS_PATH%;%COSMI_HOME%\TPB\lib\vbsec.jar
set CLASS_PATH=%CLASS_PATH%;%COSMI_HOME%\PRF\lib\cprf.jar
set CLASS_PATH=%CLASS_PATH%;%COSMI_HOME%\common\lib\hntplibMcsmj64.jar
set CLASS_PATH=%CLASS_PATH%;%COSMI_HOME%\CC\lib\hitj2ee.jar
set CLASS_PATH=%CLASS_PATH%;%COSMI_HOME%\CC\client\lib\HiEJBClientStatic.jar
set CLASS_PATH=%CLASS_PATH%;stubs.jar
set CLASS_PATH=%CLASS_PATH%;1.jar"

rem --- set properties ---
set PROPS=-Djavax.rmi.CORBA.UtilClass=com.hitachi.software.ejb.rmi.Util
set PROPS=%PROPS% -Djavax.rmi.CORBA.PortableRemoteObjectClass=com.hitachi.software.ejb.rmi.P
ortableRemoteObject
set PROPS=%PROPS% -Djava.naming.factory.initial=com.hitachi.software.ejb.jndi.InsContextFact
ory
set PROPS=%PROPS% -Djava.naming.factory.object=com.hitachi.software.ejb.jndi.InsNamingFactor
y
set PROPS=%PROPS% -Djava.naming.factory.state=com.hitachi.software.ejb.jndi.InsNamingFactory
set PROPS=%PROPS% -Djava.naming.provider.url=corbaname::localhost:900
set PROPS=%PROPS% -Dejbserver.jndi.cache=on
set PROPS=%PROPS% -Dejbserver.jndi.cache.interval=0
set PROPS=%PROPS% -Dejbserver.jndi.cache.interval.clear.option=refresh
set PROPS=%PROPS% -Dejbserver.jndi.request.timeout=0
set PROPS=%PROPS% -Dejbserver.rmi.request.timeout=0
set PROPS=%PROPS% -Dejbserver.serverName=%SERVERNAME%
set PROPS=%PROPS% -Dvbroker.agent.enableLocator=false
set PROPS=%PROPS% -Dorg.omg.PortableInterceptor.ORBInitializerClass=com.hitachi.software.ejb
.security.std.interceptor.SecurityClientInterceptorInit=1
set PROPS=%PROPS% "-Djava.security.policy=%COSMI_HOME%\CC\client\cli.policy"
set PROPS=%PROPS% -Dejbserver.server.prf.processName=EJBClient
set PROPS=%PROPS% "-Dejbserver.client.log.directory=%COSMI_HOME%\CC\client\logs"
set PROPS=%PROPS% -Dejbserver.logger.enabled.*=Error
```

```
rem --- execute client ---
%VBJ% -VBJclasspath %CLASS_PATH% %PROPS% ConverterClient

endlocal
```

(b) UNIX の場合

Java アプリケーションを開始するシェルスクリプトの記述例を次に示します。これは vbj コマンドを使って Java アプリケーションを開始する場合の例です。

なお、CLASS_PATH に指定する環境変数については、マニュアル「アプリケーションサーバリファレンス コマンド編」の「付録 H システムの環境変数」を参照してください。

```
#!/bin/csh -f

# --- environment ---
set COSMI_HOME=/opt/Cosminexus
set SERVERNAME="MyServer"
set PRF_HOME="${COSMI_HOME}/PRF"
set TPB_HOME="${COSMI_HOME}/TPB"
set JAVA_HOME="${COSMI_HOME}/jdk"
set VBJ="${TPB_HOME}/bin/vbj"
setenv PSALLOC early
setenv NODISCLAIM true
setenv AIXTHREAD SCOPE S
setenv AIXTHREAD MUTEX DEBUG OFF
setenv AIXTHREAD RWLOCK DEBUG OFF
setenv AIXTHREAD COND DEBUG OFF
setenv EXTSHM ON
setenv LDR CNTRL MAXDATA=0x40000000
setenv VBROKER ADM "${TPB_HOME}/adm"
setenv PATH "${JAVA_HOME}/bin:${PATH}"
set LPATH="${TPB_HOME}/lib:${PRF_HOME}/lib"
if ($?LIBPATH) then
    setenv LIBPATH "${LPATH}:${LIBPATH}"
else
    setenv LIBPATH "${LPATH}"
endif

# --- class path ---
set CLASS_PATH="${COSMI_HOME}/TPB/lib/vbjorb.jar"
set CLASS_PATH="${CLASS_PATH}:${COSMI_HOME}/TPB/lib/vbsec.jar"
set CLASS_PATH="${CLASS_PATH}:${COSMI_HOME}/PRF/lib/cprf.jar"
set CLASS_PATH="${CLASS_PATH}:${COSMI_HOME}/common/lib/hntrlibMcsmj64.jar"
set CLASS_PATH="${CLASS_PATH}:${COSMI_HOME}/CC/lib/hitj2ee.jar"
set CLASS_PATH="${CLASS_PATH}:${COSMI_HOME}/CC/client/lib/HiEJBClientStatic.jar"
set CLASS_PATH="${CLASS_PATH}:stubs.jar"
set CLASS_PATH="${CLASS_PATH}:1.jar"

# --- properties ---
set PROPS="-Djavax.rmi.CORBA.UtilClass=com.hitachi.software.ejb.rmi.Util"
set PROPS="${PROPS} -Djavax.rmi.CORBA.PortableRemoteObjectClass=com.hitachi.software.ejb.rmi.PortableRemoteObject"
set PROPS="${PROPS} -Djava.naming.factory.initial=com.hitachi.software.ejb.jndi.InsContextFa
```



```

ctory"
set PROPS="${PROPS} -Djava.naming.factory.object=com.hitachi.software.ejb.jndi.InsNamingFact
ory"
set PROPS="${PROPS} -Djava.naming.factory.state=com.hitachi.software.ejb.jndi.InsNamingFacto
ry"
set PROPS="${PROPS} -Djava.naming.provider.url=corbaname::localhost:900"
set PROPS="${PROPS} -Dejbserver.jndi.cache=on"
set PROPS="${PROPS} -Dejbserver.jndi.cache.interval=0"
set PROPS="${PROPS} -Dejbserver.jndi.cache.interval.clear.option=refresh"
set PROPS="${PROPS} -Dejbserver.jndi.request.timeout=0"
set PROPS="${PROPS} -Dejbserver.rmi.request.timeout=0"
set PROPS="${PROPS} -Dejbserver.serverName=${SERVERNAME}"
set PROPS="${PROPS} -Dvbroker.agent.enableLocator=false"
set PROPS="${PROPS} -Dorg.omg.PortableInterceptor.ORBInitializerClass.com.hitachi.software.e
jb.security.std.interceptor.SecurityClientInterceptorInit=1"
set PROPS="${PROPS} -Djava.security.policy==${COSMI_HOME}/CC/client/cli.policy"
set PROPS="${PROPS} -Dejbserver.server.prf.processName=EJBClient"
set PROPS="${PROPS} -Dejbserver.client.log.directory=${COSMI_HOME}/CC/client/logs"

# --- execute client ---
${VBJ} -VBJclasspath ${CLASS_PATH} ${PROPS} ConverterClient

```

- 下線が引かれている部分は、AIX の固有の記述です。
- *斜体の部分*は、OS ごとに変数名が異なります。次のとおり置き換えて使用してください。
AIX の場合：LIBPATH
Linux の場合：LD_LIBRARY_PATH

12.2.4 Java アプリケーションのユーザログ用プロパティ設定ファイル

(1) 形式

J2SE のプロパティファイル形式です。

次のようにキーを指定します。

```
<キー名称> = <値>
```

指定方法

- 改行までが値になります。
- #で始まる行はコメントとみなされます。
- 値が存在しない行を定義した場合、その行は無視されます。
- 値の後ろには、空白やコメントなどの文字列は追加できません。追加した場合、不正な値と解釈されます。

(例) <キー名称> = <値> #<コメント>

(2) 機能

vbj コマンドを使用する場合に、Java アプリケーションのユーザログ用のシステムプロパティを設定します。

ここで設定したプロパティは、Java アプリケーションで読み込み、システムプロパティに設定する必要があります。Java アプリケーションでのユーザログ出力の実装については、マニュアル「アプリケーションサーバ 機能解説 拡張編」の「8.11 EJB クライアントアプリケーションのユーザログ出力の実装と設定 (vbj コマンドを使用する場合)」を参照してください。

(3) ファイル名

格納先およびファイル名称は任意です。

(4) 指定できるキー

指定できるキーを次の表に示します。なお、キーの詳細については、「12.2.2(4)(a) ejbserver.application から始まるキー」を参照してください。また、必要に応じて、「2.2.3(5)(a) ejbserver.application から始まるキー」も参照してください。

表 12-4 Java アプリケーションのユーザログ用プロパティ設定ファイルで指定できるキー

キー名称	内容
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.appname	AppName のフィールドに出力するデフォルトのアプリケーション名を指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.autoFlush.enabled	メッセージ出力後に flush をするかどうかを指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.count	ログファイルの面数を指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.encoding	出力する文字列のエンコーディングを指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.filter	使用するフィルタを、パッケージ名を含むフルネームで指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.formatter	使用するフォーマットを、パッケージ名を含むフルネームで指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.level	ログの取得レベルの上限を指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.limit	ログファイルの容量を指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.msgid	MsgID のフィールドに出力するデフォルトのメッセージ ID を指定します。
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.path	ログファイル名のプリフィックスを指定します。

キー名称	内容
ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.separator	CJSimpleFormatter を利用して、1 文でメッセージを出力するためのセパレータを指定します。
ejbserver.application.userlog.loggers	使用するロガーの名称を宣言します。
ejbserver.application.userlog.Logger.<ロガー名称>.filter	ロガーで使用するフィルタを、パッケージ名を含むフルネームで指定します。
ejbserver.application.userlog.Logger.<ロガー名称>.handlers	指定のロガー名称に対して、ハンドラクラスを指定します。
ejbserver.application.userlog.Logger.<ロガー名称>.level	ロガーの出力レベルを指定します。
ejbserver.application.userlog.Logger.<ロガー名称>.useParentHandlers	ログレコードを受け取ったロガーから、親ロガーの接続しているハンドラへログレコードを伝播させるかどうかを指定します。

(5) サンプル

次の場所に格納されています。

- Windows の場合
 <Application Server のインストールディレクトリ>%CC%examples%userlog%sample.properties
- UNIX の場合
 /opt/Cosminexus/CC/examples/userlog/sample.properties

サンプルを次に示します。

```
# user-log handler function
ejbserver.application.userlog.CJLogHandler.conf1.appname=my_app1
ejbserver.application.userlog.CJLogHandler.conf1.path=application1/logfileA
ejbserver.application.userlog.CJLogHandler.conf1.limit=10485760
ejbserver.application.userlog.CJLogHandler.conf1.count=2
ejbserver.application.userlog.CJLogHandler.conf1.level=SEVERE

# user-log logger function
ejbserver.application.userlog.Logger.com.example.userlogger1.handlers=com.hitachi.software.ejb.application.userlog.CJMPMessageFileHandler;conf1
ejbserver.application.userlog.Logger.com.example.userlogger1.useParentHandlers=true
ejbserver.application.userlog.Logger.com.example.userlogger1.level=INFO
ejbserver.application.userlog.loggers=com.example.userlogger1
```

13

Web アプリケーションで使用するファイル

この章では、Web アプリケーションで使用するコンパイル対象外リストファイル、実行結果リストファイルについて説明します。

13.1 Web アプリケーションで使用するファイルの一覧

Web アプリケーションで使用するファイルの一覧使用するファイルの一覧を，次の表に示します。

表 13-1 Web アプリケーションで使用するファイルの一覧使用するファイルの一覧

ファイル名	分類	概要	参照先
(任意) ※	コンパイル対象外リストファイル	コマンドによる JSP 事前コンパイルの対象外とする JSP ファイルを定義します。	13.2.1
(任意) ※	実行結果リストファイル	コンパイル結果を収集するためのファイルです。	13.2.2

注※
使用できる文字は，半角の英数字，ピリオド (.)，アンダースコア (_) です。

13.2 Web アプリケーションで使用するファイルの詳細

13.2.1 コンパイル対象外リストファイル

(1) 形式

次のように対象外リストファイルを指定します。

```
# コメント
<コンパイルの対象外とするJSPファイルパス 1>
<コンパイルの対象外とするJSPファイルパス 2>
...
```

指定方法

- #で始まる行はコメントとみなされます。
- コンパイルの対象外とする JSP ファイルのパスを 1 行に一つ、記述します。

(2) 格納先およびファイル名

格納先およびファイル名は任意です。ファイル名には次の文字が使用できます。

英数字, アンダースコア (_), またはピリオド (.)

(3) 機能

コマンドによる JSP 事前コンパイル実行時, コンパイルの対象外とする JSP ファイルパスを設定します。

コンパイル対象外リストファイルを `cjjspc` コマンドの `-excludelist` オプションに指定することで, コンパイル対象外リストファイルに記述された JSP ファイルをコンパイルの対象外にできます。 `cjjspc` コマンドについては, マニュアル「アプリケーションサーバ リファレンス コマンド編」の「`cjjspc` (JSP の事前コンパイル)」を参照してください。

(4) 記述例

- JSP ファイルを単体で指定する場合

```
/jsp/title.jsp
/jsp/pieces/pageA.jsp
```

- ディレクトリに含まれるすべてのファイルを指定する場合

```
/jsp/pieces/*
```

- コンパイル対象外とするファイルの拡張子を指定する場合

```
/jsp/pieces/*.jsp
```

- コメントを書く場合

```
# JSP Fragment  
/jsp/pieces/pageA.jsp
```

13.2.2 実行結果リストファイル

(1) 形式

実行結果リストファイルの出力形式を次に示します。

```
# Excluded JSP files.  
<コンパイル対象外JSPファイルの一覧>  
# Compiled JSP files.  
<コンパイルに成功したJSPファイル>  
# <コンパイルに失敗したJSPファイル>  
...
```

(2) 格納先およびファイル名

格納先およびファイル名は任意です。ファイル名には次の文字が使用できます。

英数字, アンダースコア (_), またはピリオド (.)

(3) 機能

コマンドによる JSP 事前コンパイル実行時, cjjspc コマンドに -resultlist オプションを指定することで, コンパイル対象外 JSP ファイルのファイルパスの一覧, コンパイルに成功した JSP ファイル, およびコンパイルに失敗した JSP ファイルのファイルパスの一覧を実行結果リストファイルに出力できます。

実行結果リストファイルには, 次の順番でコンパイル結果が出力されます。

1. コンパイル対象外の JSP ファイルパスの一覧
2. コンパイルに成功, または失敗した JSP ファイルパス

コンパイルに失敗した JSP ファイルのファイルパスは, コメントとして出力されます。

(4) 出力例

```
# Excluded JSP files.  
/jsp/title.jsp  
/jsp/pieces/*.jsp  
# Compiled JSP files.  
/jsp/confirm.jsp
```

```
#/jsp/discorrect.jsp  
#/jsp/fail.jsp  
/jsp/index.jsp  
/jsp/order.jsp  
#/jsp/uncomfortable.jsp
```


14

JavaVM 起動オプション

この章では、JavaVM 起動オプションについて説明します。

Application Server では JavaVM 起動オプションとして、Java HotSpot VM のオプションのほか、JavaVM 拡張オプションを指定できます。

ここでは、JavaVM 拡張オプションの詳細について説明します。また、Application Server で指定できる Java HotSpot VM のオプション、およびデフォルト値についても説明します。

14.1 JavaVM 拡張オプションの一覧

JavaVM 拡張オプションの一覧を、次の表に示します。

注意事項

指定できるオプションについて

アプリケーションサーバに指定できるオプションは、java コマンドに対してオプション指定なしで起動した場合に表示されるオプションと、java コマンドに対して-X か-XX か-XX:+Hitachi オプションを指定したときに表示されるオプションだけです。それ以外のオプションを指定した場合、動作は保証しません。

これらのオプションは、オプション定義ファイルで指定できます。「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

「関連情報」とは、指定したキーに関する情報の参照先です。マニュアル名称の「アプリケーションサーバ」を省略しています。

表 14-1 JavaVM 拡張オプションの一覧

分類	オプション名称	概要	VR	関連情報
一覧表示オプション	-XX:+Hitachi	JavaVM 拡張オプションの一覧を表示します。	05-02	
拡張スレッドダンプ機能オプション	-XX:[+ -]HitachiThreadDump	拡張スレッドダンプ情報を出力するかどうかを指定します。	11-40	
	-XX:[+ -]HitachiThreadDumpToStdout	標準出力にスレッドダンプを出力するかどうかを指定します。	06-50	
	-XX:[+ -]HitachiThreadDumpWithHashCode	スレッド情報にハッシュコードを出力するかどうかを指定します。	07-00	
	-XX:[+ -]HitachiThreadDumpWithCpuTime	スレッド情報にユーザ CPU 時間とカーネル CPU 時間を出力するかどうかを指定します。	07-00	
	-XX:[+ -]HitachiThreadDumpWithBlockCount	スレッド情報に処理をブロックした回数と待ち状態になった回数を出力するかどうかを指定します。	07-00	
JavaVM ログファイルオプション	-XX:HitachiJavaLog ^{※1}	ログファイル名のプリフィックスを指定します。	05-02	

分類	オプション名称	概要	VR	関連情報
	-XX:HitachiJavaLogFileSize ^{※1}	1 ファイルの最大ファイルサイズを指定します。	05-02	
	-XX: [+ -]HitachiJavaLogNoMoreOutput ^{※1}	ログファイル作成時に、入出力エラーが発生した場合の動作について指定します。	05-02	
	-XX:HitachiJavaLogNumberOfFile ^{※1}	作成するログファイルの最大ファイル数を指定します。	05-02	
	-XX:[+ -]JavaLogAsynchronous	ログファイルの非同期出力機能を有効にします。	09-60	
詳細時間出力オプション	-XX:[+ -]HitachiOutputMilliTime	ミリ秒までの時間を出力するかどうかを指定します。	06-00	
拡張 verbosegc 機能オプション	-XX:[+ -]HitachiVerboseGC ^{※2}	GC が発生した時の拡張 verbosegc 情報を出力するかどうかを指定します。	11-40	
	-XX:[+ -]HitachiCommaVerboseGC	拡張 verbosegc 情報を CSV 形式で出力するかどうかを指定します。	11-40	
	-XX:HitachiVerboseGCIntervalTime	拡張 verbosegc 情報を出力する時間の間隔を指定します。	05-02	
	-XX: [+ -]HitachiVerboseGCPrintCause	GC の要因内容を出力するかどうかを指定します。	11-40	
	-XX:[+ -]HitachiVerboseGCPrintDate	拡張 verbosegc 情報に日付を出力するかどうかを指定します。	05-02	
	-XX:[+ -]HitachiVerboseGCCpuTime	GC のプロセッサ時間を出力するかどうかを指定します。	07-00	
	-XX: [+ -]HitachiVerboseGCPrintTenuring Distribution	GC 発生時に JavaVM ログファイルへ Survivor 領域のオブジェクトの年齢分布を出力します。	08-00	
	-XX: [+ -]HitachiVerboseGCPrintJVMInternalMemory	JavaVM 内部で管理しているヒープ情報を JavaVM ログファイルへ出力するかどうかを指定します。	08-53	

分類	オプション名称	概要	VR	関連情報
	-XX: [+ -]HitachiVerboseGCPrintThreadCount	Java スレッドの数を監視するために、Java スレッドの数を JavaVM ログファイルに出力するかどうかを指定します。	Linux (x86/ AMD64 & Intel EM64T) 版： 08-53 Windows x86： 08-53 Windows x64： 08-53 Windows x86 (JDK 6 ベースの 場合)： 08-70 AIX： 09-00	
	-XX: [+ -]HitachiVerboseGCPrintDeleteOnExit	java.io.File.deleteOnExit()を呼び出したことによって JavaVM が確保した累積のヒープサイズとメソッドの呼び出し回数を JavaVM ログファイルに出力するかどうかを指定します。	Linux (x86/ AMD64 & Intel EM64T) 版： 08-53 Windows x86： 08-53 Windows x64： 08-53 Windows x86 (JDK 6 ベースの 場合)： 08-70 AIX： 09-00	
コードキャッシュ領域 情報出力機能オプション	-XX:[+ -]PrintCodeCacheInfo	コードキャッシュ領域の使用量を出力するかどうか、また、使用量がしきい値に達したことを知ら	09-50	「機能解説 保守／移行編」 の「5.7.3 コードキャッ シュ領域に関

分類	オプション名称	概要	VR	関連情報
		せるメッセージを出力するかどうかを指定します。		するログの内容
	-XX:CodeCacheInfoPrintRatio	コードキャッシュ領域の使用量がしきい値に達したことを知らせるメッセージを出力する契機となる、コードキャッシュ領域の使用率を指定します。	09-50	
	-XX:[+ -]PrintCodeCacheFullMessage	Java メソッドが JIT コンパイルの対象になった場合、コードキャッシュ領域が枯渇していたときにメッセージを出力するかどうかを指定します。	09-50	「機能解説 保守／移行編」の「5.7.3 コードキャッシュ領域に関するログの内容」
OutOfMemoryError 発生時の拡張機能オプション	-XX:[+ -]HitachiOutOfMemoryCause ※2	OutOfMemoryError 発生時の発生要因種別を出力するかどうかを指定します。	05-02	
	-XX: [+ -]HitachiOutOfMemoryStackTrace ※2	OutOfMemoryError 発生時のスタックトレースを出力するかどうかを指定します。	05-02	
	-XX:HitachiOutOfMemoryStackTraceLineSize	OutOfMemoryError 発生時に出力するスタックトレースの 1 行の文字数を指定します。	05-02	
	-XX:[+ -]HitachiOutOfMemorySize※2	OutOfMemoryError 発生時に要求したメモリのサイズを出力します。	06-50	
	-XX:[+ -]HitachiOutOfMemoryAbort	OutOfMemoryError 発生時、メッセージとメモリダンプを出力して強制終了するかどうかを指定します。	06-50	
	-XX: [+ -]HitachiOutOfMemoryAbortThreadDump	OutOfMemoryError 発生時にスレッドダンプを出力かどうかを指定します。	06-50	
	-XX: [+ -]HitachiOutOfMemoryAbortThreadDumpWithJHeapProf	-XX:+HitachiOutOfMemoryAbortThreadDump で出力するスレッドダンプ	08-00	

分類	オプション名称	概要	VR	関連情報
		プログファイルにクラス別統計情報を出力します。		
	-XX: [+ -]HitachiOutOfMemoryHandling	OutOfMemory ハンドリング機能を有効にするかどうかを指定します。	08-53	
	-XX:HitachiOutOfMemoryHandlingMaxThrowCount	OutOfMemory ハンドリング機能を有効にした場合の、Java ヒープ不足または Metaspace, Compressed Class Space 不足が原因の OutOfMemory 発生回数合計値の 1 時間当たりの上限値を指定します。	08-53	
クラスライブラリトレース機能オプション	-XX:[+ -]HitachiJavaClassLibTrace※2	クラスライブラリのスタックトレースを出力するかどうかを指定します。	06-50	
	-XX:HitachiJavaClassLibTraceLineSize	クラスライブラリのスタックトレースの 1 行の文字数を指定します。	06-50	
ローカル変数情報出力機能オプション	-XX:[+ -]HitachiLocalsInThrowable	例外発生時のスタックトレースに、ローカル変数情報を出力するかどうかを指定します。	06-50	
	-XX:[+ -]HitachiLocalsInStackTrace	スレッドダンプ出力時のスタックトレースに、ローカル変数情報を出力するかどうかを指定します。	06-50	「機能解説 保守／移行編」の「5.10 JavaVM スタックトレース情報」
	-XX:[+ -]HitachiLocalsSimpleFormat	ローカル変数情報出力を、簡易フォーマットにするかどうかを指定します。	06-50	「機能解説 保守／移行編」の「5.10 JavaVM スタックトレース情報」
	-XX:[+ -]HitachiTrueTypeInLocals	ローカル変数情報出力時に、ローカル変数オブジェクトの実際の型名を文字列として出力するかどうかを指定します。	06-50	「機能解説 保守／移行編」の「5.10 JavaVM スタックトレース情報」
	-XX:HitachiCallToString	ローカル変数情報出力時に、ローカル変数オブ	06-50	「機能解説 保守／移行編」

分類	オプション名称	概要	VR	関連情報
		ジェクトの変数値を文字列として出力するかどうかを指定します。		の「5.10 JavaVM スタックトレース情報」
システムリソース解除オプション	-XX:[+ -]HitachiFullCore	システムリソース RLIMIT_CORE の設定を変更するかどうかを指定します。 このオプションは UNIX 用です。	05-02	
明示管理ヒープで指定するオプション	-XX:[+ -]HitachiUseExplicitMemory	明示管理ヒープ機能の有効、無効を指定します。	11-40	「機能解説 拡張編」の「7. 明示管理ヒープ機能を使用した FullGC の抑止」
	-XX:HitachiExplicitHeapMaxSize	Explicit ヒープ全体の最大サイズを指定します。	11-40	
	-XX:HitachiExplicitMemoryLogLevel	明示管理ヒープ機能のイベントログのレベルを指定します。	11-40	
	-XX:HitachiExplicitMemoryJavaLog	本機能によるログを出力するファイル名（パス名を含むことが可能）またはディレクトリ名を指定します。	11-40	
	-XX:HitachiExplicitMemoryJavaLogFileSize	1 ファイルの最大ファイルサイズをバイト単位で設定します。	11-40	
	-XX:HitachiExplicitMemoryJavaLogNumberOfFile	作成する最大ファイル数を指定します。最大ファイル数を超えた場合は、再度最初に作成したファイルへ出力（ラップアラウンド）します。	11-40	
	-XX:[+ -]HitachiExplicitMemoryMoveToTenuredFirst	Explicit メモリブロックの解放処理でのオブジェクトの移動先を指定します。	11-40	「機能解説 拡張編」の「7. 明示管理ヒープ機能を使用した FullGC の抑止」

分類	オプション名称	概要	VR	関連情報
	-XX: [+ -]HitachiExplicitMemoryAutoReclaim	明示管理ヒープ機能の自動解放機能を有効にするかどうか指定します。	11-40	「機能解説 拡張編」の「7.7 自動解放機能が有効な場合の Explicit メモリブロックの解放」
明示管理ヒープ機能バージョン互換オプション	-XX: [+ -]HitachiExplicitMemoryCompatibleToV8	Explicit メモリブロックを確保する方法を、08-00 と同様にするかどうか指定します。	11-40	
明示管理ヒープ自動配置設定ファイルオプション	-XX:[+ -]HitachiAutoExplicitMemory	Explicit メモリブロックの自動配置機能を有効にするかどうか指定します。	11-40	「機能解説 拡張編」の「7.13.2 自動配置設定ファイルを使った明示管理ヒープ機能の使用」
	-XX:HitachiAutoExplicitMemoryFile	Explicit メモリブロックの自動配置機能で利用する自動配置設定ファイルのパスを指定します。	11-40	「機能解説 拡張編」の「7.13.2 自動配置設定ファイルを使った明示管理ヒープ機能の使用」
明示管理ヒープ機能の Explicit メモリブロックへのオブジェクト移動制御オプション	-XX:ExplicitMemoryFullGCPolicy	FullGC 発生時に、参照関係に基づくオブジェクトの Java ヒープから Explicit メモリブロックへの移動を制御するかどうかを指定します。	11-40	「機能解説 拡張編」の「7. 明示管理ヒープ機能を使用した FullGC の抑止」
明示管理ヒープ機能適用除外クラス指定機能で指定するオプション	-XX: [+ -]ExplicitMemoryUseExcludeClass	明示管理ヒープ機能適用除外クラス指定機能を有効にするかどうかを指定します。	11-40	
	-XX:ExplicitMemoryExcludeClassListFile	明示管理ヒープ機能適用除外クラス指定機能で利用する、明示管理ヒープ機能適用除外設定ファイルのパスを指定します。	11-40	

分類	オプション名称	概要	VR	関連情報
	-XX:ExplicitMemoryNotExcludeClassListFile	明示管理ヒープ機能適用除外クラス指定機能で利用する、明示管理ヒープ機能適用除外無効設定ファイルのパスを指定します。	11-40	
HTTP セッションで利用する Explicit ヒープの省メモリ化オプション	-XX:[+ -]HitachiExplicitMemoryAutoRefReclaim	HTTP セッションで利用する Explicit ヒープの省メモリ化機能を有効にするかどうかを指定します。	11-40	「機能解説 拡張編」の「7. 明示管理ヒープ機能を使用した FullGC の抑止」
リソースの上限値指定オプション	-XX:HitachiJITCompileMaxMemorySize	JIT コンパイル時に確保するメモリの上限値を指定します。	08-53	
	-XX:HitachiThreadLimit	スレッド数の上限値を指定します。	08-53	
JIT コンパイラ稼働継続機能オプション	-XX:[+ -]JITCompilerContinuation	JIT コンパイラ稼働継続機能を有効にするかどうかを指定します。	09-50	
圧縮オブジェクトポインター機能で使用する Java オプション	-XX:[+ -]UseCompressedOops	圧縮オブジェクトポインター機能の有効または無効の設定	09-60	「機能解説 保守／移行編」の「9.18 圧縮オブジェクトポインタ機能」
ダイレクトバッファ機能オプション	-XX:[+ -]HitachiVerboseGCPrintDirectBuffer ^{※3}	ダイレクトバッファの使用状況を日立 JavaVM ログファイルに出力するかどうかを指定します。	11-40	

(凡例)

空欄：関連情報はありません。

注※1

JavaVM ログファイルについての設定です。

注※2

次のオプションを指定した場合、JavaVM ログファイルが出力されます。

-XX:+HitachiOutOfMemoryStackTrace

-XX:+HitachiOutOfMemoryCause

-XX:+HitachiOutOfMemorySize

-XX:+HitachiVerboseGC

-XX:+HitachiJavaClassLibTrace

JDK17 以降の場合にだけ指定できます。

ポイント

JavaVM 拡張オプションのデフォルト値は、サーバごとに参照先が異なります。サーバ、およびアプリケーションごとの JavaVM 拡張オプションのデフォルト値の参照先を次の表に示します。

表 14-2 サーバ、およびアプリケーションごとの JavaVM 拡張オプションのデフォルト値の参照先

使用するサーバおよびアプリケーション	参照先
J2EE サーバ	2.2.2 usrconf.cfg (J2EE サーバ用オプション定義ファイル)
バッチサーバ	3.2.1 usrconf.cfg (バッチサーバ用オプション定義ファイル)
cjclstartap コマンドで起動する Java アプリケーション	12.2.1 usrconf.cfg (Java アプリケーション用オプション定義ファイル)
そのほかの Java アプリケーション	14.4 Application Server で指定できる Java HotSpot VM のオプションのデフォルト値

なお、JavaVM 拡張オプションを簡易構築定義ファイルに指定する場合は、「param-name 指定値」に add.jvm.arg を指定した場合に指定できる「param-value の指定内容」に JavaVM 拡張オプションを指定してください。

14.2 JavaVM 拡張オプションの詳細

JavaVM 拡張オプションの詳細について説明します。

なお、本文中では、次の Java プログラムを例として使用しています。

Java プログラム例 1

```
class Example1 {
    public static void main(String[] args) {
        Example1 e1 = new Example1();
        Object obj = new Object();
        e1.method(1, 'Q', obj); // 5行目
    }

    void method(int l1, char l2, Object l3) {
        float l4 = 4.0f;
        boolean l5 = true;
        double l6 = Double.MAX_VALUE; // double型の最大値
        Object[] l7 = new Object[10];

        try {
            <例外発生!> // 15行目
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Java プログラム例 2

```
class Example2 {
    public static void main(String[] args) {
        Example2 e2 = new Example2();
        e2.method(); // 4行目
    }

    synchronized void method() {
        int l1 = 1;
        float l2 = 2.0f;
        String l3 = "local 3";
        Character l4 = new Character('X');
        Object l5 = new Thread();
        Object[] l6 = new Thread[10];

        <ここでスレッドダンプ出力!> // 15行目
    }
}
```

Java プログラム例 3

```
class Example3 {
    public static void main(String[] args) {
        Example3 e3 = new Example3();
        e3.method(); // 4行目
    }
}
```

```

void method() {
    String l1 = "local 1";
    StringBuffer l2 = new StringBuffer(l1);
    l2.append(" + local 2");
    Boolean l3 = new Boolean(false);
    Character l4 = new Character('X');
    Long l5 = new Long(Long.MIN_VALUE); // long型の最小値
    Object l6 = new Thread();
    Object[] l7 = new Thread[10];

    try {
        <例外発生!> // 18行目
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public String toString() {
    return "I am an Example3 instance.";
}
}

```

-XX:+Hitachi (一覧表示オプション)

形式

-XX:+Hitachi

説明

JavaVM 拡張オプションを一覧表示します。

このオプションは、Java プログラムを実行しません。また、J2EE サーバの起動オプションに指定した場合、J2EE サーバは起動されません。

-XX:[+|-]HitachiThreadDump (拡張スレッドダンプ情報出力オプション)

形式

-XX:+HitachiThreadDump

-Xrs オプションが指定されていない場合、スレッドダンプ出力時に拡張スレッドダンプ情報を出力します。

-XX:-HitachiThreadDump

スレッドダンプ出力時に標準のスレッドダンプ情報を出力します。

説明

拡張スレッドダンプ情報を出力するかどうかを指定します。

スレッドダンプは、標準出力、および次に示すファイルに出力されます。

デフォルト値

- -XX:+HitachiThreadDump

スレッドダンプ出力先

デフォルトでは、JavaVM 実行時のカレントディレクトリです。環境変数 JAVACOREDIR を指定することで、出力先を変更することができます。

スレッドダンプ出力ファイル名

```
javacore<プロセス番号>.<YYMMDDhhmmss>※.txt
```

注※

- YY：年（西暦で下2けた），MM：月（2けた），DD：日（2けた）
- hh：時間（24時間表記），mm：分（2けた），ss：秒（2けた）

スレッドダンプ情報の構成を、次に示します。

表 14-3 スレッドダンプ情報の構成

出力情報	内容
ヘッダ	スレッドダンプを開始した日付および時刻，JavaVM バージョン情報，起動コマンドラインを出力します。
システム設定	次の情報を出力します。 • JDK の実行環境のインストールディレクトリ • JDK を構成するライブラリのインストールディレクトリ • システムクラスパス • Java コマンドオプション
動作環境	次の情報を出力します。 • ホスト名 • OS バージョン • CPU 情報 • リソース情報（UNIX の場合だけ）
メモリ情報	Windows の場合、次の情報を出力します。 • メモリ使用状況 • 物理メモリ使用状況 • 仮想メモリ使用状況 • ページング使用量
Java ヒープ情報	Java ヒープの各世代のメモリ使用状況を出力します。
JavaVM 内部メモリマップ情報	JavaVM 自身の確保しているメモリの領域情報を出力します。

出力情報	内容
JavaVM 内部メモリサイズ情報	JavaVM 自身の確保しているメモリのサイズ情報を出力します。
アプリケーション情報	次の情報を出力します。 • シグナルハンドラ • 環境変数 • カレントディレクトリ情報
ライブラリ情報	ローディングされているライブラリの情報を出力します。
スレッド情報 <スレッド 1> : <スレッド n>	スレッドごとにスレッド情報を出力します。現存する全スレッドのスタックトレースを出力します。
Java モニタダンプ※	Java モニタオブジェクトの一覧を表示します。スレッド間の排他待ちの状況を確認できます。
JNI グローバル参照情報	JavaVM が保持している JNI のグローバル参照の数を出力します。 JNI グローバル参照は、次の場合に作成されます。 • JavaVM が自身の起動や実行のために必要な場合。 • JNI がサポートする NewGlobalRef 関数を発行した場合。
Explicit ヒープ情報	Explicit ヒープ情報では、Explicit ヒープ全体での利用状況を出力します。Explicit ヒープ利用の有無にかかわらず、-XX:+HitachiUseExplicitMemory が有効の場合は必ず出力されます。
Explicit メモリブロック情報	Explicit メモリブロック情報では、次の情報を出力します。ただし、Explicit メモリブロックが 1 つも存在しない場合は何も出力されません。 • Explicit メモリブロックごとの利用状況 • オブジェクト統計情報 (eheapprof コマンド実行時に出力) • オブジェクト解放率情報 (-freeratio オプションを指定した eheapprof コマンド実行時に出力)
フッタ	スレッドダンプを終了した日付および時刻を出力します。

注※ UNIX の場合、notify 待ちの一覧が表示されない場合があります。

出力形式

ヘッダ

EEE MMM dd hh:mm:ss yyyy※

Full thread dump Java HotSpot(TM) <VM type> (25.20-b23-CDK0970-<ビルド日> mixed mode)
<起動コマンドライン>
...

注※

EEE は曜日、MMM は月、dd は日を表します。また、hh は時間、mm は分、ss は秒、yyyy は年（西暦）を表します。

<VM type>は Client VM, Server VM または 64-Bit Server VM のどれかを表します。

システム設定

System Properties

Java Home Dir : <JDK実行環境インストールディレクトリ>
Java DLL Dir : <JDKを構成するライブラリのインストールディレクトリ>
Sys Classpath : <システムクラスパス>
User Args :
<Javaコマンドオプション1>
<Javaコマンドオプション2>
...

動作環境 (Windows の場合)

Operating Environment

Host : <ホスト名>:<IPアドレス>
OS : <OSバージョン>
CPU : <CPU種別>, <利用可能CPU数/システム全体のCPU数>

動作環境 (UNIX の場合)

Operating Environment

Host : <ホスト名>:<IPアドレス>
OS : <OSバージョン>
CPU : <CPU種別>, <利用可能CPU数/システム全体のCPU数>

Resource Limits -

RLIMIT_CPU : <プロセスで使用可能な秒数>
RLIMIT_FSIZE : <最大ファイルサイズ(単位: バイト)>
RLIMIT_DATA : <malloc可能なサイズ(単位: バイト)>
RLIMIT_STACK : <スタックの最大サイズ(単位: バイト)>
RLIMIT_CORE : <coreの最大サイズ(単位: バイト)>
RLIMIT_RSS : <プロセスの常駐サイズ(単位: バイト)>
RLIMIT_AS : <プロセストータルの利用可能メモリ(単位: バイト)>
RLIMIT_NOFILE : <最大のファイルディスクリプタ値>

メモリ情報 (Windows の場合)

Memory Status

Memory in use : <メモリ使用状況(単位: パーセント)>
Physical memory : <物理メモリ使用量>※ free
Virtual memory : <仮想メモリ使用量>※ free
Paging file : <ページング使用量>※ free

注※

<利用可能サイズ>/<総サイズ> の形式で表示されます (単位: バイト)。

Java ヒープ情報 (SerialGC を使用していて, かつ JDK11 以前の場合)

Heap Status

```

def new generation    max <最大の容量>, total <現在の容量>, used <使用中メモリ> (<最大の容量
に対する使用率>% used/max, <現在の容量に対する使用率>% used/total)
    [<領域先頭アドレス>, <コミット済み領域の末尾アドレス>, <予約済み領域の
末尾アドレス>)
    eden space <現在の容量>, <使用率>% used [<領域先頭アドレス>, <使用中領域の先頭アドレス>,
<予約済み領域の末尾アドレス>)
    from space <現在の容量>, <使用率>% used [<領域先頭アドレス>, <使用中領域の先頭アドレス>,
<予約済み領域の末尾アドレス>)
    to space <現在の容量>, <使用率>% used [<領域先頭アドレス>, <使用中領域の先頭アドレス>,
<予約済み領域の末尾アドレス>)
tenured generation    max <最大の容量>, total <現在の容量>, used <使用中メモリ> (<最大の容量
に対する使用率>% used/max, <現在の容量に対する使用率>% used/total)
    [<領域先頭アドレス>, <コミット済み領域の末尾アドレス>, <予約済み領域の
末尾アドレス>)
    the space <現在の容量>, <使用率>% used [<領域先頭アドレス>, <使用中領域の先頭アドレス>,
<次の空きブロックの先頭アドレス>, <予約済み領域の末尾アドレス>)
Metaspace              max<最大の容量>, capacity <コミット済みの領域からフリー領域を除いたメモリサ
イズ>, committed <コミット済みのメモリサイズ>, reserved <予約済みのメモリサイズ>, used <使用
中メモリ> (<最大の容量に対する使用率>% used/max, <現在の容量に対する使用率>% used/total)
    class space        max <最大の容量>, capacity <コミット済みの領域からフリー領域を除いたメモリ
サイズ>, committed <コミット済みのメモリサイズ>, reserved <予約済みのメモリサイズ>, used <使用
中メモリ> (<最大の容量に対する使用率>% used/max, <現在の容量に対する使用率>% used/committe
d)
    [<領域先頭アドレス>, <使用中領域の先頭アドレス>, <コミット済み領域の末
尾アドレス>, <予約済み領域の末尾アドレス>)

```

注

容量およびメモリサイズの単位はキロバイトです。

Java ヒープ情報 (SerialGC を使用していて、かつ JDK17 以降の場合)

Heap Status

```

def new generation    max <最大の容量>, total <現在の容量>, used <使用中メモリ> (<最大の容量
に対する使用率>% used/max, <現在の容量に対する使用率>% used/total)
    [<領域先頭アドレス>, <コミット済み領域の末尾アドレス>, <予約済み領域の
末尾アドレス>)
    eden space <現在の容量>, <使用率>% used [<領域先頭アドレス>, <使用中領域の先頭アドレス>,
<予約済み領域の末尾アドレス>)
    from space <現在の容量>, <使用率>% used [<領域先頭アドレス>, <使用中領域の先頭アドレス>,
<予約済み領域の末尾アドレス>)
    to space <現在の容量>, <使用率>% used [<領域先頭アドレス>, <使用中領域の先頭アドレス>,
<予約済み領域の末尾アドレス>)
tenured generation    max <最大の容量>, total <現在の容量>, used <使用中メモリ> (<最大の容量
に対する使用率>% used/max, <現在の容量に対する使用率>% used/total)
    [<領域先頭アドレス>, <コミット済み領域の末尾アドレス>, <予約済み領域の
末尾アドレス>)
    the space <現在の容量>, <使用率>% used [<領域先頭アドレス>, <使用中領域の先頭アドレス>,
<次の空きブロックの先頭アドレス>, <予約済み領域の末尾アドレス>)
Metaspace              max<最大の容量>, committed <コミット済みのメモリサイズ>, reserved <予約済
みのメモリサイズ>, used <使用中メモリ> (<最大の容量に対する使用率>% used/max, <現在の容量に
対する使用率>% used/total)
    class space        max <最大の容量>, committed <コミット済みのメモリサイズ>, reserved <予約
済みのメモリサイズ>, used <使用中メモリ> (<最大の容量に対する使用率>% used/max, <現在の容量に

```


対する使用率)% used/committed)
[<領域先頭アドレス>, <予約済み領域の末尾アドレス>]

注

容量およびメモリサイズの単位はキロバイトです。

Java ヒープ情報 (G1GC を使用していて、かつ JDK11 以前の場合)

Heap Status

garbage-first heap total <現在の容量>, used <使用中のメモリ> [<領域先頭アドレス>, <予約済み領域のアドレス>]
region size <1リージョンのサイズ>, <New領域の使用リージョン数> young (<New領域の使用メモリ>), <Survivor領域の使用リージョン数> survivors (<Survivor領域の使用メモリ>)
Metaspace max <最大の容量>, capacity <コミット済みの領域からフリー領域を除いたメモリサイズ>, committed <コミット済みのメモリサイズ>, reserved <予約済みのメモリサイズ>, used <使用中メモリ> (<最大の容量に対する使用率)% used/max, <現在の容量に対する使用率)% used/committed)
class space max <最大の容量>, capacity <コミット済みの領域からフリー領域を除いたメモリサイズ>, committed <コミット済みのメモリサイズ>, reserved <予約済みのメモリサイズ>, used <使用中メモリ> (<最大の容量に対する使用率)% used/max, <現在の容量に対する使用率)% used/committed)
[<領域先頭アドレス>, <使用中領域の先頭アドレス>, <コミット済み領域の末尾アドレス>, <予約済み領域の末尾アドレス>]

Java ヒープ情報 (G1GC を使用していて、かつ JDK17 以降の場合)

Heap Status

garbage-first heap total <現在の容量>, used <使用中のメモリ> [<領域先頭アドレス>, <予約済み領域のアドレス>]
region size <1リージョンのサイズ>, <New領域の使用リージョン数> young (<New領域の使用メモリ>), <Survivor領域の使用リージョン数> survivors (<Survivor領域の使用メモリ>)
Metaspace max <最大の容量>, committed <コミット済みのメモリサイズ>, reserved <予約済みのメモリサイズ>, used <使用中のメモリ> (<最大の容量に対する使用率)% used/max, <現在の容量に対する使用率)% used/committed)
class space max <最大の容量>, committed <コミット済みのメモリサイズ>, reserved <予約済みのメモリサイズ>, used <使用中のメモリ> (<最大の容量に対する使用率)% used/max, <現在の容量に対する使用率)% used/committed)
[<領域先頭アドレス>, <予約済み領域の末尾アドレス>]

Java ヒープ情報 (ZGC を使用している場合)

Heap Status

ZHeap used <使用中のメモリ>, capacity <コミット済みの領域からFree領域を除いたメモリサイズ>, max capacity <最大のコミット済みの領域からFree領域を除いたメモリサイズ>
Metaspace max <最大の容量>, committed <コミット済みのメモリサイズ>, reserved <予約済みのメモリサイズ>, used <使用中のメモリ> (<最大の容量に対する使用率)% used/max, <現在の容量に対する使用率)% used/committed)
class space max <最大の容量>, committed <コミット済みのメモリサイズ>, reserved <予約済みのメモリサイズ>, used <使用中のメモリ> (<最大の容量に対する使用率)% used/max, <現在の容量に対する使用率)% used/committed)

に対する使用率)% used/committed)
[<領域先頭アドレス>, <予約済み領域の末尾アドレス>)

注

次の情報の単位はメガバイトです。それ以外の容量およびメモリサイズの単位はキロバイトです。

- <使用中のメモリ>
- <コミット済みの領域から Free 領域を除いたメモリサイズ>
- <最大のコミット済みの領域から Free 領域を除いたメモリサイズ>

JavaVM 内部メモリマップ情報

JVM Internal Memory Map

<メモリ確保関数>:address = <開始アドレス> - <終了アドレス> (size:<サイズ>)

注

<メモリ確保関数>: mmap()かmalloc()のどちらかが出力されます。

<開始アドレス>: メモリ領域の開始アドレスが 16 進で出力されます。

<終了アドレス>: メモリ領域の終了アドレスが 16 進で出力されます。

<サイズ>: 確保しているメモリ領域のサイズが出力されます (単位: バイト)。

JavaVM 内部メモリサイズ情報

JVM Internal Memory Status

Heap Size :<確保しているメモリサイズ>※
Alloc Size :<使用中のメモリサイズ>※
Free Size :<未使用のメモリサイズ>※

注※

単位: バイト

アプリケーション情報 (Windows の場合)

Application Environment

Signal Handlers -

SIGINT :<シグナルハンドラ情報>
SIGILL :<シグナルハンドラ情報>
...
SIGBREAK :<シグナルハンドラ情報>
SIGABRT :<シグナルハンドラ情報>

Environment Variables -

<環境変数>=<値>
...

```
Cueent Directory -  
C:¥Program Files¥Hitachi¥Cosminexus¥CC¥...
```

アプリケーション環境 (UNIX の場合)

Application Environment

Signal Handlers -※1

```
SIGHUP      :<シグナルハンドラ情報>  
SIGINT      :<シグナルハンドラ情報>  
...  
SIGSOUND    :<シグナルハンドラ情報>  
SIGSAK      :<シグナルハンドラ情報>
```

Signal Handlers -

```
シグナル種別: [シグナルハンドラアドレス], sa_mask[0]=シグナルマスク, sa_flags=特殊フラグ  
...
```

Environment Variables -※2

```
<環境変数>=<値>  
...
```

Current Directory -

```
/opt/Cosminexus/CC/server/...
```

注※1

次の情報が表示されます。

- シグナルハンドラがインストールされている場合は、そのアドレス。
- SIG_DFL である場合は、default。
- SIG_IGN である場合は、ignored。

注※2

次の情報が表示されます。

- シグナル種別には、/usr/include/sys/signal.h に定義されているシグナル名。
- シグナルハンドラアドレスには、シグナルハンドラのアドレスが 16 進数で出力されます。ライブラリ名+オフセットという形式で表示されることもあります。
- シグナルマスクには、sigaction() で取り出せる構造の sa_mask フィールド値が 16 進数で出力されます。
- 特殊フラグには、sigaction() で取り出せる構造の sa_flags フィールド値が 16 進数で出力されます。

ライブラリ情報 (Windows, Linux の場合)

Loaded Libraries

Dynamic libraries :

```
<開始アドレス> - <終了アドレス>   <コマンド>  
<開始アドレス> - <終了アドレス>   <ライブラリ>  
...
```

ライブラリ情報 (AIX の場合)

Loaded Libraries

<コマンド>

text :<開始アドレス>-<終了アドレス> (サイズ)
data :<開始アドレス>-<終了アドレス> (サイズ)

<ライブラリ>

text :<開始アドレス>-<終了アドレス> (サイズ)
data :<開始アドレス>-<終了アドレス> (サイズ)

...

スレッド情報

"<スレッド名>" <daemon> prio=<優先度> jid=<ハッシュ値> tid=<スレッドID> nid=<nativeID> <status> [開始アドレス... 終了アドレス]

java.lang.Thread.State: <スレッドの現在のステータス>※1

stack=[<スタック開始アドレス>..<<YellowPageアドレス>..<<RedPageアドレス>..<<スタック終了アドレス>]

[user cpu time=<ユーザー時間>ms, kernel cpu time=<カーネル時間>ms] [blocked count=<ブロック回数>, waited count=<待機回数>]

at <クラス名>.<メソッド名>(<モジュール情報>※2<メソッド情報>)

...

注※1

<スレッドの現在のステータス>の情報は、JDK 6 がベースの場合で、バージョンが 08-10 以降のときにだけ出力されます。

注※2

<モジュール情報>は、JDK9 以降がベースの場合にだけ出力されます。

出力内容を説明します。

<スレッド名>

Thread クラスのコンストラクタに指定されたスレッド名称が出力されます。

- <daemon>：デーモンスレッドである場合に、"daemon"と出力されます。
- <優先度>：Thread#setPriority で設定された優先度が出力されます。
- <ハッシュ値>：System.identityHashCode()を呼び出して得られる値と同一の値が 8 けたの 16 進数で出力されます。
- <スレッド ID>：スレッドオブジェクトのメモリ上のアドレス。
- <nativeID>：OS レベルのスレッド ID。
- <status>：スレッドの状態。

runnable：実行中または実行可能なスレッド

in Object.wait(), waiting for monitor entry または waiting on condition：モニタロック待ちのスレッド

sleeping：中断状態のスレッド

- <開始アドレス>：Java フレームの最高位スタックアドレスが 16 進数で出力されます。
- <終了アドレス>：JavaLock のある最高位スタックアドレスが 16 進数で出力されます。
- <ユーザー時間>：スレッド開始からのユーザー時間がミリ秒単位で出力されます。
- <カーネル時間>：スレッド開始からのカーネル時間がミリ秒単位で出力されます。
- <ブロック回数>：スレッド開始から、処理がブロックされた回数が出力されます。
- <待機回数>：スレッド開始から、処理が待ち状態になった回数が出力されます。

<スレッドの現在のステータス>

スレッドの現在のステータスを表すメッセージが出力されます。メッセージの内容は `java.lang.Thread.State` 列挙型に対応します。

<スタック開始アドレス>

スタック開始アドレスが 16 進数で出力されます。

<YellowPage アドレス>

スタック Yellow ガードページ先頭アドレスが 16 進数で出力されます。

<RedPage アドレス>

スタック Red ガードページ先頭アドレスが 16 進数で出力されます。

<スタック終了アドレス>

スタック終了アドレスが 16 進数で出力されます。

<クラス名>

クラス名が出力されます。

<メソッド名>

メソッド名が出力されます。

<モジュール情報>

モジュール名がある場合にだけ、モジュール名と JDK のバージョンが出力されます。

<メソッド情報>

次のメソッド情報が出力されます。

- Native Method
ネイティブメソッドの場合に出力されます。
- ファイル名：行番号
Java メソッドで行番号付きでコンパイルされている場合に出力されます。
- Unknown Source
Java メソッドで行番号なしでコンパイルされている場合に出力されます。

Java モニタダンプ

```
Java monitor
```

```
-----  
<ロックオブジェクト>@<ハッシュコード> <オーナー情報>
```

<待機状態>:<待機スレッド数>
<待機スレッド情報>

出力内容を説明します。

<ロックオブジェクト>

ロック対象オブジェクトのクラス名が出力されます。

<ハッシュコード>

Object.hashCode で得られるハッシュコードが出力されます。

<オーナー情報>

- owner "<スレッド名>"<スレッド ID>
オーナーがある場合に出力されます。
- no owner
オーナーがない場合に出力されます。

<待機状態>

- ... waiting to enter
メソッド実行待ちの場合に出力されます。
- ... waiting to be notified
通知待ちの場合に出力されます。

<待機スレッド数>

スレッド数が出力されます。

<待機スレッド情報>

["<スレッド名>" <スレッド ID>] の形式で出力されます。

JNI グローバル参照数の情報

JNI Information

JNI global references: <JNI グローバル参照数>

出力内容を説明します。

<JNI グローバル参照数>

JavaVM が保持しているグローバル参照の数が出力されます。

注

JNI グローバル参照は JavaVM の内部でも再利用されるため、JNI がサポートする DeleteGlobalRef 関数を発行して JNI グローバル参照を削除しても、数値は減少しません。また、NewGlobalRef 関数を発行して JNI グローバル参照を新規作成しても、JavaVM が再利用した JNI グローバル参照を割り当てた場合は数値は増加しません。

Explicit ヒープ情報および Explicit メモリブロック情報

Explicit Heap Status

```
max <EH_MAX>, total <EH_TOTAL>, used <EH_USED>, garbage <EH_GARB> (<EH_PER1> used/max, <EH_PER2> ¥
used/total, <EH_PER3> garbage/used), <EM_NUMS> spaces exist
```

Explicit Memories(<EM_MGR_PTR>)

```
"<EM_NAME>" eid=<EID>(<EM_PTR>)/<EM_TYPE>, total <EM_TOTAL>, used <EM_USED>, garbage <EM_GARB> ¥
```

```
(<EM_PER1> used/total, <EM_PER2> garbage/used, <FL_BLOCKS> blocks) <EM_STAT>
```

deployed objects

Size	Instances	FreeRatio	Class
<ISIZE>	<INUM>	<FRATIO>	<CNAME>

...	...	...	...
<AISIZE>	<AINUM>	total	

...

注意事項を次に示します。

- Explicit ヒープ情報と Explicit メモリブロック情報の間には、空行が 1 行あります。
- Explicit メモリブロック情報の出力順序（どの Explicit メモリブロックから出力するか）は未定義とします。
- "<EM_NAME>"の前には、半角 2 文字分の空白があります。
- deployed objects の前には、半角 4 文字分の空白があります。
- <ISIZE>は、 _____Size_ の"e"に行末をそろえて記載しています。
- <INUM>は、 __Instances の最後"s"に行末をそろえて記載しています。
- <FRATIO>は、 __FreeRatio_ の"o"に行末をそろえて記載しています。
- <CNAME>は、 __Class_____ の最初から二つ目の"_"に行末をそろえて記載しています。
- memory map の前には、半角 4 文字分の空白があります。
- 最後の行には空行が 1 行あります。したがって、Explicit メモリブロックごとの出力の間には空行が 1 行あります。

フッタ

```
Full thread dump completed.   EEE MMM dd hh:mm:ss yyyy※
```

注※

EEE は曜日, MMM は月, dd は日を表します。また, hh は時間, mm は分, ss は秒, yyyy は年（西暦）を表します。

拡張スレッドダンプ情報との比較を、次に示します。

表 14-4 標準スレッドダンプと拡張スレッドダンプの出力情報の比較

出力情報	標準スレッドダンプ	拡張スレッドダンプ
ヘッダ	×	○
システム設定	×	○
動作環境	×	○
メモリ情報（Windows の場合だけ）	×	○
Java ヒープ情報	×	○
JavaVM 内部メモリマップ情報	×	○
JavaVM 内部メモリサイズ情報	×	○
アプリケーション環境	×	○
ライブラリ情報	×	○
スレッド情報	○	○※1
Java モニタダンプ	×	○
JNI グローバル参照数の情報	○	○
Explicit ヒープ情報	×	○
Explicit メモリブロック情報	×	○
フッタ	×	○
スレッドダンプ出力先	標準出力	標準出力※2 JavaVM ログファイル

(凡例)

○：出力されます。

×：出力されません。

注※1

スタックの開始および終了のアドレス情報などが出力されます。

注※2

-XX:+HitachiThreadDumpToStdout オプションが指定された場合に出力されます。

注意事項

- 環境変数 JAVACOREDIR で指定したディレクトリへの出力に失敗した場合、カレントディレクトリに出力されます。
- カレントディレクトリへの出力に失敗した場合、標準エラー出力に出力されます。なお、この場合、スレッドダンプは標準出力に出力されません。
- 次に示すオプションの[+|-]指定が「-」の場合、スレッド情報の一部が出力されないのをご注意ください。

オプション名称	出力されない情報
-XX:[+ -]HitachiThreadDumpWithHashCode	<ハッシュ値>
-XX:[+ -]HitachiThreadDumpWithCpuTime	<ユーザー時間>, <カーネル時間>
-XX:[+ -]HitachiThreadDumpWithBlockCount	<ブロック回数>, <待機回数>

-XX:[+|-]HitachiThreadDumpToStdout（拡張スレッドダンプ標準出力抑止オプション）

形式

-XX:+HitachiThreadDumpToStdout

拡張スレッドダンプを標準出力およびスレッドダンプ出力ファイルに出力します。

-XX:-HitachiThreadDumpToStdout

拡張スレッドダンプを標準出力に出力しません。スレッドダンプ出力ファイルだけに出力します。

説明

拡張スレッドダンプを標準出力へ出力するかどうかを指定します。

このオプションの指定にかかわらず、次のメッセージは出力されます。また、拡張スレッドダンプはJavaVM ログファイルへ出力されます。

```
Writing Java core to <ファイル名（フルパス）>... OK
```

デフォルト値

- -XX:+HitachiThreadDumpToStdout

前提オプション

- -XX:+HitachiThreadDump

-XX:[+|-]HitachiThreadDumpWithHashCode（拡張スレッドダンプハッシュコード出力オプション）

形式

-XX:+HitachiThreadDumpWithHashCode

拡張スレッドダンプのスレッド情報にハッシュコードを出力します。

-XX:-HitachiThreadDumpWithHashCode

拡張スレッドダンプのスレッド情報にハッシュコードを出力しません。

説明

拡張スレッドダンプのスレッド情報にハッシュコードを出力するかどうかを指定します。

なお、ハッシュコードは、Java プログラムを実行しているスレッドに対して出力されます。JavaVM の内部動作スレッドに対しては出力されません。

デフォルト値

- -XX:+HitachiThreadDumpWithHashCode

前提オプション

- -XX:+HitachiThreadDump

-XX:[+|-]HitachiThreadDumpWithCpuTime (拡張スレッドダンプ CPU 利用時間出力オプション)

形式

-XX:+HitachiThreadDumpWithCpuTime

拡張スレッドダンプのスレッド情報に、スレッド開始からのユーザー CPU 時間とカーネル CPU 時間を出力します。

-XX:-HitachiThreadDumpWithCpuTime

拡張スレッドダンプのスレッド情報に、スレッド開始からのユーザー CPU 時間とカーネル CPU 時間を出力しません。

説明

拡張スレッドダンプのスレッド情報に、ユーザー CPU 時間とカーネル CPU 時間を出力するかどうかを指定します。

なお、ユーザー CPU 時間とカーネル CPU 時間は、Java プログラムを実行しているスレッドに対して出力されます。JavaVM の内部動作スレッドに対しては出力されません。

デフォルト値

- -XX:+HitachiThreadDumpWithCpuTime

前提オプション

- -XX:+HitachiThreadDump

-XX:[+|-]HitachiThreadDumpWithBlockCount (拡張スレッドダンプブロック回数出力オプション)

形式

-XX:+HitachiThreadDumpWithBlockCount

拡張スレッドダンプのスレッド情報に、スレッドが処理をブロックした回数と待ち状態になった回数を出力します。

-XX:-HitachiThreadDumpWithBlockCount

拡張スレッドダンプのスレッド情報に、スレッドが処理をブロックした回数と待ち状態になった回数を出力しません。

説明

拡張スレッドダンプのスレッド情報に、スレッドが処理をブロックした回数と待ち状態になった回数を出力するかどうかを指定します。

なお、ハッシュコードは、Java プログラムを実行しているスレッドに対して出力されます。JavaVM の内部動作スレッドに対しては出力されません。

デフォルト値

- -XX:+HitachiThreadDumpWithBlockCount

前提オプション

- -XX:+HitachiThreadDump

-XX:HitachiJavaLog (ログファイル名のプリフィックス指定オプション)

形式

-XX:HitachiJavaLog:<文字列>

説明

JavaVM ログファイルのプリフィックスおよびログファイルの出力先ディレクトリを指定します。

デフォルト値

- -XX:HitachiJavaLog:java log

前提オプション

次のどれかを指定します。

- -XX:+HitachiVerboseGC

- -XX:+HitachiOutOfMemoryStackTrace
- -XX:+HitachiOutOfMemoryCause
- -XX:+HitachiOutOfMemorySize
- -XX:+HitachiJavaClassLibTrace
- -XX:+JITCompilerContinuation

引数

<文字列>

プリフィックスおよびパスを指定します。次の 3 種類の指定ができます。

プリフィックスを指定する場合

ログファイル名は、<文字列>???.log (??は 01～99 の通し番号) で生成されます。例えば、<文字列>に"Samp"を指定すると、ログファイル名は Samp01.log になります。このオプションを指定しない場合、<文字列>には、"javalog"が設定されます。また、ログファイルはカレントディレクトリに出力されます。

パスを指定する場合

<文字列>にディレクトリを指定した場合、そのディレクトリにファイルが作成されます。ログファイル名は、<文字列>javalog???.log (??は 01～99 の通し番号) で生成されます。

パスとプリフィックスを同時に指定する場合

<文字列>にディレクトリとプリフィックスを指定した場合、そのディレクトリにファイルが作成されます。ログファイル名は、<文字列>???.log (??は 01～99 の通し番号) で生成されます。例えば、<文字列>に"d:¥temp¥Samp"を指定すると、d:¥temp ディレクトリに、Samp01.log が生成されます。

-XX:HitachiJavaLogFileSize (最大ログファイルサイズ指定オプション)

形式

-XX:HitachiJavaLogFileSize=<整数値>

説明

ログファイルの単純増加を防ぐため、1 ファイルの最大ファイルサイズを指定します。最大ファイルサイズを超えた場合は、そのファイルへの出力は行いません。オプションの指定がない場合は、デフォルトで 256 キロバイトが設定されます。

デフォルト値

- -XX:HitachiJavaLogFileSize=256k

前提オプション

次のどれかを指定します。

- -XX:+HitachiVerboseGC
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:+HitachiOutOfMemoryCause
- -XX:+HitachiOutOfMemorySize
- -XX:+HitachiJavaClassLibTrace
- -XX:+JITCompilerContinuation

引数

<整数値>

1024～2147483647 の範囲で整数値（単位：バイト）を指定します。範囲外の値が指定された場合は1024 が設定されます。負の値を指定した場合はエラーとなります。

-XX:[+|-]HitachiJavaLogNoMoreOutput（ログファイル入出力エラー発生時指定オプション）

形式

-XX:+HitachiJavaLogNoMoreOutput

ログファイル出力時にファイル入出力エラーが発生した場合、次のメッセージを標準エラー出力に出力し、ログ情報の出力を停止します。

```
Java logfile output failed.(errno=<エラーとなった入出力関数名>:<エラー番号>)
```

-XX:-HitachiJavaLogNoMoreOutput

ログファイル出力時にファイル入出力エラーが発生した場合、次のメッセージを標準エラー出力に出力し、ログ情報の出力先を標準エラー出力に変更して出力を継続します。

```
Java logfile output failed.(errno=<エラーとなった入出力関数名>:<エラー番号>) Changing output to stderr
```

説明

ログファイル作成時に入出力エラーが発生した場合に、ログ情報の出力方法について指定します。なお、どちらを指定した場合も、JavaVM の処理は継続されます。

デフォルト値

- -XX:+HitachiJavaLogNoMoreOutput

前提オプション

次のどれかを指定します。

- -XX:+HitachiVerboseGC
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:+HitachiOutOfMemoryCause
- -XX:+HitachiOutOfMemorySize
- -XX:+HitachiJavaClassLibTrace
- -XX:HitachiExplicitMemoryLogLevel:none以外
- -XX:+JITCompilerContinuation

-XX:HitachiJavaLogNumberOfFile（最大ログファイル数指定オプション）

形式

```
-XX:HitachiJavaLogNumberOfFile=<整数値>
```

説明

ログファイルの単純増加を防ぐため、作成する最大ファイル数を指定します。最大ファイル数を超えた場合は、再度最初に作成したファイルへ出力を開始します。オプションの指定がない場合は、デフォルトで4が設定されます。

デフォルト値

- -XX:HitachiJavaLogNumberOfFile=4

前提オプション

次のどれかを指定します。

- -XX:+HitachiVerboseGC
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:+HitachiOutOfMemoryCause
- -XX:+HitachiOutOfMemorySize
- -XX:+HitachiJavaClassLibTrace

引数

<整数値>

1～99の範囲で指定します。100以上の値が指定された場合は99が、0以下の値が指定された場合は1が設定されます。負の値を指定した場合はエラーとなります。

-XX:[+|-]JavaLogAsynchronous

形式

-XX:+JavaLogAsynchronous

ログファイルの非同期出力機能を有効にします。

-XX:-JavaLogAsynchronous

ログファイルの非同期出力機能を無効にします。

説明

ログファイルの非同期出力機能の有効，無効を指定します。

ログファイルの非同期出力機能使用時，-XX:HitachiOutOfMemoryStackTraceLineSize オプション，または-XX:HitachiJavaClassLibTraceLineSize オプションに，4096 より大きな値を指定した場合は，出力するスタックトレース 1 行の文字数に 4096 バイトが指定されたものとして動作します。

指定したバイト数が確保できない場合は警告メッセージが出力され，スタックトレースは出力されません。また，1 行の文字数が指定した文字数を超えた場合，「at」以降の文字列の前半部分を削除して，指定された文字数分出力します。

デフォルト値

- -XX:-JavaLogAsynchronous

-XX:[+|-]HitachiOutputMilliTime（詳細時間出力オプション）

形式

-XX:+HitachiOutputMilliTime

JavaVM ログファイルに出力する日時に，ミリ秒まで出力します。

-XX:-HitachiOutputMilliTime

JavaVM ログファイルに出力する日時に，秒まで出力します。

説明

ミリ秒までの時間を出力するかどうかを指定します。

デフォルト値

- -XX:-HitachiOutputMilliTime

前提オプション

次のどれかを指定します。

- -XX:+HitachiVerboseGC
- -XX:+HitachiOutOfMemoryStackTrace
- -XX:+HitachiOutOfMemoryCause
- -XX:+HitachiOutOfMemorySize
- -XX:+HitachiJavaClassLibTrace
- -XX:HitachiExplicitMemoryLogLevel:none 以外
- -XX:+JITCompilerContinuation

出力例

- 拡張 verbosegc 情報の出力

```
[VGC]<Wed Mar 17 00:45:55.068 2004>(Skip Full:0, Copy:0)[Full GC 149K->149K(1984K), 0.0786
038 secs][DefNew::Eden: 264K->0K(512K)][DefNew::Survivor: 0K->63K(64K)][Tenured: 85K->149
K(1408K)][Metaspace: 3634K(4492K, 4492K)->3634K(4492K, 4492K)][class space: 356K(388K, 38
8K)->356K(388K, 388K)]
```

- OutOfMemoryError 発生時の出力

```
[OOM][Thread: 0x00957820]<Wed Mar 17 00:47:00.662 2004>[java.lang.OutOfMemory Error :(C H
eap) :340]
```

- クラスライブラリトレースの出力

```
[CLT][Thread: 0x00286348]<Wed Mar 17 00:47:00.662 2004>
[CLT][Thread: 0x00286348] at java.lang.Shutdown.halt0(Native Method)
[CLT][Thread: 0x00286348] at java.lang.Shutdown.halt(Shutdown.java:145)
```

- JIT コンパイラ稼働継続機能 (JIT コンパイル失敗情報)

```
[JCC][Thread: 0x05432c00]<Thu Nov 15 17:10:40.347 2012>[Method: chosa_cmp.func(Ljava/lang
/String;)V][Fail: 3][JITCT: 1]
```

-XX:[+|-]HitachiVerboseGC (拡張 verbosegc 情報出力オプション)

形式

-XX:+HitachiVerboseGC

GC が発生した場合、拡張 verbosegc 情報を JavaVM ログファイルに出力します。

GC の内部領域である Eden, Survivor, Tenured, Metaspace 種別の情報を拡張 verbosegc 情報として出力します。

-XX:-HitachiVerboseGC

GC が発生した場合、拡張 verbosegc 情報を JavaVM ログファイルに出力しません。

説明

GC が発生した時の拡張 verbosegc 情報を出力するかどうかを指定します。

デフォルト値

- -XX:-HitachiVerboseGC

出力形式 (SerialGC を使用している場合)

```
[id] <date> (Skip Full:full_count, Copy:copy_count) [gc_kind gc_info, gc_time secs][Eden: eden_info][Survivor: survivor_info][Tenured: tenured_info][Metaspace: metaspace_info][class space: class_space_info][cause:cause_info] [User: user_cpu secs] [Sys: system_cpu secs][IM: jvm_alloc_size, mmap_total_size, malloc_total_size][TC: thread_count][DOE: doe_alloc_size, called_count][CCI: cc_used_sizeK, cc_max_sizeK, cc_infoK][DirectBuffer: buffer used sizeK, buffer total sizeK, buffer max sizeK, buffer count]
```

注意事項

下線が引かれている部分は、JDK17 以降の場合にだけ出力されます。

出力内容を次に説明します。

id

VGC (JavaVM ログファイル識別子)。

date

GC 開始日時。

-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

full_count

full GC 情報出力をスキップした回数。

-XX:HitachiVerboseGCIntervalTime オプションが指定された場合に出力されます。

copy_count

copy GC 情報出力をスキップした回数。

-XX:HitachiVerboseGCIntervalTime オプションが指定された場合に出力されます。

gc_kind

GC 種別。"FullGC", "GC"が出力されます。

gc_info

GC 情報。次の形式で出力されます。

```
<GC前の領域長> -> <GC後の領域長> (<領域サイズ>)
```

gc_time

GC 経過時間。

Eden

エデンの種別。"DefNew::Eden"または"ParNew::Eden"が出力されます。

eden_info

Eden 情報。次の形式で出力されます。

```
<GC前の領域長> -> <GC後の領域長> (<領域サイズ>)
```

Survivor

Survivor の種別。"DefNew:: Survivor"または"ParNew::Survivor"が出力されます。

survivor_info

Survivor 情報。次の形式で出力されます。

```
<GC前の領域長> -> <GC後の領域長> (<領域サイズ>)
```

Tenured

Tenured の種別。"Tenured"が出力されます。

tenured_info

Tenured 情報。次の形式で出力されます。

```
<GC前の領域長> -> <GC後の領域長> (<領域サイズ>)
```

metaspace_info

Metaspace 領域の情報。次の形式で出力されます。

```
<GC前の使用サイズ>(<GC前のcapacityサイズ>, <GC前のcommitサイズ>) -> <GC後の使用サイズ>(<GC後のcapacityサイズ>, <GC後のcommitサイズ>)
```

class_space_info

Compressed Class Space の情報。次の形式で出力されます。

```
<GC前の使用サイズ>(<GC前のcapacityサイズ>, <GC前のcommitサイズ>) -> <GC後の使用サイズ>(<GC後のcapacityサイズ>, <GC後のcommitサイズ>)
```

-XX:- UseCompressedOops オプションが指定された場合、出力されません。

cause_info

GC 要因内容。

-XX:-HitachiVerboseGCPrintCause オプションが指定された場合、出力されません。

user_cpu

GC スレッドがユーザーモードで費やした CPU 時間。単位は秒です。

CPU 時間取得に失敗した場合、[User: unknown]のように、"unknown"と表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

system_cpu

GC スレッドがカーネルモードで費やした CPU 時間。単位は秒です。

CPU 時間取得に失敗した場合、[Sys: unknown]のように、"unknown"と表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

jvm_alloc_size

JavaVM 内部で管理している領域のうち、現在使用中の領域のサイズ (mmap_total_size と malloc_total_size の合計サイズのうち、現在使用中の領域のサイズ)。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

mmap_total_size

JavaVM 内部で管理している領域のうち、mmap (Windows の場合は VirtualAlloc) で割り当てた C ヒープの総サイズ。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

malloc_total_size

JavaVM 内部で管理している領域のうち、malloc で割り当てた C ヒープの総サイズ。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

thread_count

Java スレッドの数。

-XX:-HitachiVerboseGCPrintThreadCount オプションが指定された場合、出力されません。

doe_alloc_size

java.io.File.deleteOnExit() を呼び出して確保した累積のヒープサイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

called_count

java.io.File.deleteOnExit() の呼び出し回数。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

cc_used_size

GC 発生時のコードキャッシュ領域の使用サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_info

保守情報。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

buffer_used_size

ダイレクトバッファの使用サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_total_size

ダイレクトバッファの合計容量。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_max_size

ダイレクトバッファの最大値。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_count

ダイレクトバッファの数。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

出力形式 (G1GC を使用している場合)

- VGI ログ

```
[id]<date>[gc_kind gc_info, gc_time secs][Status: gc_status][G1GC:Eden: eden_info][G1GC:Survivor: survivor_info][G1GC:Tenured: tenured_info][G1GC:Humongous: humongous_info][G1GC:Free: free_info][Metaspace: metaspace_info][class space: class_space_info][cause:cause_info][RegionSize: region_size][Target: target_time secs][Predicted: predicted_time secs][TargetTenured: target_size][Reclaimable: reclaimable_info][User: user_cpu secs] [Sys: system_cpu secs][IM: jvm_alloc_size, mmap_total_size, malloc_total_size][TC: thread_count][DOE: doe_alloc_size, called_count][CCI: cc_used_sizeK, cc_max_sizeK, cc_infoK][GCID: gc_id][DirectBuffer: buffer used sizeK, buffer total sizeK, buffer max sizeK, buffer count]
```

注意事項

- 下線が引かれている部分は、JDK17 以降の場合にだけ出力されます。
- 背景色付きの太字部分は、JDK21 以降の場合には出力されません。

出力内容を次に説明します。

id

VGI (JavaVM ログファイル識別子)。

date

GC 開始日時。

-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

gc_kind

GC 種別。“Full GC” , “Mixed GC” , “Young GC” , “Young GC(initial-mark)” , “CM Remark” , “CM Cleanup” のどれかが出力されます。

gc_info

GC 情報。次の形式で出力されます。

```
<GC前の領域サイズ>/<GC前の領域サイズ(リージョン換算)>(<GC前の領域サイズ>) -> <GC後の領域サイズ>/<GC後の領域サイズ(リージョン換算)>(<GC後の領域サイズ>)
```

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ、1 リージョンのサイズの倍数で表した値です。

gc_time

GC 経過時間。

gc_status

GC の状態。” - ” , “to exhausted” が出力されます。

eden_info

Eden 情報。次の形式で出力されます。

```
<GC前の領域サイズ(リージョン換算)><GC前の最大領域サイズ(リージョン換算)>> -> <GC後の領域  
サイズ(リージョン換算)><GC後の最大領域サイズ(リージョン換算)>>
```

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ, 1 リージョンのサイズの倍数で表した値です。

survivor_info

Survivor 情報。次の形式で出力されます。

```
<GC前の領域サイズ(リージョン換算)> -> <GC後の領域サイズ(リージョン換算)>
```

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ, 1 リージョンのサイズの倍数で表した値です。

tenured_info

Tenured 情報。次の形式で出力されます。

```
<GC前の領域サイズ(リージョン換算)> -> <GC後の領域サイズ(リージョン換算)>
```

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ, 1 リージョンのサイズの倍数で表した値です。

humongous_info

Humongous 情報。次の形式で出力されます。

```
<GC前の領域サイズ(リージョン換算)> -> <GC後の領域サイズ(リージョン換算)>
```

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ, 1 リージョンのサイズの倍数で表した値です。

free_info

Free 情報。次の形式で出力されます。

```
<GC前の領域サイズ(リージョン換算)> -> <GC後の領域サイズ(リージョン換算)>
```

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ, 1 リージョンのサイズの倍数で表した値です。

metaspace_info

Metaspace 領域の情報。次の形式で出力されます。

```
<GC前の使用サイズ><GC前のcapacityサイズ>, <GC前のcommitサイズ>> -> <GC後の使用サイズ><G  
C後のcapacityサイズ>, <GC後のcommitサイズ>
```

class_space_info

Compressed Class Space の情報。次の形式で出力されます。

<GC前の使用サイズ>(<GC前のcapacityサイズ>, <GC前のcommitサイズ>) -> <GC後の使用サイズ>(<GC後のcapacityサイズ>, <GC後のcommitサイズ>)

-XX:- UseCompressedOops オプションが指定された場合、出力されません。

cause_info

GC 要因内容。

-XX:-HitachiVerboseGCPrintCause オプションが指定された場合、出力されません。

region_size

1 リージョンのサイズです。

単位はキロバイトです。

target_time

GC によるアプリケーション停止時間の目標時間です。

単位は秒です。

predicted_time

JavaVM が予測した GC によるアプリケーション停止時間です。

単位は秒です。

なお、GC 種別が” Full GC” ,” CM Remark” , “CM Cleanup” のときは予測をしないため、0 が出力されます。

target_size

Mixed GC で GC 対象となった Tenured 領域のサイズです。

単位はキロバイト。

なお、GC 種別が” Mixed GC” 以外のときは、0 が出力されます。

reclaimable_info

MixedGC で回収される Tenured 領域の予測回収サイズ情報。次の形式で出力されます。

<予測回収サイズ> (<予測回収率>)

なお、予測回収サイズ情報は CM 終了直後の Young GC または Mixed GC だけ出力されます。それ以外の場合、予測を行わないため、0 が出力されます。

user_cpu

GC スレッドがユーザーモードで費やした CPU 時間。単位は秒です。

CPU 時間取得に失敗した場合、[User: unknown]のように、"unknown"と表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

system_cpu

GC スレッドがカーネルモードで費やした CPU 時間。単位は秒です。

CPU 時間取得に失敗した場合、[Sys: unknown]のように、"unknown"と表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

jvm_alloc_size

JavaVM 内部で管理している領域のうち、現在使用中の領域のサイズ (mmap_total_size と malloc_total_size の合計サイズのうち、現在使用中の領域のサイズ)。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

mmap_total_size

JavaVM 内部で管理している領域のうち、mmap (Windows の場合は VirtualAlloc) で割り当てた C ヒープの総サイズ。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

malloc_total_size

JavaVM 内部で管理している領域のうち、malloc で割り当てた C ヒープの総サイズ。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

thread_count

Java スレッドの数。

-XX:-HitachiVerboseGCPrintThreadCount オプションが指定された場合、出力されません。

doe_alloc_size

java.io.File.deleteOnExit() を呼び出して確保した累積のヒープサイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

called_count

java.io.File.deleteOnExit() の呼び出し回数。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

cc_used_size

GC 発生時のコードキャッシュ領域の使用サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_info

保守情報。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

gc_id

GC ごとに付与される識別子。

buffer_used_size

ダイレクトバッファの使用サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_total_size

ダイレクトバッファの合計容量。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_max_size

ダイレクトバッファの最大値。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_count

ダイレクトバッファの数。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

- VCM

```
[id]<date>[cm_event][User: user_cpu secs][Sys: sys_cpu secs]
```

id

VCM (JavaVM ログファイル識別子)。

date

CM 開始日時。

-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

cm_event

CM 種別。"Concurrent Root Region Scan Start", "Concurrent Root Region Scan End", "Concurrent Mark Start", "Concurrent Mark End", "Concurrent Mark Stop", "Concurrent Cleanup Start", "Concurrent Cleanup End"のどれかが出力されます。

user_cpu

全 CM スレッドがユーザーモードで費やした CPU 時間。単位は秒です。

-XX:-HitachiVerboseGCCpuTime 指定時は出力されません。

CPU 時間取得に失敗した場合、[User: unknown]のように、"unknown"と表示されます。

CM の状態が Start の場合、0 が出力されます。

sys_cpu

全 CM スレッドがカーネルモードで費やした CPU 時間。単位は秒です。

-XX:-HitachiVerboseGCCpuTime 指定時は出力されません。

CPU 時間取得に失敗した場合、[Sys: unknown]のように、"unknown"と表示されます。

CM の状態が Start の場合、0 が出力されます。

出力形式 (ZGC を使用している場合)

- VZG ログ

```
[id]<date>[h_used_beforeK(h_size_beforeK)->h_used_afterK(h_size_afterK), gc_time msecs][ZGC::Free: f_used_beforeK->f_used_afterK][Metaspace: m_used_beforeK(m_capacity_beforeK, m
```



```
_commit_beforeK)->m_used_afterK(m_capacity_afterK, m_commit_afterK)][class space: c_used_
beforeK(c_capacity_before K, c_commit_before K)->c_used_after K(c_capacity_after K, c_com
mit_after K)][SmallPage: s_count/s_sizeM(s_emptyM, s_relocatedM, s_inPlaceinP)][MediumPag
e: m_count/m_sizeM(m_emptyM, m_relocated M, m_inPlaceinP)] [LargePage: l_count/l_sizeM(l_
emptyM, l_relocatedM, l_inPlaceinP)][cause:cause_info][User: user_cpu_stw1secs, user_cpu_
stw2secs, user_cpu_stw3secs][Sys: sys_cpu_stw1secs, sys_cpu_stw2secs, sys_cpu_stw3secs][I
M: jvm_alloc_sizeK, mmap_total_sizeK, malloc_total_sizeK][TC: thread_count][DOE: doe_allo
c_sizeK, called_count][CCI: cc_used_sizeK, cc_max_sizeK, cc_infoK][DirectBuffer: buffer_u
sed_sizeK, buffer_total_sizeK, buffer_max_sizeK, buffer_count]
```

出力内容を次に説明します。

id

VZG（日立 JavaVM ログファイル識別子）。

date

ZGC サイクルが開始された日時。

-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

h_used_before

GC 前の Java ヒープ領域の使用サイズ。単位はキロバイトです。

h_size_before

GC 前の Java ヒープ領域の領域サイズ。単位はキロバイトです。

h_used_after

GC 後の Java ヒープ領域の使用サイズ。単位はキロバイトです。

h_size_after

GC 後の Java ヒープ領域の領域サイズ。単位はキロバイトです。

gc_time

GC によるアプリケーション停止時間。STW1, STW2, STW3 の総数です。単位はミリ秒（msec）です。

f_used_before

GC 前の Free 領域のサイズ。単位はキロバイトです。

f_used_after

GC 後の Free 領域のサイズ。単位はキロバイトです。

m_used_before

GC 前の Metaspace 領域の使用サイズ。単位はキロバイトです。

m_capacity_before

GC 前の Metaspace 領域の capacity サイズ。単位はキロバイトです。

m_commit_before

GC 前の Metaspace 領域の commit サイズ。単位はキロバイトです。

m_used_after

GC 後の Metaspace 領域の使用サイズ。単位はキロバイトです。

m_capacity_after

GC 後の Metaspace 領域の capacity サイズ。単位はキロバイトです。

m_commit_after

GC 後の Metaspace 領域の commit サイズ。単位はキロバイトです。

c_used_before

GC 前の class space 領域の使用サイズ。単位はキロバイトです。

圧縮オブジェクトポインタ機能が無効の場合は出力されません。

c_capacity_before

GC 前の class space 領域の capacity サイズ。単位はキロバイトです。

圧縮オブジェクトポインタ機能が無効の場合は出力されません。

c_commit_before

GC 前の class space 領域の commit サイズ。単位はキロバイトです。

圧縮オブジェクトポインタ機能が無効の場合は出力されません。

c_used_after

GC 後の class space 領域の使用サイズ。単位はキロバイトです。

圧縮オブジェクトポインタ機能が無効の場合は出力されません。

c_capacity_after

GC 後の class space 領域の capacity サイズ。単位はキロバイトです。

圧縮オブジェクトポインタ機能が無効の場合は出力されません。

c_commit_after

GC 後の class space 領域の commit サイズ。単位はキロバイトです。

圧縮オブジェクトポインタ機能が無効の場合は出力されません。

s_count

Small ページの枚数。

s_size

Small ページの総サイズ。単位はメガバイトです。

s_empty

Small ページの空きサイズ。単位はメガバイトです。

s_relocated

Small ページに再配置されたオブジェクトサイズ。単位はメガバイトです。

s_inPlace

空きメモリがない場合に InPlace 方式でオブジェクトが再配置された Small ページ枚数。

m_count

Medium ページの枚数。

m_size

Medium ページの総サイズ。単位はメガバイトです。

m_empty

Medium ページの空きサイズ。単位はメガバイトです。

m_relocated

Medium ページに再配置されたオブジェクトサイズ。単位はメガバイトです。

m_inPlace

空きメモリがない場合に InPlace 方式でオブジェクトが再配置された Medium ページ枚数。

l_count

Large ページの枚数。

l_size

Large ページの総サイズ。単位はメガバイトです。

l_empty

Large ページの空きサイズ。単位はメガバイトです。

l_relocated

Large ページに再配置されたオブジェクトサイズ。単位はメガバイトです。

l_inPlace

空きメモリがない場合に InPlace 方式でオブジェクトが再配置された Large ページ枚数。

cause_info

GC 要因内容。

-XX:-HitachiVerboseGCPrintCause オプションが指定された場合、出力されません。

user_cpu_stw1

STW1 で全 GC スレッドがユーザモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。-XX:-

HitachiVerboseGCPrintCause オプションが指定された場合、出力されません。

user_cpu_stw2

STW2 で全 GC スレッドがユーザモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:- HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

user_cpu_stw3

STW3 で全 GC スレッドがユーザモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:- HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

sys_cpu_stw1

STW1 で全 GC スレッドがカーネルモードで費やした CPU 時間の合計。単位は秒 (sec) です。
CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

sys_cpu_stw2

STW2 で全 GC スレッドがカーネルモードで費やした CPU 時間の合計。単位は秒 (sec) です。
CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

sys_cpu_stw3

STW3 で全 GC スレッドがカーネルモードで費やした CPU 時間の合計。単位は秒 (sec) です。
CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

jvm_alloc_size

JavaVM 内部で管理している領域のうち、現在使用中の領域のサイズ (mmap_total_size と malloc_total_size の合計サイズのうち、現在使用中の領域のサイズ)。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

mmap_total_size

JavaVM 内部で管理している領域のうち、mmap (Windows の場合は VirtualAlloc) で割り当てた C ヒープの総サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

malloc_total_size

JavaVM 内部で管理している領域のうち、malloc で割り当てた C ヒープ総サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

thread_count

Java スレッドの数。

-XX:-HitachiVerboseGCPrintThreadCount オプションが指定された場合、出力されません。

doe_alloc_size

java.io.File.deleteOnExit() を呼び出して確保した累積のヒープサイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

called_count

java.io.File.deleteOnExit() の呼び出し回数。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

cc_used_size

GC 発生時のコードキャッシュ領域の使用サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_info

保守情報。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

buffer_used_size

ダイレクトバッファの使用サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_total_size

ダイレクトバッファの合計容量。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_max_size

ダイレクトバッファの最大値。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_count

ダイレクトバッファの数。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

- VZP ログ

[id]<date>[phase, phase_time msecs]

出力内容を次に説明します。

id

VZP（日立 JavaVM ログファイル識別子）。

date

各フェーズが開始された日時。

-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

phase

STW またはフェーズの種別。

phase_time

フェーズに掛かった時間。単位はミリ秒（msec）です。

出力例

SerialGC を使用している場合

- -XX:HitachiVerboseGCIntervalTime オプションが指定されている場合

```
[VGC]<Tue Nov 22 11:01:52.180 2022>(Skip Full:1, Copy:0)[Full GC 3250K->3250K(8064K), 0.0106676 secs][DefNew::Eden: 0K->0K(2304K)][DefNew::Survivor: 0K->0K(256K)][Tenured: 3250K->3250K(5504K)][Metaspace: 8613K(8896K, 8896K)->8613K(8896K, 8896K)][class space: 1131K(1280K, 1280K)->1131K(1280K, 1280K)][cause: System.gc][User: 0.0156250 secs][Sys: 0.0000000 secs][IM: 9933K, 10144K, 0K][TC: 16][DOE: 0K, 0][CCI: 895K, 49152K, 2496K][DirectBuffer: 8K, 8K, 83200K, 1]
```

注意事項

下線が引かれている部分は、JDK17 以降の場合にだけ出力されます。

G1GC を使用している場合

- VGI ログ

```
[VG1]<Tue Nov 22 11:34:45.443 2022>[Full GC 3287K/4096K(14336K)->3287K/4096K(14336K), 0.0108105 secs][Status:-][G1GC::Eden: 0K(4096K)->0K(4096K)][G1GC::Survivor: 0K->0K][G1GC::Tenured: 4096K->4096K][G1GC::Humongous: 0K->0K][G1GC::Free: 10240K->10240K][Metaspace: 8575K(8832K, 8832K)->8575K(8832K, 8832K)][class space: 1131K(1280K, 1280K)->1131K(1280K, 1280K)][cause: System.gc][RegionSize: 1024K][Target: 0.2000000 secs][Predicted: 0.0000000 secs][TargetTenured: 0K][Reclaimable: 0K(0.00%)] [User: 0.0156250 secs][Sys: 0.0000000 secs][IM: 31447K, 36256K, 0K][TC: 27][DOE: 0K, 0][CCI: 910K, 49152K, 2496K][GCID: 20][DirectBuffer: 8K, 8K, 86016K, 1]
```

注意事項

下線が引かれている部分は、JDK17 以降の場合にだけ出力されます。

- VCM ログ

```
[VCM]<Wed Jul 24 11:45:20 2013>[Concurrent Root Region Scan Start][User: 0.0000000 secs][Sys: 0.0000000 secs]
[VCM]<Wed Jul 24 11:45:20 2013>[Concurrent Root Region Scan End][User: 0.0126134 secs][Sys: 0.0146961 secs]
[VCM]<Wed Jul 24 11:45:20 2013>[Concurrent Mark Start][User: 0.0000000 secs][Sys: 0.0000000 secs]
[VCM]<Wed Jul 24 11:45:34 2013>[Concurrent Mark End][User: 0.0156250 secs][Sys: 0.2495800 secs]
```

ZGC を使用している場合

- VZG ログ

```
[VZG]<Thu Feb 16 17:16:45 2023>[18432K(18432K)->6144K(20480K), 0.0069 msecs][ZGC::Free: 67584K->79872K][Metaspace: 3849K(3968K, 3968K)->3849K(3968K, 3968K)][SmallPage: 5/10M(8M, 0M, 0inP)][MediumPage: 0/0M(0M, 0M, 0inP)][LargePage: 4/8M(6M, 0M, 0inP)][cause: ZWarmup][User: 0.0010000secs, 0.0025000secs, 0.0014000secs][Sys: 0.0012000secs, 0.0011000secs, 0.0015000secs][IM: 5098K, 6752K, 0K][TC: 20][DOE: 0K, 0][CCI: 556K, 0K, 0K]
```

- VZP ログ

-XX:[+|-]HitachiCommaVerboseGC (CSV 出力オプション)

形式

-XX:+HitachiCommaVerboseGC

拡張 verbosegc 情報の出力を、CSV ファイルで取得できるようにコンマ形式で出力します。

拡張 verbosegc 情報に出力される括弧（丸括弧(), 角括弧[], 山括弧<>）およびコロン(:) をすべて削除し、コンマ(,) で区切った数値または文字列を出力します。

-XX:-HitachiCommaVerboseGC

拡張 verbosegc 情報を通常形式で出力します。

説明

拡張 verbosegc 情報を CSV 形式で出力するかどうかを指定します。

デフォルト値

- -XX:-HitachiCommaVerboseGC

前提オプション

- -XX:+HitachiVerboseGC

出力形式 (SerialGC を使用している場合)

-XX:-HitachiVerboseGCIntervalTime オプションが指定されている場合の出力内容を次に説明します。

```
id, date, full_count, copy_count, inc_count, gc_kind, gc_info, gc_time, eden_info, survivor_
info,tenured_info, metaspace_info, classspace_info, cause_info, user_cpu, system_cpu, jvm_al
loc_size, mmap_total_size, malloc_total_size, thread_count, doe_alloc_size, called_count, cc
_used_size, cc_max_size, cc_info, buffer used size,buffer total size,buffer max size,buffer
count
```

注意事項

下線が引かれている部分は、JDK17 以降の場合にだけ出力されます。

id

JavaVM ログファイル識別子。

date

GC 開始日時。-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

full_count

full GC 情報出力をスキップした回数。

-XX:HitachiVerboseGCIntervalTime オプションが指定された場合に出力されます。

copy_count

copy GC 情報出力をスキップした回数。

-XX:HitachiVerboseGCIntervalTime オプションが指定された場合に出力されます。

inc_count

0 を表示。

-XX:HitachiVerboseGCIntervalTime オプションが指定された場合に出力されます。

gc_kind

GC 種別。"FullGC"または"GC"が出力されます。

gc_info

GC 情報。次の形式で出力されます。単位はキロバイトです。

〈GC前の領域長〉,〈GC後の領域長〉,〈領域サイズ〉

gc_time

GC 経過時間。単位は秒です。

eden_info

Eden 情報。次の形式で出力されます。単位はキロバイトです。

〈GC前の領域長〉,〈GC後の領域長〉,〈領域サイズ〉

survivor_info

Survivor 情報。次の形式で出力されます。単位はキロバイトです。

〈GC前の領域長〉,〈GC後の領域長〉,〈領域サイズ〉

tenured_info

Tenured 情報。次の形式で出力されます。単位はキロバイトです。

〈GC前の領域長〉,〈GC後の領域長〉,〈領域サイズ〉

metaspace_info

Metaspace 領域の情報。次の形式で出力されます。単位はキロバイトです。

〈GC前の使用サイズ〉,〈GC前のcapacityサイズ〉,〈GC前のcommitサイズ〉,〈GC後の使用サイズ〉,〈GC後のcapacityサイズ〉,〈GC後のcommitサイズ〉

classspace_info

Compressed Class Space 情報。次の形式で出力されます。単位はキロバイトです。

〈GC前の使用サイズ〉,〈GC前のcapacityサイズ〉,〈GC前のcommitサイズ〉,〈GC後の使用サイズ〉,〈GC後のcapacityサイズ〉,〈GC後のcommitサイズ〉

cause_info

GC 要因番号。

-XX:-HitachiVerboseGCPrintCause オプションが指定された場合、出力されません。

なお、GC 要因番号については、「-XX:[+|-]HitachiVerboseGCPrintCause (GC 要因内容出力オプション)」を参照してください。

user_cpu

GC スレッドがユーザーモードで費やした CPU 時間。単位は秒です。

CPU 時間の取得に失敗した場合は"unknown"と表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

system_cpu

GC スレッドがカーネルモードで費やした CPU 時間。単位は秒です。

CPU 時間の取得に失敗した場合は"unknown"と表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

jvm_alloc_size

JavaVM 内部で管理している領域のうち、現在使用中の領域のサイズ (mmap_total_size と malloc_total_size の合計サイズのうち、現在使用中の領域のサイズ)。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

mmap_total_size

JavaVM 内部で管理している領域のうち、mmap (Windows の場合は VirtualAlloc) で割り当てた C ヒープの総サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

malloc_total_size

JavaVM 内部で管理している領域のうち、malloc で割り当てた C ヒープ総サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

thread_count

Java スレッドの数。

-XX:-HitachiVerboseGCPrintThreadCount オプションが指定された場合、出力されません。

doe_alloc_size

java.io.File.deleteOnExit() を呼び出して確保した累積のヒープサイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

called_count

java.io.File.deleteOnExit() の呼び出し回数。

-XX:-HitachiVerboseGCPrintDeleteOnExit が指定された場合、出力されません。

cc_used_size

GC 発生時のコードキャッシュ領域の使用サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_info

保守情報。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

buffer_used_size

ダイレクトバッファの使用サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_total_size

ダイレクトバッファの合計容量。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_max_size

ダイレクトバッファの最大値。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_count

ダイレクトバッファの数。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

-XX:+HitachiVerboseGCPrintTenuringDistribution オプションが指定されている場合の出力内容を次に説明します。

id, date, size, value, max_value, total_age1, total_age2, total_agen

出力内容については、「[-XX:\[+|-\]HitachiVerboseGCPrintTenuringDistribution](#) (Survivor 領域の年齢分布出力オプション)」を参照してください。

出力形式 (G1GC を使用している場合)

- VGI ログ

id, date, gc_kind, gc_info, gc_time, gc_status, eden_info, survivor_info, tenured_info, humongous_info, free_info, metaspace_info, classspace_info, cause_info, region_size, target_time, predicted_time, target_size, reclaimable_info , user_cpu, system_cpu, jvm_alloc_size, mmap_total_size, malloc_total_size, thread_count, doe_alloc_size, called_count, cc_used_size, cc_max_size, cc_info, <u>buffer used size, buffer total size, buffer max size, buffer count</u>

注意事項

- 下線が引かれている部分は、JDK17 以降の場合にだけ出力されます。
- 背景色付きの太字部分は、JDK21 以降の場合には出力されません。

id

JavaVM ログファイル識別子。

date

GC 開始日時。-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

gc_kind

GC 種別。"Full GC", "Mixed GC", "Young GC", "Young GC(initial-mark)", "CM Remark", "CM Cleanup" のどれかが出力されます。

gc_info

GC 情報。次の形式で出力されます。単位はキロバイトです。

<GC前の領域サイズ>, <GC前の領域サイズ(リージョン換算)>, <GC前の領域サイズ>, <GC後の領域サイズ>, <GC後の領域サイズ(リージョン換算)>, <GC後の領域サイズ>

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ、1 リージョンのサイズの倍数で表した値です。

gc_time

GC 経過時間。

gc_status

GC の状態。"-", "to exhausted" が出力されます。

eden_info

Eden 情報。次の形式で出力されます。

<GC前の領域サイズ(リージョン換算)>, <GC前の最大領域サイズ(リージョン換算)>, <GC後の領域サイズ(リージョン換算)>, <GC後の最大領域サイズ(リージョン換算)>

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ、1 リージョンのサイズの倍数で表した値です。

survivor_info

Survivor 情報。次の形式で出力されます。

<GC前の領域サイズ(リージョン換算)>, <GC後の領域サイズ(リージョン換算)>

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ、1 リージョンのサイズの倍数で表した値です。

tenured_info

Tenured 情報。次の形式で出力されます。

<GC前の領域サイズ(リージョン換算)>,<GC後の領域サイズ(リージョン換算)>

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ、1 リージョンのサイズの倍数で表した値です。

humongous_info

Humongous 情報。次の形式で出力されます。

<GC前の領域サイズ(リージョン換算)>,<GC後の領域サイズ(リージョン換算)>

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ、1 リージョンのサイズの倍数で表した値です。

free_info

Free 情報。次の形式で出力されます。

<GC前の領域サイズ(リージョン換算)>,<GC後の領域サイズ(リージョン換算)>

リージョン換算とは領域サイズを 1 リージョンのサイズで切り上げ、1 リージョンのサイズの倍数で表した値です。

metaspace_info

Metaspace 領域の情報。次の形式で出力されます。単位はキロバイトです。

<GC前の使用サイズ>,<GC前のcapacityサイズ>,<GC前のcommitサイズ>,<GC後の使用サイズ>,<GC後のcapacityサイズ>,<GC後のcommitサイズ>

classspace_info

Compressed Class Space 情報。次の形式で出力されます。単位はキロバイトです。

<GC前の使用サイズ>,<GC前のcapacityサイズ>,<GC前のcommitサイズ>,<GC後の使用サイズ>,<GC後のcapacityサイズ>,<GC後のcommitサイズ>

cause_info

GC 要因番号。

-XX:-HitachiVerboseGCPrintCause オプションが指定された場合、出力されません。

なお、GC 要因番号については、「-XX:[+|-]HitachiVerboseGCPrintCause (GC 要因内容出力オプション)」を参照してください。

region_size

1 リージョンのサイズです。

単位はキロバイトです。

target_time

GC によるアプリケーション停止時間の目標時間です。

単位は秒です。

predicted_time

JavaVM が予測した GC によるアプリケーション停止時間です。

単位は秒です。

なお、GC 種別が” Full GC” , ” CM Remark” , “CM Cleanup” のときは予測をしないため、0 が出力されます。

target_size

Mixed GC で GC 対象となった Tenured 領域のサイズです。

単位はキロバイト。

なお、GC 種別が” Mixed GC” 以外のときは、0 が出力されます。

reclaimable_info

MixedGC で回収される Tenured 領域の予測回収サイズ情報。次の形式で出力されます。

<予測回収サイズ> (<予測回収率>)

なお、予測回収サイズ情報は CM 終了直後の Young GC または Mixed GC だけ出力されます。それ以外の場合、予測を行わないため、0 が出力されます。

user_cpu

GC スレッドがユーザーモードで費やした CPU 時間。単位は秒です。

CPU 時間の取得に失敗した場合は"unknown"と表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

system_cpu

GC スレッドがカーネルモードで費やした CPU 時間。単位は秒です。

CPU 時間の取得に失敗した場合は"unknown"と表示されます。

-XX:-HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

jvm_alloc_size

JavaVM 内部で管理している領域のうち、現在使用中の領域のサイズ (mmap_total_size と malloc_total_size の合計サイズのうち、現在使用中の領域のサイズ)。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

mmap_total_size

JavaVM 内部で管理している領域のうち、mmap (Windows の場合は VirtualAlloc) で割り当てた C ヒープの総サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

malloc_total_size

JavaVM 内部で管理している領域のうち、malloc で割り当てた C ヒープ総サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintJVMInternalMemory オプションが指定された場合、出力されません。

thread_count

Java スレッドの数。

-XX:-HitachiVerboseGCPrintThreadCount オプションが指定された場合、出力されません。

doe_alloc_size

java.io.File.deleteOnExit()を呼び出して確保した累積のヒープサイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

called_count

java.io.File.deleteOnExit()の呼び出し回数。

-XX:-HitachiVerboseGCPrintDeleteOnExit が指定された場合、出力されません。

cc_used_size

GC 発生時のコードキャッシュ領域の使用サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_info

保守情報。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

buffer_used_size

ダイレクトバッファの使用サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_total_size

ダイレクトバッファの合計容量。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_max_size

ダイレクトバッファの最大値。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_count

ダイレクトバッファの数。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

-XX:+HitachiVerboseGCPrintTenuringDistribution オプションが指定されている場合の出力内容を次に説明します。

id, date, size, value, max_value, total_age1, total_age2, total_agen

出力内容については、「[-XX:\[+|-\]HitachiVerboseGCPrintTenuringDistribution](#) (Survivor 領域の年齢分布出力オプション)」を参照してください。

- VCM ログ

```
id, date, cm_event, user_cpu, sys_spu
```

id

VCM (JavaVM ログファイル識別子)。

date

CM 開始日時。

-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

cm_event

CM 種別。"Concurrent Root Region Scan Start", "Concurrent Root Region Scan End", "Concurrent Mark Start", "Concurrent Mark End", "Concurrent Mark Stop", "Concurrent Cleanup Start", "Concurrent Cleanup End"のどれかが出力されます。

user_cpu

全 CM スレッドがユーザーモードで費やした CPU 時間。単位は秒です。

-XX:-HitachiVerboseGCCpuTime 指定時は出力されません。

CPU 時間取得に失敗した場合, "unknown"と表示されます。

CM の状態が Start の場合, 0 が出力されます。

sys_cpu

全 CM スレッドがカーネルモードで費やした CPU 時間。単位は秒です。

-XX:-HitachiVerboseGCCpuTime 指定時は出力されません。

CPU 時間取得に失敗した場合, "unknown"と表示されます。

CM の状態が Start の場合, 0 が出力されます。

出力形式 (ZGC を使用している場合)

- VZG ログ

```
id, date, h_used_before, h_size_before, h_used_after, h_size_after, gc_time, f_used_before, f_used_after, m_used_before, m_capacity_before, m_commit_before, m_used_after, m_capacity_after, m_commit_after, c_used_before, c_capacity_before, c_commit_before, c_used_after, c_capacity_after, c_commit_after, s_count, s_size, s_empty, s_relocated, s_inPlace, m_count, m_size, m_empty, m_relocated, m_inPlace, l_count, l_size, l_empty, l_relocated, l_inPlace, cause_info, user_cpu_stw1, user_cpu_stw2, user_cpu_stw3, sys_cpu_stw1, sys_cpu_stw2, sys_cpu_stw3, jvm_alloc_size, mmap_total_size, malloc_total_size, thread_count, doe_alloc_size, called_count, cc_used_size, cc_max_size, cc_info, buffer_used_size, buffer_total_size, buffer_max_size, buffer_count
```

id

VZG (日立 JavaVM ログファイル識別子)。

date

ZGC サイクルが開始された日時。

-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

h_used_before

GC 前の Java ヒープ領域の使用サイズ。単位はキロバイトです。

h_size_before

GC 前の Java ヒープ領域の領域サイズ。単位はキロバイトです。

h_used_after

GC 後の Java ヒープ領域の使用サイズ。単位はキロバイトです。

h_size_after

GC 後の Java ヒープ領域の領域サイズ。単位はキロバイトです。

gc_time

GC によるアプリケーション停止時間。STW1, STW2, STW3 の総数です。単位はミリ秒 (msec) です。

f_used_before

GC 前の Free 領域のサイズ。単位はキロバイトです。

f_used_after

GC 後の Free 領域のサイズ。単位はキロバイトです。

m_used_before

GC 前の Metaspace 領域の使用サイズ。単位はキロバイトです。

m_capacity_before

GC 前の Metaspace 領域の capacity サイズ。単位はキロバイトです。

m_commit_before

GC 前の Metaspace 領域の commit サイズ。単位はキロバイトです。

m_used_after

GC 後の Metaspace 領域の使用サイズ。単位はキロバイトです。

m_capacity_after

GC 後の Metaspace 領域の capacity サイズ。単位はキロバイトです。

m_commit_after

GC 後の Metaspace 領域の commit サイズ。単位はキロバイトです。

c_used_before

GC 前の class space 領域の使用サイズ。単位はキロバイトです。

圧縮オブジェクトポインタ機能が無効の場合は 0 が出力されます。

c_capacity_before

GC 前の class space 領域の capacity サイズ。単位はキロバイトです。

圧縮オブジェクトポインタ機能が無効の場合は 0 が出力されます。

c_commit_before

GC 前の class space 領域の commit サイズ。単位はキロバイトです。
圧縮オブジェクトポインタ機能が無効の場合は 0 が出力されます。

c_used_after

GC 後の class space 領域の使用サイズ。単位はキロバイトです。
圧縮オブジェクトポインタ機能が無効の場合は 0 が出力されます。

c_capacity_after

GC 後の class space 領域の capacity サイズ。単位はキロバイトです。
圧縮オブジェクトポインタ機能が無効の場合は 0 が出力されます。

c_commit_after

GC 後の class space 領域の commit サイズ。単位はキロバイトです。
圧縮オブジェクトポインタ機能が無効の場合は 0 が出力されます。

s_count

Small ページの枚数。

s_size

Small ページの総サイズ。単位はメガバイトです。

s_empty

Small ページの空きサイズ。単位はメガバイトです。

s_relocated

Small ページに再配置されたオブジェクトサイズ。単位はメガバイトです。

s_inPlace

空きメモリがない場合に InPlace 方式でオブジェクトが再配置された Small ページ枚数。

m_count

Medium ページの枚数。

m_size

Medium ページの総サイズ。単位はメガバイトです。

m_empty

Medium ページの空きサイズ。単位はメガバイトです。

m_relocated

Medium ページに再配置されたオブジェクトサイズ。単位はメガバイトです。

m_inPlace

空きメモリがない場合に InPlace 方式でオブジェクトが再配置された Medium ページ枚数。

l_count

Large ページの枚数。

L_size

Large ページの総サイズ。単位はメガバイトです。

L_empty

Large ページの空きサイズ。単位はメガバイトです。

L_relocated

Large ページに再配置されたオブジェクトサイズ。単位はメガバイトです。

L_inPlace

空きメモリがない場合に InPlace 方式でオブジェクトが再配置された Large ページ枚数。

cause_info

GC 要因内容。

-XX:-HitachiVerboseGCPrintCause オプションが指定された場合、出力されません。

user_cpu_stw1

STW1 で全 GC スレッドがユーザモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。-XX:-

HitachiVerboseGCPrintCause オプションが指定された場合、出力されません。

user_cpu_stw2

STW2 で全 GC スレッドがユーザモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:- HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

user_cpu_stw3

STW3 で全 GC スレッドがユーザモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:- HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

sys_cpu_stw1

STW1 で全 GC スレッドがカーネルモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:- HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

sys_cpu_stw2

STW2 で全 GC スレッドがカーネルモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:- HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

sys_cpu_stw3

STW3 で全 GC スレッドがカーネルモードで費やした CPU 時間の合計。単位は秒 (sec) です。

CPU 時間の取得に失敗した場合は「unknown」が表示されます。

-XX:- HitachiVerboseGCCpuTime オプションが指定された場合、出力されません。

thread_count

Java スレッドの数。

-XX:-HitachiVerboseGCPrintThreadCount オプションが指定された場合、出力されません。

doe_alloc_size

java.io.File.deleteOnExit() を呼び出して確保した累積のヒープサイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

called_count

java.io.File.deleteOnExit() の呼び出し回数。

-XX:-HitachiVerboseGCPrintDeleteOnExit オプションが指定された場合、出力されません。

cc_used_size

GC 発生時のコードキャッシュ領域の使用サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

cc_info

保守情報。単位はキロバイトです。

-XX:-PrintCodeCacheInfo オプションが指定された場合、出力されません。

buffer_used_size

ダイレクトバッファの使用サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_total_size

ダイレクトバッファの合計容量。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_max_size

ダイレクトバッファの最大値。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_count

ダイレクトバッファの数。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

- VZP ログ

id, date, phase, phase_time

出力内容を次に説明します。

id

VZP (日立 JavaVM ログファイル識別子)。

date

各フェーズが開始された日時。

-XX:-HitachiVerboseGCPrintDate オプションが指定された場合、出力されません。

phase

STW またはフェーズの種別。

phase_time

フェーズにかかった時間。単位はミリ秒 (msec) です。

出力例

SerialGC を使用している場合

- -XX:HitachiVerboseGCIntervalTime オプションが指定されている場合

```
VGC, Tue Nov 22 13:04:01.234 2022, 1, 0, 0, Full GC, 3254, 3254, 8064, 0.0149684, 0, 0, 2304, 0, 0, 256,
3254, 3254, 5504, 8613, 8896, 8896, 8613, 8896, 8896, 1131, 1280, 1280, 1131, 1280, 1280, 1, 0.0156250, 0.
0000000, 7252, 10144, 0, 15, 0, 0, 894, 49152, 2496, 17, 8, 8, 83200, 1
```

注意事項

下線が引かれている部分は、JDK17 以降の場合にだけ出力されます。

- -XX:+HitachiVerboseGCPrintTenuringDistribution オプションが指定されている場合

```
PTD, Wed May 28 11:45:23 2008, 5467547, 30, 31, 1357527, 1539661
```

G1GC を使用している場合

- VG1 ログ

```
VG1, Tue Nov 22 13:13:22.002 2022, Full GC, 3254, 4096, 14336, 3254, 4096, 14336, 0.0059093, -, 0, 40
96, 0, 4096, 0, 0, 4096, 4096, 0, 0, 10240, 10240, 8575, 8832, 8832, 8575, 8832, 8832, 1131, 1280, 1280, 1131
, 1280, 1280, 1, 1024, 0.2000000, 0.0000000, 0, 0, 0.00, 0.0000000, 0.0000000, 31450, 37280, 0, 27, 0, 0, 9
13, 49152, 2496, 20, 8, 8, 86016, 1
```

注意事項

下線が引かれている部分は、JDK17 以降の場合にだけ出力されます。

- VCM ログ

```
VCM, Fri Jul 26 21:35:50 2013, Concurrent Mark Start, 0.0000000, 0.0000000
VCM, Fri Jul 26 21:35:50 2013, Concurrent Mark End, 0.0124532, 0.0245698
```

ZGC を使用している場合

- VZG ログ

```
VZG, Fri Mar 31 14:27:52 2023, 14336, 14336, 6144, 16384, 0.0409, 71680, 79872, 3921, 4032, 4032, 3921, 4032, 4032, 0, 0, 0, 0, 0, 0, 4, 8, 4, 0, 0, 0, 0, 0, 0, 0, 3, 6, 4, 0, 0, 25, 0.0000000, 0.0000000, 0.0000000, 0.0000000, 0.0000000, 11056, 12320, 0, 23, 0, 0, 608, 49152, 2496
```

- VZP ログ

```
VZP, Fri Mar 31 14:27:53 2023, Pause Mark Start, 0.0095
```

-XX:HitachiVerboseGCIntervalTime (拡張 verbosegc 情報出力間隔指定オプション)

形式

-XX:HitachiVerboseGCIntervalTime=<整数値>

説明

拡張 verbosegc 情報を出力する時間（秒）の間隔を指定します。

オプションを指定した場合

拡張 verbosegc 情報を GC ごとに出力するのではなく、指定された時間を超えた次の GC を出力します。このとき、前回の出力から今回の出力までの間に発生した、次に示す GC の回数も出力します。これらは必ず出力されます。

表 14-5 回数を出力する GC の一覧

文字列	意味
Full	FullGC をスキップした回数
Copy	CopyGC をスキップした回数

オプションを指定しない場合

0 秒がデフォルトで設定され、GC 発生ごとに拡張 verbosegc 情報を出力します。

デフォルト値

- -XX:HitachiVerboseGCIntervalTime=0

前提オプション

- -XX:+HitachiVerboseGC

引数

<整数値>

0～2147483647 の範囲の整数値（単位：秒）を指定します。範囲外の値が指定された場合は 0 が設定されます。負の値を指定した場合はエラーとなります。

-XX:[+|-]HitachiVerboseGCPrintCause（GC 要因内容出力オプション）

形式

-XX:+HitachiVerboseGCPrintCause

GC の要因内容を，拡張 verbosegc 情報の行末に出力します。

-XX:-HitachiVerboseGCPrintCause

拡張 verbosegc 情報を通常形式で出力します。

説明

GC の要因内容を出力するかどうかを指定します。

デフォルト値

- -XX:+HitachiVerboseGCPrintCause

前提オプション

- -XX:+HitachiVerboseGC

-XX:+HitachiCommaVerboseGC オプションが指定されている場合は，次に示す要因番号が出力されます。

表 14-6 GC 要因一覧

要因番号	要因内容	説明	SerialGC 使用時の要因	G1GC 使用時の要因	ZGC 使用時の要因
0	ObjAllocFail	G1GC 未使用時，オブジェクトの割り付け失敗によって GC が発生しました。G1GC 使用時，Evacuation を実施してもオブジェクト割り付け領域が確保できなかったので GC が発生しました。	○	○	—
1	System.gc	java.lang.System.gc メソッド呼び出しによって GC が発生しました。	○	○	○
3	DelayedGC	JNI や JVMTI によって保留されていた GC が起動されました。	○	○	○

要因番号	要因内容	説明	SerialGC 使用時の要因	G1GC 使用時の要因	ZGC 使用時の要因
4	JavaGC Command	JavaGC コマンドによって GC が発生しました。	○	○	○
6	JHeapProf Command	jheapprof コマンドによって GC が発生しました。	○	—	—
8	EMReclaimFail ^{※1}	Explicit メモリの明示解放による Java ヒープへのオブジェクト移動が発生して Java ヒープがあふれました。	○	—	—
9	EMMigrateFail ^{※1}	Explicit メモリの自動解放による Java ヒープへのオブジェクト移動が発生して Java ヒープがあふれました。	○	—	—
10	JVMTIForceGC	JVMTI 関数 ForceGarbageCollection()によって GC が発生しました。	○	○	○
11	PromotionFail	CopyGC の昇格失敗によって GC が発生しました。	○	—	—
12	EMJavaGC Command ^{※1}	javagc コマンドによる Explicit メモリブロックの解放によって GC が発生しました。	○	○	—
13	EHeapProf Command ^{※1}	eheapprof コマンドによって GC が発生しました。	○	○	—
14	G1HumAllocFail	Humongous 領域へのオブジェクト割り当て失敗によって GC が発生しました。	—	○	—
15	G1EvacuationPause	オブジェクト割り当て失敗によって GC が発生しました。	—	○	—
16	Concurrent Marking	アプリケーションを停止して実行する CM 処理が発生しました。	—	○	—
17	EvacuationFail	Evacuation 失敗によって GC が発生しました。	—	○	—
18	MetaspaceAllocFail	Metaspace の領域確保失敗によって GC が発生しました。	○	○	○
19	LastMetaspaceGC	Metaspace の OutOfMemory を出す前に行う最後の GC が発生しました。	○	○	○
20	JcmdGCRun ^{※2}	jcmd コマンド (GC.run オプション) によって GC が発生しました。	○	○	○
21	G1CompactionPause ^{※2}	Free 領域に空き容量がなく、かつ Java ヒープ領域を拡張できない場合に、FullGC によって GC が発生しました。	—	○	—
22	G1PeriodicCollection ^{※2}	定期的な GC が実行されました。	—	○	—

要因番号	要因内容	説明	SerialGC 使用時の要因	G1GC 使用時の要因	ZGC 使用時の要因
23	G1PreventiveCollection ^{※3}	Free 領域のページ数と、予想されるオブジェクト生存率に基づいて、前もって GC が発生しました。	—	○	—
24	ZTimer ^{※2}	前回の GC から、 - XX:ZCollectionInterval=<seconds> オプションで指定した時間が経過したため、GC が発生しました。	—	—	○
25	ZWarmup ^{※2}	ZWarmup 以外の要因の GC が発生していないときに、Java ヒープ領域の使用率が 10%、20% または 30% を超えたため、GC が発生しました。	—	—	○
26	ZAllocationRate ^{※2}	最大割り当て率と空きメモリ量から、OOM が発生するおそれがあると判断されたため、GC が発生しました。	—	—	○
27	ZAllocationStall ^{※2}	アプリケーションの実行に必要な Java ヒープ領域の空き容量が不足しているため、GC が発生しました。	—	—	○
28	ZProactive ^{※2}	GC でスループットが低下してもシステムの動作に大きな影響がない場合に、前もって GC が発生しました。	—	—	○
29	ZHighUsage ^{※2}	Java ヒープ領域の 95% 以上が使用されたため、GC が発生しました。	—	—	○

(凡例)

- ：該当する。
- ：該当しない。

注※1

JDK11 以前の場合に出力されます。

注※2

JDK17 以降の場合に出力されます。

注※3

JDK17 の場合に出力されます。

出力例

```
[VGC]<Wed Mar 17 00:42:30 2004>(Skip Full:0, Copy:0)[Full GC 149K->149K(1984K), 0.0786038 sec
s][DefNew::Eden: 264K->0K(512K)][DefNew::Survivor: 0K->63K(64K)][Tenured: 85K->149K(1408K)][
Metaspace: 3634K(4492K, 4492K)->3634K(4492K, 4492K)][class space: 356K(388K, 388K)->356K(388
K, 388K)][cause:System.gc]
```


-XX:[+|-]HitachiVerboseGCPrintDate (拡張 verbosegc 情報日付出力オプション)

形式

-XX:+HitachiVerboseGCPrintDate

拡張 verbosegc 情報の各出力行の先頭に GC 開始日時を出力します。

-XX:-HitachiVerboseGCPrintDate

拡張 verbosegc 情報の各出力行の先頭に GC 開始日時を出力しません。

説明

拡張 verbosegc 情報に GC 開始日時を出力するかどうかを指定します。

デフォルト値

- -XX:+HitachiVerboseGCPrintDate

前提オプション

- -XX:+HitachiVerboseGC

-XX:[+|-]HitachiVerboseGCCpuTime (拡張 verbosegc 情報 CPU 利用時間出力オプション)

形式

-XX:+HitachiVerboseGCCpuTime

拡張 verbosegc 情報に、GC の開始から終了までで、GC 実行スレッドのユーザーモードおよびカーネルモードに費やされたプロセッサ時間を出力します。

-XX:-HitachiVerboseGCCpuTime

拡張 verbosegc 情報に、GC の開始から終了までで、GC 実行スレッドのユーザーモードおよびカーネルモードに費やされたプロセッサ時間を出力しません。

説明

拡張 verbosegc 情報に CPU 利用時間を出力するかどうかを指定します。

デフォルト値

- -XX:+HitachiVerboseGCCpuTime

前提オプション

- -XX:+HitachiVerboseGC

出力例

```
[VGC]<Thu Oct 02 10:38:53.658 2014>(Skip Full:1, Copy:0)[Full GC 770K->682K(8064K), 0.0050003
secs][DefNew::Eden: 88K->0K(2304K)][DefNew::Survivor: 0K->0K(256K)][Tenured: 681K->682K(550
4K)] [Metaspace: 3634K(4492K, 4492K)->3634K(4492K, 4492K)][class space: 356K(388K, 388K)->35
6K(388K, 388K)] [cause:System.gc][User: 0.0000000 secs][Sys: 0.0000000 secs]
```

-XX:[+|-]HitachiVerboseGCPrintTenuringDistribution (Survivor 領域の 年齢分布出力オプション)

形式

-XX:+HitachiVerboseGCPrintTenuringDistribution

Survivor 領域の年齢分布を JavaVM ログファイルへ出力します。

-XX:-HitachiVerboseGCPrintTenuringDistribution

Survivor 領域の年齢分布を JavaVM ログファイルへ出力しません。

説明

Survivor 領域の年齢分布を JavaVM ログファイルへ出力するかどうかを指定します。

デフォルト値

- -XX:-HitachiVerboseGCPrintTenuringDistribution

前提オプション

- -XX:+HitachiVerboseGC

関連オプション

- -XX:+HitachiVerboseGCPrintDate
- -XX:+HitachiCommaVerboseGC

出力形式

```
[id]<date>[Desired survivor:size bytes][New threshold:value][MaxTenuringThreshold: max_value]
[age1:total_age1][age2:total_age2]...[agen:total_agen]
```

出力内容を次に説明します。

id

PTD (JavaVM ログファイル識別子)。

date

GC 開始日時。

size

GC 後の Survivor 領域内オブジェクト目標サイズ。

value

次の CopyGC で Tenured 領域に昇格する Java オブジェクトの年齢のしきい値。

この値は、-XX:MaxTenuringThreshold=<value>オプションと、Survivor 領域のメモリサイズ、および、-XX:TargetSurvivorRatio=<value>オプションに設定した値を基に、CopyGC ごとに動的に設定されます。

value 値以上の年齢の Java オブジェクトが、次の CopyGC で Tenured 領域に昇格します。

max_value

CopyGC で Tenured 領域に昇格する Java オブジェクトの年齢のしきい値 (value 値) の最大値 (MaxTenuringThreshold オプションの指定値)。

value 値は、CopyGC ごとに動的に設定されますが、max_value 値を超えることはありません。

また、年齢が max_value 値以上の Java オブジェクトは、次の CopyGC で、必ず、Tenured 領域に昇格します。

total_age1

1 歳のオブジェクトのバイト数の合計。

total_age2

1 歳から 2 歳までのオブジェクトのバイト数の合計。

total_agen

1 歳から n 歳までのオブジェクトのバイト数の合計。

n が max_value に近ければ寿命の長いオブジェクトが存在するということになります。

出力例

```
[PTD]<Wed Jan 28 17:47:10 2009>[Desired survivor:32768 bytes][New threshold:30][MaxTenuringThreshhold:30][age1:6872][age2:9632][age3:25632]
```

-XX:[+|-]HitachiVerboseGCPrintJVMInternalMemory (C ヒープ情報出力オプション)

形式

-XX:+HitachiVerboseGCPrintJVMInternalMemory

JavaVM 内部で管理しているヒープ情報を JavaVM ログファイルへ出力します。

-XX:-HitachiVerboseGCPrintJVMInternalMemory

JavaVM 内部で管理しているヒープ情報を JavaVM ログファイルへ出力しません。

説明

JavaVM 内部で管理しているヒープ情報を JavaVM ログファイルに出力するかどうかを指定します。

C ヒープ領域のうち、次の 2 種類の方法で取得したヒープ領域は、JavaVM 内部で管理する方式で管理されています。

- mmap で取得した C ヒープ領域
- malloc で取得した C ヒープ領域

-XX:+HitachiVerboseGCPrintJVMInternalMemory オプションを有効にした場合、mmap で取得した C ヒープの総サイズ (mmap_total_size) と、malloc で取得した C ヒープの総サイズ (malloc_total_size) を出力できます。また、これらの割り当て済みの領域のうち、使用中の領域のサイズの合計値 (jvm_alloc_size) も出力できます。

デフォルト値

- -XX:+HitachiVerboseGCPrintJVMInternalMemory

前提オプション

- -XX:+HitachiVerboseGC

出力形式

```
[id] <date> (Skip Full:full_count, Copy:copy_count, Inc:inc_count) [gc kind gc_info, gc_time secs][Eden: eden_info][Survivor: survivor_info][Tenured: tenured_info] [Metaspace: metaspace_info][class space: class_space_info][cause:cause_info] [User: user_cpu secs][Sys: system_cpu secs][IM: jvm_alloc_size, mmap_total_size, malloc_total_size][TC: thread_count][DOE: doe_alloc_size, called_count]
```

出力内容を次に示します。なお、ここでは、このオプションによって出力される項目について説明します。ここで説明している以外の項目については、「[-XX:\[+|-\]HitachiVerboseGC \(拡張 verbosegc 情報出力オプション\)](#)」の出力形式の説明を参照してください。

jvm_alloc_size

JavaVM 内部で管理している領域のうち、現在使用中の領域のサイズ (mmap_total_size と malloc_total_size の合計サイズのうち、現在使用中の領域のサイズ)。

mmap_total_size

JavaVM 内部で管理している領域のうち、mmap (Windows の場合は VirtualAlloc) で割り当てた C ヒープの総サイズ。

malloc_total_size

JavaVM 内部で管理している領域のうち、malloc で割り当てた C ヒープの総サイズ。

出力例

```
[VGC]<Wed Jan 27 13:03:36 2010>(Skip Full:0,Copy:0)[GC 489K->152K(3520K), 0.0156080 secs][DefNew::Eden: 489K->0K(2048K)][DefNew::Survivor: 0K->63K(64K)][Tenured: 0K->88K(1408K)][Metaspace: 3634K(4492K, 4492K)->3634K(4492K, 4492K)][class space: 356K(388K, 388K)->356K(388K, 388K)][cause:ObjAllocFail][IM: 729K, 928K, 0K][TC: 509][DOE: 16K, 170]
```

-XX:[+|-]HitachiVerboseGCPrintThreadCount (スレッド数の出力オプション)

形式

-XX:+HitachiVerboseGCPrintThreadCount

Java スレッドの数を出力します。

-XX:-HitachiVerboseGCPrintThreadCount

Java スレッドの数を出力しません。

説明

Java スレッドの数を監視するために、Java スレッドの数を JavaVM ログファイルに出力するかどうかを指定します。

スレッドは、スタックに使用するためのメモリを C ヒープとして個々で確保します。このため、スレッド数が多くなると、C ヒープの確保量もスレッド数に比例して多くなります。-

XX:+HitachiVerboseGCPrintThreadCount オプションを指定することで、Java スレッドの数を監視できるようになり、C ヒープの確保量の把握ができるようになります。

デフォルト値

- -XX:+HitachiVerboseGCPrintThreadCount

前提オプション

- -XX:+HitachiVerboseGC

出力形式

```
[id] <date> (Skip Full:full_count, Copy:copy_count, Inc:inc_count) [gc_kind gc_info, gc_time secs][Eden: eden_info][Survivor: survivor_info][Tenured: tenured_info][Metaspace: metaspace_info][class space: class_space_info][cause:cause_info] [User: user_cpu secs][Sys: system_cpu secs][IM: jvm_alloc_size, mmap_total_size, malloc_total_size][TC: thread_count][DOE: doe_alloc_size, called_count]
```

出力内容を次に示します。なお、ここでは、このオプションによって出力される項目について説明します。ここで説明している以外の項目については、「[-XX:\[+|-\]HitachiVerboseGC \(拡張 verbosegc 情報出力オプション\)](#)」の出力形式の説明を参照してください。

thread_count

Java スレッドの数。

出力例

```
[VGC]<Wed Jan 27 13:03:36 2010>(Skip Full:0, Copy:0)[GC 489K->152K(3520K), 0.0156080 secs][DefNew::Eden: 489K->0K(2048K)][DefNew::Survivor: 0K->63K(64K)][Tenured: 0K->88K(1408K)][Metaspace: 3634K(4492K, 4492K)->3634K(4492K, 4492K)][class space: 356K(388K, 388K)->356K(388K, 388K)][cause:ObjAllocFail][IM: 729K, 928K, 0K][TC: 509]
```

-XX:[+|-]HitachiVerboseGCPrintDeleteOnExit (java.io.File.deleteOnExit()が使用するヒープサイズの出力オプション)

形式

-XX:+HitachiVerboseGCPrintDeleteOnExit

java.io.File.deleteOnExit()を呼び出して確保した累積のヒープサイズと、メソッドの呼び出し回数を出力します。

-XX:-HitachiVerboseGCPrintDeleteOnExit

java.io.File.deleteOnExit()を呼び出して確保した累積のヒープサイズと、メソッドの呼び出し回数を出力しません。

説明

java.io.File.deleteOnExit()を呼び出したことによってJavaVMが確保した累積のヒープサイズとメソッドの呼び出し回数を、JavaVM ログファイルに出力するかどうかを指定します。

java.io.File.deleteOnExit()は、呼び出すたびに指定されたファイルのパス情報をヒープに確保しますが、確保した領域はプロセスの終了まで解放しないため、メモリの圧迫につながるおそれがあります。

XX:+HitachiVerboseGCPrintDeleteOnExit オプションを指定すると、JavaVMがjava.io.File.deleteOnExit()を呼び出して確保したヒープサイズをログに出力して監視できるようになります。また、java.io.File.deleteOnExit()の呼び出し状況を把握するための補助的な情報として、メソッドの呼び出し回数も同時に出力できます。

出力した情報は、障害発生時、java.io.File.deleteOnExit()の呼び出しによって確保されたヒープサイズを把握してメモリ不足の原因を調査するために役立てられます。また、運用開始前の開発やテスト段階で、java.io.File.deleteOnExit()の呼び出しによって確保したヒープサイズの増加の推移を確認して、運用時にメモリを圧迫する予兆がないかを事前確認するためにも利用できます。

なお、エラーが発生した場合は、エラーメッセージが出力されます。

デフォルト値

- -XX:+HitachiVerboseGCPrintDeleteOnExit

前提オプション

- -XX:+HitachiVerboseGC

出力形式

JavaVM ログファイルの出力形式を次に示します。

```
[id] <date> (Skip Full:full_count, Copy:copy_count, Inc:inc_count) [gc_kind gc_info, gc_time
secs][Eden: eden_info][Survivor: survivor_info][Tenured: tenured_info] [Metaspace: metaspace_info][class space: class_space_info][cause:cause_info] [User: user_cpu secs][Sys: system_cpu secs][IM: jvm_alloc_size, mmap_total_size, malloc_total_size][TC: thread_count][DOE: doe_alloc_size, called_count]
```

出力内容を次に示します。なお、ここでは、このオプションによって出力される項目について説明します。ここで説明している以外の項目については、「-XX:[+|-]HitachiVerboseGC (拡張 verbosegc 情報出力オプション)」の出力形式の説明を参照してください。

doe_alloc_size

java.io.File.deleteOnExit()を呼び出して確保した累積のヒープサイズ。単位はキロバイトです。

called_count

java.io.File.deleteOnExit()の呼び出し回数。

エラーが発生した場合のエラーメッセージの出力形式を次に示します。

```
[DOE]<date>Error occurred during processing of java.io.File.deleteOnExit's heap size output
function. (<保守情報>)
[DOE]java.io.File.deleteOnExit's heap size output function stopped.
```

エラーメッセージの出力内容を次に示します。

DOE

java.io.File.deleteOnExit()のヒープサイズ出力機能でエラーが発生したことを示す識別子。

date

エラーが発生した日時。

出力例

- JavaVM ログファイルの出力例を次に示します。

```
[VGC]<Wed Jan 27 13:03:36 2010>(Skip Full:0, Copy:0)[GC 489K->152K(3520K), 0.0156080 secs]
[DefNew::Eden: 489K->0K(2048K)][DefNew::Survivor: 0K->63K(64K)][Tenured: 0K->88K(1408K)][
Metaspace: 3634K(4492K, 4492K)->3634K(4492K, 4492K)][class space: 356K(388K, 388K)->356K(
388K, 388K)][cause:ObjAllocFail][IM: 729K, 928K, 0K] [TC: 509][DOE: 16K, 170]
```

- エラーメッセージの出力例を次に示します。


```
[DOE]<Wed Jan 27 13:03:36 2010> Error occurred during processing of java.io.File.deleteOnExit's heap size output function. (FindClass:java.lang.String)
[DOE]java.io.File.deleteOnExit's heap size output function stopped.
```

注意事項

- 次の場合は、`java.io.File.deleteOnExit()` を呼び出しても累積のヒープサイズ、メソッドの呼び出し回数がカウントされません。
 - `java.io.File.deleteOnExit()` を呼び出した場合に `SecurityException` 例外が発生したとき（この例外はセキュリティマネージャの `SecurityManager.checkDelete()` がファイルへの削除アクセスを許可しない場合に発生します。この場合、メソッドの入り口で例外が挙がり、ヒープは確保されません）。
 - アプリケーションサーバのバッチアプリケーション実行基盤で作成されたアプリケーションから呼び出した場合。
 - 同一のパス名文字列で作成した `File` インスタンスを使用して `java.io.File.deleteOnExit()` を呼び出したとき。
- この機能が出力するヒープサイズを確認する際には、次の点に注意してください。
 - `java.io.File.deleteOnExit()` が確保するヒープの種類は Java ヒープです。
 - ヒープサイズはキロバイト単位で出力され、1 キロバイト未満は切り捨てられます。
`java.io.File.deleteOnExit()` の 1 回の呼び出しで確保するヒープサイズは、ファイルパスの長さに応じて数十バイトから 100 バイト程度であるため、呼び出しごとにヒープサイズの出力結果が増加しないことがあります。この場合、メソッドの呼び出し回数からメソッドの実行を確認できます。

-XX:[+|-]PrintCodeCacheInfo（コードキャッシュ領域情報出力オプション）

形式

-XX:+PrintCodeCacheInfo

コードキャッシュ領域の使用量を出力します。

また、コードキャッシュ領域の使用量がしきい値に達したことを知らせるメッセージを出力します。

-XX:-PrintCodeCacheInfo

コードキャッシュ領域の使用量を出力しません。

また、コードキャッシュ領域の使用量がしきい値に達したことを知らせるメッセージを出力しません。

説明

コードキャッシュ領域の使用量を出力するかどうか、また、使用量がしきい値に達したことを知らせるメッセージを JavaVM ログファイルに出力するかどうかを指定します。

コードキャッシュ領域については、マニュアル「アプリケーションサーバシステム設計ガイド」の「7.2.6 SerialGC 使用時の JavaVM で使用するメモリ空間の構成と JavaVM オプション」を参照してください。

このオプションを有効にすると、GC 発生時、コードキャッシュ領域の使用量が拡張 verbosegc 情報に出力されます。また、コードキャッシュ領域の使用量がしきい値に達したとき、メッセージが出力されます。

しきい値は「コードキャッシュ領域の最大サイズ×-XX:CodeCacheInfoPrintRatio オプションの値÷100」です。

-XX:-HitachiVerboseGC オプションを指定している場合でも、javagc コマンドの-v オプションや-s オプションによって出力する拡張 verbosegc 情報には、コードキャッシュ領域の使用量が出力されます。

デフォルト値

- -XX:+PrintCodeCacheInfo

前提オプション

- -XX:+HitachiVerboseGC

関連オプション

- -XX:+HitachiCommaVerboseGC
- -XX:CodeCacheInfoPrintRatio

出力形式

コードキャッシュ領域の使用量の出力形式を次に示します。

```
[id] <date> (Skip Full:full_count, Copy:copy_count, Inc:inc_count) [gc_kind gc_info, gc_time
secs][Eden: eden_info][Survivor: survivor_info][Tenured: tenured_info] [Metaspace: metaspac
e_info][class space: class_space_info][cause:cause_info] [User: user_cpu secs][Sys: system_c
pu secs][IM: jvm_alloc_size, mmap_total_size, malloc_total_size][TC: thread_count][DOE: doe_
alloc_size, called_count][CCI: cc_used_sizeK, cc_max_sizeK, cc_infoK]
```

出力内容を次に示します。なお、ここでは、このオプションによって出力される項目について説明します。ここで説明している以外の項目については、「[-XX:\[+|-\]HitachiVerboseGC \(拡張 verbosegc 情報出力オプション\)](#)」の出力形式の説明を参照してください。

cc_used_size

GC 発生時のコードキャッシュ領域の使用サイズ。単位はキロバイトです。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

cc_info

保守情報。単位はキロバイトです。

コードキャッシュ領域の使用量がしきい値に達したことを知らせるメッセージの出力形式を次に示します。

```
[cc_id]<cc_date>CodeCache usage has exceeded the threshold.[cc_used_sizeK, cc_max_sizeK, cc_
infoK]
```

メッセージの出力内容を次に示します。

cc_id

CCI (JavaVM ログファイル識別子)。

cc_date

JIT コンパイルを実行した日時。

cc_used_size

JIT コンパイル後のコードキャッシュ領域の使用サイズ。単位はキロバイトです。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

cc_info

保守情報。単位はキロバイトです。

出力例

- コードキャッシュ領域の使用量の出力例を次に示します。

```
[VGC]<Wed Mar 17 00:42:30 2004>(Skip Full:0, Copy:0)[Full GC 149K->149K(1984K), 0.0786038
secs][DefNew::Eden: 264K->0K(512K)][DefNew::Survivor: 0K->63K(64K)][Tenured: 85K->149K(14
08K)][Metaspace: 3634K(4492K, 4492K)->3634K(4492K, 4492K)][class space: 356K(388K, 388K)-
>356K(388K, 388K)][cause:System.gc][User: 0.0156250 secs][Sys: 0.0312500 secs][IM: 729K,
928K, 0K][TC: 509][DOE: 16K, 170][CCI: 2301K, 49152K, 2304K]
```

- コードキャッシュ領域の使用量がしきい値に達したことを知らせるメッセージの出力例を次に示します。

```
[CCI]<Wed Dec 26 14:27:53 2012>CodeCache usage has exceeded the threshold.[39358K, 49152K
, 39360K]
```

注意事項

- コードキャッシュ領域の使用量がしきい値に達した状態で推移している場合、Java メソッドが JIT コンパイルされてもメッセージは出力されません。
一方、コードキャッシュ領域の使用量がしきい値未満まで下がったあと、Java メソッドが JIT コンパイルされたことによって再びコードキャッシュ領域の使用量がしきい値に達した場合には、メッセージが出力されます。
- システムが使用するコードキャッシュ領域は最大 2 メガバイトです。そのため、システムのコードキャッシュ領域の使用量によっては、コードキャッシュ領域を最大サイズまで使用していない場合でも、コードキャッシュ領域が枯渇することがあります。

また、コードキャッシュ領域の使用量のしきい値に大きな値を指定している場合、メッセージを出力する前にコードキャッシュ領域が枯渇する場合があります。コードキャッシュ領域が枯渇する前にメッセージを出力したいときは、「コードキャッシュ領域の最大サイズーしきい値」の値が 4 メガバイト以上になるように、「[-XX:CodeCacheInfoPrintRatio](#) (コードキャッシュ領域使用率指定オプション)」の値を指定してください。

-XX:CodeCacheInfoPrintRatio (コードキャッシュ領域使用率指定オプション)

形式

-XX:CodeCacheInfoPrintRatio=<整数値>

説明

コードキャッシュ領域の使用量がしきい値に達したことを知らせるメッセージを出力する契機となる、コードキャッシュ領域の使用率を指定します。

ここで指定した使用率を基に、次の計算式でしきい値が計算されます。

コードキャッシュ領域の最大サイズ×-XX:CodeCacheInfoPrintRatioオプションの値÷100

コードキャッシュ領域については、マニュアル「アプリケーションサーバシステム設計ガイド」の「7.2.6 SerialGC 使用時の JavaVM で使用するメモリ空間の構成と JavaVM オプション」を参照してください。

デフォルト値

- -XX:CodeCacheInfoPrintRatio=80

前提オプション

- -XX:+PrintCodeCacheInfo

引数

<整数値>

0～100 の範囲で整数値（単位：%）を指定します。範囲外の値が指定された場合は 80 が設定されます。

-XX:[+|-]PrintCodeCacheFullMessage (コードキャッシュ領域枯渇メッセージ出力オプション)

形式

-XX:+PrintCodeCacheFullMessage

Java メソッドが JIT コンパイルの対象になった場合、コードキャッシュ領域が枯渇していたときにメッセージを出力します。メッセージは一度だけ出力されます。

-XX:-PrintCodeCacheFullMessage

Java メソッドが JIT コンパイルの対象になった場合、コードキャッシュ領域が枯渇してもメッセージを出力しません。

説明

Java メソッドが JIT コンパイルの対象になった場合、コードキャッシュ領域が枯渇していたときに、メッセージを JavaVM ログファイルに出力するかどうかを指定します。

コードキャッシュ領域については、マニュアル「アプリケーションサーバシステム設計ガイド」の「7.2.6 SerialGC 使用時の JavaVM で使用するメモリ空間の構成と JavaVM オプション」を参照してください。

デフォルト値

- -XX:+PrintCodeCacheFullMessage

前提オプション

- -XX:+HitachiVerboseGC

出力形式

メッセージの出力形式を次に示します。

```
[cc_id]<cc_date>CodeCache is full. Compiler has been disabled.[cc_used_sizeK, cc_max_sizeK, cc_infoK]
```

メッセージの出力内容を次に示します。

cc_id

CCI (JavaVM ログファイル識別子)。

cc_date

Java メソッドが JIT コンパイルの対象になった日時。

cc_used_size

Java メソッドが JIT コンパイルの対象になったときのコードキャッシュ領域の使用サイズ。単位はキロバイトです。

cc_max_size

コードキャッシュ領域の最大サイズ。単位はキロバイトです。

cc_info

保守情報。単位はキロバイトです。

出力例

メッセージの出力例を次に示します。

```
[CCI]<Wed Dec 26 14:38:29 2012>CodeCache is full. Compiler has been disabled.[49151K, 49152K, 49152K]
```

注意事項

システムが使用するコードキャッシュ領域は最大 2 メガバイトです。そのため、システムのコードキャッシュ領域の使用量によっては、コードキャッシュ領域を最大サイズまで使用していない場合でも、コードキャッシュ領域が枯渇することがあります。

-XX:[+|-]HitachiOutOfMemoryCause（例外発生要因種別出力オプション）

形式

-XX:+HitachiOutOfMemoryCause

OutOfMemoryError 発生時に、例外発生要因種別を JavaVM ログファイルに出力します。-XX:+HitachiOutOfMemoryStackTrace オプションが指定されている場合は、このオプションも設定されます。

出力される要因種別を次に示します。

スレッド数の上限値設定機能(HitachiThreadLimit オプション)を指定していて、作成したスレッド数が指定した上限値を超えた結果、OutOfMemoryError をスローする場合。

表 14-7 例外発生要因種別一覧

項番	要因メッセージ	説明
1	C Heap	C ヒープ確保時の例外
2	Java Heap	Java ヒープ確保時の例外
3	Meta Space	Metaspace 領域確保時の例外
4	Compressed Class Space	Compressed Class Space 確保時の例外
5	Unknown	上記例外を特定できない場合
6	Thread Limit	スレッド数の上限値設定機能（-XX:HitachiThreadLimit オプション）を指定した場合に、作成したスレッド数が指定した上限値を超えたときの例外

-XX:-HitachiOutOfMemoryCause

OutOfMemoryError 発生時に、例外発生要因種別を JavaVM ログファイルに出力しません。

説明

OutOfMemoryError 発生時の発生要因種別を出力するかどうかを指定します。

デフォルト値

- -XX:-HitachiOutOfMemoryCause

出力例

```
[OOM][Thread: 0x00062fd0]<Tue Dec 2 16:42:39 2003>[java.lang.OutOfMemoryError : (C Heap) : unable to create thread : 340 threads exist]
```

-XX:[+|-]HitachiOutOfMemoryStackTrace (スタックトレース出力オプション)

形式

-XX:+HitachiOutOfMemoryStackTrace

OutOfMemoryError 発生時に、例外情報とスタックトレースを JavaVM ログファイルに出力します。スタックトレースは 1 スタックごとにバッファに格納し、コード変換した後に出力します。スタックトレースの出力は、OutOfMemoryError がスローされるたびに行われるため、OutOfMemoryError をキャッチして再スローした場合には複数回出力されます。なお、スレッド作成時に OutOfMemoryError となった場合は、スタックトレースは出力されません。

-XX:-HitachiOutOfMemoryStackTrace

OutOfMemoryError 発生時に、スタックトレースを JavaVM ログファイルに出力しません。

説明

OutOfMemoryError 発生時に、例外情報とスタックトレースを JavaVM ログファイルに出力するかどうかを指定します。

デフォルト値

- -XX:-HitachiOutOfMemoryStackTrace

出力形式

```
[id] [Thread:thread_id]<date>[java.lang.OutOfMemoryError : requested size bytes (cause) : reason : thread_count threads exist]
[id] [Thread:thread_id] stack_trace
```

出力内容を次に説明します。

id

OOM (JavaVM ログファイル識別子)。

thread_id

スレッド ID (スレッドダンプに出力されている tid)。

date

OutOfMemory 例外発生日時。

size

確保しようとしたメモリのサイズ（単位：バイト）が出力されます。-XX:-HitachiOutOfMemorySize オプションが指定されている場合、出力されません。

なお、次の場合は要求したメモリサイズが取り出せません。これらの場合は、サイズとして「unknown」が出力されます。

- Java のメモリ確保機能を使用しないで、明示的に OutOfMemoryError をスローした場合
標準クラスライブラリによってスローされるものも含まれます。例えば、「throw new OutOfMemoryError();」などの処理によってスローされた場合、メモリサイズは出力できません。
- クラスロード時の verifier によって OutOfMemoryError が発生した場合

cause

例外発生要因種別。ただし、-XX:-HitachiOutOfMemoryCause オプションが指定されている場合は出力されません。例外発生要因種別については、「[-XX:\[+|-\]HitachiOutOfMemoryCause（例外発生要因種別出力オプション）](#)」を参照してください。

reason

例外発生理由。スレッドの作成に失敗した場合に出力されます。

thread_count

OutOfMemoryError 発生時のスレッド数。作成に失敗したスレッド数も含まれます。

stack_trace

スタックトレース

出力例

```
[OOM][Thread: 0x00062fd0] <Wed Mar 17 00:41:17 2004>[java.lang.OutOfMemoryError :requested 4
00000 bytes. (C Heap): unable to create thread : 1500 threads exist]
[OOM][Thread: 0x00062fd0] at java.lang.Thread.start(Native Method)
[OOM][Thread: 0x00062fd0] at sub1.<init>(Thread0012.java:22)
[OOM][Thread: 0x00062fd0] at Thread0012.test01(Thread0012.java:73)
[OOM][Thread: 0x00062fd0] at Thread0012.main(Thread0012.java:57)
```

注意事項

JavaVM 自身で作成するスレッドが、メモリ不足によってスレッドの作成に失敗した場合は、例外情報だけが出力されます。スタックトレースは出力されません。

-XX:HitachiOutOfMemoryStackTraceLineSize（スタックトレース行サイズ指定オプション）

形式

-XX:HitachiOutOfMemoryStackTraceLineSize=<整数値>

説明

OutOfMemoryError 発生時に、出力するスタックトレース 1 行の文字数をバイト数で指定します。オプションの指定がない場合は、1024 バイトがデフォルトで設定されます。指定したバイト数が確保できない場合は警告メッセージが出力され、スタックトレースは出力されません。また、1 行の文字数が指定した文字数を超えた場合、「at」以降の文字列の前半部分を削除して、指定された文字数分出力します。

デフォルト値

- -XX:HitachiOutOfMemoryStackTraceLineSize=1024

前提オプション

- -XX:+HitachiOutOfMemoryStackTrace

引数

<整数値>

1024～2147483647 の範囲で整数値（単位：バイト）を指定します。範囲外の値が指定された場合は 1024 が設定されます。負の値を指定した場合はエラーとなります。

注意事項

非同期ログ出力（-XX:+JavaLogAsynchronous）が有効な場合、-XX:HitachiOutOfMemoryStackTraceLineSize に 4096 より大きな値を指定しても、4096 が設定されます。

-XX:[+|-]HitachiOutOfMemorySize（メモリサイズ出力オプション）

形式

-XX:+HitachiOutOfMemorySize

OutOfMemoryError 発生時に、要求したメモリサイズをバイト単位で出力します。

-XX:+HitachiOutOfMemoryStackTrace オプションが指定されている場合は、このオプションも設定されます。

-XX:-HitachiOutOfMemorySize

OutOfMemoryError 発生時に、要求したメモリサイズを出力しません。

説明

OutOfMemoryError 発生時に要求したメモリのサイズを出力します。

デフォルト値

- -XX:-HitachiOutOfMemorySize

出力例

```
[OOM][Thread: 0x00062fd0]<Tue Dec 2 16:42:39 2003>[java.lang.OutOfMemoryError : requested 1024 bytes. (Java Heap) : 20 threads exist]
```

注意事項

次の場合、要求したメモリサイズが取り出せません。

- Java のメモリ確保機能を使用してなくて、明示的に OutOfMemoryError をスローした (J2SE クラスライブラリがスローするものを含みます) 場合。

```
例:throw new OutOfMemoryError();
```

- クラスロード時の verifier が OutOfMemoryError を発生させた場合。

```
[OOM][Thread: 0x00062fd0]<Tue Dec 2 16:42:39 2003>[java.lang.OutOfMemoryError : requested size unknown. (Unknown) : 10 threads exist]
```

-XX:[+|-]HitachiOutOfMemoryAbort (強制終了オプション)

形式

-XX:+HitachiOutOfMemoryAbort

OutOfMemoryError 発生時にメモリダンプを出力して、強制終了します。

-XX:-HitachiOutOfMemoryAbort

OutOfMemoryError 発生時に強制終了しません。

説明

次の原因で OutOfMemoryError が発生した場合、標準出力にメッセージを、カレントディレクトリにメモリダンプまたは core ダンプを出力して強制終了します。

- Java ヒープ不足の場合
- Metaspace 領域不足の場合
- Compressed Class Space 不足の場合
- J2SE クラスライブラリ中での C ヒープ不足の場合

なお、JavaVM 処理中に C ヒープ不足となった場合には、このオプションの指定にかかわらず、強制停止します。

強制終了時の終了コード

強制終了した場合の JavaVM の終了コードを次に示します。

Windows の場合

1

UNIX の場合

6

なお、UNIX のシェル(sh や csh など)上で実行したとき、終了コードは 0x80 が加算されて 0x86 となります。

デフォルト値

- `-XX:-HitachiOutOfMemoryAbort`

出力例

```
java.lang.OutOfMemoryError occurred.  
JavaVM aborted because of specified -XX:+HitachiOutOfMemoryAbort options.
```

注意事項

- `-XX:+HitachiOutOfMemoryStackTrace` オプションが指定されている場合は、JavaVM ログファイルにスタックトレースを出力したあとに終了します。
- このオプションが指定されている場合、`java.io.File.deleteOnExit` メソッドや `java.lang.Runtime.addShutdownHook` メソッドで登録している、JavaVM 終了時の処理は実行されずに強制終了します。

-XX:[+|-]HitachiOutOfMemoryAbortThreadDump (スレッドダンプ出力オプション)

形式

`-XX:+HitachiOutOfMemoryAbortThreadDump`

OutOfMemoryError 発生時にスレッドダンプを出力します。

`-XX:+HitachiOutOfMemoryAbort` オプションが指定されている場合に、このオプションは指定できません。

`-XX:-HitachiOutOfMemoryAbortThreadDump`

OutOfMemoryError 発生時にスレッドダンプを出力しません。

説明

OutOfMemoryError 発生時にスレッドダンプを出力します。ただし、J2SE クラスライブラリで C ヒープ不足の場合は、スレッドダンプの出力による再度の C ヒープ不足発生を避けるため、スレッドダンプは出力しません。

スレッドダンプの出力先は環境変数 JAVACOREDIR, または -XX:+HitachiThreadDumpToStdout オプションで指定します。

デフォルト値

- -XX:+HitachiOutOfMemoryAbortThreadDump

前提オプション

- -XX:+HitachiOutOfMemoryAbort
- -XX:+HitachiThreadDump

-XX:[+|-]HitachiOutOfMemoryAbortThreadDumpWithJHeapProf (クラス別統計情報出力オプション)

形式

-XX:+HitachiOutOfMemoryAbortThreadDumpWithJHeapProf

-XX:+HitachiOutOfMemoryAbortThreadDump で出力するスレッドダンプログファイルにクラス別統計情報を出力します。

-XX:-HitachiOutOfMemoryAbortThreadDumpWithJHeapProf

-XX:+HitachiOutOfMemoryAbortThreadDump で出力するスレッドダンプログファイルにクラス別統計情報を出力しません。

説明

-XX:+HitachiOutOfMemoryAbortThreadDump で出力するスレッドダンプログファイルにクラス別統計情報を出力するかどうかを指定します。

デフォルト値

-XX:-HitachiOutOfMemoryAbortThreadDumpWithJHeapProf

前提オプション

- -XX:+HitachiOutOfMemoryAbort
- -XX:+HitachiOutOfMemoryAbortThreadDump
- -XX:+HitachiThreadDump

注意事項

-XX:+HitachiOutOfMemoryAbortThreadDumpWithJHeapProf オプションは OutOfMemory 発生時に出力されるスレッドダンプに、クラス別統計情報を出力するためのオプションです。G1GC を使用した場合、クラス別統計機能が使用できないため、このオプションも使用できません。G1GC 使用時にこのオプションを指定した場合、スレッドダンプは出力されますが、クラス別統計情報は出力されません。

-XX:[+|-]HitachiOutOfMemoryHandling (OutOfMemory ハンドリングオプション)

形式

-XX:+HitachiOutOfMemoryHandling

OutOfMemory ハンドリング機能を有効にします。

-XX:-HitachiOutOfMemoryHandling

OutOfMemory ハンドリング機能を無効にします。

説明

OutOfMemory ハンドリング機能を有効にするかどうかを指定します。このオプションは、J2EE サーバを J2EE サーバモードで実行した場合にだけ有効です。

OutOfMemory ハンドリング機能は、OutOfMemory 発生時強制終了機能 (-XX:+HitachiOutOfMemoryAbort) と組み合わせて使用します。OutOfMemory 発生時強制終了機能が無効 (-XX:-HitachiOutOfMemoryAbort) になっている場合、OutOfMemory ハンドリング機能は無効になります。

OutOfMemory ハンドリング機能を有効にした場合、OutOfMemory 発生時に OutOfMemoryError スロー条件が判定されます。具体的には、次に示す処理を実行中に Java ヒープ不足や Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory が発生した場合に、J2EE サーバの実行を継続するかどうか判定されます。

- Web コンテナ上の Web アプリケーション (Servlet/JSP) が実行中のリクエスト処理
- EJB クライアントアプリケーションから呼び出された Enterprise Bean が実行中の処理
- Message-driven Bean が実行中の処理
- Timer Service から呼び出された Enterprise Bean が実行中の処理

判定の結果、J2EE サーバの実行が継続される場合は、java.lang.OutOfMemoryError がスローされて、OutOfMemory が発生したリクエスト処理だけが中止されます。

JavaVM は、判定結果によって次のように動作します。ただし、Web アプリケーションで java.lang.OutOfMemoryError をキャッチしている場合は、その処理に従います。

- OutOfMemoryError スロー条件を満たしている場合
java.lang.OutOfMemoryError をスローして、OutOfMemory が発生したリクエスト処理だけを中止します。
- OutOfMemoryError スロー条件を満たしていない場合
OutOfMemory 発生時強制終了機能によって、JavaVM を強制終了します。

OutOfMemoryError スロー条件を次に示します。なお、OutOfMemoryError スロー条件に合致しない場合、OutOfMemory 発生時強制終了機能によって、JavaVM は強制終了します。

OutOfMemoryError スロー条件

- OutOfMemory が発生した場合に、次に示す条件すべてに該当するとき、JavaVM は OutOfMemoryError スロー条件を満たしているものと判定して、異常終了しないで java.lang.OutOfMemoryError をスローします。
- Java ヒープ不足、または Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory である。
 - Web コンテナ上の Web アプリケーション（Servlet/JSP）が実行中のリクエスト処理、EJB クライアントアプリケーションから呼び出された Enterprise Bean が実行中の処理、Message-driven Bean が実行中の処理、または Timer Service から呼び出された Enterprise Bean が実行中の処理で発生した OutOfMemory である。
 - OutOfMemoryError スロー除外条件に該当しない。

OutOfMemoryError スロー除外条件

今回の OutOfMemory が発生した時刻から過去 1 時間以内の Java ヒープ不足が原因の OutOfMemory の発生回数と Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory の発生回数の合計値（今回の OutOfMemory を含む）が、 - XX:HitachiOutOfMemoryHandlingMaxThrowCount オプション値に指定した値よりも大きい。

OutOfMemory 発生時強制終了機能と OutOfMemory ハンドリング機能の OutOfMemory 発生要因ごとの動作を次の表に示します。OutOfMemory ハンドリング機能が有効かどうかは、OutOfMemory の発生原因が Java ヒープ不足または Metaspace 領域、Compressed Class Space 不足の場合の動作に影響します。

表 14-8 OutOfMemory 発生時強制終了機能と OutOfMemory ハンドリング機能の OutOfMemory 発生要因ごとの動作

OutOfMemory 発生要因	OutOfMemory 発生時強制終了機能の対象となるかどうか (OutOfMemory 発生時の動作※1)	OutOfMemory ハンドリング機能の対象となるかどうか (OutOfMemory 発生時の動作)
Java ヒープ不足	対象になります (JavaVM は強制終了します)。	対象になります (判定結果に従って動作します※2)。
Metaspace 領域不足	対象になります (JavaVM は強制終了します)。	対象になります (判定結果に従って動作します※2)。
Compressed Class Space 不足	対象になります (JavaVM は強制終了します)。	対象になります (判定結果に従って動作します※2)。
C ヒープ不足	対象になります (JavaVM は強制終了します)。	対象になりません (JavaVM は強制終了します※3)。
Unkown	対象になりません	対象になりません

OutOfMemory 発生要因	OutOfMemory 発生時強制終了機能の対象となるかどうか (OutOfMemory 発生時の動作※1)	OutOfMemory ハンドリング機能の対象となるかどうか (OutOfMemory 発生時の動作)
	(java.lang.OutOfMemoryError がスローされます)。	(java.lang.OutOfMemoryError がスローされます)。
Thread Limit	対象になりません (java.lang.OutOfMemoryError がスローされます)。	対象になりません (java.lang.OutOfMemoryError がスローされます)。

注※1 OutOfMemory ハンドリング機能は無効 (-XX:-HitachiOutOfMemoryHandling) の場合の動作です。

注※2 OutOfMemoryError スロー条件を満たしている場合、java.lang.OutOfMemoryError がスローされます。

OutOfMemoryError スロー条件を満たしていない場合、OutOfMemory 発生時強制終了機能の処理に移行して、JavaVM は強制終了します。

注※3 OutOfMemory 発生時強制終了機能によって、JavaVM は強制終了します。

また、OutOfMemory ハンドリング機能が有効な場合、Java ヒープ不足、および Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory 発生時に OutOfMemory の発生頻度に関する情報が JavaVM ログファイルに出力されます。

デフォルト値

- -XX:-HitachiOutOfMemoryHandling

前提オプション

- -XX:+HitachiOutOfMemoryAbort

出力形式

```
[id][Thread: thread_id]<date>[Handling: oom_count(max_oom_count)]
```

出力内容を次に説明します。

id

OMH (Java VMログファイル識別子)。

thread_id

スレッド ID (スレッドダンプに出力されている tid)。

date

OutOfMemory をハンドリングした日時。

oom_count

今回の OutOfMemory が発生した時刻から過去 1 時間以内の、Java ヒープ不足が原因の OutOfMemory の発生回数と Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory の発生回数の合計値 (今回の OutOfMemory を含む)。

ただし、今回の OutOfMemory も含めて、1 時間以内の発生回数の合計値が 3601 を超えた場合でも、出力項目の最大値は 3601 になります。

max_oom_count

-XX:HitachiOutOfMemoryHandlingMaxThrowCount オプションに指定した値。

出力例

```
[0MH][Thread: 0x00927f48]<Tue Aug 24 19:02:19 2010>[Handling: 1(60)]
```

注意事項

- OutOfMemory ハンドリング機能は、OutOfMemory の根本原因を解決したり、OutOfMemory 発生による J2EE サーバの終了を確実に回避したりする機能ではありません。また、OutOfMemory 発生後に J2EE サーバの実行を継続できることを確実に保証する機能でもありません。この機能は、ユーザープログラム処理の問題による突発的な OutOfMemory が発生した場合に、J2EE サーバの終了を一時的に防ぐための機能です。

java.lang.OutOfMemoryError は、OutOfMemory ハンドリング機能の設定に関係なく、ヒープが枯渇している場合に発生します。java.lang.OutOfMemoryError が発生した場合には、J2EE サーバをできるだけ速やかに再起動して回復すること、そのあとで OutOfMemory の根本原因を解決することを推奨します。

なお、java.lang.OutOfMemoryError をスローすることによって、リソースのリークやロックなどが発生し、J2EE サーバに予期しない動作が発生するおそれがあります。このため、この機能を有効にしている場合に OutOfMemory が発生したときには、適切なタイミングで J2EE サーバを再起動する運用にしてください。また、J2EE サーバに予期しない動作が発生することを避けたい場合は、この機能を無効にしてください。この場合、突発的な OutOfMemory が発生したときには、J2EE サーバは終了します。

この機能を有効にした場合に、予期しない動作になってしまったときは、J2EE サーバを再起動してください。また、その後の運用ではこの機能を無効にしてください。

- OutOfMemory ハンドリング機能が有効な場合、OutOfMemory 発生時強制終了機能が有効（-XX:+HitachiOutOfMemoryAbort）でも、jsp や Servlet での処理中に OutOfMemory が発生したときは、強制終了しないで Java SE の仕様どおりに java.lang.OutOfMemoryError がスローされます。そのため、例えば、finally 節を使用して適切にリソースを解放していないようなときには、リソースの解放漏れなどが発生することがあります。

このような問題を避けて、従来どおり強制終了させたい場合は、OutOfMemory ハンドリング機能を無効にしてください。

-XX:HitachiOutOfMemoryHandlingMaxThrowCount（最大発生回数の設定オプション）

形式

-XX:HitachiOutOfMemoryHandlingMaxThrowCount=<整数値>

説明

OutOfMemory ハンドリング機能を有効にした場合の、Java ヒープ不足または Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory 発生回数合計値の 1 時間当たりの上限値を指定します。

このオプションで指定した値は、OutOfMemoryError スロー除外条件の判定で使用されます。

Java ヒープ不足または Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory が発生した場合、次のように処理が実行されます。

- 今回の OutOfMemory が発生した時刻から過去 1 時間以内の Java ヒープ不足が原因の OutOfMemory の発生回数と Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory の発生回数の合計値（今回の OutOfMemory を含む）が、このオプションに指定した上限値よりも大きい場合、OutOfMemoryError はスローされません。OutOfMemory 発生時強制終了機能（-XX:+HitachiOutOfMemoryAbort）の処理に移行して、JavaVM は強制終了します。
- このオプションの値として 0 を指定した場合は、OutOfMemory が発生したときに必ず OutOfMemory 発生時強制終了機能の処理に移行して、JavaVM は強制終了します。
- このオプションの値として 3600 を指定した場合は、OutOfMemoryError スロー除外条件の判定が実行されません。この場合、OutOfMemory が発生したときに OutOfMemoryError スロー除外条件以外の OutOfMemoryError スロー条件を満たしている場合、必ず java.lang.OutOfMemoryError がスローされます。

デフォルト値

- -XX:HitachiOutOfMemoryHandlingMaxThrowCount=60

前提オプション

- -XX:+HitachiOutOfMemoryHandling

引数

<整数値>

Java ヒープ不足または Metaspace 領域、Compressed Class Space 不足が原因の OutOfMemory 発生回数合計値の 1 時間当たりの上限値を指定します。

0~3600 (3601~ $2^{63}-1$ の場合は 3600 として扱います)。

注意事項

自然数以外の値が指定された場合は、定義していないオプションを指定した場合と同様の動作になります。

-XX:[+|-]HitachiJavaClassLibTrace (クラスライブラリのスタックトレース出力オプション)

形式

-XX:+HitachiJavaClassLibTrace

クラスライブラリのスタックトレースを出力します。

-XX:-HitachiJavaClassLibTrace

クラスライブラリのスタックトレースを出力しません。

説明

次に示すシステム全体に影響を与えるメソッドが使用された場合、これらの API のスタックトレースを、JavaVM ログファイルへ出力します。

- java.lang.System.gc
- java.lang.System.exit
- java.lang.System.runFinalizersOnExit
- java.lang.Runtime.exit
- java.lang.Runtime.halt
- java.lang.Runtime.runFinalizersOnExit

デフォルト値

- -XX:-HitachiJavaClassLibTrace

出力形式

```
[id] [Thread:thread_id]<date>  
[id] [Thread:thread_id] stack_trace
```

出力内容を次に説明します。

id :

CLT (Java VMログファイル識別子)。

thread_id :

スレッド ID (スレッドダンプに出力されている tid)。

date :

クラスライブラリ使用日時。

stack_trace :

スタックトレース。

出力例 1

```
[CLT][Thread: 0x00062fd0]<Mon Sep 27 12:10:03 2004>
[CLT][Thread: 0x00062fd0] at at java.lang.Shutdown.halt0(Native Method)
[CLT][Thread: 0x00062fd0] at java.lang.Shutdown.halt(Shutdown.java:145)
[CLT][Thread: 0x00062fd0] - locked <0x100101a0> (a java.lang.Shutdown$Lock)
[CLT][Thread: 0x00062fd0] at java.lang.Shutdown.exit(Shutdown.java:222)
[CLT][Thread: 0x00062fd0] - locked <0x1413c0a0> (a java.lang.Class)
[CLT][Thread: 0x00062fd0] at java.lang.Terminator$1.handle(Terminator.java:35)
[CLT][Thread: 0x00062fd0] at sun.misc.Signal$1.run(Signal.java:195)
[CLT][Thread: 0x00062fd0] at java.lang.Thread.run(Thread.java:534)
```

出力例 2

```
[CLT][Thread: 0x009c4000]<Tue Oct 09 15:36:18 2012>
[CLT][Thread: 0x009c4000] at java.lang.Runtime.outputJavaClassLibTrace(Native Method)
[CLT][Thread: 0x009c4000] at java.lang.Runtime.runFinalizersOnExit(Runtime.java:378)
[CLT][Thread: 0x009c4000] at java.lang.System.runFinalizersOnExit(System.java:978)
[CLT][Thread: 0x009c4000] at Program.main(Program.java:8)
```

-XX:HitachiJavaClassLibTraceLineSize (クラスライブラリのスタックトレース行サイズ指定オプション)

形式

```
-XX:HitachiJavaClassLibTraceLineSize=<整数値>
```

説明

-XX:+HitachiJavaClassLibTrace オプションを指定した場合に出力される、クラスライブラリのスタックトレースの 1 行の文字数をバイト数で指定します。

デフォルト値

- -XX:HitachiJavaClassLibTraceLineSize=1024

前提オプション

- -XX:+HitachiJavaClassLibTrace

引数

<整数値>

1024～2147483647 の範囲で整数値（単位：バイト）を指定します。範囲外の値が指定された場合は 1024 が設定されます。

出力例

```
[CLT][Thread: 0x00286c58]<Thu Oct 21 14:56:24 2004>
[CLT][Thread: 0x00286c58] at java.lang.Runtime.gc(Native Method)
[CLT][Thread: 0x00286c58] at java.lang.System.gc(System.java:737)
[CLT][Thread: 0x00286c58] at mple.func_012345678～省略～xyz(Sample.java:9)
[CLT][Thread: 0x00286c58] at Sample.main(Sample.java:5)
```

1 行の文字数が指定したバイト数を超える場合、「at」以降の文字列の前半部分が削除されます。出力例の場合、4 行目の Sample の前半が削除されて mple となります。

-XX:[+|-]HitachiLocalsInThrowable（例外発生時のローカル変数情報収集オプション）

形式

-XX:+HitachiLocalsInThrowable

スタックトレース中のメソッドのローカル変数情報を出力します。

ただし、java.lang.StackOverflowError 発生時には、このオプションは無視されます。

-XX:-HitachiLocalsInThrowable

スタックトレース中のメソッドのローカル変数情報を出力しません。

説明

java.lang.Throwable.fillInStackTrace メソッド実行時に、スタックトレース中のメソッドのローカル変数情報を収集します。

デフォルト値

- -XX:-HitachiLocalsInThrowable

出力形式

```
locals:
  name: <name>
  type: <type>
  value: <value>
...
```

出力内容を次に示します。

- 1 行目に「locals:」という見出しが出力されます。
- 2 行目以降は、収集可能であった各ローカル変数について、次の情報が 1 行ずつ出力されます。
 1. 変数名

2. 型名（基本型名，クラス名または配列型名）

3. 変数値を表現する文字列

なお，各ローカル変数の出力内容は，空行で区切られています。

<name> :

ローカル変数名。

メソッドに渡される引数の場合は，変数名に続いて[arg***]（***は引数番号）が表示されます。

<type> :

ローカル変数の型名（基本型名，クラス名または配列型名）。

<value> :

ローカル変数の値を表現する文字列。

- 基本型：
値をそのまま文字列化したもの
- クラスまたは配列型：
変数値が null の場合：(null)
それ以外の場合：<オブジェクトの存在するアドレス>

値表現の最大文字列長は 64 です。これを超える場合は 64 文字目までを出力した後，「...」という文字列が出力されます。クラスまたは配列型の場合，次の追加オプションを指定することでより詳細な表現が追加できます。

- -XX:+HitachiLocalsSimpleFormat
- -XX:+HitachiTrueTypeInLocals
- -XX:HitachiCallToString

出力例

Java プログラム例 1 を使用した出力例を，次に示します。

すべてのローカル変数情報が出力される場合

```
at Example1.method(Example1.java:15)
  locals:
    name: this
    type: Example1
    value: <0x922f42d0>

    name: l1 [arg1]
    type: int
    value: 1

    name: l2 [arg2]
    type: char
    value: 'Q'
```

```

    name: l3 [arg3]
    type: java.lang.Object
    value: <0xaf112f08>

    name: l4
    type: float
    value: 4.000000

    name: l5
    type: boolean
    value: true

    name: l6
    type: double
    value: 1.79769E+308

    name: l7
    type: java.lang.Object[]
    value: <0x922f42d8>

at Example1.main(Example1.java:5)
  locals:
...

```

ローカル変数情報が存在しない場合

- -g オプションまたは-g:vars オプションを付加しないで class ファイルを生成した場合
- -g オプションまたは-g:vars オプションを付加して生成した class ファイルの、native メソッドの場合

```

at Example1.method(Example1.java:15)
  locals:
    name: this
    type: Example1
    value: <0x922f42d0>

    name: [arg1]
    type: int
    value: 1

    name: [arg2]
    type: char
    value: 'Q'

    name: [arg3]
    type: java.lang.Object
    value: <0xaf112f08>

at Example1.main(Example1.java:5)
  locals:
...

```

注意事項

- ローカル変数情報を完全に収集するためには、javac で class ファイルを生成する際に、-g オプションまたは-g:vars オプションを付加して class ファイル内にローカル変数情報を埋め込んでおく必要がある

ります。-g オプションまたは-g:vars オプションを付加しないで作成された class ファイルについては、収集可能な範囲でローカル変数情報が出力されます。

- -g オプションまたは-g:vars オプションを付加して生成された class ファイルでも、native メソッドの場合はローカル変数情報が存在しません。
- JIT コンパイラがメソッドを JIT コンパイルする際、最適化の一環として、不要と判断したローカル変数を除去することがあります。

(例) `int not_used = 12345` といった宣言および初期化以降未使用のローカル変数

この場合、例外発生時点のローカル変数情報には、次の値が出力されます。

型名	出力情報
boolean 型	false
char 型	'¥0'
byte 型 short 型 int 型 long 型 float 型 double 型	0
クラス型 配列型	(null)

- 制御構造が複雑で行数が多いメソッドのローカル変数情報を出力する場合、解析に時間が掛かるため、例外発生時の例外オブジェクト生成処理に時間が掛かることがあります。
- `java.lang.Thread` クラスの `getStackTrace` メソッドを使用して取得した、カレントスレッドのスタックトレースにローカル変数情報を出力するためには、例外発生時のスタックトレースにローカル変数情報を出力する、例外発生時のローカル変数情報収集オプション (`-XX:+HitachiLocalsInThrowable`) が必要になります。

-XX:[+|-]HitachiLocalsInStackTrace (スレッドダンプ出力時のローカル変数出力オプション)

形式

`-XX:+HitachiLocalsInStackTrace`

スレッドダンプ出力時のスタックトレースに、ローカル変数情報を出力します。

`-XX:-HitachiLocalsInStackTrace`

スレッドダンプ出力時のスタックトレースに、ローカル変数情報を出力しません。

説明

スレッドダンプ出力時のスタックトレースに、各メソッドのローカル変数情報を追加して出力します。ローカル変数情報の出力内容については、「[-XX:\[+|-\]HitachiLocalsInThrowable](#)（例外発生時のローカル変数情報収集オプション）」を参照してください。

デフォルト値

- `-XX:-HitachiLocalsInStackTrace`

出力例

Java プログラム例 2 を使用した出力例を、次に示します。

- `-XX:+HitachiLocalsSimpleFormat` オプションおよび `-XX:+HitachiTrueTypeInLocals` オプションが指定されている場合

```
"main" prio=1 tid=0xb6e88d20 nid=0xb7492080 runnable [bffffb000..bffffb474]
  at Example2.method(Example2.java:15)
    - locked <0xab040550> (a Example2)
    locals:
      (Example2) this = <0xab040550> (Example2)
      (int) l1 = 1
      (float) l2 = 2.000000
      (java.lang.String) l3 = <0xaf112cc0> (java.lang.String)
      (java.lang.Character) l4 = <0xab040698> (java.lang.Character)
      (java.lang.Object) l5 = <0xab0407c8> (java.lang.Thread)
      (java.lang.Object[]) l6 = <0xab0408b8> (java.lang.Thread[])
  at Example2.main(Example2.java:4)
    locals:
      (java.lang.String[]) args [arg1] = <0xab040540> (java.lang.String[])
      (Example2) e2 = <0xab040550> (Example2)
```

注意事項

- ローカル変数情報を完全に収集するためには、`javac` で class ファイルを生成する際に、`-g` オプションまたは `-g:vars` オプションを付加して class ファイル内にローカル変数情報を埋め込んでおく必要があります。`-g` オプションまたは `-g:vars` オプションを付加しないで作成された class ファイルについては、収集可能な範囲でローカル変数情報が出力されます。
- 一般にスタックトレース情報の収集を行うスレッドと収集対象のスレッドは一致しません。このため、情報収集を行うためには、対象スレッドを停止させる必要があります、`toString` メソッドを呼び出すことはできません。このため、`-XX:HitachiCallToString` オプションの指定は無効になります。
- 制御構造が複雑で行数が多いメソッドのローカル変数情報を出力する場合、解析に時間が掛かるため、拡張スレッドダンプの出力やスレッドスタックトレースの取得に時間が掛かることがあります。

-XX:[+|-]HitachiLocalsSimpleFormat (ローカル変数情報の出力フォーマット変更オプション)

形式

-XX:+HitachiLocalsSimpleFormat

ローカル変数情報出力を、簡易フォーマットで出力します。

-XX:-HitachiLocalsSimpleFormat

ローカル変数情報出力を、通常フォーマットで出力します。

説明

ローカル変数情報の出力フォーマットを、1 変数 1 行で出力する簡易出力フォーマットに変更します。

デフォルト値

- -XX:-HitachiLocalsSimpleFormat

前提オプション

- -XX:+HitachiLocalsInThrowable
- -XX:+HitachiLocalsInStackTrace

出力形式

```
locals:
  (type) name = value
  (type) name = value
...
```

type, name および value の出力内容については、[「-XX:\[+|-\]HitachiLocalsInThrowable \(例外発生時のローカル変数情報収集オプション\)」](#)を参照してください。

出力例

Java プログラム例 1 を使用した出力例を、次に示します。

```
at Example1.method(Example1.java:15)
  locals:
    (Example1) this = <0x922f42d0>
    (int) l1 [arg1] = 1
    (char) l2 [arg2] = 'Q'
    (java.lang.Object) l3 [arg3] = <0xaf112f08>
    (float) l4 = 4.000000
    (boolean) l5 = true
    (double) l6 = 1.79769E+308
    (java.lang.Object[]) l7 = <0x922f42d8>
at Example1.main(Example1.java:5)
```



```
    locals:  
    ...
```

-XX:[+|-]HitachiTrueTypeInLocals (ローカル変数情報の実型名出力オプション)

形式

-XX:+HitachiTrueTypeInLocals

ローカル変数情報に、実際のオブジェクト型名を出力します。

-XX:-HitachiTrueTypeInLocals

ローカル変数情報に、実際のオブジェクト型名を出力しません。

説明

ローカル変数情報収集時に、クラスまたは配列型のローカル変数について、実際に代入されているオブジェクトの型名を出力します。変数値を表現する文字列の後ろの丸括弧内に表示されます。

なお、ローカル変数に格納されているクラスまたは配列型のオブジェクトが JavaVM 内部のオブジェクトの場合は、"internal type"と出力されます。

デフォルト値

- -XX:-HitachiTrueTypeInLocals

前提オプション

- -XX:+HitachiLocalsInThrowable
- -XX:+HitachiLocalsInStackTrace

出力例

Java プログラム例 3 を使用した出力例を、次に示します。

- -XX:+HitachiLocalsSimpleFormat オプションおよび-XX:HitachiCallToString=full が指定されている場合

```
at Example3.method(Example3.java:18)
  locals:
    (Example3) this = <0xaa07db58> "I am an Example3 instance." (Example3)
    (java.lang.String) l1 = <0xaa07df68> "local 1" (java.lang.String)
    (java.lang.StringBuffer) l2 = <0xaa07dca0> "local 1 + local 2" (java.lang.StringBuffer)
    (java.lang.Boolean) l3 = <0xaa07de18> "false" (java.lang.Boolean)
    (java.lang.Character) l4 = <0xaa07df68> "X" (java.lang.Character)
    (java.lang.Long) l5 = <0xaa07e078> "-9223372036854775808" (java.lang.Long)
    (java.lang.Object) l6 = <0xaa07e1a8> "Thread[Thread-0,5,main]" (java.lang.Thread)
    (java.lang.Object[]) l7 = <0xaa07e298> "[Ljava.lang.Thread;@26e431" (java.lang.Thre
```

```
ad[])  
  at Example3.main(Example3.java:4)  
    locals:  
  ...
```

-XX:HitachiCallToString (ローカル変数情報出力オプション)

形式

-XX:HitachiCallToString=<適用範囲>

説明

<適用範囲>に該当するクラスのローカル変数オブジェクトに対して取得した String オブジェクトの文字列を、変数値を表現する文字列として出力します。

なお、ローカル変数に格納されているオブジェクトがない場合、または JavaVM 内部のオブジェクトの場合は、出力されません。

デフォルト値

- -XX:HitachiCallToString=minimal

前提オプション

- -XX:+HitachiLocalsInThrowable

引数

<適用範囲>

minimal または full を指定します。

minimal :

java.lang パッケージ内の次に示すクラスが対象になります。

- String
- StringBuffer
- Boolean
- Byte
- Character
- Short
- Integer
- Long
- Float

- Double

なお、空文字 ("") を指定した場合も、minimal と同じです。

full :

すべてのクラスおよび配列型が対象になります。

出力例

Java プログラム例 3 を使用した出力例（簡易出力フォーマット）を、次に示します。

-XX:HitachiCallToString=minimalの場合

```
at Example3.method(Example3.java:18)
  locals:
    (Example3) this = <0xaa07db58>
    (java.lang.String) l1 = <0xae173a28> "local 1"
    (java.lang.StringBuffer) l2 = <0xaa07dca0> "local 1 + local 2"
    (java.lang.Boolean) l3 = <0xaa07de18> "false"
    (java.lang.Character) l4 = <0xaa07df68> "X"
    (java.lang.Long) l5 = <0xaa07e078> "-9223372036854775808"
    (java.lang.Object) l6 = <0xaa07e1a8>
    (java.lang.Object[]) l7 = <0xaa07e298>
at Example3.main(Example3.java:4)
  locals:
...
```

-XX:HitachiCallToString=fullの場合

```
at Example3.method(Example3.java:18)
  locals:
    (Example3) this = <0xaa07db58> "I am an Example3 instance."
    (java.lang.String) l1 = <0xae173a28> "local 1"
    (java.lang.StringBuffer) l2 = <0xaa07dca0> "local 1 + local 2"
    (java.lang.Boolean) l3 = <0xaa07de18> "false"
    (java.lang.Character) l4 = <0xaa07df68> "X"
    (java.lang.Long) l5 = <0xaa07e078> "-9223372036854775808"
    (java.lang.Object) l6 = <0xaa07e1a8> "Thread[Thread-0,5,main]"
    (java.lang.Object[]) l7 = <0xaa07e298> "[Ljava.lang.Thread;@26e431"
at Example3.main(Example3.java:4)
  locals:
...
```

注意事項

- ローカル変数情報を完全に収集するためには、javac で class ファイルを生成する際に、-g オプションまたは-g:vars オプションを付加して class ファイル内にローカル変数情報を埋め込んでおく必要があります。-g オプションまたは-g:vars オプションを付加しないで作成された class ファイルについては、収集可能な範囲でローカル変数情報が出力されます。

- このオプションを指定して、AWT または Swing を利用した Java のプログラムを実行する場合、＜適用範囲＞には「minimal」（デフォルト）を指定してください。＜適用範囲＞に「full」を指定すると、プログラムが正しく動作しません。
- ユーザプログラムでの例外発生時の原因調査時に限り、＜適用範囲＞に「full」を指定してください。この機能は、例外オブジェクト生成時、ローカル変数オブジェクトの toString() メソッドを呼び出す仕様です。toString() メソッドの呼び出しにより、原因調査に有用な情報が得られる反面、本来とは異なるタイミングで toString() メソッドを呼び出すことで、製品やユーザプログラムが正しく動作しない場合があるためです。

-XX:[+|-]HitachiFullCore（システムリソース解除オプション）

このオプションは、UNIX 用です。

形式

-XX:+HitachiFullCore

システムリソース RLIMIT_CORE を強制的に最大値に引き上げて、障害発生時にユーザーリミットを無視して core ファイルを作成します。

-XX:-HitachiFullCore

システムリソース RLIMIT_CORE は変更しません。

説明

システムリソース RLIMIT_CORE の設定を変更するかどうかを指定します。

デフォルト値

- -XX:-HitachiFullCore

-XX:[+|-]HitachiUseExplicitMemory（明示管理ヒープ機能オプション）

形式

-XX:+HitachiUseExplicitMemory

明示管理ヒープ機能を有効にします。

-XX:-HitachiUseExplicitMemory

明示管理ヒープ機能を無効にします。

説明

明示管理ヒープ機能の有効、無効を指定します。なお、明示管理ヒープ機能を有効にする場合は、-Xms と-Xmx の指定値が同じであることを推奨します。

デフォルト値

- -XX:-HitachiUseExplicitMemory

関連オプション

- -Xms
- -Xmx

注意事項

-XX:+UseParNewGC オプションを指定している場合、このオプションは指定できません。-XX:+UseParNewGC オプションと同時に指定した場合、メッセージが標準出力に出力されて、JavaVM がリターンコード 1 で終了します。

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:HitachiExplicitHeapMaxSize (Explicit メモリブロックの最大サイズ指定オプション)

形式

-XX:HitachiExplicitHeapMaxSize=<適用範囲>

説明

Explicit ヒープ全体の最大サイズを指定します。

デフォルト値

- -XX:HitachiExplicitHeapMaxSize=64m

前提オプション

- -XX:+HitachiUseExplicitMemory

引数

<適用範囲>

1 ~ 2⁶³-1 (9223372036854775807)

注意事項

- 適用範囲はバイト単位で指定します。
- 設定した値が有効な値の範囲に含まれていない場合、次のメッセージを標準出力に出力して JavaVM をリターンコード 1 で終了します。

```
Invalid maximum explicit heap size: -XX:HitachiExplicitHeapMaxSize=[指定した値]Could not create the Java virtual machine.
```

- このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:HitachiExplicitMemoryLogLevel (明示管理ヒープ機能のログ出力レベル指定オプション)

形式

-XX:HitachiExplicitMemoryLogLevel:<適用範囲>

説明

明示管理ヒープ機能のイベントログのレベルを指定します。

デフォルト値

- -XX:HitachiExplicitMemoryLogLevel:none

前提オプション

- -XX:+HitachiUseExplicitMemory

引数

<適用範囲>

none

明示管理ヒープ機能のイベントログを出力しません。

normal

通常運用での指定を想定しています。Explicit ヒープの状態を定期的（GC ごと）に出力します。また、Explicit ヒープの状態が大きく変化するイベント時に出力します。

verbose

normal の内容に加え、Explicit ヒープの状態が変化するイベント時に出力します。ログ量が normal よりも多くなります。

debug

verbose の内容に加え、一部のイベント発生時の詳細な情報を出力します。ログ量が大量になるため、パフォーマンスが著しく低下します。

注意事項

none, normal, verbose, debug 以外の値が指定された場合、次のメッセージを標準出力に出力して JavaVM をリターンコード 1 で終了します。

```
Invalid explicit heap log level: -XX:HitachiExplicitMemoryLogLevel:<適用範囲>Could not create the Java virtual machine.(log levels: none normal verbose debug)
```

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:HitachiExplicitMemoryJavaLog (明示管理ヒープ機能のログファイル出力先指定オプション)

形式

-XX:HitachiExplicitMemoryJavaLog:<文字列>

説明

明示管理ヒープ機能のログファイルのプリフィックスおよびログファイルの出力先ディレクトリを指定します。

デフォルト値

- -XX:HitachiExplicitMemoryJavaLog:ehjavalog

関連オプション

- -XX:HitachiJavaLog

引数

<文字列>

プリフィックスおよびパスを指定します。次の 3 種類の指定ができます。

プリフィックスを指定する場合

ログファイル名は、<文字列>???.log (?? は 01 ~ 99 の通し番号) で生成されます。例えば、<文字列> に"Samp" を指定すると、ログファイル名は Samp01.log になります。このオプションを指定しない場合、<文字列>には、"ehjavalog"が設定されます。また、ログファイルはカレントディレクトリに出力されます。

パスを指定する場合

<文字列>にディレクトリを指定した場合、そのディレクトリにファイルが作成されます。ログファイル名は、<文字列>ehjavalog???.log (?? は 01 ~ 99 の通し番号) で生成されます。

パスとプリフィックスを同時に指定する場合

<文字列> にディレクトリとプリフィックスを指定した場合、そのディレクトリにファイルが作成されます。ログファイル名は、<文字列>?.log (?? は 01 ~ 99 の通し番号) で生成されます。例えば、<文字列> に"d:¥temp¥Samp"を指定すると、d:¥temp ディレクトリに、Samp01.log が生成されます。

注意事項

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:HitachiExplicitMemoryJavaLogFileSize (明示管理ヒープ機能の最大ログファイルサイズ指定オプション)

形式

-XX:HitachiExplicitMemoryJavaLogFileSize=<整数値>

説明

1 ファイルの最大ファイルサイズをバイト単位で設定します。オプションの指定がない場合は、デフォルトで 256 キロバイトが設定されます。

デフォルト値

- -XX:HitachiExplicitMemoryJavaLogFileSize=256k

関連オプション

- -XX:HitachiJavaLogFileSize

引数

<整数値>

1024~2147483647 の範囲で整数値 (単位: バイト) 指定します。

範囲外の値が指定された場合は 1024 が設定されます。

注意事項

自然数以外の値が指定された場合は、定義していないオプションを指定した場合と同様の動作になります。

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:HitachiExplicitMemoryJavaLogNumberOfFile (明示管理ヒープ機能の最大ログファイル数指定オプション)

形式

-XX:HitachiExplicitMemoryJavaLogNumberOfFile=<整数値>

説明

作成する最大ファイル数を指定します。最大ファイル数を超えた場合は、再度最初に作成したファイルへ出力（ラップアラウンド）します。

デフォルト値

- -XX:HitachiExplicitMemoryJavaLogNumberOfFile=4

関連オプション

- -XX:HitachiJavaLogNumberOfFile

引数

<整数値>

1～99 の範囲で指定します。

ただし、100 以上の値が指定された場合は 99、0 が指定された場合は 1 となります。

注意事項

自然数以外の値が指定された場合は、定義していないオプションを指定した場合と同様の動作になります。

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:[+|-]HitachiExplicitMemoryMoveToTenuredFirst (明示管理ヒープ機能の明示解放機能オプション)

形式

-XX:+HitachiExplicitMemoryMoveToTenuredFirst

Explicit メモリブロックの解放処理でのオブジェクトの移動先が、Tenured 領域になります。

-XX:-HitachiExplicitMemoryMoveToTenuredFirst

Explicit メモリブロックの解放処理でのオブジェクトの移動先が、New 領域になります。

説明

Explicit メモリブロックの解放処理でのオブジェクトの移動先を指定します。

デフォルト値

- `-XX:+HitachiExplicitMemoryMoveToTenuredFirst`

前提オプション

- `-XX:HitachiUseExplicitMemory`

注意事項

このオプションが有効になっている場合、解放処理が実行されるときに被参照のオブジェクトは、Tenured 領域へ移動します。

参照元のオブジェクトが一時的に短命なオブジェクトの場合、このオプションを無効とし、Explicit メモリブロック内の被参照のオブジェクトを New 領域に移動することで、次の CopyGC の際に参照元オブジェクトと同時に回収できます。そうすることにより、このオプションが無効となっている場合に、Tenured 領域の利用済みサイズが上昇することを防げます。

ただし、次の点に注意が必要です。

- CopyGC 頻度の増加

Explicit メモリブロック解放時に、被参照のオブジェクトを New 領域に移動するため、移動したオブジェクトが New 領域を消費します。その結果、次の CopyGC までの間隔が短くなり、スループットの低下や、オブジェクトの加齢速度が上昇するおそれがあります。

- Tenured 領域に移動するオブジェクトが New 領域と明示管理ヒープ領域間で循環

複数の Explicit メモリブロック内のオブジェクトから参照されるオブジェクト（例えばアプリケーション共通のデータ定義）が存在する場合、そのオブジェクトは長寿命オブジェクトとなります。そのため、このオプションが有効な場合に、オブジェクトが Explicit メモリブロックへ移動したあと、その Explicit メモリブロックの解放処理が実行される際に Tenured 領域へ移動し、オブジェクトの寿命（アプリケーションの終了時など）まで Tenured 領域に存在し続けることになります。

一方、このオプションが有効な場合に、Explicit メモリブロックの解放処理が実行される際に New 領域へ移動し、その後しきい値までの CopyGC が実行されると、別の Explicit メモリブロックへ移動します。この繰り返しにより、オブジェクトが New 領域と明示管理ヒープ領域を循環し、CopyGC や Explicit メモリブロックの解放処理のオーバーヘッドが増大するおそれがあります。

このオプションは、JDK11 を使用する場合だけ指定できます。

`-XX:[+|-]HitachiExplicitMemoryAutoReclaim`（明示管理ヒープ機能の自動解放機能オプション）

形式

`-XX:+HitachiExplicitMemoryAutoReclaim`

明示管理ヒープ機能の自動解放機能を有効にします。

`-XX:-HitachiExplicitMemoryAutoReclaim`

明示管理ヒープ機能の自動解放機能を無効にします。

説明

明示管理ヒープ機能の自動解放機能を利用するかどうか指定します。

デフォルト値

- `-XX:+HitachiExplicitMemoryAutoReclaim`

前提オプション

- `-XX:-HitachiExplicitMemoryCompatibleToV8`

関連オプション

- `-XX:+HitachiAutoExplicitMemory`

注意事項

このオプションは、JDK11 を使用する場合だけ指定できます。

`-XX:[+|-]HitachiExplicitMemoryCompatibleToV8` (明示管理ヒープ機能のバージョン互換設定オプション)

形式

`-XX:+HitachiExplicitMemoryCompatibleToV8`

08-00 と同様の方法で Explicit メモリブロックを確保します。

`-XX:-HitachiExplicitMemoryCompatibleToV8`

08-00 と異なる方法で Explicit メモリブロックを確保します。

説明

Explicit メモリブロックを確保する方法を、08-00 と同様にするかどうか指定します。08-50 以降の新機能を利用しないで、08-00 で動作するアプリケーションをそのまま 08-50 で動作させる場合、このオプションを有効にします。

なお、このオプションを利用する場合、明示管理ヒープの自動解放機能は無効になります。

デフォルト値

- `-XX:-HitachiExplicitMemoryCompatibleToV8`

前提オプション

- `-XX:+HitachiUseExplicitMemory`

関連オプション

- `-XX:+HitachiExplicitMemoryAutoReclaim`
- `-XX:+HitachiAutoExplicitMemory`

注意事項

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:[+|-]HitachiAutoExplicitMemory (明示管理ヒープ機能の自動配置機能オプション)

形式

`-XX:+HitachiAutoExplicitMemory`

明示管理ヒープ機能の自動配置機能を有効にします。

`-XX:-HitachiAutoExplicitMemory`

明示管理ヒープ機能の自動配置機能を無効にします。

説明

Explicit メモリブロックの自動配置機能の有効、無効を指定します。

なお、このオプションを有効にした場合、Explicit メモリブロックの最小サイズは 16 キロバイトになります。無効に指定した場合、Explicit メモリブロックの最小サイズは 64 キロバイトになります。

デフォルト値

- `-XX:-HitachiAutoExplicitMemory`

前提オプション

- `-XX:+HitachiUseExplicitMemory`
- `-XX:+HitachiExplicitMemoryAutoReclaim`
- `-XX:-HitachiExplicitMemoryCompatibleToV8`

関連オプション

- `-XX:HitachiAutoExplicitMemoryFile`

注意事項

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:HitachiAutoExplicitMemoryFile (明示管理ヒープ機能の自動配置機能ファイルパス指定オプション)

形式

-XX:HitachiAutoExplicitMemoryFile:<文字列>

説明

Explicit メモリブロックの自動配置機能で利用する自動配置設定ファイルのパスを指定します。

デフォルト値

- -XX:HitachiAutoExplicitMemoryFile:<空文字>

前提オプション

- -XX:+HitachiAutoExplicitMemory

引数

<文字列>

JavaVM プロセスのカレントディレクトリからの相対パス、および絶対パスを指定します。パスはディレクトリ名を含めて指定できます。

なお、指定できる自動配置設定ファイルは一つだけです。

1 文字以上の文字列を指定した場合

明示管理ヒープ自動配置設定のデフォルト設定およびこのオプションで指定したファイルの設定が有効になります。

注意事項

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:ExplicitMemoryFullGCPolicy (明示管理ヒープ機能の Explicit メモリブロックへのオブジェクト移動制御オプション)

形式

-XX:ExplicitMemoryFullGCPolicy=<数値>

説明

FullGC 発生時に、参照関係に基づくオブジェクトの Java ヒープから Explicit メモリブロックへの移動を制御するかどうかを指定します。このオプションを使用すると、FullGC 発生時に Explicit ヒープへ移動していたオブジェクトを、Tenured 領域へ移動できます。

デフォルト値

- -XX:ExplicitMemoryFullGCPolicy=0

前提オプション

- -XX:+HitachiUseExplicitMemory

関連オプション

- -XX:+HitachiAutoExplicitMemory
- -XX:+ExplicitMemoryUseExcludeClass

引数

<数値>

0 または 1 を指定します。範囲外の値が指定された場合は 0 が設定されます。

0 を指定すると、FullGC 発生時、Explicit メモリブロック内にあるオブジェクトから参照されている Java ヒープ内のオブジェクトは、Explicit メモリブロックへ移動します。

1 を指定すると、FullGC 発生時、Explicit メモリブロック内にあるオブジェクトから参照されている Java ヒープ内のオブジェクトは、Explicit メモリブロックへ移動しません。New 領域にあるオブジェクトは Tenured 領域へ移動します。

注意事項

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:[+|-]ExplicitMemoryUseExcludeClass (明示管理ヒープ機能適用除外クラス指定機能オプション)

形式

-XX:+ExplicitMemoryUseExcludeClass

明示管理ヒープ機能適用除外クラス指定機能を有効にします。

-XX:-ExplicitMemoryUseExcludeClass

明示管理ヒープ機能適用除外クラス指定機能を無効にします。

説明

明示管理ヒープ機能適用除外クラス指定機能の有効、無効を指定します。明示管理ヒープ機能適用除外クラス指定機能を有効にすると、次のファイルに記述されているクラスには、明示管理ヒープ機能が適用されません。このため、ファイルに記述されているクラスのオブジェクトは、Explicit ヒープではなく、Tenured 領域に移動します。

Windows の場合

- <JDK インストールディレクトリ>%lib%explicitmemory%sysexmemexcludeclass.cfg
- <JDK インストールディレクトリ>%usrconf%exmemexcludeclass.cfg

UNIX の場合

- /opt/Cosminexus/jdk/lib/explicitmemory/sysexmemexcludeclass.cfg
- /opt/Cosminexus/jdk/usrconf/exmemexcludeclass.cfg

デフォルト値

互換モードの場合

- -XX:-ExplicitMemoryUseExcludeClass

推奨モードの場合

- -XX:+ExplicitMemoryUseExcludeClass

前提オプション

- -XX:+HitachiUseExplicitMemory

関連オプション

- -XX:+HitachiAutoExplicitMemory
- -XX:ExplicitMemoryFullGCPolicy
- -XX:ExplicitMemoryExcludeClassListFile
- -XX:ExplicitMemoryNotExcludeClassListFile

注意事項

- 明示管理ヒープ機能適用除外クラス指定機能を有効にすると、Explicit メモリブロックへのオブジェクト移動制御機能も有効になります。
- JavaVM 内のクラス (java.lang, java.util など) は、明示管理ヒープ機能適用除外クラスとして指定できません。
- このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:ExplicitMemoryExcludeClassListFile (明示管理ヒープ機能適用除外設定ファイルパス指定オプション)

形式

-XX:ExplicitMemoryExcludeClassListFile:<文字列>

説明

明示管理ヒープ機能適用除外クラス指定機能で利用する、明示管理ヒープ機能適用除外設定ファイルのファイルパスを指定します。このファイルに記述されているクラスには、明示管理ヒープ機能が適用されません。

JavaVM 起動時に、このオプションに指定したファイルパスのファイルを明示管理ヒープ機能適用除外設定ファイルとして読み込みます。なお、このオプションを省略、または空文字を指定した場合は、デフォルトのファイルパスのファイルを明示管理ヒープ機能適用除外設定ファイルとして読み込みます。

デフォルト値

- -XX:ExplicitMemoryExcludeClassListFile:<空文字>

Windows の場合

<JDK インストールディレクトリ>%usrconf%\exmemexcludeclass.cfg

UNIX の場合

/opt/Cosminexus/jdk/usrconf/exmemexcludeclass.cfg

前提オプション

- -XX:+HitachiUseExplicitMemory
- -XX:+ExplicitMemoryUseExcludeClass

関連オプション

- -XX:ExplicitMemoryFullGCPolicy
- -XX:+HitachiAutoExplicitMemory
- -XX:ExplicitMemoryNotExcludeClassListFile

引数

<文字列>

JavaVM プロセスのカレントディレクトリからの相対パス、または絶対パスを指定します。パスはディレクトリ名を含めて指定できます。

なお、指定できる明示管理ヒープ機能適用除外設定ファイルは一つだけです。また、ファイルの文字エンコーディングには、ASCII コードを使用してください。

注意事項

- 指定されたファイルがオープンできない、または読み込めない場合は、明示管理ヒープ機能のイベントログにワーニングメッセージを出力し、デフォルトパスにある設定ファイルを読み込みます。デフォルトパスにあるファイルをオープンできない、または読み込めない場合は、明示管理ヒープ機能適用除外クラス指定機能は有効になりません。
- 指定されたファイルの記述フォーマットに誤りがある場合は、明示管理ヒープ機能のイベントログにワーニングメッセージを出力して、次の行から構文解析処理を継続します。

- 明示管理ヒープ機能適用除外設定ファイルと、明示管理ヒープ機能適用除外無効設定ファイルの両方に、同じのクラス名が記述されていた場合は、明示管理ヒープ機能適用除外無効設定ファイルの指定が優先されます。この場合、そのクラスに対する明示管理ヒープ機能適用除外クラス指定機能は、無効になります。
- このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:ExplicitMemoryNotExcludeClassListFile (明示管理ヒープ機能適用除外無効設定ファイルパス指定オプション)

形式

`-XX:ExplicitMemoryNotExcludeClassListFile:<文字列>`

説明

明示管理ヒープ機能適用除外クラス指定機能で利用する、明示管理ヒープ機能適用除外無効設定ファイルのファイルパスを指定します。明示管理ヒープ機能適用除外設定ファイルに記述されていても、このファイルに記述されているクラスには、明示管理ヒープ機能適用除外クラス指定機能が適用されません。

JavaVM 起動時に、このオプションに指定したファイルパスのファイルを明示管理ヒープ機能適用除外無効設定ファイルとして読み込みます。なお、このオプションを省略、または空文字を指定した場合は、デフォルトのファイルパスのファイルを明示管理ヒープ機能適用除外無効設定ファイルとして読み込みます。

デフォルト値

- `-XX:ExplicitMemoryNotExcludeClassListFile:<空文字>`

Windows の場合

`<JDK インストールディレクトリ>%usrconf%exmemnotexcludeclass.cfg`

UNIX の場合

`/opt/Cosminexus/jdk/usrconf/exmemnotexcludeclass.cfg`

前提オプション

- `-XX:+HitachiUseExplicitMemory`
- `-XX:+ExplicitMemoryUseExcludeClass`

関連オプション

- `-XX:ExplicitMemoryFullGCPolicy`
- `-XX:HitachiAutoExplicitMemory`
- `-XX:ExplicitMemoryExcludeClassListFile`

引数

<文字列>

JavaVM プロセスのカレントディレクトリからの相対パス、および絶対パスを指定します。パスはディレクトリ名を含めて指定できます。

なお、指定できる明示管理ヒープ機能適用除外設定ファイルは一つだけです。また、ファイルの文字エンコーディングには、ASCII コードを使用してください。

注意事項

- 指定されたファイルがオープンできない、または読み込めない場合は、明示管理ヒープ機能のイベントログにワーニングメッセージを出力し、デフォルトパスにある設定ファイルを読み込みます。デフォルトパスにあるファイルをオープンできない、または読み込めない場合は、明示管理ヒープ機能適用除外クラス指定機能は無効になりません。
- 指定されたファイルの記述フォーマットに誤りがある場合は、明示管理ヒープ機能のイベントログにワーニングメッセージを出力して、次の行から構文解析処理を継続します。
- 明示管理ヒープ機能適用除外設定ファイルと、明示管理ヒープ機能適用除外無効設定ファイルの両方に、同じのクラス名が記述されていた場合は、明示管理ヒープ機能適用除外無効設定ファイルの指定が優先されます。この場合、そのクラスに対する明示管理ヒープ機能適用除外クラス指定機能は、無効になります。
- このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:[+|-]HitachiExplicitMemoryAutoRefReclaim (HTTP セッションで利用する Explicit ヒープの省メモリ化オプション)

形式

-XX:+HitachiExplicitMemoryAutoRefReclaim

HTTP セッションで利用する Explicit ヒープの省メモリ化機能を有効にします。

-XX:-HitachiExplicitMemoryAutoRefReclaim

HTTP セッションで利用する Explicit ヒープの省メモリ化機能を無効にします。

説明

HTTP セッションで利用する Explicit ヒープの省メモリ化機能の有効、無効を指定します。

無効にした場合は、アプリケーションサーバが作成した Explicit メモリブロックの自動解放の自動予約をしません。

有効にした場合は、自動配置設定ファイルで作成された Explicit メモリブロックに加えて、HTTP セッションオブジェクトを配置するためにアプリケーションサーバが作成した Explicit メモリブロックに対して自動解放の自動予約をします。これによって、必要な Explicit ヒープサイズを減少させることができます。

ただし、有効にした場合、稼働情報の出力内容に注意が必要です。詳細は、マニュアル「アプリケーションサーバ 機能解説 拡張編」の「7.11 HTTP セッションで利用する Explicit ヒープのメモリ使用量の削減」を参照してください。

デフォルト値

- `-XX:-HitachiExplicitMemoryAutoRefReclaim`

前提オプション

- `-XX:+HitachiExplicitMemoryAutoReclaim`

注意事項

このオプションは、JDK11 を使用する場合だけ指定できます。

-XX:HitachiJITCompileMaxMemorySize (JIT コンパイル時の確保メモリ上限値指定オプション)

形式

`-XX:HitachiJITCompileMaxMemorySize=<整数値>`

説明

JIT コンパイル時に確保する C ヒープのメモリサイズの上限值を指定します。JIT コンパイルによって確保する C ヒープのメモリが指定値を超えた場合、JavaVM ログファイルにログを出力します。JIT コンパイル処理の対象となっていた Java メソッドは、以降、インタプリタでだけ実行されるようになります。なお、その場合でも JavaVM は強制終了しないで処理を続行します。

0 を指定した場合、JIT コンパイル時に確保するメモリは上限値で制限されません。

なお、JIT コンパイルは JavaVM の内部スレッド (JIT コンパイラスレッド) で実行されます。JIT コンパイラスレッドは二つあるため、このオプションで指定した上限値を 2 で割った値が、一つの JIT コンパイラスレッドに対する上限値となります。

また、このオプションで上限値を設定した場合も、実際に使用できる C ヒープがそれよりも少ないときには、上限値に達する前に C ヒープ不足が発生することがあります。この場合、JavaVM は強制終了します。

デフォルト値

- `-XX:HitachiJITCompileMaxMemorySize=0`

前提オプション

- -server

引数

<整数値>

JIT コンパイル時に確保するメモリの上限値を指定します。単位はバイトです。単位文字として、「k」（キロ）、「m」（メガ）、「g」（ギガ）も指定できます。指定できる範囲は次のとおりです。範囲外の値を指定した場合は、0 が指定されます。

- $0 \sim 2^{64}-1$ (18446744073709551615)

0 を指定した場合は、JIT コンパイル時に確保するメモリは上限値で制限されません。JIT コンパイル中に C ヒープ不足が発生した場合、JavaVM は強制終了します。

出力形式

```
[id][Thread: thread_id]<date>["thread_name" exceeded max memory size.]
[current_sizeK->new_sizeK/limit_size_per_threadK/limit_sizeK]
[compile_target][byte_code_size]
```

出力内容を次に説明します。

id

JMS（JavaVM ログファイル識別子）。

thread_id

JIT コンパイルを抑制した JIT コンパイラスレッドのスレッド ID。

date

JIT コンパイルを抑制した日時。

-XX:+HitachiOutputMilliTime オプションが指定された場合、ミリ秒単位で出力されます。

thread_name

JIT コンパイルを抑制した JIT コンパイラスレッドのスレッド名。

current_size

JIT コンパイルを抑制した JIT コンパイラスレッドの、現在のメモリ確保サイズ（単位：キロバイト）。

new_size

JIT コンパイルを抑制した JIT コンパイラスレッドの、現在のメモリ確保サイズと追加で確保しようとしたサイズの合計値（単位：キロバイト）。

limit_size_per_thread

一つの JIT コンパイラスレッドの上限値（単位：キロバイト）。

limit_size

JIT コンパイラスレッド全体の上限値（単位：キロバイト）。

compile_target

JIT コンパイル処理の対象となっていた Java メソッド。

byte_code_size

JIT コンパイル処理の対象となっていた Java メソッドのバイトコードのサイズ（単位：バイト）。

出力例

オプションとして「-XX:HitachiJITCompileMaxMemorySize=536870912」を指定して、一つの JIT コンパイラスレッドに対して 262145 キロバイトの C ヒープを確保した場合に、上限値に達したときの出力例を示します。

```
[JMS][Thread: 0x03bf1150]<Wed Feb 24 14:33:58 2010>["CompilerThread0" exceeded max memory size. ][262143K->262145K/262144K/524288K][test1.func][213]
```

注意事項

- JIT コンパイル時に確保するメモリが、このオプションで指定した上限値を超えて、JIT コンパイルが抑制されると、アプリケーションのスループットが低下します。
- 自然数以外の値が指定された場合は、定義していないオプションを指定した場合と同様の動作になります。
- このオプションで指定した上限値を超えた時に JIT コンパイルが抑制される対象は、超えた時にコンパイルしていたメソッドだけです。それ以外のメソッドについては抑制されず、実行状況に応じて JIT コンパイル対象となります。
- このオプションによるメモリ確保量のチェックは、JIT コンパイル中の幾つかのチェックポイントで実施します。このチェックポイントの間に一時的に上限値を超えて確保することがあります。

-XX:[+|-]JITCompilerContinuation（JIT コンパイラ稼働継続機能オプション）

形式

-XX:+JITCompilerContinuation

JIT コンパイラ稼働継続機能を有効にします。

-XX:-JITCompilerContinuation

JIT コンパイラ稼働継続機能を無効にします。

説明

JIT コンパイラ稼働継続機能を有効にするかどうかを指定します。

JIT コンパイルがアプリケーションを構成するメソッドの論理矛盾で失敗しても、アプリケーションを正常に継続できるようにするためには、この機能を有効にすることをお勧めします。

JIT コンパイラ稼働継続機能を有効にした場合、JIT コンパイルがアプリケーションを構成するメソッドの論理矛盾で失敗しても、JavaVM は JavaVM ログファイルにこの機能のログを出力して、処理を続行します。この場合、JIT コンパイルに失敗したメソッドでは、以降のコンパイルはインタプリタ方式で実行されるようになります。アプリケーションを構成しているメソッドのうち、JIT コンパイルに失敗したメソッド以外のメソッドは JIT コンパイルで実行されるため、アプリケーションは正常に継続されます。ただし、JIT コンパイルが 6 回以上失敗すると、JavaVM はエラーリポートファイル、およびメモリダンプまたは core ダンプを出力して、強制終了します。

JIT コンパイラ稼働継続機能を無効にした場合、JIT コンパイルがアプリケーションを構成するメソッドの論理矛盾で失敗すると、JavaVM はエラーリポートファイル、およびメモリダンプまたは core ダンプを出力して、強制終了します。

デフォルト値

- -XX:+JITCompilerContinuation

前提オプション

- -server
- -XX:+HitachiVerboseGC

出力形式

[id]

注

[id]以降には JIT コンパイラ稼働継続機能のログが出力されます。

出力内容を次に説明します。

id :

JCC (Java VMログファイル識別子)。

出力例

```
[JCC][Thread: 0x05432c00]<Thu Nov 15 17:10:40 2012>[Method: jit_sample.func()V][Fail: 3][JITCT: 1]
[JCC][Thread: 0x05432c00][PC: 0x083aff9a][Lib: D:¥work¥jdk¥bin¥server¥jvm.dll+0x3aff9a][VM: Java HotSpot(TM) Server VM (20.8-b03-CDK0950-20121115 mixed mode windows-x86 )]
[JCC][Thread: 0x05432c00][EAX=0x00000000, EBX=0x00618128, ECX=0x00000000, EDX=0x05485340]
[JCC][Thread: 0x05432c00][ESP=0x0566d3c0, EBP=0x0566d3c4, ESI=0x00618278, EDI=0x00000000]
[JCC][Thread: 0x05432c00][EIP=0x083aff9a, EFLAGS=0x00010202]
[JCC][Thread: 0x05432c00][siginfo: read 0x00000000]
[JCC][Thread: 0x05432c00][Unlock: MethodCompileQueue_lock]
[JCC][Thread: 0x05432c00][NewJITCT: 0x05438800][JITCT: 2]
[JCC][Thread: 0x05432c00][Free: "ResourceArea" 524288 bytes.]
```

```
[JCC][Thread: 0x05432c00][stop]
[JCC][Fail: 1][date: Thu Nov 15 10:10:40 2012][Method: jit_sample.func1(Ljava/lang/String;)V]
[PC: 0x083ff00a][Lib: D:¥work¥jdk¥bin¥server¥jvm.dll+0x3ff00a]
[JCC][Fail: 2][date: Thu Nov 15 11:11:16 2012][Method: jit_sample.func2()V][PC: 0x083afe3a][
Lib: D:¥work¥jdk¥bin¥server¥jvm.dll+0x3afe3a]
```

-XX:[+|-]UseCompressedOops（圧縮オブジェクトポインター機能で使用する Java オプション）

形式

-XX:+UseCompressedOops

圧縮オブジェクトポインター機能を有効にします。

-XX:-UseCompressedOops

圧縮オブジェクトポインター機能を無効にします。

説明

圧縮オブジェクトポインター機能の有効、無効を指定します。圧縮オブジェクトポインター機能は、Java オブジェクトのサイズを圧縮して管理することで、JavaVM 実行時の Java ヒープ領域、および Explicit ヒープ領域の使用サイズを削減します。

なお、Explicit ヒープ領域は、明示管理ヒープ機能が有効な場合にだけ該当します。この機能を有効にするには、次の条件を満たしていることが前提となります。

- Java ヒープ領域、Metaspace 領域、および Explicit ヒープ領域の指定サイズの合計値が 32 ギガバイト未満であること

JavaVM 起動時、Java ヒープ領域、Metaspace 領域、および Explicit ヒープ領域の指定サイズの合計値が、32 ギガバイト以上の場合、JavaVM は次のメッセージを標準出力に出力し、圧縮オブジェクトポインター機能を無効とします。

Java HotSpot(TM) 64-Bit Server VM warning: Max heap size too large for Compressed Oops

デフォルト値

- -XX:-UseCompressedOops

-XX:HitachiThreadLimit（スレッド数の上限値指定オプション）

形式

-XX:HitachiThreadLimit=<整数値>

説明

スレッド数の上限値を指定します。スレッド数の上限値が指定値を超えた場合に、OutOfMemoryError 例外がスローされます。ただし、JavaVM が起動する前に上限値を超えた場合、例外はスローされません。また、スレッド数には、アプリケーションが JNI などを使用して JavaVM の管理外の範囲で作成したスレッド数は含みません。0 を指定した場合、上限値は設定されません。

なお、-XX:+HitachiOutOfMemoryAbort オプションと同時に指定した場合、このオプションの機能によって OutOfMemoryError 例外がスローされた場合、JavaVM の強制終了は実行されません。

また、-XX:+HitachiOutOfMemoryStackTrace オプションと同時に指定した場合、例外メッセージが JavaVM ログファイルに出力されます。

デフォルト値

- -XX:HitachiThreadLimit=0

引数

<整数値>

スレッド数の上限値を 0～2147483647 の範囲で整数値（単位：スレッド数）を指定します。範囲外の値が指定された場合は 0 が設定されます。負の値を指定した場合はエラーとなります。

出力形式

```
Could not create "name" thread. Threadlimit Exceeded. num threads exist.
```

このメッセージは、java.lang.Throwable.getMessage() で取得できます。また、メッセージ全体が半角 128 文字以上になった場合は、メッセージの末尾が省略されます。

name

作成に失敗したスレッド名。

num

現時点でのスレッド数。

-XX:+HitachiOutOfMemoryStackTrace オプションが指定されている場合、JavaVM ログファイルに例外情報とスタックトレースを出力できます。例外発生要因種別は、"Thread Limit" となります。例外発生要因種別については、[「-XX:\[+|-\]HitachiOutOfMemoryCause（例外発生要因種別出力オプション）」](#)を参照してください。

出力例

```
Could not create "Thread-1" thread. Threadlimit Exceeded. 9 threads exist.
```

この例は、main スレッドによって java.lang.Thread.start が呼び出され、この機能によって例外をスローした場合に、java.lang.Throwable.getMessage() でメッセージを取り出したメッセージの例です。

注意事項

上限値に小さな値を設定すると、起動前に例外をスローします。J2EE サーバが使用するスレッド数より小さな値を設定しないでください。

J2EE サーバが使用するスレッド数については、マニュアル「アプリケーションサーバ システム設計ガイド」の「5. 使用するリソースの見積もり (J2EE アプリケーション実行基盤)」または「6. 使用するリソースの見積もり (バッチアプリケーション実行基盤)」を参照してください。

-XX:[+|-]HitachiVerboseGCPrintDirectBuffer (ダイレクトバッファの使用状況出力オプション)

形式

-XX:+HitachiVerboseGCPrintDirectBuffer

ダイレクトバッファの使用状況を日立 JavaVM ログファイルに出力します。

-XX:-HitachiVerboseGCPrintDirectBuffer

ダイレクトバッファの使用状況を日立 JavaVM ログファイルに出力しません。

説明

ダイレクトバッファの使用状況を日立 JavaVM ログファイルに出力するかどうかを指定します。

デフォルト値

-XX:+HitachiVerboseGCPrintDirectBuffer

出力形式

日立 JavaVM ログファイル

[DirectBuffer: buffer_used_sizeK, buffer_total_sizeK, buffer_max_sizeK, buffer_count]

出力内容を次に説明します。

buffer_used_size

ダイレクトバッファの使用サイズ。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_total_size

ダイレクトバッファの合計容量。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_max_size

ダイレクトバッファの最大値。単位はキロバイトです。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

buffer_count

ダイレクトバッファの数。

-XX:-HitachiVerboseGCPrintDirectBuffer オプションが指定された場合、出力されません。

標準出力とエラーリポートファイル (hs_err ログ)

```
Direct Buffer:
Used : <使用サイズ>
Total: <合計容量>
Max   : <最大サイズ>
Count: <バッファ数>
```

出力内容を次に説明します。

使用サイズ

JavaAPI「ByteBuffer#allocateDirect()」で確保したダイレクトバッファの使用サイズ。単位はバイトです。

なお、ダイレクトバッファに書き込むデータのサイズではないため、使用サイズと合計容量は同じ値になります。

合計容量

ダイレクトバッファの合計容量。単位はバイトです。

最大サイズ

ダイレクトバッファの最大値。単位はバイトです。

バッファ数

ダイレクトバッファの数。

出力例

日立 JavaVM ログファイル

```
[DirectBuffer: 8K, 8K, 83200K, 1]
```

標準出力とエラーリポートファイル (hs_err ログ)

```
Direct Buffer:
Used : 102400
Total: 102400
Max   : 1048576
Count: 100
```

注意事項

- JavaAPI「ByteBuffer#allocateDirect()」以外の方法で確保した Java ヒープ以外の領域についての情報は、この機能では出力されません。
- 日立 JavaVM ログに出力される情報では、1 キロバイト未満の値は切り捨てられます。そのため、再度ダイレクトバッファを確保しても、出力される値が変わらないことがあります。ただし、buffer_count の値からダイレクトバッファを確保していることは確認できます。

14.3 JavaVM で使用するプロパティ

JavaVM で使用するプロパティの一覧を、次の表に示します。

表 14-9 JavaVM で使用するプロパティの一覧

分類	オプション名称	概要	VR	関連情報
JavaVM で使用する プロパティ	jvm.userprf.Enable	ユーザ拡張性能解析トレースを有効にするかどうかを指定します。	09-00	
	jvm.userprf.ExtendedSetting	ユーザ拡張性能解析トレース設定ファイルの記述形式に、次の方式を有効にするかどうかを指定します。 - パッケージ名とクラス名を指定する方式 - パッケージ名だけを指定する方式	09-50	「機能解説 保守／移行編」の「7. 性能解析トレースを使用した性能解析」
	jvm.userprf.File	ユーザ拡張性能解析トレース設定ファイルのファイルパスを指定します。	09-00	
	jvm.userprf.Limit	ユーザ拡張性能解析トレースの対象となるメソッド数の上限値を指定します。	09-00	
	jvm.userprf.Trace	ユーザ拡張性能解析トレース設定ファイルで指定したクラスファイルの書き換えの成功ログを出力するかどうかを指定します。	09-00	
	jvm.userprf.LineNumber	メソッドの最後に実行した行の行番号をトレース情報に出力するかどうかを指定します。	09-00	
	jvm.userprf.ThrowableName	例外またはエラーのクラス名をトレース情報に出力するかどうかを指定します。	09-00	
	jvm.userprf.ThrowableNameEditMethod	例外またはエラーのクラス名が文字数制限 32 文字を超えた場合の名前の編集方法を指定します。	09-00	
	jvm.userprf.LogLevel	ユーザ拡張性能解析トレースのトレース出力レベルを指定します。	09-00	
	JP.co.Hitachi.soft.jvm.autofinalizer	ファイナライズ滞留解消機能の有効または無効を設定します。	09-60	「機能解説 保守／移行編」の「9.16 ファイナライズ滞留解消機能」
	sun.nio.cs.map ※	エンコーディング名 shift_jis, csshiftjis, ms_kanji,x-sjis を MS932 と同等とするかどうかを指定します。 なお、JDK17 以降の場合、このプロパティは使用できません。	02-00	

注※

JDK17 以降の場合は使用できません。

jvm.userprf.Enable

形式

`jvm.userprf.Enable={true | false}`

説明

ユーザ拡張性能解析トレースを有効にするかどうかを指定します。

デフォルト値

`jvm.userprf.Enable=false`

前提プロパティ

なし。

引数

true

ユーザ拡張性能解析トレースを有効にします。

false

ユーザ拡張性能解析トレースを無効にします。

true, false 以外を指定した場合、デフォルト値となります。

注意事項

このプロパティに true を指定した場合、ユーザ拡張性能解析トレースは自動的にユーザ拡張性能解析トレース設定ファイルを読み込みます。事前にユーザ拡張性能解析トレース設定ファイルを設定してください。ユーザ拡張性能解析トレース設定ファイルのファイル名の指定については、jvm.userprf.File プロパティを参照してください。

指定例

プロパティの指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.Enable=true
```

jvm.userprf.ExtendedSetting

形式

`jvm.userprf.ExtendedSetting={true | false}`

説明

ユーザ拡張性能解析トレース設定ファイルの記述形式に、次の方式を有効にするかどうかを指定します。

- パッケージ名とクラス名を指定する方式
- パッケージ名だけを指定する方式

デフォルト値

`jvm.userprf.ExtendedSetting=false`

前提プロパティ

`jvm.userprf.Enable=true`

引数

true

ユーザ拡張性能解析トレース設定ファイルに、次の方式で記述できます。

- パッケージ名とクラス名を指定する方式
- パッケージ名だけを指定する方式

false

ユーザ拡張性能解析トレース設定ファイルに、次の方式は記述できません。

- パッケージ名とクラス名を指定する方式
- パッケージ名だけを指定する方式

true, false 以外を指定した場合、デフォルト値となります。

注意事項

このプロパティに false を指定して、true で有効になる方式でユーザ拡張性能解析トレース設定ファイルに記述した場合、ユーザ拡張性能解析トレース設定ファイルのフォーマットエラーになります。

指定例

プロパティの指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.ExtendedSetting=true
```

jvm.userprf.File

形式

jvm.userprf.File=<文字列>

説明

ユーザ拡張性能解析トレース設定ファイルのファイルパスを指定します。

ユーザ拡張性能解析トレースは、このプロパティに指定したファイルパスのファイルをユーザ拡張性能解析トレース設定ファイルとして読み込みます。なお、このプロパティを指定していない場合、デフォルトのファイルパスのファイルをユーザ拡張性能解析トレース設定ファイルとして読み込みます。

ポイント

ユーザ拡張性能解析トレース設定ファイルに記述されたメソッドがユーザ拡張性能解析トレースのトレース対象となります。

デフォルト値

Windows の場合

<JDKインストールディレクトリ>%usrconf%userprf.cfg

UNIX の場合

/opt/Cosminexus/jdk/usrconf/userprf.cfg

前提プロパティ

jvm.userprf.Enable=true

引数

<文字列>

ユーザ拡張性能解析トレース設定ファイルのファイルパスを指定します。

ユーザ拡張性能解析トレース設定ファイルのファイルパスは次のように指定します。

- 指定できるファイルパスは一つだけです。
- カレントディレクトリからの相対パスおよび絶対パスがパス文字列として有効です。カレントディレクトリの例を次に示します。

(例) cjstartsv で実行した場合のデフォルトのカレントディレクトリ

- Windows の場合
<Application Server のインストールディレクトリ>%CC%server%public%ejb%<サーバ名称>
- UNIX の場合
/opt/Cosminexus/CC/server/public/ejb/<サーバ名称>

注意事項

- ユーザ拡張性能解析トレース設定ファイルを読み込むときの文字エンコーディングは、JDK の動作プラットフォームのデフォルトエンコーディングです。文字エンコーディングがデフォルトエンコーディング以外の場合、デフォルトエンコーディングで記述されているものとして処理されます。
- サーバごとにユーザ拡張性能解析トレースを異なる設定にする場合は、サーバごとにユーザ拡張性能解析トレース設定ファイルを用意して、このプロパティでサーバごとのファイルパスを指定してください。
- 指定されたファイルパスのファイルが読み込めない場合や読み込んだ設定が有効でない場合、ユーザ拡張性能解析トレース設定ファイルの読み込み時のログが JavaVM ログファイルに出力されます。

指定例

プロパティの指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.File=/test/setting.txt
```

jvm.userprf.Limit

形式

jvm.userprf.Limit=<整数値>

説明

ユーザ拡張性能解析トレースの対象となるメソッド数の上限値を指定します。

デフォルト値

2147483647

前提プロパティ

jvm.userprf.Enable=true

引数

<整数値>

0～2147483647 までの整数値で指定します。それ以外を指定した場合はデフォルト値となります。

注意事項

ユーザ拡張性能解析トレース設定ファイルで指定したメソッドのうち、このプロパティで指定した値のメソッド数までがトレースの対象になります。指定したメソッドがこのプロパティで指定した値をはじめて超えたとき、ログメッセージが JavaVM ログファイルに出力されます。

指定例

プロパティの指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.Limit=100
```

jvm.userprf.Trace

形式

```
jvm.userprf.Trace={true | false}
```

説明

ユーザ拡張性能解析トレース設定ファイルで指定したクラスファイルの書き換えの成功ログを出力するかどうかを指定します。

デフォルト値

```
jvm.userprf.Trace=false
```

前提プロパティ

```
jvm.userprf.Enable=true
```

引数

true

ユーザ拡張性能解析トレースの対象クラスの書き換えの成功ログを出力します。

false

ユーザ拡張性能解析トレースの対象クラスの書き換えの成功ログを出力しません。

true, false 以外を指定した場合、デフォルト値となります。

注意事項

ユーザ拡張性能解析トレースの対象クラスの書き換えに失敗した場合は、このプロパティの指定に関係なく、ログメッセージがJavaVM ログファイルに出力されます。

指定例

プロパティの指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.Trace=true
```

jvm.userprf.LineNumber

形式

`jvm.userprf.LineNumber={true | false}`

説明

メソッドが正常に終了した場合、メソッドの最後に実行した行の行番号をトレース情報に出力するかどうかを指定します。

デフォルト値

`jvm.userprf.LineNumber=false`

前提プロパティ

`jvm.userprf.Enable=true`

引数

`true`

トレース情報のオペレーション情報（OPR）に、メソッドの最後に実行した行の行番号を出力します。

`false`

メソッドの最後に実行した行の行番号をトレース情報に出力しません。

`true`、`false` 以外を指定した場合、デフォルト値となります。

注意事項

JavaC コマンドの `-g` オプションで行番号のデバッグ情報を生成しない場合は、このプロパティに `true` を指定しても行番号は出力されません。

指定例

プロパティの指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.LineNumber=true
```

jvm.userprf.ThrowableName

形式

`jvm.userprf.ThrowableName={true | false}`

説明

メソッドが異常終了した場合、例外またはエラーのクラス名をトレース情報のオペレーション情報（OPR）に出力するかどうかを指定します。

デフォルト値

```
jvm.userprf.ThrowableName=false
```

前提プロパティ

```
jvm.userprf.Enable=true
```

引数

true

例外またはエラーのクラス名をトレース情報に出力します。

false

例外またはエラーのクラス名をトレース情報に出力しません。

true, false 以外を指定した場合、デフォルト値となります。

注意事項

このプロパティに true を指定した場合、jvm.userprf.ThrowableNameEditMethod プロパティに指定した編集方法でクラス名が出力されます。

指定例

プロパティの指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.ThrowableName=true
```

jvm.userprf.ThrowableNameEditMethod

形式

```
jvm.userprf.ThrowableNameEditMethod={FRONT_CUT | BACK_CUT | CENTER_CUT}
```

説明

例外またはエラーのクラス名が 32 文字を超える場合のクラス名の編集方法を指定します。例外またはエラーのクラス名が 32 文字以内であれば、このプロパティの指定に関係なく、すべての文字がトレース情報に出力されます。

デフォルト値

```
jvm.userprf.ThrowableNameEditMethod=FRONT_CUT
```

前提プロパティ

- `jvm.userprf.Enable=true`
- `jvm.userprf.ThrowableName=true`

引数

<文字列>

例外またはエラーのクラス名が 32 文字を超える場合のクラス名の編集方法を次の文字列で指定します。

FRONT_CUT

例外またはエラーのクラス名の後方 32 文字が出力されます。なお、文字の先頭には省略を示す「*」が付与されます。

BACK_CUT

例外またはエラーのクラス名の前方 32 文字が出力されます。なお、文字の末尾には省略を示す「*」が付与されます。

CENTER_CUT

例外またはエラーのクラス名の前方 16 文字と後方 16 文字が出力されます。なお、文字の前方 16 文字と後方 16 文字の間に、省略を示す「*」が付与されます。

上記以外の文字列を指定した場合、デフォルト値となります。

指定例

パッケージ名が「com.sample」、クラス名が「Class0123456789012345678901234567890」の例外で終了するとき、例外のクラス名または完全修飾クラス名の前方 32 文字をトレース情報のオペレーション情報（OPR）に出力する場合の指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.ThrowableNameEditMethod=BACK_CUT
```

`jvm.userprf.LogLevel` プロパティの指定によって、トレース情報のオペレーション情報（OPR）には次のように出力されます。[jvm.userprf.LogLevel](#) プロパティについては、「`jvm.userprf.LogLevel`」を参照してください。

class を指定した場合

```
Class012345678901234567890123456*
```

package または method または signature を指定した場合

```
com.sample.Class0123456789012345*
```

jvm.userprf.LogLevel

形式

jvm.userprf.LogLevel={class | package | method | signature}

説明

ユーザ拡張性能解析トレースの出力レベルを指定します。

デフォルト値

jvm.userprf.LogLevel=class

前提プロパティ

jvm.userprf.Enable=true

引数

<文字列>

ユーザ拡張性能解析トレースのトレース情報のオプション情報（OPT）に出力する，トレース対象のメソッド名の出力レベルを次の文字列で指定します。

class

クラス名

package

完全修飾クラス名

method

完全修飾クラス名+メソッド名

signature

完全修飾クラス名+メソッド名+メソッドの引数の型

上記以外の文字列を指定した場合，デフォルト値となります。

なお，jvm.userprf.ThrowableName プロパティに true を指定している場合，このプロパティの指定によってメソッドの異常出口で，次のようにオペレーション情報（OPR）が出力されます。

- class を指定した場合：例外またはエラーのクラス名
- package または method または signature を指定した場合：例外またはエラーの完全修飾クラス名

指定例

パッケージ名「com.sample」，クラス名「ClassA」，およびメソッド名「methodA」のトレース情報を，完全修飾クラス名+メソッド名でトレース情報のオプション情報（OPT）に出力する場合の指定例を次に示します。

```
add.jvm.arg= -Djvm.userprf.LogLevel=method
```

トレース情報には次のように出力されます。

- com.sample.ClassA.methodA

JP.co.Hitachi.soft.jvm.autofinalizer

形式

```
JP.co.Hitachi.soft.jvm.autofinalizer={true|false}
```

説明

ファイナライズ滞留解消機能の有効，無効を指定します。

true, false 以外の値が設定された場合，デフォルト値になります。

デフォルト値

```
JP.co.Hitachi.soft.jvm.autofinalizer=true
```

出力例

- ファイナライズ処理の滞留を検知して，ファイナライズ処理監視スレッドを新たに生成する場合

```
# FinalizerWatcherThread: Create: create secondary finalizer thread.  
[queue length = 128] <Mon May 26 18:00:36 JST 2008>
```

- 生成したファイナライズ処理監視スレッドが終了した場合

```
# FinalizerWatcherThread: Finish: secondary finalizer thread is finished.  
<Mon May 26 20:12:26 JST 2008>
```

sun.nio.cs.map

形式

```
sun.nio.cs.map=Windows-31J/Shift_JIS
```

説明

エンコーディング名 shift_jis, csshiftjis, ms_kanji, x-sjis を MS932 と同等とするかどうかを指定します。

なお，JDK17 以降の場合は，このプロパティを使用できません。

文字エンコーディングの別名である shift_jis, csshiftjis, ms_kanji, x-sjis は、JDK1.4.0 では MS932 の別名でしたが、JDK1.4.2 以降では SJIS の別名に変更になりました。今まで MS932 の別名として shift_jis, csshiftjis, ms_kanji, x-sjis を使用していたプログラムをそのままこのバージョンで実行すると文字化けすることがあります。

shift_jis, csshiftjis, ms_kanji, x-sjis を以前のバージョンと同じく、MS932 の別名として実行させるためには、sun.nio.cs.map プロパティに Windows-31J/Shift_JIS を指定してください。

(例)

```
import java.io.*;
class encode_Shift_JIS {
    public static void main( String arg[] ) {
        try {
            String string_data = "-①~";
            byte[] data = string_data.getBytes();
            InputStreamReader isr =
                new InputStreamReader(
                    new ByteArrayInputStream( data ), "shift_jis");
            char[] read_data = new char[6];
            isr.read( read_data, 0, 6 );
            System.out.println(new String(read_data));
        }
        catch ( Exception e ) {
            e.printStackTrace();
        }
    }
}
```

JDK1.4.0 では、shift_jis は MS932 の別名であったため、上記のプログラムは文字化けしませんでした。JDK1.4.2 のバージョンで実行すると次のように文字化けします。

```
java encode_Shift_JIS
??@?
```

文字化けを回避するためには、sun.nio.cs.map プロパティに Windows-31J/Shift_JIS を指定して実行してください。

指定例

J2EE サーバで設定する場合は usrconf.cfg ファイルに次の指定をしてください。

```
add.jvm.arg=-Dsun.nio.cs.map=Windows-31J/Shift_JIS
```

J2EE サーバなど各環境での usrconf.cfg ファイルの記載方法の詳細については「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」を参照してください。

14.4 Application Server で指定できる Java HotSpot VM のオプションのデフォルト値

Java HotSpot VM のオプションのデフォルト値を示します。デフォルト値は、OS ごとに異なります。

■

ポイント

ここで説明するデフォルト値は、Java HotSpot VM で定義している初期値になります。

実際に J2EE サーバ、バッチサーバ、Java アプリケーションを起動した場合は、この表で示したデフォルト値ではなく、各オプション定義ファイルの `add.jvm.arg` キーのデフォルト値が有効になるキーがあります。

各オプション定義ファイルのデフォルト値については、「[2.2.2 usrconf.cfg \(J2EE サーバ用オプション定義ファイル\)](#)」, または「[3.2.1 usrconf.cfg \(バッチサーバ用オプション定義ファイル\)](#)」, または「[12.2.1 usrconf.cfg \(Java アプリケーション用オプション定義ファイル\)](#)」を参照してください。

Java HotSpot VM のオプションのデフォルト値を OS ごとに次に示します。Application Server で指定できる Java HotSpot VM のオプションについては、「[14.5 Application Server で指定できる Java HotSpot VM のオプション](#)」を参照してください。

表 14-10 Java HotSpot VM のオプションのデフォルト値

項番	オプション名	使用する OS と JavaVM の種類		
		Windows	Linux	AIX
1	-Xmx<size>*1	83M	83M	83M
2	-Xms<size>*1	7.8M	7.8M	7.8M
3	-XX:MaxMetaspaceSize=<size>*1	2 ⁶⁴ -1	2 ⁶⁴ -1	2 ⁶⁴ -1
4	-XX:MetaspaceSize=<size>*1	16M	16M	16M
5	-XX:CompressedClassSpaceSize=<size>*1	1G	1G	1G
6	-Xmn<size>	*3	*3	*3
7	-Xss<size>	1M	1M	1M
8	-XX:NewRatio=<value>*1	2	2	2
9	-XX:SurvivorRatio=<value>*2	8	32	8
10	-XX:TargetSurvivorRatio=<value>	50	50	50

項番	オプション名	使用する OS と JavaVM の種類		
		Windows	Linux	AIX
11	-XX:MaxTenuringThreshold=<value>	14	14	14
12	-XX:ReservedCodeCacheSize=<size>	48M	48M	48M

注

<size>の単位はバイトです。

注※1

J2EE サーバの場合、J2EE サーバ用オプション定義ファイル（usrconf.cfg）の add.jvm.arg キーのデフォルト値が有効になります。add.jvm.arg キーで定義されていないオプションのデフォルト値はこの表で示した値となります。

注※2

J2EE サーバモードで実行した J2EE サーバでは、JavaVM のデフォルト値は上書きされます。上書き後のデフォルト値は-XX:SurvivorRatio=8 となります。これは、すべてのプラットフォームに共通です。

注※3

Java ヒープと-XX:NewRatio によって決まります。

14.5 Application Server で指定できる Java HotSpot VM のオプション

ここでは、add.jvm.arg に指定できる Java HotSpot VM のオプション、およびオプションを指定する際の注意事項について説明します。

add.jvm.arg に指定できる Java HotSpot VM のオプションについて次の表に示します。なお、「VR」とは、パラメタが導入・変更されたアプリケーションサーバのバージョンです。

表 14-11 add.jvm.arg を指定した場合に指定できる Java HotSpot VM のオプション

オプション名	内容	指定可能値	VR
-D<property>	JavaVM のシステムプロパティを指定します。	入力には制限されません。	05-00
-agentlib:<libname>[=<options>]	ネイティブエージェントライブラリ<libname>をロードします。	入力には制限されません。	08-00
-verbose:<情報種別>	<情報種別>に指定した情報を出力します。<情報種別>に指定できる値を示します。 class : クラスがロードされるたびにクラスに関する情報を出力します。 gc : GC イベントが発生するたびに報告します。 jni : ネイティブメソッドの使用およびその他の Java Native Interface (JNI) アクティビティに関する情報を報告します。	指定できる文字列を次に示します。 • class • gc • jni	05-00
-Xloggc:<ファイル>	-verbose:gc と同様に GC イベントが発生するたびに報告しますが、そのデータを<ファイル>に記録します。-verbose:gc を指定したときに報告される情報のほかに、報告される各イベントの先頭に、最初の GC イベントからの経過時間（秒単位）が付け加えられます。	入力には制限されません。	05-00
-Xms<size>	Java ヒープの初期サイズを設定します。	自然数の値を次に示す単位を使って指定します。 • キロ [k] • メガ [m] • ギガ [g] • テラ [t] なお、大文字・小文字は区別されません。	05-00
-Xmx<size>	Java ヒープの最大サイズを設定します。	自然数の値を次に示す単位を使って指定します。 • キロ [k] • メガ [m]	05-00

オプション名	内容	指定可能値	VR
		- ギガ [g] - テラ [t] なお、大文字・小文字は区別されません。	
-Xmn<size>	New 領域の初期値および最大値を設定します。	自然数の値を次に示す単位を使って指定します。 - キロ [k] - メガ [m] - ギガ [g] - テラ [t] なお、大文字・小文字は区別されません。	05-00
-Xss<size>	1 スタック領域の最大サイズを設定します。	自然数の値を次に示す単位を使って指定します。 - キロ [k] - メガ [m] - ギガ [g] - テラ [t] なお、大文字・小文字は区別されません。	05-00
-Xprof	このオプションを指定した場合、実行中のプログラムのプロファイルを生成し、プロファイリングデータを標準出力に出力します。このオプションは、プログラム開発用のユーティリティとして提供されています。本番稼働システムでの使用を目的としたものではありません。	—	05-00
-Xrunhprof[:<suboption>=<value>,...]	CPU、ヒープ、またはモニタのプロファイリングを有効にします。-Xrunhprof の後ろにコロン「:」を指定して「<suboption>=<value>」を記述します。「<suboption>=<value>」はコンマ「,」で区切って複数指定できます。 サブオプションとそのデフォルト値のリストを取得するには、コマンド java-Xrunhprof:help を実行します。	任意の文字列を指定します。 <suboption>には「=」および「,」は指定できません。 また、<value>には「,」は指定できません。	05-00
-Xrun<libraryName>	JVMPI, JVMDI などのエージェントを含むライブラリと、起動時に渡されるオプションの文字列を指定します。	入力は制限されません。	05-00
-Xdebug	指定した場合、JVMDI のサポートを有効にして開始します。JVMDI は推奨されていません。また、J2SE 5.0 でのデバッグには使用されません。したがって、J2SE 5.0 でのデバッグには、このオプションは必要ありません。	—	05-00

オプション名	内容	指定可能値	VR
-XX:NewRatio=<value>	New 領域に対する Tenured 領域の割合を指定します。 <value>が 2 の場合は、New 領域と Tenured 領域の割合が、1:2 になります。 「New 領域の使用サイズ ≥ Tenured 領域の空き領域サイズ」になると FullGC が発生します。このオプションに 1 を設定すると、FullGC が多発するので注意してください。	自然数の値を指定します。	05-00
-XX:MetaspaceSize=<size>	Metaspace 領域の初期サイズを指定します。	自然数の値を次に示す単位を使って指定します。 • キロ [k] • メガ [m] • ギガ [g] • テラ [t] なお、大文字・小文字は区別されません。	09-70
-XX:MaxMetaspaceSize=<size>	Metaspace 領域の最大サイズを指定します。	自然数の値を次に示す単位を使って指定します。 • キロ [k] • メガ [m] • ギガ [g] • テラ [t] なお、大文字・小文字は区別されません。	09-70
-XX:CompressedClassSpaceSize	Compressed Class Space の最大サイズを指定します。	自然数の値を次に示す単位を使って指定します。 • キロ [k] • メガ [m] • ギガ [g] なお、大文字・小文字は区別されません。	09-70
-XX:SurvivorRatio=<value>	New::Survivor 領域の From 空間と To 空間に対する New::Eden 領域の割合を指定します。 <value>に 8 を設定した場合は、New::Eden 領域、From 空間、To 空間の割合が、8:1:1 になります。	自然数の値を指定します。	05-00
-XX:TargetSurvivorRatio=<value>	GC 実行後の New::Survivor 領域内で Java オブジェクトが占める割合の目標値 (0~100 (単位: %)) を指定します。	自然数の値を指定します。	05-00
-XX:MaxTenuringThreshold=<value>	CopyGC 実行時に、From 空間と To 空間で Java オブジェクトを入れ替える回数のしきい値を指定します。指定した回数を超えて入れ替え対象になった Java オブジェクトは、Tenured 領域に移動されます。	自然数の値を指定します。	05-00

オプション名	内容	指定可能値	VR
	このオプションの有効範囲は、0～デフォルト値です。範囲外の値を指定した場合、しきい値を超えた場合に Tenured 領域へ移動する機能は無効になります。		
-Xrunhndlwrap	JDK のバージョンが 5 以下の環境で Windows をログオフしたとき、JavaVM の実行を継続するかどうか設定します。 -XX:+EagerXrunInit を指定した場合、このオプションの指定は無効になります。また、-agentlib:hndlwrap2 を設定した場合、このオプションは設定しないでください。 このオプションは、ほかの JVMTI 版プログラムと同時に使用できません。 コマンドプロンプト上でこのオプションを使用した Java を実行している状態でログオフすると、「プログラムが反応しない」旨のエラーポップアップが表示され、ログオフできません。	—	05-01
-agentlib:hndlwrap2	JDK のバージョンが 6 の環境で Windows をログオフしたとき、JavaVM の実行を継続するかどうか設定します。 -Xrunhndlwrap を設定した場合、このオプションは設定しないでください。 このオプションは、ほかの JVMTI プログラムと同時に使用できません。 コマンドプロンプト上でこのオプションを使用した Java を実行している状態でログオフすると、「プログラムが反応しない」旨のエラーポップアップが表示され、ログオフできません。	—	08-50
-XX:[+ -]UseSerialGC ※	-XX:+UseSerialGC SerialGC を実行します。 -XX:-UseSerialGC SerialGC を実行しません。	指定できる文字を次に示します。 • プラス「+」 • マイナス「-」	08-70
-XX:[+ -]UseG1GC	-XX:+UseG1GC G1GC を実行します。 -XX:-UseG1GC G1GC を実行しません。 -XX:+HitachiUseExplicitMemory オプションとは同時に指定できません。指定した場合、メッセージが標準出力に出力され、JavaVM がリターンコード 1 で終了します。	指定できる文字を次に示します。 • プラス「+」 • マイナス「-」	09-60
-XX:[+ -]UseZGC	-XX:+UseZGC ZGC を実行します。	指定できる文字を次に示します。	11-40

オプション名	内容	指定可能値	VR
	-XX:-UseZGC ZGC を実行しません。 このオプションを-XX:+UseSerialGC または-XX:+UseG1GC と同時に指定した場合は、エラーメッセージを標準出力に出力して、コマンドの起動に失敗します。	• プラス「+」 • マイナス「-」	
-XX:ParallelGCThreads=<threads> (G1GC の場合)	G1GC の Evacuation を並列実行するスレッドの数を指定します。-XX:+UseG1GC を指定したときに有効になります。なお、このオプションは、デフォルト値で使用することを推奨します。変更する場合は、実際に JavaVM を動作させて測定した値を基に、最適な値を算出してから変更してください。 JavaVM 起動時に、このオプションに指定した数の Evacuation 処理用のスレッドが作成され、処理が実行されます。 このオプションの指定を省略した場合のデフォルト値は次のとおりです。 • 実行環境の論理 CPU 数が 8 以下の場合は、CPU 数になります。 • 実行環境の論理 CPU 数が 9 以上の場合は、「8 + (CPU 数 - 8) × (5 ÷ 8)」(小数点以下は切り捨て)の値になります。 例えば、4CPU の場合は、「ParallelGCThreads = 4」でデフォルト値は 4 になります。また、72CPU の場合は、「ParallelGCThreads = 8 + (72 - 8) × (5 ÷ 8) = 48」で 48 になります。 このオプションに 0 を指定した場合は、デフォルト値が使用されます。 指定時には、次の点に注意してください。 • このオプションの値を大きくすると、Evacuation 用のスレッド数が増え、Evacuation に割り当てられるリソースが増加します。そのため、値を大きくするとスループットが低下する可能性があります。このオプションの値を変更した場合、性能の要件を満たしているか確認をしてください。 • 実行環境で作成できるスレッド数を超える値を指定した場合は、JavaVM 起動時に Evacuation 処理用のスレッドの作成に失敗するため、JavaVM を起動できません。	自然数の値を指定します。	11-40
-XX:ParallelGCThreads=<threads> (ZGC の場合)	ZGC サイクルの STW フェーズでの GC スレッド数を指定します。このオプションと-XX:ConcGCThreads オプションに指定した値のうち、大きい方の個数のスレッドが、JavaVM を起動したときに作成されます。	1～1000 の自然数の値を指定します。	11-40

オプション名	内容	指定可能値	VR
	このオプションの指定を省略した場合、実行環境の論理 CPU 数の 60%がデフォルト値として設定されます。ただし、小数点以下の値は切り上げられます。なお、次の場合は、JavaVM を起動できません。 - 自然数以外の値を指定した場合 標準エラー出力にメッセージを出力します。 -1 以下または 1001 以上の数値を指定した場合 標準出力および標準エラー出力にメッセージを出力します。 0 を指定した場合 標準出力にメッセージを出力します。 - 実行環境で作成できるスレッド数以上の値を指定した場合 出力されるメッセージについては、マニュアル「アプリケーションサーバ 機能解説 保守／移行編」の「5.8 JavaVM が出力するメッセージログ（標準出力およびエラーリポートファイル）」を参照してください。		
-XX:ConcGCThreads=<threads> (G1GC の場合)	G1GC の Concurrent Marking の処理を並列実行するスレッドの数を指定します。 -XX:+UseG1GC を指定したときに有効になります。なお、このオプションは、デフォルト値を使用することを推奨します。変更する場合は、実際に JavaVM を動作させて測定した値を基に、最適な値を算出してから変更してください。 JavaVM 起動時に、このオプションに指定した数の Concurrent Marking 処理用のスレッドが作成され、処理が実行されます。 このオプションの指定を省略した場合のデフォルト値は次のとおりです。 $\text{Max}(\text{ParallelGCThreads} + 2)/4, 1)$ $\text{Max}(A,B)$は A と B のうち、大きい値を取ることを意味します。また、ParallelGCThreads は -XX:ParallelGCThreads の値を意味します。 このオプションに 0 を指定した場合はデフォルト値が使用されます。 指定時には次の点に注意して下さい。 - 実行環境で作成できるスレッド数を超える値を指定した場合は、JavaVM 起動時に Concurrent Marking 処理用のスレッドの作成に失敗するため、JavaVM を起動できません。 - Concurrent Marking 処理用スレッドのスレッド数は Evacuation 処理用のスレッド数を超えて作成することはできません。Evacuation 処理用のスレッド数を超えて指定した場合、メッセージが	自然数の値を指定します。	11-40

オプション名	内容	指定可能値	VR
	標準出力に出力され JavaVM の起動に失敗します。		
-XX:ConcGCThreads= threads> (ZGC の 場合)	ZGC サイクルの M/R, RP, EC および RE フェーズでの GC スレッド数の最大値を指定します。このオプションと-XX:ParallelGCThreads オプションに指定した値のうち、大きい方の個数のスレッドが、JavaVM を起動したときに作成されます。 このオプションの指定を省略した場合、実行環境の論理 CPU 数の 25%がデフォルト値として設定されます。ただし、小数点以下の値は切り上げられます。なお、次の場合は、JavaVM を起動できません。 - 自然数以外の値を指定した場合 標準エラー出力にメッセージを出力します。 -1 以下または 1001 以上の数値を指定した場合 標準出力および標準エラー出力にメッセージを出力します。 0 を指定した場合 標準出力にメッセージを出力します。 - 実行環境で作成できるスレッド数以上の値を指定した場合 出力されるメッセージについては、マニュアル「アプリケーションサーバ 機能解説 保守／移行編」の「5.8 JavaVM が出力するメッセージログ (標準出力およびエラーリポートファイル)」を参照してください。	1~1000 の自然数の値を指定します。	11-40
-XX:MaxGCPauseMilli s	G1GC の GC によるアプリケーション停止時間の目標時間を [ms]の単位で指定できます。 -XX:+UseG1GC を指定したときに有効になります。 このオプションに 0 を指定した場合は、メッセージが標準出力に出力され JavaVM の起動に失敗します。 このオプションに 100 以下の値を指定する場合、ログファイルの非同期出力機能 (-XX:+JavaLogAsynchronous) を使用することを推奨します。	自然数の値を指定します。	09-60
-XX:ReservedCodeCa cheSize	コードキャッシュ領域の最大サイズを指定します。	自然数の値を次に示す単位を使って指定します。 - キロ [k] - メガ [m] - ギガ [g] なお、大文字・小文字は区別されません。	09-50
-XX:ZAllocationSpike Tolerance=<factor>	ZGC を使用している場合に、最大割り当て率と空きメモリ量から OOM が発生するまでの時間を計算するときに、割り当て率に掛ける数値を指定します。	0 以上の数値を指定します。	11-40

オプション名	内容	指定可能値	VR
	<factor>の数値を掛けて割り当て率を大きく見積もることで、割り当て率が急に増えた場合でも、GC を実行して OOM を防ぐことができます。 このオプションの指定を省略した場合、1.0 がデフォルト値として設定されます。		
-XX:ZCollectionInterval=<seconds>	ZGC を使用している場合に、GC 発生要因「ZTimer」のインターバルの秒数を指定します。 このオプションの指定を省略した場合、0 がデフォルト値として設定されます。	自然数の値を指定します。	11-40
-XX:ZFragmentationLimit=<percent>	ZGC を使用している場合に、ページを回収するかどうかの基準となる使用済みのオブジェクトの割合を指定します。ページにある使用済みのオブジェクトの割合が<percent>以上の場合、ZGC サイクルの EC フェーズで、回収するページとして選択されます。 このオプションの指定を省略した場合、25 がデフォルト値として設定されます。	0~100 の整数を指定します。	11-40
-XX:[+ -]ZProactive	-XX:+ZProactive GC 発生要因「ZProactive」による GC が発生します。 -XX:-ZProactive GC 発生要因「ZProactive」による GC が発生しません。 ZGC を使用している場合に、GC 発生要因「ZProactive」での GC の発生を有効にするかどうかを指定します。アプリケーションが頻繁に実行されていないとき、または割り当てられるオブジェクトが少ないときに、前もって GC を発生させることで、Java ヒープ領域の空き容量を確保できます。 このオプションの指定を省略した場合、ON がデフォルト値として設定されます。	指定できる文字を次に示します。 • プラス「+」 • マイナス「-」	11-40
-XX:[+ -]ZUncommit	-XX:+ZUncommit 未使用メモリを OS に返します。 -XX:-ZUncommit 未使用メモリを OS に返しません。 ZGC を使用している場合に、未使用メモリを OS に返すかどうかを指定します。未使用メモリを OS に返すときは、ヒープサイズが最小ヒープサイズを下回らないように設定されています。 なお、-XX:[+ -]ZUncommit オプションを使用する場合は、-Xms オプションの値よりも大きい値を、-Xmx オプションに指定してください。-Xms オプションと-Xmx オプションに同じ値を指定すると、-XX:[+ -]ZUncommit オプションは無効になります。	指定できる文字を次に示します。 • プラス「+」 • マイナス「-」	11-40

オプション名	内容	指定可能値	VR
	このオプションの指定を省略した場合、ON がデフォルト値として設定されます。		
-XX:ZUncommitDelay= =<seconds>	ZGC を使用している場合に、未使用メモリを OS に返す判断をする時間の長さを指定します。 なお、-XX:ZUncommitDelay=<seconds> オプションを使用する場合は、-Xms オプションの値よりも大きい値を、-Xmx オプションに指定してください。-Xms オプションと-Xmx オプションに同じ値を指定すると、-XX:ZUncommitDelay=<seconds> オプションは無効になります。 このオプションの指定を省略した場合、300 がデフォルト値として設定されます。	自然数の値を指定します。	11-40
-XX:MaxDirectMemorySize= =<size>	ダイレクトバッファの最大サイズ（バイト単位）を設定します。 このオプションのデフォルト値は、Java ヒープ領域の最大値と同じ値です。そのため、ダイレクトバッファを大量に使用する場合は、このオプションの値を大きくしてください。 ダイレクトバッファの最大サイズを見積もるときは、日立 JavaVM ログファイルにあるダイレクトバッファ情報出力機能のログから、ダイレクトバッファの使用サイズを確認してください。なお、日立 JavaVM ログファイルにダイレクトバッファ情報を出力するためには、- XX:+HitachiVerboseGCPrintDirectBuffer オプションを指定してください。詳細は、「14.2 JavaVM 拡張オプションの詳細」の「-XX:[+ -]HitachiVerboseGCPrintDirectBuffer（ダイレクトバッファの使用状況出力オプション）」を参照してください。	自然数の値を次に示す単位を使って指定します。 • キロ [k] • メガ [m] • ギガ [g] • テラ [t] なお、大文字・小文字は区別されません。	11-40
-XX: [+ -]AlwaysPreTouch	-XX:+AlwaysPreTouch JVM 初期化時に Java ヒープ上の全メモリへのアクセスを有効にします。 -XX:-AlwaysPreTouch JVM 初期化時に Java ヒープ上の全メモリへのアクセスを無効にします。 JVM 初期化時に Java ヒープ上の全メモリへのアクセスを有効にするかどうかを指定します。 このオプションの指定を省略した場合、OFF がデフォルト値として設定されます。	指定できる文字を次に示します。 • プラス「+」 • マイナス「-」	11-60
-XX:SoftMaxHeapSize= =<size>	Java ヒープが拡大できる大きさにソフト制限を設定します。ZGC はこのヒープサイズをできるだけ超えないようにしますが、拡大はできます。	0 以上の整数を指定します。	11-60

オプション名	内容	指定可能値	VR
	このオプションの指定を省略した場合、MaxHeapSize の値がデフォルト値として設定されます。		

(凡例)

－：該当なし

注※

-XX:[+|-]UseSerialGC オプションと-XX:[+|-]UseG1GC オプションの指定の組み合わせごとに実行される処理を次に示します。

表 14-12 -XX:[+|-]UseSerialGC オプションと-XX:[+|-]UseG1GC オプションの指定の組み合わせごとの処理

-XX:[+ -]UseSerialGC オプションの指定	-XX:[+ -]UseG1GC オプションの指定	実行される処理
指定なし	指定なし	SerialGC を実行 (JDK11 以前の場合) G1GC の実行 (JDK17 以降の場合 (ただし、論理プロセッサが 2 未満または物理メモリが約 1792 メガバイト未満の場合は、SerialGC が実行されます))
-XX:+UseSerialGC	-XX:+UseG1GC	プロセスの起動に失敗
-XX:+UseSerialGC	-XX:-UseG1GC	SerialGC を実行 (デフォルトと同様)
-XX:-UseSerialGC	-XX:-UseG1GC	プロセスの起動に失敗
-XX:-UseSerialGC	-XX:+UseG1GC	G1GC を実行

注意事項

- デバッグまたはプロファイリング用オプションについて
JavaVM オプションの-Xprof や-Xdebug, JVMTI エージェントの hprof や jdwp (-agentlib:<libname>などで指定) は、プログラムの開発用ユーティリティとして提供されているものです。システムの運用では指定しないようにしてください。
- GC の指定について
アプリケーションサーバの GC は、SerialGC (UseSerialGC) と G1GC (UseG1GC) を選択できます。この二つの GC を同時に指定しないようにしてください。同時に指定した場合は、Java プロセスが起動できません。

ポイント

Java HotSpot VM のオプションのデフォルト値は、サーバごとに参照先が異なります。サーバ、およびアプリケーションごとの Java HotSpot VM のオプションのデフォルト値の参照先を次の表に示します。

表 14-13 サーバ、およびアプリケーションごとの Java HotSpot VM のオプションのデフォルト値の参照先

使用するサーバおよびアプリケーション	参照先
J2EE サーバ	2.2.2 usrconf.cfg (J2EE サーバ用オプション定義ファイル)
バッチサーバ	3.2.1 usrconf.cfg (バッチサーバ用オプション定義ファイル)
cjclstartap コマンドで起動する Java アプリケーション	12.2.1 usrconf.cfg (Java アプリケーション用オプション定義ファイル)
その他の Java アプリケーション	14.4 Application Server で指定できる Java HotSpot VM のオプションのデフォルト値

なお、Java HotSpot VM のオプションを簡易構築定義ファイルに指定する場合は、「param-name 指定値」に add.jvm.arg を指定した場合に指定できる「param-value の指定内容」に Java HotSpot VM のオプションを指定してください。

14.6 JavaVM で使用する環境変数の一覧

JavaVM で使用する環境変数の一覧を，次の表に示します。

「VR」とは，環境変数が導入・変更されたアプリケーションサーバのバージョンです。

表 14-14 JavaVM で使用する環境変数の一覧

分類	環境変数	説明	VR
JavaVM で使用する環境変数	JAVACOREDIR	スレッドダンプファイルの出力先ディレクトリを指定します。	Windows の場合 05-05 上記以外の OS の場合 05-00

14.7 JavaVM で使用する環境変数の詳細

JavaVM で使用する環境変数の詳細について説明します。

JAVACOREDIR

形式

Windows の場合（バッチファイル）

```
set JAVACOREDIR=<スレッドダンプファイルの出力先ディレクトリ>
```

UNIX の場合（C シェル）

```
setenv JAVACOREDIR <スレッドダンプファイルの出力先ディレクトリ>
```

説明

スレッドダンプファイルを出力する場合の、出力先ディレクトリを指定します。

デフォルト値

デフォルトでは JAVACOREDIR は未設定です。

スレッドダンプ出力先

デフォルトでは、スレッドダンプファイルはカレントディレクトリに出力されます。
カレントディレクトリを次に示します。

Windows の場合

<作業ディレクトリ>%ejb%<サーバ名称>

UNIX の場合

<作業ディレクトリ>/ejb/<サーバ名称>

前提オプション

-XX:+HitachiThreadDump

指定例

Windows の場合（バッチファイル）

スレッドダンプを C:%home%user%threaddump に出力します。

```
set JAVACOREDIR=C:%home%user%threaddump
```

UNIX の場合（C シェル）

スレッドダンプを/home/user/threaddump に出力します。

```
setenv JAVACOREDIR /home/user/threaddump
```

注意事項

- JAVACOREDIR で指定したディレクトリへのスレッドダンプファイルの出力に失敗した場合は、カレントディレクトリにスレッドダンプファイルを出力します。
- カレントディレクトリへの出力も失敗した場合は、標準エラー出力だけにスレッドダンプを出力します。この場合は、標準出力にはスレッドダンプは出力されません。

付録

付録 A 拡張 MIB オブジェクト定義ファイル

拡張 MIB オブジェクト定義ファイルは、Management Server が出力する稼働性能などの情報を、MIB オブジェクトとして出力するための定義ファイルです。MIB オブジェクトは、SNMP Manager 製品と連携する場合に使用します。

付録 A.1 拡張 MIB オブジェクト定義ファイルのテーブル

拡張 MIB オブジェクト定義ファイルのテーブルについて説明します。

各テーブルの OID を次に示します。

.iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).hitachi(116).systemExMib(5).cosminexus Mibs(33).manager(1)

(1) j2eeContainerTable

j2eeContainerTable を次の表に示します。

表 A-1 j2eeContainerTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
j2eeContainerTable (1)	j2eeContainerEntry (1)	j2eeContainerIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	なし
		j2eeContainerFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>"	
		j2eeContainerServerName (3)	DisplayString	RO	—	J2EE サーバのサーバ名	
		j2eeContainerNamingServiceHost (4)	DisplayString	RO	—	J2EE サーバの接続するネーミングサービスのホスト名	
		j2eeContainerNamingServicePort (5)	INTEGER	RO	—	J2EE サーバの接続するネーミングサービスのポート番号	
		j2eeContainerStartTime (6)	INTEGER	RO	秒	J2EE サーバの起動時刻※	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

注※

00:00 Jan. 1, 1970 UTC からの経過秒数

(2) jvmTable

jvmTable を次の表に示します。

表 A-2 jvmTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/ W	単位	説明	前提条件
jvmTable (2)	jvmEntry (1)	jvmIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	なし
		jvmFullyQualified Name (2)	DisplaySt ring	RO	－	インスタンスを一意に識別 する完全修飾名 "<論理サーバ名>"	
		jvmName (3)	DisplaySt ring	RO	－	JavaVM の実装名	
		jvmVersion (4)	DisplaySt ring	RO	－	JavaVM の実装バージョン	
		jvmCurrentFreeM emory (5)	Gauge	RO	KB	JavaVM の空きメモリ所要 量（現在値）	
		jvmFreeMemoryH ighWaterMark (6)	Gauge	RO	KB	JavaVM の空きメモリ所要 量（最大値）	
		jvmFreeMemoryL owWaterMark (7)	Gauge	RO	KB	JavaVM の空きメモリ所要 量（最小値）	
		jvmFreeMemoryA verag (8)	Gauge	RO	KB	JavaVM の空きメモリ所要 量（平均値※ ¹ ）	
		jvmFreeMemoryP eak (9)	Gauge	RO	KB	JavaVM の空きメモリ所要 量（ピーク値※ ² ）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/ W	単位	説明	前提条件
		jvmCurrentTotal Memory (10)	Gauge	RO	KB	JavaVM の総メモリ所要量 (使用メモリ所要量と空きメ モリ所要量の総和)の現在値	
		jvmTotalMemory HighWaterMark (11)	Gauge	RO	KB	JavaVM の総メモリ所要量 (使用メモリ所要量と空きメ モリ所要量の総和)の最大値	
		jvmTotalMemory LowWaterMark (12)	Gauge	RO	KB	JavaVM の総メモリ所要量 (使用メモリ所要量と空きメ モリ所要量の総和)の最小値	
		jvmTotalMemory Average (13)	Gauge	RO	KB	JavaVM の総メモリ所要量 (使用メモリ所要量と空きメ モリ所要量の総和)の平均値 ※1	
		jvmTotalMemory Peak (14)	Gauge	RO	KB	JavaVM の総メモリ所要量 (使用メモリ所要量と空きメ モリ所要量の総和)のピー ク値※2	
		jvmMaxMemory (15)	INTEGER	RO	KB	JavaVM の使用する最大メ モリ所要量 指定されていない場合は, $(2^{63}-1)/2^{10}$	
		jvmStatisticsStarte dTime (16)	INTEGER	RO	秒	統計開始時刻※3	
		jvmSamplingTime (17)	INTEGER	RW	秒	統計のサンプリング時間※4	

(凡例)

R/W : Read-Only / Read-Write 種別

— : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

値の取得要求時点でのピーク値

このピーク値は、記録されてからサンプリング時間の有効期間を持つ最大値のことを示します。

注※3

00:00 Jan. 1, 1970 UTC からの経過秒数

注※4

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(3) transactionTable

transactionTable を次の表に示します。

表 A-3 transactionTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
transactionTable (3)	transactionEntry (1)	transactionIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	トランザクションの開始
		transactionFullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>"	
		transactionDefaultTransactionTimeout (3)	INTEGER	RO	秒	トランザクションタイムアウトのデフォルト値	
		transactionActiveTransactionCount (4)	Gauge	RO	－	アクティブトランザクション数（現在値）	
		transactionAverageTime (5)	Gauge	RO	ミリ秒	トランザクション時間（平均値※1）	
		transactionStatisticsStartTime (6)	INTEGER	RO	秒	統計開始時刻※2	
		transactionSamplingTime (7)	INTEGER	RW	秒	統計のサンプリング時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(4) httpServerTable

httpServerTable を次の表に示します。

表 A-4 httpServerTable

テーブル 名称 (OID)	エントリ名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
httpServer Table (4)	httpServerE ntry (1)	httpServerIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	なし
		httpServerFullyQual ifiedName (2)	DisplayStr ing	RO	—	インスタンスを一意に識 別する完全修飾名 "<論理サーバ名>"	
		httpServerPortNum ber (3)	INTEGER	RO	—	管理用サーバのポート 番号	
		httpServerActiveThr eadCountHighWate rMark (4)	Gauge	RO	—	リクエスト処理中のス レッド数（最大値）	
		httpServerActiveThr eadCountLowWate rMark (5)	Gauge	RO	—	リクエスト処理中のス レッド数（最小値）	
		httpServerActiveThr eadCountAverage (6)	Gauge	RO	—	リクエスト処理中のス レッド数（平均値※ ¹ ）	
		httpServerActiveThr eadCountPeak (7)	Gauge	RO	—	リクエスト処理中のス レッド数（ピーク値※ ² ）	
		httpServerStatisticsS tartedTime (8)	INTEGER	RO	秒	統計開始時刻※ ³	

テーブル 名称 (OID)	エントリ名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		httpServerSampling Time (9)	INTEGER	RW	秒	統計のサンプリング時間 ※4	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

値の取得要求時点でのピーク値

このピーク値は、記録されてからサンプリング時間の有効期間を持つ最大値のことを示します。

注※3

00:00 Jan. 1, 1970 UTC からの経過秒数

注※4

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(5) httpServerConnectorTable

httpServerConnectorTable を次の表に示します。

表 A-5 httpServerConnectorTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
httpServer Connector Table (5)	httpServer Connector Entry (1)	httpServerConnect orIndex (1)	INTEGE R	RO	－	シーケンス番号[1, …]	Web サーバ連 携機能の使用
		httpServerConnect orFullyQualifiedNa me (2)	DisplaySt ring	RO	－	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名>"	
		httpServerConnect orPortNumber (3)	INTEGE R	RO	－	Web サーバとの通信 に使用するポート	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		httpServerConnect orBackLogUpperB ound (4)	INTEGE R	RO	－	Web サーバとの通信 ソケットのバックログ	
		httpServerConnect orConnectionCoun t (5)	Gauge	RO	－	Web サーバと Web コンテナとのコネク ション数（現在値）	
		httpServerConnect orConnectionCoun tHighWaterMark (6)	Gauge	RO	－	Web サーバと Web コンテナとのコネク ション数（最大値）	
		httpServerConnect orConnectionCoun tLowWaterMark (7)	Gauge	RO	－	Web サーバと Web コンテナとのコネク ション数（最小値）	
		httpServerConnect orActiveThreadCo untUpperBound (8)	Gauge	RO	－	リクエスト処理中のス レッド数（上限値）	
		httpServerConnect orActiveThreadCo untHighWaterMar k (9)	Gauge	RO	－	リクエスト処理中のス レッド数（最大値）	
		httpServerConnect orActiveThreadCo untLowWaterMark (10)	Gauge	RO	－	リクエスト処理中のス レッド数（最小値）	
		httpServerConnect orActiveThreadCo untAverage (11)	Gauge	RO	－	リクエスト処理中のス レッド数（平均値※ ¹ ）	
		httpServerConnect orActiveThreadCo untPeak (12)	Gauge	RO	－	リクエスト処理中のス レッド数（ピーク値※ ² ）	
		httpServerConnect orWaitingRequest	Gauge	RO	－	稼働スレッド数の上限 値の設定によって、実	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		CountHighWaterMark (13)				行を待たされているリクエスト数（最大値）	
		httpServerConnectorWaitingRequestCountLowWaterMark (14)	Gauge	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数（最小値）	
		httpServerConnectorWaitingRequestCountAverage (15)	Gauge	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数（平均値※1)	
		httpServerConnectorWaitingRequestCountPeak (16)	Gauge	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数（ピーク値※2)	
		httpServerConnectorStatisticsStartTime (17)	INTEGER	RO	秒	統計開始時刻※3	
		httpServerConnectorSamplingTime (18)	INTEGER	RW	秒	統計のサンプリング時間※4	
		httpServerConnectorActiveThreadCount (19)	Gauge	RO	－	リクエスト処理中のスレッド数	
		httpServerConnectorWaitingRequestCount (20)	Gauge	RO	－	実行を待たされているリクエスト数	

(凡例)

R/W : Read-Only／Read-Write 種別

－ : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

値の取得要求時点でのピーク値

このピーク値は、記録されてからサンプリング時間の有効期間を持つ最大値のことを示します。

注※3

00:00 Jan. 1, 1970 UTC からの経過秒数

注※4

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(6) dataSourceTable

dataSourceTable を次の表に示します。

表 A-6 dataSourceTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
dataSourceTable (6)	dataSourceEntry (1)	dataSourceIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	データソースのインポート
		dataSourceFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<データソース名>"	
		dataSourceName (3)	DisplayString	RO	—	データソース名	
		dataSourceAuthenticationType (4)	INTEGER	RO	—	認証タイプ 0：アプリケーションごとの認証 1：コンテナごとの認証	
		dataSourceLoginTimeout (5)	INTEGER	RO	秒	ログインタイムアウト値	
		dataSourceUserID (6)	DisplayString	RO	—	ユーザ ID	
		dataSourcePoolUpperBound (7)	INTEGER	RO	—	コネクションプール数 (上限値)	
		dataSourcePoolLowerBound (8)	INTEGER	RO	—	コネクションプール数 (下限値)	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		dataSourceCurrent PoolSize (9)	Gauge	RO	－	コネクションプール数 (現在値)	
		dataSourceActiveC onnectionCount (10)	Gauge	RO	－	使用中のコネクション 数 (現在値)	
		dataSourceFreeCon nectionCount (11)	Gauge	RO	－	未使用のコネクション 数 (現在値)	

(凡例)

R/W : Read-Only／Read-Write 種別

－ : 該当しない。

RO : Read-Only

(7) j2eeApplicationTable

j2eeApplicationTable を次の表に示します。

表 A-7 j2eeApplicationTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
j2eeApplic ationTable (7)	j2eeApplic ationEntry (1)	j2eeApplicationInd ex (1)	INTEGER	RO	－	シーケンス番号[1, …]	J2EE アプ リケーショ ン (EAR) の開始
		j2eeApplicationFull yQualifiedName (2)	DisplaySt ring	RO	－	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名 >:<J2EE アプリケー ション名>"※	
		j2eeApplicationNa me (3)	DisplaySt ring	RO	－	J2EE アプリケーショ ン名	
		j2eeApplicationDe ployedTime (4)	INTEGER	RO	秒	デプロイされた時刻	

(凡例)

R/W : Read-Only／Read-Write 種別

－ : 該当しない。

RO : Read-Only

注※

アプリケーションがテストモードで開始されている場合

"<論理サーバ名>:<J2EE アプリケーション名>_TEST"となります。

(8) ejbApplicationTable

ejbApplicationTable を次の表に示します。

表 A-8 ejbApplicationTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
ejbApplica tionTable (8)	ejbApplica tionEntry (1)	ejbApplicationInde x (1)	INTEGER	RO	—	シーケンス番号[1, …]	EJB アプリ ケーション (EJB-JAR) の開始
		ejbApplicationFully QualifiedName (2)	DisplaySt ring	RO	—	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名 >:<J2EE アプリケー ション名>:<EJB アプ リケーション名>"	
		ejbApplicationNam e (3)	DisplaySt ring	RO	—	EJB アプリケーション 名(EJB-JAR 名)	

(凡例)

R/W : Read-Only/Read-Write 種別

— : 該当しない。

RO : Read-Only

(9) statelessSessionBeanTable

statelessSessionBeanTable を次の表に示します。

表 A-9 statelessSessionBeanTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSe ssionBean Table (9)	statelessSe ssionBeanE ntry (1)	statelessSessionBea nIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateless Session Bean の 使用

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statelessSessionBeanFullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	
		statelessSessionBeanName (3)	DisplayString	RO	－	Enterprise Bean 名	
		statelessSessionBeanHomeInterfaceName (4)	DisplayString	RO	－	RemoteHome インタフェースクラス名※1	
		statelessSessionBeanRemoteInterfaceName (5)	DisplayString	RO	－	RemoteComponent インタフェースクラス名※1	
		statelessSessionBeanLocalHomeInterfaceName (6)	DisplayString	RO	－	LocalHome インタフェースクラス名 (EJB2.0 対応)※2	
		statelessSessionBeanLocalComponentInterfaceName (7)	DisplayString	RO	－	LocalComponent インタフェースクラス名 (EJB2.0 対応)※2	
		statelessSessionBeanEjbClassName (8)	DisplayString	RO	－	EJB クラス名	
		statelessSessionBeanPoolUpperBound (9)	INTEGER	RO	－	インスタンスプール数 (上限値)	
		statelessSessionBeanPoolLowerBound (10)	INTEGER	RO	－	インスタンスプール数 (下限値)	
		statelessSessionBeanCurrentPoolSize (11)	Gauge	RO	－	インスタンスプール数 (現在値)	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statelessSessionBeanWaitingSessionCount (12)	Gauge	RO	－	接続待ちセッション数 (現在値)	
		statelessSessionBeanActiveSessionBeanCount (13)	Gauge	RO	－	使用中の Stateless Session Bean 数 (現在 値)	
		statelessSessionBeanFreeSessionBeanCount (14)	Gauge	RO	－	未使用の Stateless Session Bean 数 (現在 値)	
		statelessSessionBeanFreeSessionTransactionType (15)	DisplayString	RO	－	トランザクションタイプ "CMT"または"BMT"	

(凡例)

R/W : Read-Only / Read-Write 種別

－ : 該当しない。

RO : Read-Only

注※1

Local インタフェースだけの場合は, "" (空文字) を表示します。

注※2

Remote インタフェースだけの場合は, "" (空文字) を表示します。

(10) statelessSessionBeanHomeTable

statelessSessionBeanHomeTable を次の表に示します。

表 A-10 statelessSessionBeanHomeTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSessionBeanHomeTable (10)	statelessSessionBeanHomeEntry (1)	statelessSessionBeanHomeIndex (1)	INTEGER	RO	－	シーケンス番号[1, ...]	Stateless Session Bean Remote Home Interface の使用
		statelessSessionBeanHomeFullyQualifiedname	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		(2)				"<論理サーバ名 >:<J2EE アプリケー ション名>:<EJB アプ リケーション名 >:<Enterprise Bean 名>"	
		statelessSessionBea nHomeName (3)	DisplaySt ring	RO	ー	RemoteHome インタ フェースクラス名	
		statelessSessionBea nHomeResponseMa xTime (4)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する呼び 出しのレスポンス時間 (最大値)	
		statelessSessionBea nHomeResponseMi nTime (5)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する呼び 出しのレスポンス時間 (最小値)	
		statelessSessionBea nHomeResponseAv eTime (6)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する呼び 出しのレスポンス時間 (平均値※1)	
		statelessSessionBea nHomeStatisticsSta rtedTime (7)	INTEGER	RO	秒	統計開始時刻※2	
		statelessSessionBea nHomeSamplingTi me (8)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W : Read-Only/Read-Write 種別

ー : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(11) statelessSessionBeanRemoteTable

statelessSessionBeanRemoteTable を次の表に示します。

表 A-11 statelessSessionBeanRemoteTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSessionBeanRemoteTable (11)	statelessSessionBeanRemoteEntry (1)	statelessSessionBeanRemoteIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateless Session Bean Remote Component Interface の使用
		statelessSessionBeanRemoteFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	
		statelessSessionBeanRemoteName (3)	DisplayString	RO	—	RemoteComponent インタフェースクラス名	
		statelessSessionBeanRemoteResponseMaxTime (4)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statelessSessionBeanRemoteResponseMinTime (5)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statelessSessionBeanRemoteResponseAveTime (6)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（平均値※ ¹ ）	
		statelessSessionBeanRemoteResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	Remote インタフェースに対する EJB メソッドの実行時間（最大値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statelessSessionBeanRemoteResponseEjbMinTime (8)	Gauge	RO	ミリ 秒	Remote インタフェースに対する EJB メソッドの実行時間（最小値）	
		statelessSessionBeanRemoteResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	Remote インタフェースに対する EJB メソッドの実行時間（平均値※ 1)	
		statelessSessionBeanRemoteStatisticsStartedTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		statelessSessionBeanRemoteSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(12) statefulSessionBeanTable

statefulSessionBeanTable を次の表に示します。

表 A-12 statefulSessionBeanTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSessionBeanTable (12)	statefulSessionBeanEntry (1)	statefulSessionBeanIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateful Session Bean の使用
		statefulSessionBeanFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	
		statefulSessionBeanName (3)	DisplayString	RO	—	Enterprise Bean 名	
		statefulSessionBeanHomeInterfaceName (4)	DisplayString	RO	—	RemoteHome インタフェースクラス名※1	
		statefulSessionBeanRemoteInterfaceName (5)	DisplayString	RO	—	RemoteComponent インタフェースクラス名※1	
		statefulSessionBeanLocalHomeInterfaceName (6)	DisplayString	RO	—	LocalHome インタフェースクラス名※2	
		statefulSessionBeanLocalComponentInterfaceName (7)	DisplayString	RO	—	LocalComponent インタフェースクラス名※2	
		statefulSessionBeanEjbClassName (8)	DisplayString	RO	—	EJB クラス名	
		statefulSessionBeanSessionUpperBound (9)	INTEGER	RO	—	同時接続数（上限値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBeanActiveSessionUpperBound (10)	INTEGER	RO	－	同時実行数（上限値）	
		statefulSessionBeanPassivateTimeout (11)	INTEGER	RO	秒	非活性セッションのタイムアウト値	
		statefulSessionBeanActiveTimeout (12)	INTEGER	RO	秒	実行中セッションのタイムアウト値	
		statefulSessionBeanCurrentSessionCount (13)	Gauge	RO	－	接続セッション数（現在値）	
		statefulSessionBeanWaitingSessionCount (14)	Gauge	RO	－	接続待ちセッション数（現在値）	
		statefulSessionBeanActiveSessionCount (15)	Gauge	RO	－	実行中セッション数（現在値）	
		statefulSessionBeanPassiveSessionCount (16)	Gauge	RO	－	非活性セッション数（現在値）	
		statefulSessionBeanTransactionType (17)	DisplayString	RO	－	トランザクションタイプ "CMT"または"BMT"	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

注※1

Local インタフェースだけの場合は、""（空文字）を表示します。

注※2

Remote インタフェースだけの場合は、""（空文字）を表示します。

(13) statefulSessionBeanHomeTable

statefulSessionBeanHomeTable を次の表に示します。

表 A-13 statefulSessionBeanHomeTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSessionBeanHomeTable (13)	statefulSessionBeanHomeEntry (1)	statefulSessionBeanHomeIndex (1)	INTEGER	RO	—	シーケンス番号[1, ...]	StatefulSessionBeanRemoteHomeInterface の使用
		statefulSessionBeanHomeFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	
		statefulSessionBeanHomeName (3)	DisplayString	RO	—	RemoteHome インタフェースクラス名	
		statefulSessionBeanHomeResponseMaxTime (4)	Gauge	RO	ミリ秒	RemoteHome インタフェースに対する呼び出しのレスポンス時間 (最大値)	
		statefulSessionBeanHomeResponseMinTime (5)	Gauge	RO	ミリ秒	RemoteHome インタフェースに対する呼び出しのレスポンス時間 (最小値)	
		statefulSessionBeanHomeResponseAverageTime (6)	Gauge	RO	ミリ秒	RemoteHome インタフェースに対する呼び出しのレスポンス時間 (平均値※1)	
		statefulSessionBeanHomeResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	RemoteHome インタフェースに対する EJB メソッド実行時間 (最大値)	
		statefulSessionBeanHomeResponseEjbMinTime (8)	Gauge	RO	ミリ秒	RemoteHome インタフェースに対する EJB メソッド実行時間 (最小値)	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBeanHomeResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する EJB メソッド実行時間（平 均値※1）	
		statefulSessionBeanHomeStatisticsStar tedTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		statefulSessionBeanHomeSamplingTi me (11)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(14) statefulSessionBeanRemoteTable

statefulSessionBeanRemoteTable を次の表に示します。

表 A-14 statefulSessionBeanRemoteTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSessionBeanRemoteTable (14)	statefulSessionBeanRemoteEntry (1)	statefulSessionBeanRemoteIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	Stateful Session Bean Remote Component

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBeanRemoteFullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	Interface の使用
		statefulSessionBeanRemoteName (3)	DisplayString	RO	－	RemoteComponent インタフェースクラス名	
		statefulSessionBeanRemoteResponseMaxTime (4)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statefulSessionBeanRemoteResponseMinTime (5)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statefulSessionBeanRemoteResponseAveTime (6)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statefulSessionBeanRemoteResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する EJB メソッド実行時間（最大値）	
		statefulSessionBeanRemoteResponseEjbMinTime (8)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する EJB メソッド実行時間（最小値）	
		statefulSessionBeanRemoteResponseEjbAveTime (9)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する EJB メソッド実行時間（平均値※1）	
		statefulSessionBeanRemoteStatisticsStartedTime (10)	INTEGER	RO	秒	統計開始時刻※2	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBeanRemoteSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(15) entityBeanTable

entityBeanTable を次の表に示します。

表 A-15 entityBeanTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
entityBeanTable (15)	entityBeanEntry (1)	entityBeanIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	Entity Bean の使用
		entityBeanFullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	
		entityBeanName (3)	DisplayString	RO	－	Enterprise Bean 名	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		entityBeanHomeInterfaceName (4)	DisplayString	RO	－	RemoteHome インタフェースクラス名※1	
		entityBeanRemoteInterfaceName (5)	DisplayString	RO	－	RemoteComponent インタフェースクラス名※1	
		entityBeanLocalHomeInterfaceName (6)	DisplayString	RO	－	LocalHome インタフェースクラス名※2	
		entityBeanLocalComponentInterfaceName (7)	DisplayString	RO	－	LocalComponent インタフェースクラス名※2	
		entityBeanEjbClassName (8)	DisplayString	RO	－	EJB クラス名	
		entityBeanPersistenceType (9)	DisplayString	RO	－	永続化タイプ CMP または BMP	
		entityBeanCacheOption (10)	DisplayString	RO	－	キャッシュモデル A : Full Cache B : Cache C : No Cache	
		entityBeanSessionUpperBound (11)	INTEGER	RO	－	同時接続数（上限値）	
		entityBeanPoolUpperBound (12)	INTEGER	RO	－	インスタンスプール数（上限値）	
		entityBeanPoolLowerBound (13)	INTEGER	RO	－	インスタンスプール数（下限値）	
		entityBeanCurrentPoolSize (14)	Gauge	RO	－	インスタンスプール数（現在値）	
		entityBeanSessionTimeout (15)	INTEGER	RO	－	接続タイムアウト値	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		entityBeanCurrentSessionCount (16)	Gauge	RO	－	接続セッション数（現在値）	
		entityBeanWaitingSessionCount (17)	Gauge	RO	－	接続待ちセッション数（現在値）	
		entityBeanActiveEntityBeanCount (18)	Gauge	RO	－	使用中の Entity Bean 数（現在値）	
		entityBeanFreeEntityBeanCount (19)	Gauge	RO	－	未使用の Entity Bean 数（現在値）	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

注※1

Local インタフェースだけの場合は、""（空文字）を表示します。

注※2

Remote インタフェースだけの場合は、""（空文字）を表示します。

(16) entityBeanHomeTable

entityBeanHomeTable を次の表に示します。

表 A-16 entityBeanHomeTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
entityBeanHomeTable (16)	entityBeanHomeEntry (1)	entityBeanHomeIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	Entity Bean Remote Home Interface の使用
		entityBeanHomeFullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>"	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
						>:<Enterprise Bean 名>"	
		entityBeanHomeName (3)	DisplayString	RO	ー	RemoteHome インタ フェースクラス名	
		entityBeanHomeResponseMaxTime (4)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する呼び 出しのレスポンス時間 (最大値)	
		entityBeanHomeResponseMinTime (5)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する呼び 出しのレスポンス時間 (最小値)	
		entityBeanHomeResponseAveTime (6)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する呼び 出しのレスポンス時間 (平均値※1)	
		entityBeanHomeResponseEjbMaxTime (7)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する EJB メソッド実行時間 (最 大値)	
		entityBeanHomeResponseEjbMinTime (8)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する EJB メソッド実行時間 (最 小値)	
		entityBeanHomeResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	RemoteHome インタ フェースに対する EJB メソッド実行時間 (平 均値※1)	
		entityBeanHomeStatisticsStartedTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		entityBeanHomeSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W : Read-Only/Read-Write 種別

ー : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(17) entityBeanRemoteTable

entityBeanRemoteTable を次の表に示します。

表 A-17 entityBeanRemoteTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
entityBeanRemoteTable (17)	entityBeanRemoteEntry (1)	entityBeanRemoteIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Entity Bean Remote Component Interface の使用
		entityBeanRemoteFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	
		entityBeanRemoteName (3)	DisplayString	RO	—	RemoteComponent インタフェースクラス名	
		entityBeanRemoteResponseMaxTime (4)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（最大値）	
		entityBeanRemoteResponseMinTime (5)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（最小値）	
		entityBeanRemoteResponseAveTime (6)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する呼び出しのレスポンス時間（平均値※1）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		entityBeanRemoteResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する EJB メソッド実行時間（最大値）	
		entityBeanRemoteResponseEjbMinTime (8)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する EJB メソッド実行時間（最小値）	
		entityBeanRemoteResponseEjbAvgTime (9)	Gauge	RO	ミリ秒	RemoteComponent インタフェースに対する EJB メソッド実行時間（平均値※1）	
		entityBeanRemoteStatisticsStartTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		entityBeanRemoteSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(18) webApplicationTable

webApplicationTable を次の表に示します。

表 A-18 webApplicationTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
webApplicationTable (18)	webApplicationEntry (1)	webApplicationIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Web アプリケーション (WAR) の使用
		webApplicationFullyQualifiedNames (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<コンテキストルート>"	
		webApplicationContextRoot (3)	DisplayString	RO	—	コンテキストルート	
		webApplicationActiveSessionCount (4)	Gauge	RO	—	使用中のセッション数 (現在値)	
		webApplicationTotalJSPServiceCount (5)	Counter	RO	—	JSP の実行回数 (累積値)	
		webApplicationTotalJSPServiceFailedCount (6)	Counter	RO	—	JSP の実行失敗回数 (累積値)	
		webApplicationTotalJSPServiceMaxTime (7)	Gauge	RO	ミリ秒	JSP の実行時間 (最大値)	
		webApplicationTotalJSPServiceMinTime (8)	Gauge	RO	ミリ秒	JSP の実行時間 (最小値)	
		webApplicationTotalJSPServiceAverageTime (9)	Gauge	RO	ミリ秒	JSP の実行時間 (平均値 ※1)	
		webApplicationTotalJSPServicePeak (10)	Gauge	RO	ミリ秒	JSP の実行時間 (ピーク値 ※2)	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		webApplicationTotalJSPServiceOutputDataSizeMax (11)	Gauge	RO	KB	JSP の出力データサイズ (最大値)	
		webApplicationTotalJSPServiceOutputDataSizeMin (12)	Gauge	RO	KB	JSP の出力データサイズ (最小値)	
		webApplicationTotalJSPServiceOutputDataSizeAve (13)	Gauge	RO	KB	JSP の出力データサイズ (平均値※1)	
		webApplicationTotalJSPServiceOutputDataSizePeak (14)	Gauge	RO	KB	JSP の出力データサイズ (ピーク値※2)	
		webApplicationTotalStaticFileServiceCount (15)	Counter	RO	—	静的コンテンツのアクセス回数 (累積値)	
		webApplicationTotalStaticFileServiceFailedCount (16)	Counter	RO	—	静的コンテンツのアクセス失敗回数 (累積値)	
		webApplicationTotalStaticFileServiceMaxTime (17)	Gauge	RO	ミリ 秒	静的コンテンツの取得 処理時間 (最大値)	
		webApplicationTotalStaticFileServiceMinTime (18)	Gauge	RO	ミリ 秒	静的コンテンツの取得 処理時間 (最小値)	
		webApplicationTotalStaticFileServiceAveTime (19)	Gauge	RO	ミリ 秒	静的コンテンツの取得 処理時間 (平均値※1)	
		webApplicationTotalStaticFileServicePeak (20)	Gauge	RO	ミリ 秒	静的コンテンツの取得 処理時間 (ピーク値※2)	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		webApplicationTotalStaticFileServiceOutputDataSizeMax (21)	Gauge	RO	KB	静的コンテンツの出力 データサイズ（最大値）	
		webApplicationTotalStaticFileServiceOutputDataSizeMin (22)	Gauge	RO	KB	静的コンテンツの出力 データサイズ（最小値）	
		webApplicationTotalStaticFileServiceOutputDataSizeAverage (23)	Gauge	RO	KB	静的コンテンツの出力 データサイズ（平均値※ 1)	
		webApplicationTotalStaticFileServiceOutputDataSizePeak (24)	Gauge	RO	KB	静的コンテンツの出力 データサイズ（ピーク 値※2)	
		webApplicationStatisticsStartedTime (25)	INTEGER	RO	秒	統計開始時刻※3	
		webApplicationSamplingTime (26)	INTEGER	RW	秒	統計のサンプリング 時間※4	
		webApplicationExclusiveThreadCountUpperBound (27)	INTEGER	RO	－	占有スレッド数	• Web アプリケーション (WAR) の使用 • 同時実行スレッド数制御機能の使用
		webApplicationActiveThreadCountUpperBound (28)	INTEGER	RO	－	リクエスト処理中のス レッド数（上限値）	
		webApplicationCurrentThreadCountUpperBound (29)	Gauge	RO	－	リクエスト処理中のス レッド数（現在上限値※ 5)	
		webApplicationActiveThreadCount (30)	Gauge	RO	－	リクエスト処理中のス レッド数（現在値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		webApplicationActiveThreadCountHighWaterMark (31)	Gauge	RO	ー	リクエスト処理中のスレッド数（最大値）	
		webApplicationActiveThreadCountLowWaterMark (32)	Gauge	RO	ー	リクエスト処理中のスレッド数（最小値）	
		webApplicationActiveThreadCountAverage (33)	Gauge	RO	ー	リクエスト処理中のスレッド数（平均値※1）	
		webApplicationActiveThreadCountPeak (34)	Gauge	RO	ー	リクエスト処理中のスレッド数（ピーク値※2）	
		webApplicationWaitingRequestCountUpperBound (35)	INTEGER	RO	ー	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数（上限値）	
		webApplicationWaitingRequestCount (36)	Gauge	RO	ー	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数（現在値）	
		webApplicationWaitingRequestCountHighWaterMark (37)	Gauge	RO	ー	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数（最大値）	
		webApplicationWaitingRequestCountLowWaterMark (38)	Gauge	RO	ー	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数（最小値）	
		webApplicationWaitingRequestCountAverage (39)	Gauge	RO	ー	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数（平均値※1）	
		webApplicationWaitingRequestCountPeak (40)	Gauge	RO	ー	稼働スレッド数の上限値の設定によって、実行を待たされているリ	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
						クエスト数（ピーク値※ 2)	
		webApplicationOver flowRequestCount (41)	Counter	RO	－	URL グループ単位および Web アプリケーション単位の実行待ちリクエストの上限値の設定によって、エラーを返したリクエスト数（累積値）	

(凡例)

- R/W：Read-Only／Read-Write 種別
- －：該当しない。
- RO：Read-Only
- RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

- 値の取得要求時点でのピーク値
- このピーク値は、記録されてからサンプリング時間の有効期間を持つ最大値のことを示します。

注※3

00:00 Jan. 1, 1970 UTC からの経過秒数

注※4

- サンプリング時間の設定方法
- 0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

注※5

現在の同時実行できるスレッド数の上限

(19) servletTable

servletTable を次の表に示します。

表 A-19 servletTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
servletTable (19)	servletEntry (1)	servletIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	サーブレットの使用

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		servletFullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<コンテキストルート>:<Servlet クラス名>"	
		servletName (3)	DisplayString	RO	－	web.xml に記述されたサーブレット名 設定されていない場合は、空文字	
		servletServletClassName (4)	DisplayString	RO	－	Servlet クラス名	
		servletServiceCount (5)	Counter	RO	－	サーブレットの実行回数（累積値）	
		servletServiceFailedCount (6)	Counter	RO	－	サーブレットの実行失敗回数（累積値）	
		servletServiceMaxTime (7)	Gauge	RO	ミリ秒	サーブレットの実行時間（最大値）	
		servletServiceMinTime (8)	Gauge	RO	ミリ秒	サーブレットの実行時間（最小値）	
		servletServiceAverageTime (9)	Gauge	RO	ミリ秒	サーブレットの実行時間（平均値※1）	
		servletServicePeak (10)	Gauge	RO	ミリ秒	サーブレットの実行時間（ピーク値※2）	
		servletServiceOutputDataSizeMax (11)	Gauge	RO	KB	サーブレットの出力データサイズ（最大値）	
		servletServiceOutputDataSizeMin (12)	Gauge	RO	KB	サーブレットの出力データサイズ（最小値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		servletServiceOutputDataSizeAverage (13)	Gauge	RO	KB	サーブレットの出力データサイズ（平均値※1）	
		servletServiceOutputDataSizePeak (14)	Gauge	RO	KB	サーブレットの出力データサイズ（ピーク値※2）	
		servletStatisticsStartTime (15)	INTEGER	RO	秒	統計開始時刻※3	
		servletSamplingTime (16)	INTEGER	RW	秒	統計のサンプリング時間※4	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

値の取得要求時点でのピーク値

このピーク値は、記録されてからサンプリング時間の有効期間を持つ最大値のことを示します。

注※3

00:00 Jan. 1, 1970 UTC からの経過秒数

注※4

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(20) requestUrlTable

requestUrlTable を次の表に示します。

表 A-20 requestUrlTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
requestUrl Table (20)	requestUrl Entry (1)	requestUrlIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	requestUrl の統計開始、 かつ該当 Web アプリ ケーションへ の URL アク セス
		requestUrlFullyQu alifiedName (2)	DisplaySt ring	RO	—	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名 >:<J2EE アプリケー ション名>:<コンテキ ストルート>:<リクエ スト URL>"	
		requestUrl (3)	DisplaySt ring	RO	—	リクエスト URL	
		requestUrlServiceC ount (4)	Counter	RO	—	リクエスト実行回数 (累積値)	
		requestUrlServiceF ailedCount (5)	Counter	RO	—	リクエスト失敗回数 (累積値)	
		requestUrlServiceM axTime (6)	Gauge	RO	ミリ 秒	リクエスト実行時間 (最大値)	
		requestUrlServiceM inTime (7)	Gauge	RO	ミリ 秒	リクエスト実行時間 (最小値)	
		requestUrlServiceA veTime (8)	Gauge	RO	ミリ 秒	リクエスト実行時間 (平均値※1)	
		requestUrlServiceP eak (9)	Gauge	RO	ミリ 秒	リクエスト実行時間 (ピーク値※2)	
		requestUrlServiceO utputDataSizeMax (10)	Gauge	RO	KB	リクエスト出力データ サイズ (最大値)	
		requestUrlServiceO utputDataSizeMin (11)	Gauge	RO	KB	リクエスト出力データ サイズ (最小値)	
		requestUrlServiceO utputDataSizeAve	Gauge	RO	KB	リクエスト出力データ サイズ (平均値※1)	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		(12)					
		requestUrlServiceOutputDataSizePeak (13)	Gauge	RO	KB	リクエスト出力データ サイズ (ピーク値※2)	
		requestUrlStatistics StartedTime (14)	INTEGER	RO	秒	統計開始時刻※3	
		requestUrlSamplingTime (15)	INTEGER	RW	秒	統計のサンプリング 時間※4	

(凡例)

R/W : Read-Only／Read-Write 種別

－ : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

値の取得要求時点でのピーク値

このピーク値は、記録されてからサンプリング時間の有効期間を持つ最大値のことを示します。

注※3

00:00 Jan. 1, 1970 UTC からの経過秒数

注※4

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト (10 秒) に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(21) dataSourceFullJTATable

dataSourceFullJTATable を次の表に示します。

表 A-21 dataSourceFullJTATable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
dataSourceFullJTATable	dataSourceFullJTATableEntry	dataSourceFullJTAIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	データソース (XA) のインポート

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
(21)	(1)	dataSourceFullJTA FullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<データソース名>"	
		dataSourceFullJTA Name (3)	DisplayString	RO	－	データソース名	
		dataSourceFullJTA LoginTimeout (4)	INTEGER	RO	秒	ログインタイムアウト値	
		dataSourceFullJTA UserID (5)	DisplayString	RO	－	ユーザ ID	
		dataSourceFullJTA PoolUpperBound (6)	INTEGER	RO	－	コネクションプールの数（上限値）	
		dataSourceFullJTA PoolLowerBound (7)	INTEGER	RO	－	コネクションプールの数（下限値）	
		dataSourceFullJTA CurrentPoolSize (8)	Gauge	RO	－	コネクションプールの数（現在値）	
		dataSourceFullJTA ActiveConnectionCount (9)	Gauge	RO	－	使用中のコネクション数（現在値）	
		dataSourceFullJTA FreeConnectionCount (10)	Gauge	RO	－	未使用のコネクション数（現在値）	
		dataSourceFullJTA GetConnectionMaxTime (11)	Gauge	RO	ミリ秒	getConnection メソッドの実行時間（最大値）	
		dataSourceFullJTA GetConnectionMinTime (12)	Gauge	RO	ミリ秒	getConnection メソッドの実行時間（最小値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		dataSourceFullJTA GetConnectionAve Time (13)	Gauge	RO	ミリ 秒	getConnection メ ソッドの実行時間（平 均値※1）	
		dataSourceFullJTA CreateConnection MaxTime (14)	Gauge	RO	ミリ 秒	getXAConnection メ ソッド実行時間（最大 値）	
		dataSourceFullJTA CreateConnection MinTime (15)	Gauge	RO	ミリ 秒	getXAConnection メ ソッド実行時間（最小 値）	
		dataSourceFullJTA CreateConnection AveTime (16)	Gauge	RO	ミリ 秒	getXAConnection メ ソッド実行時間（平 均値※1）	
		dataSourceFullJTA GetConnectionFail edCount (17)	Counter	RO	－	getConnection メ ソッドの失敗回数	
		dataSourceFullJTA ConnectionErrorCo unt (18)	Counter	RO	－	Connection で FATAL エラーが発生 した回数 ConnectionEventList ener の connectionErrorOcc urred メソッドが呼ば れた回数	
		dataSourceFullJTA StatisticsStartedTi me (19)	INTEGER	RO	秒	統計開始時刻※2	
		dataSourceFullJTA SamplingTime (20)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(22) resourceAdapterTable

resourceAdapterTable を次の表に示します。

表 A-22 resourceAdapterTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
resourceAd apterTable (22)	resourceAd apterEntry (1)	resourceAdapterIn dex (1)	INTEGER	RO	—	シーケンス番号[1, …]	リソースアダ プタの使用
		resourceAdapterFu llyQualifiedName (2)	DisplaySt ring	RO	—	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名>:<リ ソースアダプタ名>": コンテナ "<論理サーバ名 >:<J2EE アプリケー ション名>:<リソース アダプタ名>": アプリ ケーション"	
		resourceAdapterNa me (3)	DisplaySt ring	RO	—	リソースアダプタ名	
		resourceAdapterVe ndorName (4)	DisplaySt ring	RO	—	リソースアダプタ提供 ベンダ名	
		resourceAdapterSp ecVersion (5)	DisplaySt ring	RO	—	リソースアダプタが準 拠する JCA 使用のバー ジョン	
		resourceAdapterEI SType (6)	DisplaySt ring	RO	—	リソースアダプタの接 続先 EIS のタイプ	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		resourceAdapterVersion (7)	DisplayString	RO	－	リソースアダプタのバージョン	
		resourceAdapterConnectionFactoryInterfaceName (8)	DisplayString	RO	－	ConnectionFactoryのインタフェースクラス名	
		resourceAdapterConnectionFactoryImplName (9)	DisplayString	RO	－	ConnectionFactoryの実装クラス名	
		resourceAdapterManagedConnectionFactoryImplName (10)	DisplayString	RO	－	ManagedConnectionFactoryの実装クラス名	
		resourceAdapterConnectionFactoryInterfaceName (11)	DisplayString	RO	－	Connectionのインタフェースクラス名	
		resourceAdapterConnectionFactoryImplName (12)	DisplayString	RO	－	Connectionの実装クラス名	
		resourceAdapterTransactionSupport (13)	DisplayString	RO	－	トランザクションのサポートモデル NoTransaction : トランザクションなし LocalTransaction : ローカルトランザクション XATransaction : XA トランザクション	
		resourceAdapterConfigProperties (14)	DisplayString	RO	－	ConfigurationProperty 情報※1	
		resourceAdapterManagedConnectionFactoryCreateCount (15)	Counter	RO	－	ManagedConnectionFactoryのcreateManagedConnectionメソッドの実行回数	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		resourceAdapterConnCreateCount (16)	Counter	RO	－	ManagedConnection の getConnection メソッドの実行回数	
		resourceAdapterConnCleanupCount (17)	Counter	RO	－	ManagedConnection の cleanup メソッドの実行回数	
		resourceAdapterConnDestroyCount (18)	Counter	RO	－	ManagedConnection の destroy メソッドの実行回数	
		resourceAdapterUserId (19)	DisplayString	RO	－	ユーザ ID	
		resourceAdapterPoolUpperBound (20)	INTEGER	RO	－	コネクションプールの数（上限値）	
		resourceAdapterPoolLowerBound (21)	INTEGER	RO	－	コネクションプールの数（下限値）	
		resourceAdapterCurrentPoolSize (22)	Gauge	RO	－	コネクションプールの数（現在値）	
		resourceAdapterActiveConnectionCount (23)	Gauge	RO	－	使用中のコネクション数（現在値）	
		resourceAdapterFreeConnectionCount (24)	Gauge	RO	－	未使用のコネクション数（現在値）	
		resourceAdapterGetConnectionMaxTime (25)	Gauge	RO	ミリ秒	ConnectionManager の allocateConnection メソッド実行時間（最大値）	
		resourceAdapterGetConnectionMinTime (26)	Gauge	RO	ミリ秒	ConnectionManager の allocateConnection メソッド実行時間（最小値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		resourceAdapterGetConnectionAverageTime (27)	Gauge	RO	ミリ 秒	ConnectionManager の allocateConnection メソッド実行時間（平 均値※2）	
		resourceAdapterCreateConnectionMaxTime (28)	Gauge	RO	ミリ 秒	ManagedConnection Factory の createManagedConn ection メソッド実行時 間（最大値）	
		resourceAdapterCreateConnectionMinTime (29)	Gauge	RO	ミリ 秒	ManagedConnection Factory の createManagedConn ection メソッド実行時 間（最小値）	
		resourceAdapterCreateConnectionAverageTime (30)	Gauge	RO	ミリ 秒	ManagedConnection Factory の createManagedConn ection メソッド実行時 間（平均値※2）	
		resourceAdapterGetConnectionFailedCount (31)	Counter	RO	－	allocateConnection メソッドの失敗回数	
		resourceAdapterConnectionErrorCount (32)	Counter	RO	－	ManagedConnection で FATAL エラーが発 生した回数 ConnectionEventList ener の connectionErrorOcc urred メソッドが呼ば れた回数	
		resourceAdapterStatisticsStartTime (33)	INTEGER	RO	秒	統計開始時刻※3	
		resourceAdapterSamplingTime (34)	INTEGER	RW	秒	統計のサンプリング 時間※4	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO : Read-Only
RW : Read-Write

注※1

出力形式は、<プロパティ名>=<プロパティ値>:<プロパティ名>=<プロパティ値>:・・・とします。プロパティ名、またはプロパティ値に"=", ":", "¥"が含まれる場合は、"¥=", "¥:", "¥¥"として表示します。

(例) プロパティ名・・・"abc" プロパティ値・・・"ab:bc=ef¥gh"の場合
abc=ab¥:bc¥=ef¥gh

注※2

値の取得要求時刻から過去サンプリング時間の平均値

注※3

00:00 Jan. 1, 1970 UTC からの経過秒数

注※4

サンプリング時間の設定方法

0以下の値が指定された場合は、統計が停止されます。1～9の値を指定した場合は、サンプリング時間がデフォルト（10秒）に設定されて、統計が開始されます。10以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(23) statelessSessionBeanLocalHomeTable

statelessSessionBeanLocalHomeTable を次の表に示します。

表 A-23 statelessSessionBeanLocalHomeTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSessionBeanLocalHomeTable (23)	statelessSessionBeanLocalHomeEntry (1)	statelessSessionBeanLocalHomeIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateless Session Bean Local Home Interface の使用
		statelessSessionBeanLocalHomeFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	
		statelessSessionBeanLocalHomeName (3)	DisplayString	RO	—	LocalHome インタフェースクラス名	
		statelessSessionBeanLocalHomeResponseMaxTime (4)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する呼び出しのレスポンス時間（最大値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statelessSessionBeanLocalHomeResponseMinTime (5)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する呼び 出しのレスポンス時間 (最小値)	
		statelessSessionBeanLocalHomeResponseAveTime (6)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する呼び 出しのレスポンス時間 (平均値※1)	
		statelessSessionBeanLocalHomeStatisticsStartedTime (7)	INTEGER	RO	秒	統計開始時刻※2	
		statelessSessionBeanLocalHomeSamplingTime (8)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W : Read-Only／Read-Write 種別

－ : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト (10 秒) に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(24) statelessSessionBeanLocalComponentTable

statelessSessionBeanLocalComponentTable を次の表に示します。

表 A-24 statelessSessionBeanLocalComponentTable

テーブル 名称 (OID)	エン트리 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSessionBeanLocalComponentTable (24)	statelessSessionBeanLocalComponentTable (1)	statelessSessionBeanLocalComponentIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateless Session Bean Local Component Interface の使用
		statelessSessionBeanLocalComponentFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>"	
		statelessSessionBeanLocalComponentFullyQualifiedName (3)	DisplayString	RO	—	LocalComponent インタフェースクラス名	
		statelessSessionBeanLocalComponentResponseMaxTime (4)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statelessSessionBeanLocalComponentResponseMinTime (5)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statelessSessionBeanLocalComponentResponseAveTime (6)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statelessSessionBeanLocalComponentResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（最大値）	
		statelessSessionBeanLocalComponentResponseEjbMinTime (8)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（最小値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statelessSessionBeanLocalComponentResponseEjbAverageTime (9)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（平均値※1）	
		statelessSessionBeanLocalComponentStatisticsStartTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		statelessSessionBeanLocalComponentSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(25) statefulSessionBeanLocalHomeTable

statefulSessionBeanLocalHomeTable を次の表に示します。

表 A-25 statefulSessionBeanLocalHomeTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSession	statefulSessionBeanLocalHomeEntry	statefulSessionBeanLocalHomeIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	Stateful Session Bean Local

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
BeanLocal HomeTable (25)	(1)	statefulSessionBean LocalHomeFullyQualified Name (2)	DisplayString	RO	ー	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名 >:<J2EE アプリケー ション名>:<EJB アプ リケーション名 >:<Enterprise Bean 名>"	Home Interface の使用
		statefulSessionBean LocalHomeName (3)	DisplayString	RO	ー	LocalHome インタ フェースクラス名	
		statefulSessionBean LocalHomeResponseMaxTime (4)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する呼び 出しのレスポンス時間 (最大値)	
		statefulSessionBean LocalHomeResponseMinTime (5)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する呼び 出しのレスポンス時間 (最小値)	
		statefulSessionBean LocalHomeResponseAveTime (6)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する呼び 出しのレスポンス時間 (平均値※1)	
		statefulSessionBean LocalHomeResponseEjbMaxTime (7)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する EJB メソッドの実行時間 (最大値)	
		statefulSessionBean LocalHomeResponseEjbMinTime (8)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する EJB メソッドの実行時間 (最小値)	
		statefulSessionBean LocalHomeResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する EJB メソッドの実行時間 (平均値※1)	
		statefulSessionBean LocalHomeStatisticsStartedTime (10)	INTEGER	RO	秒	統計開始時刻※2	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBean LocalHomeSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(26) statefulSessionBeanLocalComponentTable

statefulSessionBeanLocalComponentTable を次の表に示します。

表 A-26 statefulSessionBeanLocalComponentTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSession BeanLocal ComponentTable (26)	statefulSession BeanLocal ComponentEntry (1)	statefulSessionBean LocalComponentIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	Stateful Session Bean Local Component Interface の使用
		statefulSessionBean LocalComponentFullyQualified Name (2)	DisplayString	RO	－	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名 >:<J2EE アプリケー ション名>:<EJB アプ リケーション名 >:<Enterprise Bean 名>"	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBean LocalComponentName (3)	DisplayString	RO	－	LocalComponent インタフェースクラス名	
		statefulSessionBean LocalComponentResponseMaxTime (4)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statefulSessionBean LocalComponentResponseMinTime (5)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statefulSessionBean LocalComponentResponseAveTime (6)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statefulSessionBean LocalComponentResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対するEJB メソッドの実行時間（最大値）	
		statefulSessionBean LocalComponentResponseEjbMinTime (8)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対するEJB メソッドの実行時間（最小値）	
		statefulSessionBean LocalComponentResponseEjbAveTime (9)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対するEJB メソッドの実行時間（平均値※1）	
		statefulSessionBean LocalComponentStatisticsStartTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		statefulSessionBean LocalComponentSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(27) entityBeanLocalHomeTable

entityBeanLocalHomeTable を次の表に示します。

表 A-27 entityBeanLocalHomeTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
entityBean LocalHome Table (27)	entityBean LocalHome Entry (1)	entityBeanLocalHo meIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Entity Bean Local Home Interface の使用
		entityBeanLocalHo meFullyQualifiedN ame (2)	DisplaySt ring	RO	—	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名 >:<J2EE アプリケー ション名>:<EJB アプ リケーション名 >:<Enterprise Bean 名>"	
		entityBeanLocalHo meName (3)	DisplaySt ring	RO	—	LocalHome インタ フェースクラス名	
		entityBeanLocalHo meResponseMaxTi me (4)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する呼び 出しのレスポンス時間 (最大値)	
		entityBeanLocalHo meResponseMinTi me (5)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する呼び 出しのレスポンス時間 (最小値)	
		entityBeanLocalHo meResponseAveTi me	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する呼び	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		(6)				出しのレスポンス時間 (平均値※1)	
		entityBeanLocalHomeResponseEjbMaxTime (7)	Gauge	RO	ミリ 秒	LocaHome インタ フェースに対する EJB メソッドの実行時間 (最大値)	
		entityBeanLocalHomeResponseEjbMinTime (8)	Gauge	RO	ミリ 秒	LocaHome インタ フェースに対する EJB メソッドの実行時間 (最小値)	
		entityBeanLocalHomeResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	LocaHome インタ フェースに対する EJB メソッドの実行時間 (平均値※1)	
		entityBeanLocalHomeStatisticsStartedTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		entityBeanLocalHomeSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W : Read-Only/Read-Write 種別

— : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト (10 秒) に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(28) entityBeanLocalComponentTable

entityBeanLocalComponentTable を次の表に示します。

表 A-28 entityBeanLocalComponentTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
entityBean Local Component Table (28)	entityBean Local Component Entry (1)	entityBeanLocalCo mponentIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Entity Bean Local Compone nt Interface の使用
		entityBeanLocalCo mponentFullyQuali fiedName (2)	DisplaySt ring	RO	—	インスタンスを一意に 識別する完全修飾名 "<論理サーバ名 >:<J2EE アプリケー ション名>:<EJB アプ リケーション名 >:<Enterprise Bean 名>"	
		entityBeanLocalCo mponentName (3)	DisplaySt ring	RO	—	LocalComponent イ ンタフェースクラス名	
		entityBeanLocalCo mponentResponse MaxTime (4)	Gauge	RO	ミリ 秒	LocalComponent イ ンタフェースに対する 呼び出しのレスポンス 時間（最大値）	
		entityBeanLocalCo mponentResponse MinTime (5)	Gauge	RO	ミリ 秒	LocalComponent イ ンタフェースに対する 呼び出しのレスポンス 時間（最小値）	
		entityBeanLocalCo mponentResponse AveTime (6)	Gauge	RO	ミリ 秒	LocalComponent イ ンタフェースに対する 呼び出しのレスポンス 時間（平均値※1）	
		entityBeanLocalCo mponentResponseE jbMaxTime (7)	Gauge	RO	ミリ 秒	LocalComponent イ ンタフェースに対する EJB メソッドの実行時 間（最大値）	
		entityBeanLocalCo mponentResponseE jbMinTime (8)	Gauge	RO	ミリ 秒	LocalComponent イ ンタフェースに対する EJB メソッドの実行時 間（最小値）	
		entityBeanLocalCo mponentResponseE jbAveTime (9)	Gauge	RO	ミリ 秒	LocalComponent イ ンタフェースに対する EJB メソッドの実行時 間（平均値※1）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		entityBeanLocalComponentStatisticsStartedTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		entityBeanLocalComponentSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(29) messageDrivenBeanTable

messageDrivenBeanTable を次の表に示します。

表 A-29 messageDrivenBeanTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
messageDrivenBeanTable (29)	messageDrivenBeanEntry (1)	messageDrivenBeanIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	Message-driven Bean の使用
		messageDrivenBeanFullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名"	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
						>:<Enterprise Bean 名>"	
		messageDrivenBeanName (3)	DisplayString	RO	－	Enterprise Bean 名	
		messageDrivenBeanEjbClassName (4)	DisplayString	RO	－	EJB クラス名	
		messageDrivenBeanTransactionType (5)	DisplayString	RO	－	トランザクションタイプ CMT または BMT	
		messageDrivenBeanDestinationType (6)	DisplayString	RO	－	デスティネーションタイプ Queue : Point-To-Point モデルのデスティネーション Topic : Publish-Subscribe モデルのデスティネーション	
		messageDrivenBeanPoolUpperBound (7)	INTEGER	RO	－	インスタンスプールの数（上限値）	
		messageDrivenBeanCurrentPoolSize (8)	Gauge	RO	－	インスタンスプールの数（現在値）	
		messageDrivenBeanCurrentSessionCount (9)	Gauge	RO	－	接続セッション数（現在値）	
		messageDrivenBeanResponseEjbMaxTime (10)	Gauge	RO	ミリ秒	EJB メソッドの実行時間（最大値）	
		messageDrivenBeanResponseEjbMinTime (11)	Gauge	RO	ミリ秒	EJB メソッドの実行時間（最小値）	
		messageDrivenBeanResponseEjbAverageTime	Gauge	RO	ミリ秒	EJB メソッドの実行時間（平均値※1）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		(12)					
		messageDrivenBeanStatisticsStartTime (10)	INTEGER	RO	秒	統計開始時刻※2	
		messageDrivenBeanSamplingTime (11)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(30) queueTable

queueTable を次の表に示します。

表 A-30 queueTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
queueTable (30)	queueEntry (1)	queueIndex (1)	INTEGER	RO	－	シーケンス番号[1, …]	なし
		queueFullyQualifiedName (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<キュー名>"	
		queueName (3)	DisplayString	RO	－	キュー名	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		queueStatus (4)	DisplayString	RO	－	状態 scheduling：スケジュール可能状態 preparing：準備中 stopping：キュー終了処理中 closing[in]：キュー入り口閉塞中 closing[out]：キュー出口閉塞中	
		queueSharedApplicationCount (5)	INTEGER	RO	－	キューを共有しているアプリケーション数	
		queueStayedRequestCount (6)	Gauge	RO	－	キューに滞留しているリクエスト数	
		queueStayedRequestMax (7)	Gauge	RO	－	キューに滞留した最大リクエスト数	
		queueLength (8)	INTEGER	RO	－	同時に登録できるリクエスト数	
		queueLookupName (9)	DisplayString	RO	－	キューが管理している登録名情報	
		queueInterfaceName (10)	DisplayString	RO	－	キューが管理しているインタフェース情報	
		queueParallelCount (11)	INTEGER	RO	－	キューが管理している同時実行数	
		queueTotalThreadCount (12)	INTEGER	RO	－	キューが管理している現在の総常駐スレッド数	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

(31) statelessSessionBeanHomeMethodTable

statelessSessionBeanHomeMethodTable を次の表に示します。

表 A-31 statelessSessionBeanHomeMethodTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSessionBeanHomeMethodTable (31)	statelessSessionBeanHomeMethodEntry (1)	statelessSessionBeanHomeMethodIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateless Session Bean Remote Home Interface Method の使用
		statelessSessionBeanHomeMethodFullyQualifiedNames (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>:<Home インタフェースクラス名>:<メソッド名>"	
		statelessSessionBeanHomeMethodName (3)	DisplayString	RO	—	メソッド名	
		statelessSessionBeanHomeMethodResponseMaxTime (4)	Gauge	RO	ミリ秒	Home インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statelessSessionBeanHomeMethodResponseMinTime (5)	Gauge	RO	ミリ秒	Home インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statelessSessionBeanHomeMethodResponseAvgTime (6)	Gauge	RO	ミリ秒	Home インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statelessSessionBeanHomeMethodResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	Home インタフェースに対する EJB メソッドの実行時間（最大値）	
		statelessSessionBeanHomeMethodResponseEjbMinTime (8)	Gauge	RO	ミリ秒	Home インタフェースに対する EJB メソッドの実行時間（最小値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statelessSessionBeanHomeMethodResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	Home インタフェース に対する EJB メソッド の実行時間（平均値※1）	
		statelessSessionBeanHomeMethodCall Count (10)	Counter	RO	—	メソッドの実行回数	
		statelessSessionBeanHomeMethodStatisticsStartedTime (11)	INTEGER	RO	秒	統計開始時刻※2	
		statelessSessionBeanHomeMethodSamplingTime (12)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

—：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(32) statelessSessionBeanRemoteMethodTable

statelessSessionBeanRemoteMethodTable を次の表に示します。

表 A-32 statelessSessionBeanRemoteMethodTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSessionBeanRemoteMethodTable (32)	statelessSessionBeanRemoteMethodEntry (1)	statelessSessionBeanRemoteMethodIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateless Session Bean Remote Component Interface Method の使用
		statelessSessionBeanRemoteMethodFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>:<Remote インタフェースクラス名>:<メソッド名>"	
		statelessSessionBeanRemoteMethodName (3)	DisplayString	RO	—	メソッド名	
		statelessSessionBeanRemoteMethodResponseMaxTime (4)	Gauge	RO	ミリ秒	Remote インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statelessSessionBeanRemoteMethodResponseMinTime (5)	Gauge	RO	ミリ秒	Remote インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statelessSessionBeanRemoteMethodResponseAveTime (6)	Gauge	RO	ミリ秒	Remote インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statelessSessionBeanRemoteMethodResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	Remote インタフェースに対する EJB メソッドの実行時間（最大値）	
		statelessSessionBeanRemoteMethodResponseEjbMinTime (8)	Gauge	RO	ミリ秒	Remote インタフェースに対する EJB メソッドの実行時間（最小値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statelessSessionBeanRemoteMethodResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	Remote インタフェースに対する EJB メソッドの実行時間（平均値※1）	
		statelessSessionBeanRemoteMethodCallCount (10)	Counter	RO	－	メソッドの実行回数	
		statelessSessionBeanRemoteMethodStatisticsStartTime (11)	INTEGER	RO	秒	統計開始時刻※2	
		statelessSessionBeanRemoteMethodSamplingTime (12)	INTEGER	RW	秒	統計のサンプリング時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(33) statelessSessionBeanLocalHomeMethodTable

statelessSessionBeanLocalHomeMethodTable を次の表に示します。

表 A-33 statelessSessionBeanLocalHomeMethodTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSessionBeanLocalHomeMethodTable (33)	statelessSessionBeanLocalHomeMethodEntry (1)	statelessSessionBeanLocalHomeMethodIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateless Session Bean Local Home Interface Method の使用
		statelessSessionBeanLocalHomeMethodFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>:<LocalHome インタフェースクラス名>:<メソッド名>"	
		statelessSessionBeanLocalHomeMethodName (3)	DisplayString	RO	—	メソッド名	
		statelessSessionBeanLocalHomeMethodResponseMaxTime (4)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する呼び出しのレスポンス時間(最大値)	
		statelessSessionBeanLocalHomeMethodResponseMinTime (5)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する呼び出しのレスポンス時間(最小値)	
		statelessSessionBeanLocalHomeMethodResponseAveTime (6)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する呼び出しのレスポンス時間(平均値※1)	
		statelessSessionBeanLocalHomeMethodResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する EJB メソッドの実行時間(最大値)	
		statelessSessionBeanLocalHomeMethod	Gauge	RO	ミリ秒	LocalHome インタフェースに対する EJB	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		dResponseEjbMinTime (8)				メソッドの実行時間 (最小値)	
		statelessSessionBeanLocalHomeMethodResponseEjbAverageTime (9)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する EJB メソッドの実行時間 (平均値※1)	
		statelessSessionBeanLocalHomeMethodCallCount (10)	Counter	RO	－	メソッドの実行回数	
		statelessSessionBeanLocalHomeMethodStatisticsStartTime (11)	INTEGER	RO	秒	統計開始時刻※2	
		statelessSessionBeanLocalHomeMethodSamplingTime (12)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W : Read-Only／Read-Write 種別

－ : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト (10 秒) に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(34) statelessSessionBeanLocalComponentMethodTable

statelessSessionBeanLocalComponentMethodTable を次の表に示します。

表 A-34 statelessSessionBeanLocalComponentMethodTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statelessSessionBeanLocalComponentMethodTable (34)	statelessSessionBeanLocalComponentMethodEntry (1)	statelessSessionBeanLocalComponentMethodIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateless Session Bean Local Component Interface Method の使用
		statelessSessionBeanLocalComponentMethodFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>:<LocalComponent インタフェースクラス名>:<メソッド名>"	
		statelessSessionBeanLocalComponentMethodName (3)	DisplayString	RO	—	メソッド名	
		statelessSessionBeanLocalComponentMethodResponseMaxTime (4)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statelessSessionBeanLocalComponentMethodResponseMinTime (5)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statelessSessionBeanLocalComponentMethodResponseAverageTime (6)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statelessSessionBeanLocalComponentMethodResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（最大値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statelessSessionBeanLocalComponentMethodResponseEjbMinTime (8)	Gauge	RO	ミリ 秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（最小値）	
		statelessSessionBeanLocalComponentMethodResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（平均値※ ¹ ）	
		statelessSessionBeanLocalComponentMethodCallCount (10)	Counter	RO	—	メソッドの実行回数	
		statelessSessionBeanLocalComponentMethodStatisticsStartedTime (11)	INTEGER	RO	秒	統計開始時刻※ ²	
		statelessSessionBeanLocalComponentMethodSamplingTime (12)	INTEGER	RW	秒	統計のサンプリング時間※ ³	

(凡例)

R/W：Read-Only／Read-Write 種別

—：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(35) statefulSessionBeanHomeMethodTable

statefulSessionBeanHomeMethodTable を次の表に示します。

表 A-35 statefulSessionBeanHomeMethodTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSessionBeanHomeMethodTable (35)	statefulSessionBeanHomeMethodEntry (1)	statefulSessionBeanHomeMethodIndex (1)	INTEGER	RO	—	シーケンス番号[1, ...]	Stateful Session Bean Remote Home Interface Method の使用
		statefulSessionBeanHomeMethodFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>:<Home インタフェースクラス名>:<メソッド名>"	
		statefulSessionBeanHomeMethodName (3)	DisplayString	RO	—	メソッド名	
		statefulSessionBeanHomeMethodResponseMaxTime (4)	Gauge	RO	ミリ秒	Home インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statefulSessionBeanHomeMethodResponseMinTime (5)	Gauge	RO	ミリ秒	Home インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statefulSessionBeanHomeMethodResponseAveTime (6)	Gauge	RO	ミリ秒	Home インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statefulSessionBeanHomeMethodResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	Home インタフェースに対する EJB メソッドの実行時間（最大値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBeanHomeMethodResponseEjbMinTime (8)	Gauge	RO	ミリ 秒	Home インタフェース に対する EJB メソッド の実行時間（最小値）	
		statefulSessionBeanHomeMethodResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	Home インタフェース に対する EJB メソッド の実行時間（平均値※ 1)	
		statefulSessionBeanHomeMethodCall Count (10)	Counter	RO	－	メソッドの実行回数	
		statefulSessionBeanHomeMethodStatisticsStartedTime (11)	INTEGER	RO	秒	統計開始時刻※2	
		statefulSessionBeanHomeMethodSamplingTime (12)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(36) statefulSessionBeanRemoteMethodTable

statefulSessionBeanRemoteMethodTable を次の表に示します。

表 A-36 statefulSessionBeanRemoteMethodTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSessionBeanRemoteMethodTable (36)	statefulSessionBeanRemoteMethodEntry (1)	statefulSessionBeanRemoteMethodIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateful Session Bean Remote Component Interface Method の使用
		statefulSessionBeanRemoteMethodFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>:<Remote インタフェースクラス名>:<メソッド名>"	
		statefulSessionBeanRemoteMethodName (3)	DisplayString	RO	—	メソッド名	
		statefulSessionBeanRemoteMethodResponseMaxTime (4)	Gauge	RO	ミリ秒	Remote インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statefulSessionBeanRemoteMethodResponseMinTime (5)	Gauge	RO	ミリ秒	Remote インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statefulSessionBeanRemoteMethodResponseAveTime (6)	Gauge	RO	ミリ秒	Remote インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statefulSessionBeanRemoteMethodResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	Remote インタフェースに対する EJB メソッドの実行時間（最大値）	
		statefulSessionBeanRemoteMethodResponseEjbMinTime (8)	Gauge	RO	ミリ秒	Remote インタフェースに対する EJB メソッドの実行時間（最小値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBeanRemoteMethodResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	Remote インタフェースに対する EJB メソッドの実行時間（平均値※1）	
		statefulSessionBeanRemoteMethodCallCount (10)	Counter	RO	－	メソッドの実行回数	
		statefulSessionBeanRemoteMethodStatisticsStartedTime (11)	INTEGER	RO	秒	統計開始時刻※2	
		statefulSessionBeanRemoteMethodSamplingTime (12)	INTEGER	RW	秒	統計のサンプリング時間※3	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(37) statefulSessionBeanLocalHomeMethodTable

statefulSessionBeanLocalHomeMethodTable を次の表に示します。

表 A-37 statefulSessionBeanLocalHomeMethodTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSessionBeanLocalHomeMethodTable (37)	statefulSessionBeanLocalHomeMethodEntry (1)	statefulSessionBeanLocalHomeMethodIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateful Session Bean Local Home Interface Method の使用
		statefulSessionBeanLocalHomeMethodFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>:<LocalHome インタフェースクラス名>:<メソッド名>"	
		statefulSessionBeanLocalHomeMethodName (3)	DisplayString	RO	—	メソッド名	
		statefulSessionBeanLocalHomeMethodResponseMaxTime (4)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する呼び出しのレスポンス時間 (最大値)	
		statefulSessionBeanLocalHomeMethodResponseMinTime (5)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する呼び出しのレスポンス時間 (最小値)	
		statefulSessionBeanLocalHomeMethodResponseAveTime (6)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する呼び出しのレスポンス時間 (平均値※1)	
		statefulSessionBeanLocalHomeMethodResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	LocalHome インタフェースに対する EJB メソッドの実行時間 (最大値)	
		statefulSessionBeanLocalHomeMethod	Gauge	RO	ミリ秒	LocalHome インタフェースに対する EJB	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		dResponseEjbMin Time (8)				メソッドの実行時間 (最小値)	
		statefulSessionBeanLocalHomeMethodResponseEjbAverageTime (9)	Gauge	RO	ミリ 秒	LocalHome インタ フェースに対する EJB メソッドの実行時間 (平均値※1)	
		statefulSessionBeanLocalHomeMethodCallCount (10)	Counter	RO	－	メソッドの実行回数	
		statefulSessionBeanLocalHomeMethodStatisticsStartTime (11)	INTEGER	RO	秒	統計開始時刻※2	
		statefulSessionBeanLocalHomeMethodSamplingTime (12)	INTEGER	RW	秒	統計のサンプリング 時間※3	

(凡例)

R/W : Read-Only／Read-Write 種別

－ : 該当しない。

RO : Read-Only

RW : Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト (10 秒) に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(38) statefulSessionBeanLocalComponentMethodTable

statefulSessionBeanLocalComponentMethodTable を次の表に示します。

表 A-38 statefulSessionBeanLocalComponentMethodTable

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
statefulSessionBeanLocalComponentMethodTable (38)	statefulSessionBeanLocalComponentMethodEntry (1)	statefulSessionBeanLocalComponentMethodIndex (1)	INTEGER	RO	—	シーケンス番号[1, …]	Stateful Session Bean Local Component Interface Method の使用
		statefulSessionBeanLocalComponentMethodFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名 "<論理サーバ名>:<J2EE アプリケーション名>:<EJB アプリケーション名>:<Enterprise Bean 名>:<LocalComponent インタフェースクラス名>:<メソッド名>"	
		statefulSessionBeanLocalComponentMethodName (3)	DisplayString	RO	—	メソッド名	
		statefulSessionBeanLocalComponentMethodResponseMaxTime (4)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（最大値）	
		statefulSessionBeanLocalComponentMethodResponseMinTime (5)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（最小値）	
		statefulSessionBeanLocalComponentMethodResponseAverageTime (6)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する呼び出しのレスポンス時間（平均値※1）	
		statefulSessionBeanLocalComponentMethodResponseEjbMaxTime (7)	Gauge	RO	ミリ秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（最大値）	

テーブル 名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		statefulSessionBeanLocalComponentMethodResponseEjbMinTime (8)	Gauge	RO	ミリ 秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（最小値）	
		statefulSessionBeanLocalComponentMethodResponseEjbAveTime (9)	Gauge	RO	ミリ 秒	LocalComponent インタフェースに対する EJB メソッドの実行時間（平均値※ ¹ ）	
		statefulSessionBeanLocalComponentMethodCallCount (10)	Counter	RO	－	メソッドの実行回数	
		statefulSessionBeanLocalComponentMethodStatisticsStartedTime (11)	INTEGER	RO	秒	統計開始時刻※ ²	
		statefulSessionBeanLocalComponentMethodSamplingTime (12)	INTEGER	RW	秒	統計のサンプリング時間※ ³	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

値の取得要求時刻から過去サンプリング時間の平均値

注※2

00:00 Jan. 1, 1970 UTC からの経過秒数

注※3

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

(39) webContainerTable

webContainerTable を次の表に示します。

表 A-39 webContainerTable

テーブル名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
webContainerTable (39)	webContainerEntry (1)	webContainerIndex (1)	INTEGER	RO	—	シーケンス番号[1, ...]	なし
		webContainerFullyQualifiedName (2)	DisplayString	RO	—	インスタンスを一意に識別する完全修飾名"<論理サーバ名>"	
		webContainerServerName (3)	DisplayString	RO	—	Web コンテナのサーバ名	
		webContainerStartTime (4)	INTEGER	RO	秒	Web コンテナの起動時刻※1	
		webContainerStatisticsStartTime (5)	INTEGER	RO	秒	統計開始時刻※1	
		webContainerSamplingTime (6)	INTEGER	RW	秒	統計のサンプリング時間※2	
		webContainerActiveThreadCountUpperBound (7)	INTEGER	RO	—	リクエスト処理中のスレッド数の上限値	• Web アプリケーション (WAR) の使用 • 同時実行スレッド数制御機能の使用 • 同時実行スレッド数を設定していない Web アプリケーションが存在
		webContainerCurrentThreadCountUpperBound (8)	Gauge	RO	—	リクエスト処理中のスレッド数の現在上限値※3	
		webContainerActiveThreadCount (9)	Gauge	RO	—	リクエスト処理中のスレッド数の現在値	
		webContainerActiveThreadCountHighWaterMark (10)	Gauge	RO	—	リクエスト処理中のスレッド数の最大値	

テーブル名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		webContainerActiveThreadCountLowWaterMark (11)	Gauge	RO	－	リクエスト処理中のスレッド数の最小値	
		webContainerActiveThreadCountAverage (12)	Gauge	RO	－	リクエスト処理中のスレッド数の平均値※ ⁴	
		webContainerActiveThreadCountPeak (13)	Gauge	RO	－	リクエスト処理中のスレッド数のピーク値※ ⁵	
		webContainerWaitingRequestCountUpperBound (14)	INTEGER	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数の上限値	
		webContainerWaitingRequestCount (15)	Gauge	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数の現在値	
		webContainerWaitingRequestCountHighWaterMark (16)	Gauge	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数の最大値	
		webContainerWaitingRequestCountLowWaterMark (17)	Gauge	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数の最小値	
		webContainerWaitingRequestCountAverage (18)	Gauge	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数の平均値※ ⁴	
		webContainerWaitingRequestCountPeak (19)	Gauge	RO	－	稼働スレッド数の上限値の設定によって、実行を待たされているリクエスト数のピーク値※ ⁵	
		webContainerOverflowRequestCount	Counter	RO	－	デフォルトの実行待ちリクエストの上限値の設定によって、エラー	

テーブル名称 (OID)	エントリ 名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		(20)				を返したリクエスト数の 累積値	

(凡例)

R/W：Read-Only／Read-Write 種別

－：該当しない。

RO：Read-Only

RW：Read-Write

注※1

00:00 Jan. 1, 1970 UTC からの経過秒数

注※2

サンプリング時間の設定方法

0 以下の値が指定された場合は、統計が停止されます。1～9 の値を指定した場合は、サンプリング時間がデフォルト（10 秒）に設定されて、統計が開始されます。10 以上の値を指定した場合は、指定した値をサンプリング時間として、統計が開始されます。

注※3

現在の同時実行可能なスレッド数の上限

注※4

値の取得要求時刻から過去サンプリング時間の平均値

注※5

値の取得要求時点でのピーク値（記録されてからサンプリング時間の有効期間を持つ最大値）

(40) queueAppTable

queueAppTable を次の表に示します。

表 A-40 queueAppTable

テーブル名称 (OID)	エントリ名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
queueAppTable (41)	queueAppEntry (1)	queueAppIndex (1)	INTEGER	RO	－	シーケンス番号[1, ...]	なし
		queueAppFullyQualified Name (2)	DisplayString	RO	－	インスタンスを一意に識別する完全修飾名"<論理 CTM サーバ名>:<キュー名>:<論理 J2EE サーバ名>"	
		queueAppQueue Name (3)	DisplayString	RO	－	キュー名	

テーブル名称 (OID)	エントリ名称 (OID)	メトリック名称 (OID)	TYPE	R/W	単位	説明	前提条件
		queueAppJ2EENAME (4)	DisplayString	RO	—	J2EE 実サーバ名	
		queueAppStartParallelCount (5)	INTEGER	RO	—	J2EE アプリケーション開始時の同時実行数	
		queueAppResidentThreadCount (6)	INTEGER	RO	—	J2EE アプリケーションに対する現在の常駐スレッド数	

(凡例)

R/W : Read-Only／Read-Write 種別

— : 該当しない。

RO : Read-Only

付録 B.1 拡張子と MIME タイプの対応づけ

web.xml に mime-mapping タグで指定する拡張子と MIME タイプの対応づけのデフォルト値を表 B-1 に示します。

表 B-1 拡張子と MIME タイプの対応づけ

拡張子	MIME タイプ
abs	audio/x-mpeg
ai	application/postscript
aif	audio/x-aiff
aifc	audio/x-aiff
aiff	audio/x-aiff
aim	application/x-aim
art	image/x-jg
asf	video/x-ms-asf
asx	video/x-ms-asf
au	audio/basic
avi	video/x-msvideo
avx	video/x-rad-screenplay
bcpio	application/x-bcpio
bin	application/octet-stream
bmp	image/bmp
body	text/html
cdf	application/x-cdf
cer	application/x-x509-ca-cert
class	application/java
cpio	application/x-cpio
csh	application/x-csh
css	text/css
dib	image/bmp
doc	application/msword

拡張子	MIME タイプ
dtd	text/plain
dv	video/x-dv
dvi	application/x-dvi
eps	application/postscript
etx	text/x-setext
exe	application/octet-stream
gif	image/gif
gtar	application/x-gtar
gz	application/x-gzip
hdf	application/x-hdf
hqx	application/mac-binhex40
htc	text/x-component
htm	text/html
html	text/html
ief	image/ief
jad	text/vnd.sun.j2me.app-descriptor
jar	application/octet-stream
java	text/plain
jnlp	application/x-java-jnlp-file
jpe	image/jpeg
jpeg	image/jpeg
jpg	image/jpeg
js	text/javascript
kar	audio/x-midi
latex	application/x-latex
m3u	audio/x-mpegurl
mac	image/x-macpaint
man	application/x-troff-man
me	application/x-troff-me
mid	audio/x-midi
midi	audio/x-midi

拡張子	MIME タイプ
mif	application/x-mif
mov	video/quicktime
movie	video/x-sgi-movie
mp1	audio/x-mpeg
mp2	audio/x-mpeg
mp3	audio/x-mpeg
mpa	audio/x-mpeg
mpe	video/mpeg
mpeg	video/mpeg
mpega	audio/x-mpeg
mpg	video/mpeg
mpv2	video/mpeg2
ms	application/x-wais-source
nc	application/x-netcdf
oda	application/oda
pbm	image/x-portable-bitmap
pct	image/pict
pdf	application/pdf
pgm	image/x-portable-graymap
pic	image/pict
pict	image/pict
pls	audio/x-scpls
png	image/png
pnm	image/x-portable-anymap
pnt	image/x-macpaint
ppm	image/x-portable-pixmap
ps	application/postscript
psd	image/x-photoshop
qt	video/quicktime
qti	image/x-quicktime
qtif	image/x-quicktime

拡張子	MIME タイプ
ras	image/x-cmu-raster
rgb	image/x-rgb
rm	application/vnd.rn-realmedia
roff	application/x-troff
rtf	application/rtf
rtx	text/richtext
sh	application/x-sh
shar	application/x-shar
smf	audio/x-midi
snd	audio/basic
src	application/x-wais-source
sv4cpio	application/x-sv4cpio
sv4crc	application/x-sv4crc
swf	application/x-shockwave-flash
t	application/x-troff
tar	application/x-tar
tcl	application/x-tcl
tex	application/x-tex
texi	application/x-texinfo
texinfo	application/x-texinfo
tif	image/tiff
tiff	image/tiff
tr	application/x-troff
tsv	text/tab-separated-values
txt	text/plain
ulw	audio/basic
ustar	application/x-ustar
xbm	image/x-bitmap
xpm	image/x-pixmap
xwd	image/x-windowdump
wav	audio/x-wav

拡張子	MIME タイプ
wbmp	image/vnd.wap.wbmp
wml	text/vnd.wap.wml
wmlc	application/vnd.wap.wmlc
wmls	text/vnd.wap.wmlscript
wmlscriptc	application/vnd.wap.wmlscriptc
wrl	x-world/x-vrml
Z	application/x-compress
z	application/x-compress
zip	application/zip

索引

記号

- .cmxrc 176
- .mngsvrmonitorrc 451
- .mngsvrutilrc (mngsvrutil コマンドのクライアント側定義ファイル) のキー 450
- .vmirc 489
- <LB 接続情報の識別名>.properties 470
- XX:[+|-]HitachiVerboseGCPrintDirectBuffer 673
- XX:[+|-]ExplicitMemoryUseExcludeClass 662
- XX:[+|-]HitachiAutoExplicitMemory 660
- XX:[+|-]HitachiCommaVerboseGC 599
- XX:[+|-]HitachiExplicitMemoryAutoReclaim 658
- XX:[+|-]HitachiExplicitMemoryAutoRefReclaim 666
- XX:[+|-]HitachiExplicitMemoryCompatibleToV8 659
- XX:
[+|-]HitachiExplicitMemoryMoveToTenuredFirst 657
- XX:[+|-]HitachiFullCore 652
- XX:[+|-]HitachiJavaClassLibTrace 641
- XX:[+|-]HitachiJavaLogNoMoreOutput 581
- XX:[+|-]HitachiLocalsInStackTrace 646
- XX:[+|-]HitachiLocalsInThrowable 643
- XX:[+|-]HitachiLocalsSimpleFormat 648
- XX:[+|-]HitachiOutOfMemoryAbort 633
- XX:
[+|-]HitachiOutOfMemoryAbortThreadDump 634
- XX:
[+|-]HitachiOutOfMemoryAbortThreadDumpWithJHeapProf 635
- XX:[+|-]HitachiOutOfMemoryCause 629
- XX:[+|-]HitachiOutOfMemoryHandling 636
- XX:[+|-]HitachiOutOfMemorySize 632
- XX:[+|-]HitachiOutOfMemoryStackTrace 630
- XX:[+|-]HitachiOutputMilliTime 583
- XX:[+|-]HitachiThreadDump 564
- XX:[+|-]HitachiThreadDumpToStdout 577
- XX:[+|-]HitachiThreadDumpWithBlockCount 579
- XX:[+|-]HitachiThreadDumpWithCpuTime 578
- XX:[+|-]HitachiThreadDumpWithHashCode 577
- XX:[+|-]HitachiTrueTypeInLocals 649
- XX:[+|-]HitachiUseExplicitMemory 652
- XX:[+|-]HitachiVerboseGC 584
- XX:[+|-]HitachiVerboseGCCpuTime 617
- XX:[+|-]HitachiVerboseGCPrintCause 614
- XX:[+|-]HitachiVerboseGCPrintDate 617
- XX:[+|-]HitachiVerboseGCPrintDeleteOnExit 622
- XX:
[+|-]HitachiVerboseGCPrintJVMInternalMemory 619
- XX:
[+|-]HitachiVerboseGCPrintTenuringDistribution 618
- XX:[+|-]HitachiVerboseGCPrintThreadCount 621
- XX:[+|-]JavaLogAsynchronous 583
- XX:[+|-]JITCompilerContinuation 669
- XX:[+|-]PrintCodeCacheFullMessage 627
- XX:[+|-]PrintCodeCacheInfo 624
- XX:[+|-]UseCompressedOops 671
- XX:+Hitachi 564
- XX:CodeCacheInfoPrintRatio 627
- XX:ExplicitMemoryExcludeClassListFile 663
- XX:ExplicitMemoryFullGCPolicy 661
- XX:ExplicitMemoryNotExcludeClassListFile 665
- XX:HitachiAutoExplicitMemoryFile 661
- XX:HitachiCallToString 650
- XX:HitachiExplicitHeapMaxSize 653
- XX:HitachiExplicitMemoryJavaLog 655
- XX:HitachiExplicitMemoryJavaLogFileSize 656
- XX:HitachiExplicitMemoryJavaLogNumberOfFile 657

-XX:HitachiExplicitMemoryLogLevel 654
-XX:HitachiJavaClassLibTraceLineSize 642
-XX:HitachiJavaLog 579
-XX:HitachiJavaLogFileSize 580
-XX:HitachiJavaLogNumberOfFile 582
-XX:HitachiJITCompileMaxMemorySize 667
-XX:HitachiOutOfMemoryHandlingMaxThrowCount 639
-XX:HitachiOutOfMemoryStackTraceLineSize 631
-XX:HitachiThreadLimit 671
-XX:HitachiVerboseGCIntervalTime 613

A

add.class.path [J2EE サーバ用オプション] 25, 318
add.class.path [Java アプリケーション用のキー] 514
add.class.path [運用管理エージェント用オプション定義ファイルのキー] 407
add.class.path [バッチアプリケーション用のキー] 167
add.class.path [バッチサーバ用のキー] 143
add.env [Management Server 用環境変数定義ファイルのキー] 428
add.jvm.arg [J2EE サーバ用オプション] 25, 312
add.jvm.arg [Java アプリケーション用のキー] 513
add.jvm.arg [運用管理エージェント用オプション定義ファイルのキー] 407
add.jvm.arg [バッチアプリケーション用のキー] 167
add.jvm.arg [バッチサーバ用のキー] 143
add.library.path [J2EE サーバ用オプション] 26, 318
add.library.path [Java アプリケーション用のキー] 514
add.library.path [運用管理エージェント用オプション定義ファイルのキー] 408
add.library.path [バッチサーバ用のキー] 143
add.network.drive [運用管理エージェント用オプション定義ファイルのキー] 408

additional.startcmd 256
admin.logger.ExceptionLogFile.filenum [管理コマンドプロパティファイルのキー] 370
admin.logger.ExceptionLogFile.filepath [管理コマンドプロパティファイルのキー] 370
admin.logger.ExceptionLogFile.filesize [管理コマンドプロパティファイルのキー] 370
admin.logger.MessageLogFile.filenum [管理コマンドプロパティファイルのキー] 370
admin.logger.MessageLogFile.filepath [管理コマンドプロパティファイルのキー] 371
admin.logger.MessageLogFile.filesize [管理コマンドプロパティファイルのキー] 371
admin.logger.MessageLogFile.trace.level [管理コマンドプロパティファイルのキー] 371
adminagent.<サーバ種別>.usr_cmd.abnormal_end [運用管理エージェントプロパティファイルのキー] 401
adminagent.<サーバ種別>.watch.interval [運用管理エージェントプロパティファイルのキー] 402
adminagent.<サーバ種別>.watch.level [運用管理エージェントプロパティファイルのキー] 403
adminagent.<サーバ種別>.watch.retry_count [運用管理エージェントプロパティファイルのキー] 403
adminagent.<サーバ種別>.watch.start_time [運用管理エージェントプロパティファイルのキー] 404
adminagent.<サーバ種別>.watch.timeout [運用管理エージェントプロパティファイルのキー] 404
adminagent.adapter.allowedHosts [運用管理エージェントプロパティファイルのキー] 389
adminagent.adapter.bind_host [運用管理エージェントプロパティファイルのキー] 390
adminagent.adapter.port [運用管理エージェントプロパティファイルのキー] 390
adminagent.cluster.localaddress.check [運用管理エージェントプロパティファイルのキー] 390
adminagent.connector.comm.state.cache_max_time [Management Server 環境設定ファイルのキー] 413
adminagent.finalization.stop_servers [運用管理エージェントプロパティファイルのキー] 391
adminagent.forcestop.threaddump.interval [運用管理エージェントプロパティファイルのキー] 392

adminagent.forcestop.threaddump.timeout [運用管理エージェントプロパティファイルのキー] 392

adminagent.forcestop.threaddump [運用管理エージェントプロパティファイルのキー] 391

adminagent.hws.group [運用管理エージェントプロパティファイルのキー] 392

adminagent.hws.owner [運用管理エージェントプロパティファイルのキー] 392

adminagent.hws.sys_cmd.abnormal_end.traceinfo [運用管理エージェントプロパティファイルのキー] 393

adminagent.hws.watch.method [運用管理エージェントプロパティファイルのキー] 393

adminagent.hws.watch.url [運用管理エージェントプロパティファイルのキー] 394

adminagent.j2ee.process.console_event.enabled [運用管理エージェントプロパティファイルのキー] 394

adminagent.j2ee.process.console_log.enabled [運用管理エージェントプロパティファイルのキー] 395

adminagent.j2ee.sys_cmd.abnormal_end.javatrace [運用管理エージェントプロパティファイルのキー] 395

adminagent.j2ee.sys_cmd.abnormal_end.threaddump [運用管理エージェントプロパティファイルのキー] 395

adminagent.jp1event_enabled [運用管理エージェントプロパティファイルのキー] 396

adminagent.jp1event.event_server_name [運用管理エージェントプロパティファイルのキー] 395

adminagent.log.filenum [運用管理エージェントプロパティファイルのキー] 396

adminagent.log.filesize [運用管理エージェントプロパティファイルのキー] 396

adminagent.log.level [運用管理エージェントプロパティファイルのキー] 396

adminagent.lsinfo_dir [運用管理エージェントプロパティファイルのキー] 396

adminagent.maintenance.log.filenum [運用管理エージェントプロパティファイルのキー] 397

adminagent.maintenance.log.filesize [運用管理エージェントプロパティファイルのキー] 397

adminagent.prfttrace_dir [運用管理エージェントプロパティファイルのキー] 397

adminagent.process.consolelog.enabled [運用管理エージェントプロパティファイルのキー] 398

adminagent.process.consolelog.event.queue_size [運用管理エージェントプロパティファイルのキー] 398

adminagent.process.consolelog.filenum [運用管理エージェントプロパティファイルのキー] 398

adminagent.process.consolelog.filesize [運用管理エージェントプロパティファイルのキー] 398

adminagent.process.consolelog.style [運用管理エージェントのプロパティファイルのキー] 399

adminagent.process.consolelog.time [運用管理エージェントのプロパティファイルのキー] 399

adminagent.properties 389

adminagent.rmi.log.filenum [運用管理エージェントプロパティファイルのキー] 399

adminagent.rmi.log.filesize [運用管理エージェントプロパティファイルのキー] 399

adminagent.rmi.log.level [運用管理エージェントプロパティファイルのキー] 399

adminagent.snapshotlog.listfile.2.num_snapshots [運用管理エージェントプロパティファイルのキー] 400

adminagent.snapshotlog.log_dir [運用管理エージェントプロパティファイルのキー] 400

adminagent.snapshotlog.num_snapshots [運用管理エージェントプロパティファイルのキー] 400

adminagent.sys_cmd.abnormal_end.prfttrace [運用管理エージェントプロパティファイルのキー] 400

adminagent.userserver.process.console_log.enabled [運用管理エージェントプロパティファイルのキー] 401

adminagent.watch.retry_timeout.enabled [運用管理エージェントプロパティファイルのキー] 401

adminagent.xml 408

AdminAgentrc 405

adminagentuser.cfg 406

AllText 266

AppendDirectives 265

B

batch.ctm.enabled [バッチアプリケーション用のキー] 168

batch.log.directory [バッチアプリケーション用のキー] 168

batch.log.lockInterval [バッチアプリケーション用のキー] 168

batch.log.lockRetryCount [バッチアプリケーション用のキー] 168

batch.log.maintenance.filenum [バッチアプリケーション用のキー] 169

batch.log.maintenance.filesize [バッチアプリケーション用のキー] 169

batch.log.message.filenum [バッチアプリケーション用のキー] 169

batch.log.message.filesize [バッチアプリケーション用のキー] 169

batch.log.stdout.enabled [バッチアプリケーション用のキー] 169

batch.request.timeout [バッチアプリケーション用のキー] 169

batch.schedule.group.name [バッチアプリケーション用のキー] 169

batch.service.enabled [J2EE サーバ用オプション] 318

batch.service.enabled [バッチサーバ用のキー] 143

batch.vbroker.agent.addr [バッチアプリケーション用のキー] 169

batch.vbroker.agent.port [バッチアプリケーション用のキー] 169

broker.logger.ExceptionLogFile.filenum [CJMSP ブローカー共通プロパティファイルのキー] 373

broker.logger.ExceptionLogFile.filenum [CJMSP ブローカー個別プロパティファイルのキー] 378

broker.logger.ExceptionLogFile.filesize [CJMSP ブローカー共通プロパティファイルのキー] 373

broker.logger.ExceptionLogFile.filesize [CJMSP ブローカー個別プロパティファイルのキー] 378

broker.logger.MessageLogFile.filenum [CJMSP ブローカー共通プロパティファイルのキー] 373

broker.logger.MessageLogFile.filenum [CJMSP ブローカー個別プロパティファイルのキー] 379

broker.logger.MessageLogFile.filesize [CJMSP ブローカー共通プロパティファイルのキー] 373

broker.logger.MessageLogFile.filesize [CJMSP ブローカー個別プロパティファイルのキー] 379

broker.logger.MessageLogFile.trace.level [CJMSP ブローカー共通プロパティファイルのキー] 373

broker.logger.MessageLogFile.trace.level [CJMSP ブローカー個別プロパティファイルのキー] 379

C

CJMSP ブローカー個別プロパティファイル 373

CJMS プロバイダで使用するファイル 367

cmxclient.properties 179

cmxserver.properties 175

com.cosminexus.manager.cmdtracelog.fnum [Manager 設定ファイルのキー] 432

com.cosminexus.manager.cmdtracelog.size [Manager 設定ファイルのキー] 432

com.cosminexus.manager.log.compatible [Manager 設定ファイルのキー] 432

com.cosminexus.manager.log.dir [Manager 設定ファイルのキー] 430

com.cosminexus.manager.messagelog.fnum [Manager 設定ファイルのキー] 430

com.cosminexus.manager.messagelog.size [Manager 設定ファイルのキー] 430

com.cosminexus.manager.messagelog.style [Manager 設定ファイルのキー] 431

com.cosminexus.manager.messagelog.time [Manager 設定ファイルのキー] 431

com.cosminexus.manager.tracelog.fnum [Manager 設定ファイルのキー] 431

com.cosminexus.manager.tracelog.size [Manager 設定ファイルのキー] 431

com.cosminexus.manager.tracelog.style [Manager 設定ファイルのキー] 432

com.cosminexus.manager.tracelog.time [Manager 設定ファイルのキー] 432

com.cosminexus.mngsvr.agent.read_timeout
〔Management Server 環境設定ファイルのキー〕
413

com.cosminexus.mngsvr.compat.operation_ap
p 〔Management Server 環境設定ファイルのキー〕
423

com.cosminexus.mngsvr.jp1event.alert
〔Management Server 環境設定ファイルのキー〕
413

com.cosminexus.mngsvr.jp1event.critical
〔Management Server 環境設定ファイルのキー〕
414

com.cosminexus.mngsvr.jp1event.emergency
〔Management Server 環境設定ファイルのキー〕
414

com.cosminexus.mngsvr.jp1event.enabled
〔Management Server 環境設定ファイルのキー〕
414

com.cosminexus.mngsvr.jp1event.error
〔Management Server 環境設定ファイルのキー〕
414

com.cosminexus.mngsvr.jp1event.information
〔Management Server 環境設定ファイルのキー〕
414

com.cosminexus.mngsvr.jp1event.notice
〔Management Server 環境設定ファイルのキー〕
414

com.cosminexus.mngsvr.jp1event.warning
〔Management Server 環境設定ファイルのキー〕
415

com.cosminexus.mngsvr.log.display_number
〔Management Server 環境設定ファイルのキー〕
415

com.cosminexus.mngsvr.log.level
〔Management Server 環境設定ファイルのキー〕
415

com.cosminexus.mngsvr.log.rotate
〔Management Server 環境設定ファイルのキー〕
415

com.cosminexus.mngsvr.log.size 〔Management
Server 環境設定ファイルのキー〕 415

com.cosminexus.mngsvr.logical_server_abnorm
al_stop.exit 〔Management Server 環境設定ファイ
ルのキー〕 416

com.cosminexus.mngsvr.maintenance.log.filenu
m 〔Management Server 環境設定ファイルのキー〕
416

com.cosminexus.mngsvr.maintenance.log.filesiz
e 〔Management Server 環境設定ファイルのキー〕
416

com.cosminexus.mngsvr.management_user_ac
count.enabled 〔Management Server 環境設定
ファイルのキー〕 417

com.cosminexus.mngsvr.management.connect
or.enabled 〔Management Server 環境設定ファイ
ルのキー〕 416

com.cosminexus.mngsvr.management.enabled
〔Management Server 環境設定ファイルのキー〕
416

com.cosminexus.mngsvr.management.host
〔Management Server 環境設定ファイルのキー〕
416

com.cosminexus.mngsvr.management.listen.po
rt 〔Management Server 環境設定ファイルのキー〕
417

com.cosminexus.mngsvr.management.port
〔Management Server 環境設定ファイルのキー〕
417

com.cosminexus.mngsvr.management.read_tim
eout 〔Management Server 環境設定ファイルの
キー〕 417

com.cosminexus.mngsvr.on_start
〔Management Server 環境設定ファイルのキー〕
417

com.cosminexus.mngsvr.snapshot.auto_collect.
enabled 〔Management Server 環境設定ファイ
ルのキー〕 418

com.cosminexus.mngsvr.snapshot.auto_collect.
timeout 〔Management Server 環境設定ファイ
ルのキー〕 420

com.cosminexus.mngsvr.snapshot.collect.point
〔Management Server 環境設定ファイルのキー〕
418

com.cosminexus.mngsvr.sys_cmd.abnormal_en
d.enabled 〔Management Server 環境設定ファイ
ルのキー〕 418

com.cosminexus.mngsvr.sys_cmd.abnormal_end.timeout [Management Server 環境設定ファイルのキー] 418

com.cosminexus.mngsvr.trace [Management Server 環境設定ファイルのキー] 418

com.cosminexus.mngsvr.upload_app.directory [Management Server 環境設定ファイルのキー] 420

com.cosminexus.mngsvr.upload_app.enabled [Management Server 環境設定ファイルのキー] 419

com.cosminexus.mngsvr.upload_app.maxsize [Management Server 環境設定ファイルのキー] 420

com.cosminexus.mngsvr.usr_cmd.abnormal_end.enabled [Management Server 環境設定ファイルのキー] 419

com.cosminexus.mngsvr.usr_cmd.abnormal_end.timeout [Management Server 環境設定ファイルのキー] 419

com.cosminexus.mngsvr.vmi.enabled [Management Server 環境設定ファイルのキー] 419

config.properties 373

CoreDumpDirectory 261

cpp.library.version [J2EE サーバ用オプション] 26, 318

cpp.library.version [Java アプリケーション用のキー] 514

cpp.library.version [バッチサーバ用のキー] 143

criticalList.cfg 138, 163

CTM コマンドオプションファイル 384

CTM で使用するファイル 380

CTM ユーザ環境変数定義ファイル 382

CustomDivideFileNum 268

CustomDivideTimeDifference 268

CustomDivideTimeInterval 268

CustomWraparoundFileNum 268

CustomWraparoundFilesize 268

D

dataSourceFullJTATable 740

dataSourceTable 713

DocumentRoot 265

E

ejb.client.directory.shareable [Java アプリケーション用のキー] 514

ejb.client.ejb.log [Java アプリケーション用のキー] 514

ejb.client.log.appid [Java アプリケーション用のキー] 514

ejb.client.log.directory [Java アプリケーション用のキー] 515

ejb.client.log.lockInterval [Java アプリケーション用のキー] 515

ejb.client.log.lockRetryCount [Java アプリケーション用のキー] 516

ejb.client.log.stdout.enabled [Java アプリケーション用のキー] 516

ejb.public.directory [J2EE サーバ用オプション] 26, 312

ejb.public.directory [バッチサーバ用のキー] 143

ejb.server.corefilenum [J2EE サーバ用オプション] 26, 318

ejb.server.corefilenum [バッチサーバ用のキー] 144

ejb.server.log.directory [J2EE サーバ用オプション] 27, 313

ejb.server.log.directory [バッチサーバ用のキー] 144

ejb.server.log.stderr.filesize [J2EE サーバ用オプション] 28, 318

ejb.server.log.stderr.filesize [バッチサーバ用のキー] 144

ejb.server.log.stdout.filesize [J2EE サーバ用オプション] 28, 318

ejb.server.log.stdout.filesize [バッチサーバ用のキー] 144

ejbApplicationTable 715

ejbserver.application.InitTermProcessClasses [J2EE サーバのカスタマイズ用キー] 37

ejbserver.application.InitTermProcessClasses [バッチサーバ用のキー] 148

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.appname [J2EE サーバのカスタマイズ用キー] 37

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.appname [Java アプリケーション] 521

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.appname [バッチサーバ用のキー] 148

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.autoFlush.enabled [Java アプリケーション] 522

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.count [J2EE サーバのカスタマイズ用キー] 37

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.count [Java アプリケーション] 522

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.count [バッチサーバ用のキー] 148

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.encoding [J2EE サーバのカスタマイズ用キー] 37

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.encoding [Java アプリケーション] 522

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.encoding [バッチサーバ用のキー] 148

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.filter [J2EE サーバのカスタマイズ用キー] 38

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.filter [Java アプリケーション] 522

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.filter [バッチサーバ用のキー] 148

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.formatter [J2EE サーバのカスタマイズ用キー] 38

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.formatter [Java アプリケーション] 522

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.formatter [バッチサーバ用のキー] 149

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.level [J2EE サーバのカスタマイズ用キー] 39

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.level [Java アプリケーション] 522

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.level [バッチサーバ用のキー] 149

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.limit [J2EE サーバのカスタマイズ用キー] 39

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.limit [Java アプリケーション] 522

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.limit [バッチサーバ用のキー] 149

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.msgid [J2EE サーバのカスタマイズ用キー] 39

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.msgid [Java アプリケーション] 523

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.msgid [バッチサーバ用のキー] 149

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.path [J2EE サーバのカスタマイズ用キー] 40

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.path [Java アプリケーション] 523

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.path [バッチサーバ用のキー] 149

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.separator [J2EE サーバのカスタマイズ用キー] 40

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.separator [Java アプリケーション] 523

ejbserver.application.userlog.CJLogHandler.<ハンドラ名称>.separator [バッチサーバ用のキー] 149

ejbserver.application.userlog.Logger.<ロガー名称>.filter [J2EE サーバのカスタマイズ用キー] 41

ejbserver.application.userlog.Logger.<ロガー名称>.filter [Java アプリケーション] 523

ejbserver.application.userlog.Logger.<ロガー名称>.filter [バッチサーバ用のキー] 149

ejbserver.application.userlog.Logger.<ロガー名称>.handlers [J2EE サーバのカスタマイズ用キー] 41

ejbserver.application.userlog.Logger.<ロガー名称>.handlers [Java アプリケーション] 524

ejbserver.application.userlog.Logger.<ロガー名称>.handlers [バッチサーバ用のキー] 149

ejbserver.application.userlog.Logger.<ロガー名称>.level [J2EE サーバのカスタマイズ用キー] 42

ejbserver.application.userlog.Logger.<ロガー名称>.level [Java アプリケーション] 524

ejbserver.application.userlog.Logger.<ロガー名称>.level [バッチサーバ用のキー] 149

ejbserver.application.userlog.Logger.<ロガー名称>.useParentHandlers [Java アプリケーション] 524

ejbserver.application.userlog.Logger.<ロガー名称>.useParentHandlers [バッチサーバ用のキー] 149

ejbserver.application.userlog.Logger.<ロガー名称>.useParentHandler [J2EE サーバのカスタマイズ用キー] 42

ejbserver.application.userlog.loggers [J2EE サーバのカスタマイズ用キー] 41

ejbserver.application.userlog.loggers [Java アプリケーション] 523

ejbserver.application.userlog.loggers [バッチサーバ用のキー] 149

ejbserver.batch.application.exit.enabled [バッチサーバ用のキー] 149

ejbserver.batch.gc.watch.threshold [バッチサーバ用のキー] 150

ejbserver.batch.queue.length [バッチサーバ用のキー] 150

ejbserver.batch.schedule.group.name [バッチサーバ用のキー] 150

ejbserver.bv.limit.max_validation_message [J2EE サーバのカスタマイズ用キー] 43

ejbserver.client.ctm.RequestPriority [J2EE サーバのカスタマイズ用キー] 44

ejbserver.client.ctm.RequestPriority [Java アプリケーション] 525

ejbserver.client.ctm.RequestPriority [バッチアプリケーション用のキー] 171

ejbserver.client.ctm.RequestPriority [バッチサーバ用のキー] 150

ejbserver.client.ejb.log [Java アプリケーション] 525

ejbserver.client.log.appid [Java アプリケーション] 526

ejbserver.client.log.directory [Java アプリケーション] 526

ejbserver.client.log.lockInterval [Java アプリケーション] 526

ejbserver.client.log.lockRetryCount [Java アプリケーション] 526

ejbserver.client.transaction.clientName [Java アプリケーション] 527

ejbserver.client.transaction.enabled [Java アプリケーション] 527

ejbserver.commonj.WorkManager.non_daemon_work_threads [J2EE サーバのカスタマイズ用キー] 44

ejbserver.compiler.jvm.maxHeapSize 283

ejbserver.compiler.jvm.maxHeapSize [J2EE サーバのカスタマイズ用キー] 44

ejbserver.compiler.jvm.minHeapSize 283

ejbserver.compiler.jvm.minHeapSize [J2EE サーバのカスタマイズ用キー] 45

ejbserver.connectionpool.applicationAuthentication.disabled [J2EE サーバのカスタマイズ用キー] 45, 314

ejbserver.connectionpool.applicationAuthentication.disabled [バッチサーバ用のキー] 150

ejbserver.connectionpool.association.enabled DespiteUnshareableSetting [J2EE サーバのカスタマイズ用キー] 46, 314

ejbserver.connectionpool.association.enabled [J2EE サーバのカスタマイズ用キー] 46

ejbserver.connectionpool.association.enabled [バッチサーバ用のキー] 150

ejbserver.connectionpool.sharingOutsideTransactionScope.enabled [J2EE サーバのカスタマイズ用キー] 46, 314

`ejbserver.connectionpool.sharingOutsideTransactionScope.enabled` [バッチサーバ用のキー] 151

`ejbserver.connectionpool.validation.timeout` [J2EE サーバのカスタマイズ用キー] 47

`ejbserver.connectionpool.validation.timeout` [バッチサーバ用のキー] 151

`ejbserver.connector.logwriter.filenum` [J2EE サーバのカスタマイズ用キー] 47

`ejbserver.connector.logwriter.filenum` [バッチサーバ用のキー] 151

`ejbserver.connector.logwriter.filesize` [J2EE サーバのカスタマイズ用キー] 47

`ejbserver.connector.logwriter.filesize` [バッチサーバ用のキー] 151

`ejbserver.connector.statementpool.clear.backcompat` [J2EE サーバのカスタマイズ用キー] 47

`ejbserver.container.audit_trail.enabled` [J2EE サーバのカスタマイズ用キー] 48

`ejbserver.container.audit_trail.enabled` [バッチサーバ用のキー] 151

`ejbserver.container.bmp.backcompatible` [J2EE サーバのカスタマイズ用キー] 48, 314

`ejbserver.container.ejbhome.sessionbean.reconnect.enabled` [J2EE サーバのカスタマイズ用キー] 48, 314

`ejbserver.container.passivate.scan.interval` [J2EE サーバのカスタマイズ用キー] 49

`ejbserver.container.rebindpolicy` [J2EE サーバのカスタマイズ用キー] 49, 314

`ejbserver.container.rebindpolicy` [Java アプリケーション] 527

`ejbserver.container.rebindpolicy` [バッチアプリケーション用のキー] 171

`ejbserver.container.remove.scan.interval` [J2EE サーバのカスタマイズ用キー] 50

`ejbserver.container.security.disabled` [J2EE サーバのカスタマイズ用キー] 49, 314

`ejbserver.ctm.ActivateTimeOut` [J2EE サーバのカスタマイズ用キー] 50

`ejbserver.ctm.ActivateTimeOut` [バッチサーバ用のキー] 151

`ejbserver.ctm.CTMDomain` [J2EE サーバのカスタマイズ用キー] 50, 314

`ejbserver.ctm.CTMDomain` [バッチサーバ用のキー] 151

`ejbserver.ctm.CTMID` [J2EE サーバのカスタマイズ用キー] 51

`ejbserver.ctm.CTMID` [バッチサーバ用のキー] 151

`ejbserver.ctm.CTMMMyHost` [J2EE サーバのカスタマイズ用キー] 51, 315

`ejbserver.ctm.CTMMMyHost` [バッチサーバ用のキー] 151

`ejbserver.ctm.DeactivateTimeOut` [J2EE サーバのカスタマイズ用キー] 51

`ejbserver.ctm.DeactivateTimeOut` [バッチサーバ用のキー] 151

`ejbserver.ctm.enabled` [J2EE サーバのカスタマイズ用キー] 51, 315

`ejbserver.ctm.enabled` [バッチサーバ用のキー] 152

`ejbserver.ctm.QueueLength` [J2EE サーバのカスタマイズ用キー] 52

`ejbserver.ctm.useGlobalJNDI` [J2EE サーバのカスタマイズ用キー] 52

`ejbserver.cui.checkmethod.compatible` [サーバ管理コマンドのカスタマイズ用キー] 361

`ejbserver.cui.exitcode.compatible` [サーバ管理コマンドのカスタマイズ用キー] 362

`ejbserver.cui.logfile.compatible` [サーバ管理コマンドのカスタマイズ用キー] 362

`ejbserver.cui.optionalname.enabled` [サーバ管理コマンドのカスタマイズ用キー] 363

`ejbserver.deploy.annotations.load_check.enabled` [J2EE サーバのカスタマイズ用キー] 53

`ejbserver.deploy.annotations.load_check.enabled` [サーバ管理コマンドのカスタマイズ用キー] 363

`ejbserver.deploy.annotations.load_libjars.enabled` [J2EE サーバのカスタマイズ用キー] 53

`ejbserver.deploy.app.stopforcibly.disabled` [J2EE サーバのカスタマイズ用キー] 53, 315

`ejbserver.deploy.context.check_interval` [J2EE サーバのカスタマイズ用キー] 54

ejbserver.deploy.context.reload_scope [J2EE サーバのカスタマイズ用キー] 54
 ejbserver.deploy.context.update.interval [J2EE サーバのカスタマイズ用キー] 54
 ejbserver.deploy.exclusive.lockAliveInterval [J2EE サーバのカスタマイズ用キー] 54, 315
 ejbserver.deploy.exclusive.lockAliveInterval [バッチサーバ用のキー] 152
 ejbserver.deploy.LibraryConflictAvoid.<J2EE アプリケーション名>.enabled [J2EE サーバのカスタマイズ用キー] 56
 ejbserver.deploy.LibraryConflictAvoid.<J2EE アプリケーション名>.package [J2EE サーバのカスタマイズ用キー] 56
 ejbserver.deploy.LibraryConflictAvoid.<J2EE アプリケーション名>.resource_dir [J2EE サーバのカスタマイズ用キー] 57
 ejbserver.deploy.multi_release_jar.enabled [J2EE サーバのカスタマイズ用キー] 57
 ejbserver.deploy.resourcefile.scramble.enabled [J2EE サーバのカスタマイズ用キー] 55
 ejbserver.deploy.resourcefile.scramble.enabled [バッチサーバ用のキー] 152
 ejbserver.deploy.session.work.directory [J2EE サーバのカスタマイズ用キー] 55
 ejbserver.deploy.stub.generation.scope [J2EE サーバのカスタマイズ用キー] 56, 315
 ejbserver.distributedtx.ots.recoverFailMessageCount [J2EE サーバのカスタマイズ用キー] 58
 ejbserver.distributedtx.ots.status.directory1 [J2EE サーバのカスタマイズ用キー] 59
 ejbserver.distributedtx.ots.status.directory1 [Java アプリケーション] 528
 ejbserver.distributedtx.ots.status.directory2 [J2EE サーバのカスタマイズ用キー] 59
 ejbserver.distributedtx.ots.status.directory2 [Java アプリケーション] 528
 ejbserver.distributedtx.recovery.completionCheckOnStopping.timeout [J2EE サーバのカスタマイズ用キー] 59, 315
 ejbserver.distributedtx.recovery.port [J2EE サーバのカスタマイズ用キー] 60
 ejbserver.distributedtx.recovery.port [Java アプリケーション] 529
 ejbserver.distributedtx.rollbackClientTxOnSystemException [J2EE サーバのカスタマイズ用キー] 60
 ejbserver.distributedtx.XATransaction.enabled [J2EE サーバのカスタマイズ用キー] 60
 ejbserver.DynamicStubLoading.Enabled [J2EE サーバのカスタマイズ用キー] 61
 ejbserver.ejb.cmp20.cmr.use.existing_table [J2EE サーバのカスタマイズ用キー] 62, 315
 ejbserver.ejb.timerservice.maxCallbackThreads [J2EE サーバのカスタマイズ用キー] 62
 ejbserver.ejb.timerservice.retryCount [J2EE サーバのカスタマイズ用キー] 62
 ejbserver.ejb.timerservice.retryInterval [J2EE サーバのカスタマイズ用キー] 62
 ejbserver.ext.method_observation.interval [J2EE サーバのカスタマイズ用キー] 62
 ejbserver.ext.method_observation.interval [バッチサーバ用のキー] 152
 ejbserver.http.port [J2EE サーバのカスタマイズ用キー] 63
 ejbserver.http.port [バッチサーバ用のキー] 152
 ejbserver.instrumentation.enabled [J2EE サーバのカスタマイズ用キー] 63
 ejbserver.instrumentation.enabled [バッチサーバ用のキー] 153
 ejbserver.javaee.batch.<J2EE アプリケーション名>.appTag [J2EE サーバのカスタマイズ用キー] 64
 ejbserver.javaee.batch.<J2EE アプリケーション名>.executorService.jndiName [J2EE サーバのカスタマイズ用キー] 64
 ejbserver.javaee.batch.executorService.<JNDI名>.keepAliveTime [J2EE サーバのカスタマイズ用キー] 64
 ejbserver.javaee.batch.executorService.<JNDI名>.maxThreads [J2EE サーバのカスタマイズ用キー] 64
 ejbserver.javaee.batch.executorService.<JNDI名>.minThreads [J2EE サーバのカスタマイズ用キー] 64

ejbserver.javaee.batch.executorService.<JNDI 名>.queueSize [J2EE サーバのカスタマイズ用キー] 64

ejbserver.javaee.batch.jobRepository.autoCreate.enabled [J2EE サーバのカスタマイズ用キー] 64

ejbserver.javaee.batch.jobRepository.jndiName [J2EE サーバのカスタマイズ用キー] 65

ejbserver.javaee.cdi.beanIdentifierIndexOptimization [J2EE サーバのカスタマイズ用キー] 65

ejbserver.javaee.cdi.beansXmlRequired [J2EE サーバのカスタマイズ用キー] 65

ejbserver.javaee.cdi.webInfLibJarsEnabled [J2EE サーバのカスタマイズ用キー] 65

ejbserver.javaee.concurrent.hungCheckIntervalSeconds [J2EE サーバのカスタマイズ用キー] 66

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.awaitTerminationSeconds [J2EE サーバのカスタマイズ用キー] 66

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.corePoolSize [J2EE サーバのカスタマイズ用キー] 66

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.hungAfterSeconds [J2EE サーバのカスタマイズ用キー] 66

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.keepAliveSeconds [J2EE サーバのカスタマイズ用キー] 66

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.longRunningTasks [J2EE サーバのカスタマイズ用キー] 66

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.maxPoolSize [J2EE サーバのカスタマイズ用キー] 67

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.taskQueueCapacity [J2EE サーバのカスタマイズ用キー] 67

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.threadLifeTimeSeconds [J2EE サーバのカスタマイズ用キー] 67

ejbserver.javaee.concurrent.managedExecutorService.<JNDI 名>.threadPriority [J2EE サーバのカスタマイズ用キー] 67

ejbserver.javaee.concurrent.managedExecutorService.jndiNames [J2EE サーバのカスタマイズ用キー] 67

ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.awaitTerminationSeconds [J2EE サーバのカスタマイズ用キー] 67

ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.corePoolSize [J2EE サーバのカスタマイズ用キー] 67

ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.hungAfterSeconds [J2EE サーバのカスタマイズ用キー] 68

ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.keepAliveSeconds [J2EE サーバのカスタマイズ用キー] 68

ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.longRunningTasks [J2EE サーバのカスタマイズ用キー] 68

ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.threadLifeTimeSeconds [J2EE サーバのカスタマイズ用キー] 68

ejbserver.javaee.concurrent.managedScheduledExecutorService.<JNDI 名>.threadPriority [J2EE サーバのカスタマイズ用キー] 68

ejbserver.javaee.concurrent.managedScheduledExecutorService.jndiNames [J2EE サーバのカスタマイズ用キー] 68

ejbserver.javaee.concurrent.managedThreadFactory.<JNDI 名>.threadPriority [J2EE サーバのカスタマイズ用キー] 68

ejbserver.javaee.concurrent.managedThreadFactory.jndiNames [J2EE サーバのカスタマイズ用キー] 69

ejbserver.javaee.jaxrs.config.client.connectTimeout [J2EE サーバのカスタマイズ用キー] 69

ejbserver.javaee.jaxrs.config.client.readTimeout [J2EE サーバのカスタマイズ用キー] 69

ejbserver.javaeeLogfilter.msgids [J2EE サーバのカスタマイズ用キー] 69

ejbserver.jca.adapter.tp1.bind_host [J2EE サーバのカスタマイズ用キー] 69

`ejbserver.jndi.cache.interval.clear.option` [J2EE サーバのカスタマイズ用キー] 70

`ejbserver.jndi.cache.interval.clear.option` [Java アプリケーション] 530

`ejbserver.jndi.cache.interval.clear.option` [バッチサーバ用のキー] 153

`ejbserver.jndi.cache.interval` [J2EE サーバのカスタマイズ用キー] 70

`ejbserver.jndi.cache.interval` [Java アプリケーション] 529

`ejbserver.jndi.cache.interval` [バッチサーバ用のキー] 153

`ejbserver.jndi.cache.reference` [J2EE サーバのカスタマイズ用キー] 70, 315

`ejbserver.jndi.cache.reference` [バッチサーバ用のキー] 153

`ejbserver.jndi.cache` [J2EE サーバのカスタマイズ用キー] 70

`ejbserver.jndi.cache` [Java アプリケーション] 529

`ejbserver.jndi.cache` [バッチサーバ用のキー] 153

`ejbserver.jndi.global.enabled` [J2EE サーバのカスタマイズ用キー] 71

`ejbserver.jndi.log.message.verbosemode` 540

`ejbserver.jndi.namingservice.group.<Specify group name>.providerurls` [J2EE サーバのカスタマイズ用キー] 71

`ejbserver.jndi.namingservice.group.<Specify group name>.providerurls` [Java アプリケーション] 530

`ejbserver.jndi.namingservice.group.<Specify group name>.providerurls` [バッチサーバ用のキー] 153

`ejbserver.jndi.namingservice.group.list` [J2EE サーバのカスタマイズ用キー] 71

`ejbserver.jndi.namingservice.group.list` [Java アプリケーション] 530

`ejbserver.jndi.namingservice.group.list` [バッチサーバ用のキー] 153

`ejbserver.jndi.request.timeout` [J2EE サーバのカスタマイズ用キー] 71

`ejbserver.jndi.request.timeout` [Java アプリケーション] 530

`ejbserver.jndi.request.timeout` [バッチアプリケーション用のキー] 171

`ejbserver.jndi.request.timeout` [バッチサーバ用のキー] 153

`ejbserver.jpa.cdiEnabled` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jpa.defaultJtaDsName` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jpa.defaultNonJtaDsName` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jpa.defaultProviderClassName` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jpa.disable` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jpa.emfprop.<property key>` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jpa.overrideJtaDsName` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jpa.overrideNonJtaDsName` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jpa.overrideProvider` [J2EE サーバのカスタマイズ用キー] 72

`ejbserver.jta.TransactionManager.defaultTime Out` [J2EE サーバのカスタマイズ用キー] 73

`ejbserver.jta.TransactionManager.defaultTime Out` [Java アプリケーション] 530

`ejbserver.jta.TransactionManager.defaultTime Out` [バッチサーバ用のキー] 153

`ejbserver.logger.access_log.nio_http.enabled` [J2EE サーバのカスタマイズ用キー] 73

`ejbserver.logger.access_log.nio_http.format` [J2EE サーバのカスタマイズ用キー] 73

`ejbserver.logger.access_log.websocket.enabled` [J2EE サーバのカスタマイズ用キー] 74

`ejbserver.logger.access_log.websocket.format` [J2EE サーバのカスタマイズ用キー] 74

`ejbserver.logger.channels.define.<チャンネル名>.filenum` [J2EE サーバのカスタマイズ用キー] 74

`ejbserver.logger.channels.define.<チャンネル名>.filenum` [Java アプリケーション] 531

`ejbserver.logger.channels.define.<チャンネル名>.filenum` [バッチサーバ用のキー] 153

`ejbserver.logger.channels.define.<チャンネル名>.filesize` [J2EE サーバのカスタマイズ用キー] 75

`ejbserver.logger.channels.define.<チャンネル名>.filesize` [Java アプリケーション] 531

`ejbserver.logger.channels.define.<チャンネル名>.filesize` [バッチサーバ用のキー] 154

`ejbserver.logger.DevelopmentLogFile.level` [J2EE サーバのカスタマイズ用キー] 74

`ejbserver.logger.enabled.*` [J2EE サーバのカスタマイズ用キー] 76

`ejbserver.logger.enabled.*` [Java アプリケーション] 531

`ejbserver.logger.enabled.*` [サーバ管理コマンドのカスタマイズ用キー] 364

`ejbserver.logger.enabled.*` [バッチサーバ用のキー] 154

`ejbserver.logger.rotationStyle` [J2EE サーバのカスタマイズ用キー] 76

`ejbserver.logger.rotationStyle` [バッチサーバ用のキー] 154

`ejbserver.logger.rotationTime` [J2EE サーバのカスタマイズ用キー] 76

`ejbserver.logger.systemlog.enabled` [J2EE サーバのカスタマイズ用キー] 76

`ejbserver.management.JVM.stats_monitor.FullGCCCount.enabled` [J2EE サーバのカスタマイズ用キー] 77

`ejbserver.management.JVM.stats_monitor.FullGCCCount.enabled` [バッチサーバ用のキー] 154

`ejbserver.management.JVM.stats_monitor.FullGCCCount.interval` [J2EE サーバのカスタマイズ用キー] 78

`ejbserver.management.JVM.stats_monitor.FullGCCCount.interval` [バッチサーバ用のキー] 154

`ejbserver.management.JVM.stats_monitor.FullGCCCount.threshold` [J2EE サーバのカスタマイズ用キー] 77

`ejbserver.management.JVM.stats_monitor.FullGCCCount.threshold` [バッチサーバ用のキー] 154

`ejbserver.management.statistics.interval` [J2EE サーバのカスタマイズ用キー] 78

`ejbserver.management.statistics.interval` [バッチサーバ用のキー] 155

`ejbserver.management.stats_file.base_time` [J2EE サーバのカスタマイズ用キー] 78

`ejbserver.management.stats_file.base_time` [バッチサーバ用のキー] 155

`ejbserver.management.stats_file.dir` [J2EE サーバのカスタマイズ用キー] 78

`ejbserver.management.stats_file.dir` [バッチサーバ用のキー] 155

`ejbserver.management.stats_file.enabled` [J2EE サーバのカスタマイズ用キー] 78

`ejbserver.management.stats_file.enabled` [バッチサーバ用のキー] 155

`ejbserver.management.stats_file.num` [J2EE サーバのカスタマイズ用キー] 78

`ejbserver.management.stats_file.num` [バッチサーバ用のキー] 155

`ejbserver.management.stats_file.period` [J2EE サーバのカスタマイズ用キー] 79

`ejbserver.management.stats_file.period` [バッチサーバ用のキー] 155

`ejbserver.management.webcontainer.stats_monitor.waiting_request_count.enabled` [J2EE サーバのカスタマイズ用キー] 79

`ejbserver.management.webcontainer.stats_monitor.waiting_request_count.high_threshold` [J2EE サーバのカスタマイズ用キー] 80

`ejbserver.management.webcontainer.stats_monitor.waiting_request_count.low_threshold` [J2EE サーバのカスタマイズ用キー] 80

`ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.enabled` [J2EE サーバのカスタマイズ用キー] 79

`ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.high_threshold` [J2EE サーバのカスタマイズ用キー] 79

`ejbserver.management.webcontainer.stats_monitor.whole_waiting_request_count.low_threshold` [J2EE サーバのカスタマイズ用キー] 79

`ejbserver.manager.agent.Agent.conf` [J2EE サーバのカスタマイズ用キー] 80, 315

`ejbserver.manager.agent.Agent.conf` [バッチサーバ用のキー] 155

ejbserver.manager.agent.Agent.enabled [J2EE サーバのカスタマイズ用キー] 80, 316

ejbserver.manager.agent.Agent.enabled [バッチサーバ用のキー] 155

ejbserver.manager.agent.JP1EventAgent.conf [J2EE サーバのカスタマイズ用キー] 81

ejbserver.manager.agent.JP1EventAgent.conf [バッチサーバ用のキー] 155

ejbserver.manager.agent.JP1EventAgent.enabled [J2EE サーバのカスタマイズ用キー] 81

ejbserver.manager.agent.JP1EventAgent.enabled [バッチサーバ用のキー] 155

ejbserver.manager.agent.MEventAgent.conf [J2EE サーバのカスタマイズ用キー] 81

ejbserver.manager.agent.MEventAgent.conf [バッチサーバ用のキー] 155

ejbserver.manager.agent.MEventAgent.enabled [J2EE サーバのカスタマイズ用キー] 81

ejbserver.manager.agent.MEventAgent.enabled [バッチサーバ用のキー] 155

ejbserver.manager.jp1event.event_server_name [J2EE サーバのカスタマイズ用キー] 81, 316

ejbserver.naming.exec.args [J2EE サーバのカスタマイズ用キー] 81, 316

ejbserver.naming.host [J2EE サーバのカスタマイズ用キー] 82

ejbserver.naming.host [サーバ管理コマンドのカスタマイズ用キー] 364

ejbserver.naming.host [バッチサーバ用のキー] 156

ejbserver.naming.nameroot [J2EE サーバのカスタマイズ用キー] 82, 316

ejbserver.naming.nameroot [バッチサーバ用のキー] 156

ejbserver.naming.port [J2EE サーバのカスタマイズ用キー] 83

ejbserver.naming.port [サーバ管理コマンドのカスタマイズ用キー] 364

ejbserver.naming.port [バッチサーバ用のキー] 156

ejbserver.naming.protocol [J2EE サーバのカスタマイズ用キー] 83, 316

ejbserver.naming.protocol [サーバ管理コマンドのカスタマイズ用キー] 365

ejbserver.naming.startupMode [J2EE サーバのカスタマイズ用キー] 83

ejbserver.naming.startupRetryCount [J2EE サーバのカスタマイズ用キー] 84

ejbserver.naming.startupRetryCount [バッチサーバ用のキー] 156

ejbserver.naming.startupWaitTime [J2EE サーバのカスタマイズ用キー] 84

ejbserver.naming.startupWaitTime [バッチサーバ用のキー] 156

ejbserver.rmi.localinvocation.scope [J2EE サーバのカスタマイズ用キー] 84

ejbserver.rmi.logger.filenum [J2EE サーバのカスタマイズ用キー] 84

ejbserver.rmi.logger.filenum [Management Server 環境設定ファイルのキー] 420

ejbserver.rmi.logger.filesize [J2EE サーバのカスタマイズ用キー] 85

ejbserver.rmi.logger.filesize [Management Server 環境設定ファイルのキー] 420

ejbserver.rmi.naming.host [J2EE サーバのカスタマイズ用キー] 85

ejbserver.rmi.naming.host [バッチサーバ用のキー] 156

ejbserver.rmi.naming.port [J2EE サーバのカスタマイズ用キー] 85

ejbserver.rmi.naming.port [バッチサーバ用のキー] 156

ejbserver.rmi.passbyreference [J2EE サーバのカスタマイズ用キー] 85

ejbserver.rmi.remote.listener.port [J2EE サーバのカスタマイズ用キー] 85

ejbserver.rmi.remote.listener.port [バッチサーバ用のキー] 156

ejbserver.rmi.request.timeout [J2EE サーバのカスタマイズ用キー] 86

ejbserver.rmi.request.timeout [Java アプリケーション] 531

ejbserver.rmi.request.timeout [サーバ管理コマンドのカスタマイズ用キー] 365

ejbserver.rmi.request.timeout [バッチアプリケーション用のキー] 171

ejbserver.rmi.request.timeout [バッチサーバ用のキー] 156

ejbserver.rmi.stateless.unique_id.enabled [J2EE サーバのカスタマイズ用キー] 86, 316

ejbserver.security.digest.algorithm [J2EE サーバのカスタマイズ用キー] 87

ejbserver.security.service.url [Java アプリケーション] 532

ejbserver.server.eheap.httpsession.enabled [J2EE サーバのカスタマイズ用キー] 87

ejbserver.server.mutex.invocation.timeout [J2EE サーバのカスタマイズ用キー] 88, 316

ejbserver.server.prf.PRIFID [J2EE サーバのカスタマイズ用キー] 88

ejbserver.server.prf.PRIFID [Java アプリケーション] 532

ejbserver.server.prf.PRIFID [バッチサーバ用のキー] 156

ejbserver.server.prf.processName 540

ejbserver.server.threaddump.filenum [J2EE サーバのカスタマイズ用キー] 88

ejbserver.server.threaddump.filenum [バッチサーバ用のキー] 156

ejbserver.serverName [Java アプリケーション] 532

ejbserver.stateful.passivate.switch [J2EE サーバのカスタマイズ用キー] 89

ejbserver.stdoutlog.autoflush [J2EE サーバのカスタマイズ用キー] 89

ejbserver.stdoutlog.autoflush [Java アプリケーション] 533

ejbserver.stdoutlog.autoflush [バッチサーバ用のキー] 156

ejbserver.watch.defaultRequestQueue.enabled [J2EE サーバのカスタマイズ用キー] 90

ejbserver.watch.defaultRequestQueue.interval [J2EE サーバのカスタマイズ用キー] 90

ejbserver.watch.defaultRequestQueue.threshold [J2EE サーバのカスタマイズ用キー] 90

ejbserver.watch.defaultRequestQueue.writefile.enabled [J2EE サーバのカスタマイズ用キー] 90

ejbserver.watch.enabled [バッチサーバ用のキー] 157

ejbserver.watch.fileDescriptor.enabled [J2EE サーバのカスタマイズ用キー] 91

ejbserver.watch.fileDescriptor.enabled [バッチサーバ用のキー] 157

ejbserver.watch.fileDescriptor.interval [J2EE サーバのカスタマイズ用キー] 91

ejbserver.watch.fileDescriptor.interval [バッチサーバ用のキー] 157

ejbserver.watch.fileDescriptor.threshold [J2EE サーバのカスタマイズ用キー] 91

ejbserver.watch.fileDescriptor.threshold [バッチサーバ用のキー] 157

ejbserver.watch.fileDescriptor.writefile.enabled [J2EE サーバのカスタマイズ用キー] 91

ejbserver.watch.fileDescriptor.writefile.enabled [バッチサーバ用のキー] 157

ejbserver.watch.memory.enabled [J2EE サーバのカスタマイズ用キー] 91

ejbserver.watch.memory.enabled [バッチサーバ用のキー] 157

ejbserver.watch.memory.interval [J2EE サーバのカスタマイズ用キー] 92

ejbserver.watch.memory.interval [バッチサーバ用のキー] 157

ejbserver.watch.memory.rate2alert.enabled [J2EE サーバのカスタマイズ用キー] 93

ejbserver.watch.memory.rate2alert.enabled [バッチサーバ用のキー] 158

ejbserver.watch.memory.threshold [J2EE サーバのカスタマイズ用キー] 92

ejbserver.watch.memory.threshold [バッチサーバ用のキー] 157

ejbserver.watch.memory.writefile.enabled [J2EE サーバのカスタマイズ用キー] 92

ejbserver.watch.memory.writefile.enabled [バッチサーバ用のキー] 157

`ejbserver.watch.thread.enabled` [J2EE サーバのカスタマイズ用キー] 92

`ejbserver.watch.thread.enabled` [バッチサーバ用のキー] 157

`ejbserver.watch.thread.interval` [J2EE サーバのカスタマイズ用キー] 93

`ejbserver.watch.thread.interval` [バッチサーバ用のキー] 157

`ejbserver.watch.thread.threshold` [J2EE サーバのカスタマイズ用キー] 93

`ejbserver.watch.thread.threshold` [バッチサーバ用のキー] 157

`ejbserver.watch.thread.writefile.enabled` [J2EE サーバのカスタマイズ用キー] 93

`ejbserver.watch.thread.writefile.enabled` [バッチサーバ用のキー] 157

`ejbserver.watch.threaddump.enabled` [J2EE サーバのカスタマイズ用キー] 93

`ejbserver.watch.threaddump.enabled` [バッチサーバ用のキー] 157

`ejbserver.watch.threaddump.interval` [J2EE サーバのカスタマイズ用キー] 93

`ejbserver.watch.threaddump.interval` [バッチサーバ用のキー] 157

`ejbserver.watch.threaddump.threshold` [J2EE サーバのカスタマイズ用キー] 93

`ejbserver.watch.threaddump.threshold` [バッチサーバ用のキー] 157

`ejbserver.watch.threaddump.writefile.enabled` [J2EE サーバのカスタマイズ用キー] 93

`ejbserver.watch.threaddump.writefile.enabled` [バッチサーバ用のキー] 157

`ejbserver.webj2ee.connectionAutoClose.enabled` [J2EE サーバのカスタマイズ用キー] 94

`entityBeanHomeTable` 728

`entityBeanLocalComponentTable` 755

`entityBeanLocalHomeTable` 754

`entityBeanRemoteTable` 730

`entityBeanTable` 726

`ErrorDivideFileNum` 268

`ErrorDivideTimeDifference` 269

`ErrorDivideTimeInterval` 269

`ErrorWraparoundFileNum` 269

`ErrorWraparoundFilesize` 269

G

`Group` 262

H

`hitachi_web.properties` 23

`HttpsdCustomLogFileDir` 269

`HttpsdCustomlogFormat` 269

`HttpsdCustomMethod` 270

`HttpsdErrorLogFileDir` 270

`HttpsdErrorMethod` 270

`HttpsdLogFileDir` 270

`HttpsdRequestLogFileDir` 271

`HttpsdRequestMethod` 271

`httpServerConnectorTable` 710

`httpServerTable` 709

HTTP Server のログの出力方法を設定するパラメタ 267

`HWSKeepStartServers` 262

`HWSLogTimeVerbose` 263

`HWSPrfld` 266

`HWSRequestlofilter` 271

`HWSRequestLogLevel` 271

`HWSRequestModuleDebug` 272

`HWSRequestModuleInfo` 272

`HWSRequestProxy` 272

`HWSRequestRequest` 272

I

`imq.admin.tcp.port` [CJMSP ブローカー個別プロパティファイルのキー] 375

`imq.autocreate.destination.maxNumMsgs` [CJMSP ブローカー共通プロパティファイルのキー] 373

`imq.autocreate.destination.maxNumMsgs` [CJMSP ブローカー個別プロパティファイルのキー] 377

imq.autocreate.destination.maxTotalMsgBytes
[CJMSP ブローカー共通プロパティファイルのキー]
373

imq.autocreate.destination.maxTotalMsgBytes
[CJMSP ブローカー個別プロパティファイルのキー]
377

imq.autocreate.queue.consumerFlowLimit
[CJMSP ブローカー共通プロパティファイルのキー]
373

imq.autocreate.queue.consumerFlowLimit
[CJMSP ブローカー個別プロパティファイルのキー]
378

imq.autocreate.topic.consumerFlowLimit
[CJMSP ブローカー共通プロパティファイルのキー]
373

imq.autocreate.topic.consumerFlowLimit
[CJMSP ブローカー個別プロパティファイルのキー]
378

imq.hostname [CJMSP ブローカー共通プロパティ
ファイルのキー] 372

imq.hostname [CJMSP ブローカー個別プロパティ
ファイルのキー] 375

imq.instanceconfig.version [CJMSP ブローカー個
別プロパティファイルのキー] 379

imq.jms.tcp.port [CJMSP ブローカー個別プロパ
ティファイルのキー] 375

imq.metrics.interval [CJMSP ブローカー共通プロ
パティファイルのキー] 373

imq.metrics.interval [CJMSP ブローカー個別プロ
パティファイルのキー] 378

imq.persist.file.sync.enabled [CJMSP ブローカー
共通プロパティファイルのキー] 373

imq.persist.file.sync.enabled [CJMSP ブローカー
個別プロパティファイルのキー] 377

imq.portmapper.port [CJMSP ブローカー個別プロ
パティファイルのキー] 376

J

j2eeApplicationTable 714

j2eeContainerTable 705

J2EE アプリケーションを実行するシステムで指定で
きるタグ 233

J2EE アプリケーションを実行するシステムで使用する
ファイルの一覧 173

J2EE サーバ共通用メッセージマッピングファイル463

J2EE サーバ個別用メッセージマッピングファイル464

J2EE サーバで使用するファイル 21

J2EE サーバの JavaVM のシステムプロパティに設定
できるパラメタ 313

J2EE サーバの拡張パラメタ 317

J2EE サーバのカスタマイズ用キー 35

webserver.jsp.tld.validation 121

J2EE サーバの互換モードを設定するパラメタ 277

J2EE サーバ用 Web アプリケーションプロパティファ
イル 23

J2EE サーバ用オプション定義ファイル 24

J2EE サーバ用オプション定義を設定するパラメタ311

J2EE サーバ用セキュリティポリシーファイル 133

J2EE サーバ用ユーザプロパティファイル 34

J2EE サーバ用ユーザプロパティを設定するパラメタ
278

java.endorsed.dirs 540

java.naming.factory.initial [J2EE サーバのカスタ
マイズ用キー] 94

java.naming.factory.initial [Java アプリケーショ
ン] 533

java.naming.factory.initial [バッチサーバ用のキー]
158

java.naming.factory.object 541

java.naming.factory.state 541

java.naming.provider.url [Java アプリケーション]
533

java.naming.provider.url [バッチアプリケーション
用のキー] 171

java.rmi.server.hostname [Management Server
環境設定ファイルのキー] 421

java.security.policy 542

javax.rmi.CORBA.PortableRemoteObjectClass
541

javax.rmi.CORBA.StubClass 541

javax.rmi.CORBA.UtilClass 541

javax.xml.datatype.DatatypeFactory [Java アプ
リケーション] 534

javax.xml.parsers.DocumentBuilderFactory
[Java アプリケーション] 534

javax.xml.parsers.SAXParserFactory [Java アプリケーション] 534

javax.xml.transform.TransformerFactory [Java アプリケーション] 534

javax.xml.validation.SchemaFactory¥:http¥://
www.w3.org/2001/XMLSchema [Java アプリケーション] 534

javax.xml.xpath.XPathFactory¥:http¥://
java.sun.com/jaxp/xpath/dom [Java アプリケーション] 534

Java アプリケーションで使用するファイル 509

Java アプリケーション用オプション定義ファイル 512

Java アプリケーション用ユーザプロパティファイル 519

JP1/IM 連携用システムログメッセージマッピングファイル 462

JP1/IM 連携用モニタ起動コマンドの設定ファイル 451

JP1 イベントへの変換 465

JP1 連携を設定するパラメタ 333

jvm.type [J2EE サーバ用オプション] 28, 318

jvm.type [Java アプリケーション用のキー] 517

jvm.type [バッチサーバ用のキー] 144

jvmTable 706

K

KeepAliveTimeout 263

L

lb.ACOS.privilegedexec.password [負荷分散機定義プロパティファイルのキー] 471, 484

lb.host [負荷分散機定義プロパティファイルのキー] 470, 484

lb.password [負荷分散機定義プロパティファイルのキー] 471, 484

lb.persistence.cookie-insert.cookieName [ティア別プロパティファイルのキー] 482

lb.persistence.method [ティア別プロパティファイルのキー] 482

lb.port [負荷分散機定義プロパティファイルのキー] 471, 484

lb.properties 183

lb.protocol [負荷分散機定義プロパティファイルのキー] 471, 484

lb.timeout [負荷分散機定義プロパティファイルのキー] 471, 484

lb.type [負荷分散機定義プロパティファイルのキー] 470, 484

lb.user [負荷分散機定義プロパティファイルのキー] 471, 484

lb.use [ティア別プロパティファイルのキー] 481

lb.virtualserver.host [ティア別プロパティファイルのキー] 482

lb.virtualserver.name [ティア別プロパティファイルのキー] 481

lb.virtualserver.port [ティア別プロパティファイルのキー] 482

Listen 263

LogLevel 264

M

maction.<Management アクション ID>.command [Management アクション実行用プロパティファイルのキー] 433

maction.<Management アクション ID>.exclusive_time [Management アクション実行用プロパティファイルのキー] 434

maction.<Management アクション ID>.max_executable_actions [Management アクション実行用プロパティファイルのキー] 434

maction.<Management アクション ID>.timeout.forced_stop [Management アクション実行用プロパティファイルのキー] 433

maction.<Management アクション ID>.timeout [Management アクション実行用プロパティファイルのキー] 433

maction.message.<メッセージ ID>.mactions [Management アクション実行用プロパティファイルのキー] 434

maction.mevent.receiving_info.keep_size.max [Management アクション実行用プロパティファイルのキー] 435

maction.properties 433
 maction.server.<論理サーバ名>.mactions
 [Management アクション実行用プロパティファイルのキー] 434
 maction.tier.<Web システム名>.<物理ティア種別名>.mactions [Management アクション実行用プロパティファイルのキー] 434
 maction.unit.<Web システム名>.<サービスユニット名>.mactions [Management アクション実行用プロパティファイルのキー] 434
 mail.mime.charset [J2EE サーバのカスタマイズ用キー] 131
 mail.mime.decodefilename [J2EE サーバのカスタマイズ用キー] 132
 mail.mime.decodeparameters [J2EE サーバのカスタマイズ用キー] 132
 mail.mime.decodetext.strict [J2EE サーバのカスタマイズ用キー] 131
 mail.mime.encodeeol.ignoremissingboundaryparameter [J2EE サーバのカスタマイズ用キー] 132
 mail.mime.encodeeol.strict [J2EE サーバのカスタマイズ用キー] 131
 mail.mime.encodefilename [J2EE サーバのカスタマイズ用キー] 131
 mail.mime.encodeparameters [J2EE サーバのカスタマイズ用キー] 132
 mail.mime.multipart.ignoremissingendboun [J2EE サーバのカスタマイズ用キー] 132
 Management Server 環境設定ファイル 413
 Management Server 管理ファイル用退避対象定義ファイル 442
 Management Server 用オプション定義ファイル 424
 Management Server 用環境変数定義ファイル 427
 Management Server 用メッセージマッピングファイル 462
 Management アクション実行用プロパティファイル 433
 Management イベント発行用プロパティファイル 437
 Management イベント発行用プロパティを設定するパラメタ 332
 Management イベント発行用メッセージ ID リストファイル 439
 manager.<論理サーバ名>.jp1event.system.mapping.properties 464
 manager.cfg 429
 manager.j2ee.compat [J2EE サーバのカスタマイズ用キー] 132
 manager.jp1event.system.filtering.severity.alert 333
 manager.jp1event.system.filtering.severity.critical 334
 manager.jp1event.system.filtering.severity.emergency 334
 manager.jp1event.system.filtering.severity.error 334
 manager.jp1event.system.filtering.severity.information 334
 manager.jp1event.system.filtering.severity.notice 334
 manager.jp1event.system.filtering.severity.warning 335
 manager.jp1event.system.mapping.properties 463
 manager.jp1event.user.filtering.enabled 335
 manager.jp1event.user.filtering.filter 335
 manager.jp1event.user.mapping.level.config 335
 manager.jp1event.user.mapping.level.fine 335
 manager.jp1event.user.mapping.level.finer 336
 manager.jp1event.user.mapping.level.finest 336
 manager.jp1event.user.mapping.level.info 336
 manager.jp1event.user.mapping.level.severe 336
 manager.jp1event.user.mapping.level.warning 337
 manager.mevent.logical_server_name
 [Management アクション実行用プロパティファイルのキー] 437
 manager.mevent.message_id.list 333
 manager.mevent.message_id.list [Management アクション実行用プロパティファイルのキー] 438
 manager.mevent.retry.interval 333

manager.mevent.retry.interval [Management アクション実行用プロパティファイルのキー] 438

manager.mevent.retry.limit 333

manager.mevent.retry.limit [Management アクション実行用プロパティファイルのキー] 438

manager.mevent.send.host [Management アクション実行用プロパティファイルのキー] 437

manager.mevent.send.max 333

manager.mevent.send.max [Management アクション実行用プロパティファイルのキー] 438

manager.mevent.send.port [Management アクション実行用プロパティファイルのキー] 437

manager.mevent.send.timeout 333

manager.mevent.send.timeout [Management アクション実行用プロパティファイルのキー] 438

manager.mevent.sender.bind.host 333

manager.mevent.sender.bind.host [Management アクション実行用プロパティファイルのキー] 438

manager.web.send_request_method 259

Manager 設定ファイル 429

Manager で使用するファイル 386

Manager で使用するファイルの一覧 387

MaxClients 264

MaxClients.worker 266

messageDrivenBeanTable 757

mngagent.<実サーバ名>.properties 411

mngagent.agent_name [運用監視エージェントプロパティファイルのキー] 411

mngagent.connector.host 332

mngagent.connector.host [運用監視エージェントプロパティファイルのキー] 412

mngagent.connector.port 332

mngagent.connector.port [運用監視エージェントプロパティファイルのキー] 412

mngagent.domain_name [運用監視エージェントプロパティファイルのキー] 411

mngagent.log.filename [運用監視エージェントプロパティファイルのキー] 412

mngagent.log.filesize 332

mngagent.log.filesize [運用監視エージェントプロパティファイルのキー] 412

mngsvr.jp1event.event_server_name [Management Server 環境設定ファイルのキー] 421

mngsvr.myhost.name [Management Server 環境設定ファイルのキー] 421

mngsvrmonitor.browser [JP1/IM 連携用モニタ起動コマンドの設定ファイルのキー] 451

mngsvrmonitor.connect.host [JP1/IM 連携用モニタ起動コマンドの設定ファイルのキー] 452

mngsvrmonitor.connect.password [JP1/IM 連携用モニタ起動コマンドの設定ファイルのキー] 452

mngsvrmonitor.connect.userid [JP1/IM 連携用モニタ起動コマンドの設定ファイルのキー] 452

mngsvrutil.apply_user [mngsvrutil コマンドのクライアント側共通定義ファイルのキー] 450

mngsvrutil.auth.failed_wait [mngsvrutil コマンドのサーバ側定義ファイルのキー] 447

mngsvrutil.auth.log_challenged [mngsvrutil コマンドのサーバ側定義ファイルのキー] 447

mngsvrutil.auth.log_denied [mngsvrutil コマンドのサーバ側定義ファイルのキー] 448

mngsvrutil.auth.log_failed [mngsvrutil コマンドのサーバ側定義ファイルのキー] 448

mngsvrutil.auth.log_succeeded [mngsvrutil コマンドのサーバ側定義ファイルのキー] 448

mngsvrutil.compat.monitoring_tree [mngsvrutil コマンドのサーバ側定義ファイルのキー] 448

mngsvrutil.connect.host [mngsvrutil コマンドのクライアント側定義ファイルのキー] 444

mngsvrutil.connect.password [mngsvrutil コマンドのクライアント側定義ファイルのキー] 445

mngsvrutil.connect.userid [mngsvrutil コマンドのクライアント側定義ファイルのキー] 445

mngsvrutil.output.file [mngsvrutil コマンドのクライアント側定義ファイルのキー] 445

mngsvrutil.output.format [mngsvrutil コマンドのクライアント側定義ファイルのキー] 445

mngsvrutil.output.suppress_header [mngsvrutil コマンドのクライアント側定義ファイルのキー] 445

mngsvrutil.session.max_inactive_interval
[mngsvrutil コマンドのサーバ側定義ファイルの
キー] 449

mngsvrutil.target_kind [mngsvrutil コマンドのク
ライアント側定義ファイルのキー] 446

mngsvrutil.target_name [mngsvrutil コマンドのク
ライアント側定義ファイルのキー] 445

mngsvrutil.target_server_name [mngsvrutil コマ
ンドのクライアント側定義ファイルのキー] 446

mngsvrutilcl.properties 449

mngsvrutil コマンドのクライアント側共通定義ファ
イル 449

mngsvrutil コマンドのクライアント側定義ファイル
443

mngsvrutil コマンドのサーバ側定義ファイル 447

mserver.cfg 424

mserver.jp1event.system.mapping.properties
462

mserver.properties 413

mserverenv.cfg 427

mstartup.force.watchtime 256

mstartup.no 256

mstartup.premised.serverName 257

mstartup.restartcnt 257

mstartup.retrywait 257

mstartup.start.watchtime 257

mstartup.watchtime 257

O

org.omg.CORBA.ORBClass 542

org.omg.CORBA.ORBSingletonClass 542

org.omg.PortableInterceptor.ORBInitializerClass
.com.hitachi.software.ejb.security.std.intercept
or.SecurityClientInterceptorInit 542

org.w3c.dom.DOMImplementationSourceList
[Java アプリケーション] 535

org.xml.sax.driver [Java アプリケーション] 535

org.xml.sax.parser [Java アプリケーション] 535

P

PRFID 341

prfspool 341

PrfTraceBufferSize 341

PrfTraceCount 341

PrfTraceFileSize 341

PrfTraceLevel 341

Q

queueTable 759

R

rasetup.cfg 471

rasetup コマンド用設定ファイル 471

realservername 257

RequestDivideFileNum 273

RequestDivideTimeDifference 273

RequestDivideTimeInterval 273

requestUrlTable 738

RequestWraparoundFileNum 273

RequestWraparoundFilesize 273

resourceAdapterTable 743

S

SecurityManager の使用を設定するパラメタ 337

server.policy 133, 159

ServerName 263

servletTable 736

SetBy 261

setup.CC.group [管理者設定用プロパティファイル
のキー] 492

setup.CC.user [管理者設定用プロパティファイルの
キー] 492

setup.cfg (セットアップウィザード用設定ファイル)
452

setup.log.dir [rasetup コマンド用設定ファイルの
キー] 472

setup.log.dir [セットアップウィザード用設定ファ
イルのキー] 453

setup.log.filenum [rasetup コマンド用設定ファ
イルのキー] 473

setup.log.filenum [セットアップウィザード用設定ファイルのキー] 453

setup.log.filesize [rasetup コマンド用設定ファイルのキー] 473

setup.log.filesize [セットアップウィザード用設定ファイルのキー] 453

setup.log.style [rasetup コマンド用設定ファイルのキー] 473

setup.log.time [rasetup コマンド用設定ファイルのキー] 473

setup.maintenance.log.filenum [rasetup コマンド用設定ファイルのキー] 473

setup.maintenance.log.filenum [セットアップウィザード用設定ファイルのキー] 454

setup.maintenance.log.filesize [rasetup コマンド用設定ファイルのキー] 473

setup.maintenance.log.filesize [セットアップウィザード用設定ファイルのキー] 454

setup.manager.group [管理者設定用プロパティファイルのキー] 492

setup.mng.administrator.password [管理者設定用プロパティファイルのキー] 491

setup.mng.administrator.userid [管理者設定用プロパティファイルのキー] 491

sinaviagent.cfg 473

sinaviagent.log.dir [サーバ通信エージェント用オプション定義ファイル] 474

sinaviagent.log.filenum [サーバ通信エージェント用オプション定義ファイル] 475

sinaviagent.log.filesize [サーバ通信エージェント用オプション定義ファイル] 475

sinaviagent.log.level [サーバ通信エージェント用オプション定義ファイル] 475

sinaviagent.log.style [サーバ通信エージェント用オプション定義ファイル] 476

sinaviagent.log.time [サーバ通信エージェント用オプション定義ファイル] 476

sinaviagent.maintenance.log.filenum [サーバ通信エージェント用オプション定義ファイル] 477

sinaviagent.maintenance.log.filesize [サーバ通信エージェント用オプション定義ファイル] 477

sinaviagent.process.consolelog.enabled [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.process.consolelog.filenum [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.process.consolelog.filesize [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.process.consolelog.style [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.process.consolelog.time [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.service.log.filenum [サーバ通信エージェント用オプション定義ファイル] 476

sinaviagent.service.log.filesize [サーバ通信エージェント用オプション定義ファイル] 476

sinaviagent.service.log.level [サーバ通信エージェント用オプション定義ファイル] 476

sinaviagent.service.maintenance.log.filenum [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.service.maintenance.log.filesize [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.snactl.log.filenum [サーバ通信エージェント用オプション定義ファイル] 477

sinaviagent.snactl.log.filesize [サーバ通信エージェント用オプション定義ファイル] 477

sinaviagent.snactl.log.level [サーバ通信エージェント用オプション定義ファイル] 477

sinaviagent.snactl.maintenance.log.filenum [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.snactl.maintenance.log.filesize [サーバ通信エージェント用オプション定義ファイル] 478

sinaviagent.spool.dir [サーバ通信エージェント用オプション定義ファイル] 474

sinaviagent.work.dir [サーバ通信エージェント用オプション定義ファイル] 475

snapshot ログ収集対象定義ファイル 497

snapshot ログ収集対象定義ファイルで使用できる変数 498

StartServers 264

StartServers.worker 267

statefulSessionBeanHomeMethodTable 769
statefulSessionBeanHomeTable 723
statefulSessionBeanLocalComponentMethodTable 774
statefulSessionBeanLocalComponentTable 752
statefulSessionBeanLocalHomeMethodTable 772
statefulSessionBeanLocalHomeTable 750
statefulSessionBeanRemoteMethodTable 770
statefulSessionBeanRemoteTable 724
statefulSessionBeanTable 720
statelessSessionBeanHomeMethodTable 760
statelessSessionBeanHomeTable 717
statelessSessionBeanLocalComponentMethodTable 766
statelessSessionBeanLocalComponentTable 748
statelessSessionBeanLocalHomeMethodTable 764
statelessSessionBeanLocalHomeTable 747
statelessSessionBeanRemoteMethodTable 762
statelessSessionBeanRemoteTable 719
statelessSessionBeanTable 715

T

ThreadsPerChild 265
ThreadsPerChild.worker 266
tier.properties 480
tier.startup.no [ティア別プロパティファイルのキー] 482
tier.target.os [ティア別プロパティファイルのキー] 482
tierlb.properties 483
transactionTable 708

U

use.security 337
User 265
user.env.variable 258
usrconf 355

USRCONF_JVM_ARGS [サーバ管理コマンド用オプション定義ファイルのキー (UNIX 用)] 355
USRCONF_JVM_ARGS [サーバ管理コマンド用オプション定義ファイルのキー (Windows 用)] 358
USRCONF_JVM_CLASSPATH [サーバ管理コマンド用オプション定義ファイルのキー (Windows 用)] 358
USRCONF_JVM_CLPATH [サーバ管理コマンド用オプション定義ファイルのキー (UNIX 用)] 355
USRCONF_JVM_LIBPATH [サーバ管理コマンド用オプション定義ファイルのキー (UNIX 用)] 355
USRCONF_JVM_LIBPATH [サーバ管理コマンド用オプション定義ファイルのキー (Windows 用)] 358
usrconf.bat 357
usrconf.cfg 24, 142, 165, 512
usrconf.properties 34, 146, 170, 360, 519

V

vbj.java2iop.jvm.maxHeapSize 301
vbj.java2iop.jvm.maxHeapSize [J2EE サーバのカスタマイズ用キー] 95
vbj.java2iop.jvm.minHeapSize 301
vbj.java2iop.jvm.minHeapSize [J2EE サーバのカスタマイズ用キー] 95
vbroker.agent.enableLocator [J2EE サーバのカスタマイズ用キー] 96, 316
vbroker.agent.enableLocator [Java アプリケーション] 535
vbroker.agent.enableLocator [バッチサーバ用のキー] 158
vbroker.agent.port [J2EE サーバのカスタマイズ用キー] 96
vbroker.agent.port [バッチサーバ用のキー] 158
vbroker.ce.iop.ccm.htc.readerPerConnection [J2EE サーバのカスタマイズ用キー] 96
vbroker.ce.iop.ccm.htc.readerPerConnection [Java アプリケーション] 536
vbroker.ce.iop.ccm.htc.readerPerConnection [バッチサーバ用のキー] 158
vbroker.ce.iop.ccm.htc.threadStarter [J2EE サーバのカスタマイズ用キー] 96

vbroker.ce.iiopt.ccm.htc.threadStarter [バッチサーバ用のキー] 158

vbroker.orb.htc.comt.entryCount [J2EE サーバのカスタマイズ用キー] 97, 316

vbroker.orb.htc.comt.entryCount [Java アプリケーション] 536

vbroker.orb.htc.comt.entryCount [バッチサーバ用のキー] 158

vbroker.orb.htc.comt.fileCount [J2EE サーバのカスタマイズ用キー] 97, 316

vbroker.orb.htc.comt.fileCount [Java アプリケーション] 536

vbroker.orb.htc.comt.fileCount [バッチサーバ用のキー] 158

vbroker.orb.htc.mtr.trace [Java アプリケーション] 536

vbroker.orb.htc.tracePath [J2EE サーバのカスタマイズ用キー] 97

vbroker.orb.htc.tracePath [J2EE サーバのカスタマイズ用キー] 316

vbroker.orb.htc.tracePath [Java アプリケーション] 536

vbroker.orb.htc.tracePath [サーバ管理コマンドのカスタマイズ用キー] 365

vbroker.orb.htc.tracePath [バッチサーバ用のキー] 158

vbroker.se.iiopt_tp.host [J2EE サーバのカスタマイズ用キー] 97

vbroker.se.iiopt_tp.host [バッチサーバ用のキー] 159

vbroker.se.iiopt_tp.proxyHost [J2EE サーバのカスタマイズ用キー] 98

vbroker.se.iiopt_tp.scm.iiopt_tp.listener.port [J2EE サーバのカスタマイズ用キー] 98

vbroker.se.iiopt_tp.scm.iiopt_tp.listener.port [サーバ管理コマンドのカスタマイズ用キー] 366

vbroker.se.iiopt_tp.scm.iiopt_tp.listener.port [バッチサーバ用のキー] 159

vbroker.se.iiopt_ts.proxyHost [J2EE サーバのカスタマイズ用キー] 98

version.properties 485

vm.interface.version [インタフェースバージョン定義ファイルのキー] 486

vm.target.edition [インタフェースバージョン定義ファイルのキー] 486

vmi.apply_user [仮想サーバマネージャのクライアント共通設定プロパティファイルのキー] 488

vmi.connect.m [仮想サーバマネージャのクライアント共通設定プロパティファイルのキー] 489

vmi.connect.m [仮想サーバマネージャのクライアント設定プロパティファイルのキー] 490

vmi.connect.p [仮想サーバマネージャのクライアント共通設定プロパティファイルのキー] 489

vmi.connect.p [仮想サーバマネージャのクライアント設定プロパティファイルのキー] 490

vmi.connect.u [仮想サーバマネージャのクライアント共通設定プロパティファイルのキー] 489

vmi.connect.u [仮想サーバマネージャのクライアント設定プロパティファイルのキー] 490

vmi.connect.wait [仮想サーバマネージャのクライアント共通設定プロパティファイルのキー] 489

vmi.connect.wait [仮想サーバマネージャのクライアント設定プロパティファイルのキー] 490

vmi.jp1.base.auth.enabled [仮想サーバマネージャプロパティファイルのキー] 487

vmi.properties 486

vmi.serverconnectionagent.connect.timeout [仮想サーバマネージャプロパティファイルのキー] 487

vmi.serverconnectionagent.exec.timeout [仮想サーバマネージャプロパティファイルのキー] 487

vmi.serverconnectionagent.socket.timeout [仮想サーバマネージャプロパティファイルのキー] 487

vmi.spool.dir [仮想サーバマネージャプロパティファイルのキー] 487

vmiclient.properties 488

vmsetup.properties 490

W

webApplicationTable 731

webContainerTable 777

webserver.application.lower_version [J2EE サーバのカスタマイズ用キー] 99

webserver.connector.ajp13.bind_host [バッチサーバ用のキー] 159
 webserver.connector.ajp13.port [Management Server 環境設定ファイルのキー] 422
 webserver.connector.ajp13.port [バッチサーバ用のキー] 159
 webserver.connector.http.bind_host [J2EE サーバのカスタマイズ用キー] 100
 webserver.connector.http.bind_host [Management Server 環境設定ファイルのキー] 422
 webserver.connector.http.permitted.hosts [J2EE サーバのカスタマイズ用キー] 100
 webserver.connector.http.permitted.hosts [Management Server 環境設定ファイルのキー] 422
 webserver.connector.http.port [Management Server 環境設定ファイルのキー] 423
 webserver.connector.limit.max_parameter_count [J2EE サーバのカスタマイズ用キー] 100
 webserver.connector.limit.max_post_form_data [J2EE サーバのカスタマイズ用キー] 101
 webserver.connector.nio_http.backlog [J2EE サーバのカスタマイズ用キー] 102
 webserver.connector.nio_http.bind_host [J2EE サーバのカスタマイズ用キー] 102
 webserver.connector.nio_http.hostname_lookups [J2EE サーバのカスタマイズ用キー] 102
 webserver.connector.nio_http.idle_thread_timeout [J2EE サーバのカスタマイズ用キー] 102
 webserver.connector.nio_http.keep_alive.max_requests [J2EE サーバのカスタマイズ用キー] 102
 webserver.connector.nio_http.keep_alive.timeout [J2EE サーバのカスタマイズ用キー] 103
 webserver.connector.nio_http.limit.max_headers [J2EE サーバのカスタマイズ用キー] 103
 webserver.connector.nio_http.limit.max_request_body [J2EE サーバのカスタマイズ用キー] 103
 webserver.connector.nio_http.limit.max_request_header [J2EE サーバのカスタマイズ用キー] 103
 webserver.connector.nio_http.limit.max_request_trailer 103
 webserver.connector.nio_http.limit.max_trailers 104
 webserver.connector.nio_http.max_connections [J2EE サーバのカスタマイズ用キー] 104
 webserver.connector.nio_http.max_servlet_execute_threads [J2EE サーバのカスタマイズ用キー] 104
 webserver.connector.nio_http.max_threads [J2EE サーバのカスタマイズ用キー] 104
 webserver.connector.nio_http.min_threads [J2EE サーバのカスタマイズ用キー] 104
 webserver.connector.nio_http.permitted.hosts [J2EE サーバのカスタマイズ用キー] 105
 webserver.connector.nio_http.port [J2EE サーバのカスタマイズ用キー] 105
 webserver.connector.nio_http.receive_timeout [J2EE サーバのカスタマイズ用キー] 105
 webserver.connector.nio_http.response.header.server [J2EE サーバのカスタマイズ用キー] 105
 webserver.connector.nio_http.send_timeout [J2EE サーバのカスタマイズ用キー] 105
 webserver.connector.nio_http.setQueryEncoding.enabled [J2EE サーバのカスタマイズ用キー] 105
 webserver.container.jaxws.webservice.no_webxml.enabled [J2EE サーバのカスタマイズ用キー] 106
 webserver.container.jaxws.webservice.wsee.no_webxml.enabled [J2EE サーバのカスタマイズ用キー] 107
 webserver.container.jaxws.webservice.wsee.warnname [J2EE サーバのカスタマイズ用キー] 107
 webserver.container.redirectWelcomeFile [J2EE サーバのカスタマイズ用キー] 107
 webserver.container.server_id.enabled [J2EE サーバのカスタマイズ用キー] 108
 webserver.container.server_id.name [J2EE サーバのカスタマイズ用キー] 108
 webserver.container.server_id.value [J2EE サーバのカスタマイズ用キー] 108
 webserver.container.servlet.default_mapping.enabled [J2EE サーバのカスタマイズ用キー] 109
 webserver.container.thread_control.queue_size [J2EE サーバのカスタマイズ用キー] 109

webserver.context.check_interval [J2EE サーバのカスタマイズ用キー] 110
 webserver.context.reload_delay_timeout [J2EE サーバのカスタマイズ用キー] 110
 webserver.context.stop_asyncwait_timeout [J2EE サーバのカスタマイズ用キー] 110
 webserver.context.update.interval [J2EE サーバのカスタマイズ用キー] 111
 webserver.dbsfo.attribute_data_size.max [J2EE サーバのカスタマイズ用キー] 111
 webserver.dbsfo.check_size.mode [J2EE サーバのカスタマイズ用キー] 111
 webserver.dbsfo.connector.name [J2EE サーバのカスタマイズ用キー] 112
 webserver.dbsfo.enabled [J2EE サーバのカスタマイズ用キー] 112
 webserver.dbsfo.exception_type_backcompat [J2EE サーバのカスタマイズ用キー] 112
 webserver.dbsfo.exclude.extensions [J2EE サーバのカスタマイズ用キー] 112
 webserver.dbsfo.exclude.uris [J2EE サーバのカスタマイズ用キー] 112
 webserver.dbsfo.integrity_mode.enabled [J2EE サーバのカスタマイズ用キー] 112
 webserver.dbsfo.negotiation.high_level [J2EE サーバのカスタマイズ用キー] 113
 webserver.dbsfo.session_read_only.uris [J2EE サーバのカスタマイズ用キー] 113
 webserver.dbsfo.thread_control_queue.enabled [J2EE サーバのカスタマイズ用キー] 113
 webserver.errorpage.stack_trace.enabled [J2EE サーバのカスタマイズ用キー] 113, 317
 webserver.http.request.encoding [J2EE サーバのカスタマイズ用キー] 114
 webserver.http.request.uri_decode.enabled [J2EE サーバのカスタマイズ用キー] 114
 webserver.http.response.encoding [J2EE サーバのカスタマイズ用キー] 114
 webserver.http2.enabled 115
 webserver.http2.initialWindowSize 115
 webserver.http2.keepAliveTimeout 115
 webserver.http2.max_headers 115
 webserver.http2.max_request_header 115
 webserver.http2.max_request_trailer 115
 webserver.http2.max_trailers 116
 webserver.http2.maxConcurrentStreamExecution 115
 webserver.http2.maxConcurrentStreams 115
 webserver.http2.readTimeout 116
 webserver.http2.streamReadTimeout 116
 webserver.http2.streamWriteTimeout 116
 webserver.http2.writeTimeout 116
 webserver.jsp.additional.import.list [J2EE サーバのカスタマイズ用キー] 116
 webserver.jsp.check_interval [J2EE サーバのカスタマイズ用キー] 117
 webserver.jsp.compile.backcompat [J2EE サーバのカスタマイズ用キー] 117
 webserver.jsp.debugging.enabled [J2EE サーバのカスタマイズ用キー] 122
 webserver.jsp.jsp_page.bom.enabled [J2EE サーバのカスタマイズ用キー] 121
 webserver.jsp.keepgenerated [J2EE サーバのカスタマイズ用キー] 118
 webserver.jsp.keepgenerated [J2EE サーバ用オプション] 23
 webserver.jsp.pageEncoding [J2EE サーバのカスタマイズ用キー] 118
 webserver.jsp.precompile.jsp_work_dir [J2EE サーバのカスタマイズ用キー] 118, 317
 webserver.jsp.tld.mapping.java_ee_tag_library.enabled [J2EE サーバのカスタマイズ用キー] 121
 webserver.jsp.translation.backcompat.customAction.declareVariable [J2EE サーバのカスタマイズ用キー] 119
 webserver.jsp.translation.backcompat.tag.noCheckRtexprvalue [J2EE サーバのカスタマイズ用キー] 119
 webserver.jsp.translation.backcompat.tag.rtexprvalueTerminate [J2EE サーバのカスタマイズ用キー] 119
 webserver.jsp.translation.backcompat.taglib.noCheckPrefix [J2EE サーバのカスタマイズ用キー] 120

webserver.jsp.translation.backcompat.useBean.noCheckClass [J2EE サーバのカスタマイズ用キー] 120

webserver.jsp.translation.customAction.ignoreCaseAttributeName [J2EE サーバのカスタマイズ用キー] 120

webserver.jsp.translation.useBean.noCheckDuplicateld [J2EE サーバのカスタマイズ用キー] 121

webserver.jsp.update.interval [J2EE サーバのカスタマイズ用キー] 120

webserver.prf.output_httpheader [J2EE サーバのカスタマイズ用キー] 122

webserver.servlet_api.exception.getCause.backcompat [J2EE サーバのカスタマイズ用キー] 122

webserver.servlet_api.unsupported.throwUnsupportedOperationException [J2EE サーバのカスタマイズ用キー] 123

webserver.ServletContainerInitializer_jar.include.path [J2EE サーバのカスタマイズ用キー] 123

webserver.session.cookie_config.http_only [J2EE サーバのカスタマイズ用キー] 124

webserver.session.cookie_config.name [J2EE サーバのカスタマイズ用キー] 124

webserver.session.delete_cookie.backcompat [J2EE サーバのカスタマイズ用キー] 125, 317

webserver.session.max.log_interval [J2EE サーバのカスタマイズ用キー] 125, 317

webserver.session.max.throwHttpSessionLimitExceededException [J2EE サーバのカスタマイズ用キー] 126

webserver.session.server_id.enabled [J2EE サーバのカスタマイズ用キー] 126, 317

webserver.session.server_id.value [J2EE サーバのカスタマイズ用キー] 126

webserver.session.tracking_mode [J2EE サーバのカスタマイズ用キー] 126

webserver.sessionManager.<コンテキストルート名>.className [J2EE サーバのカスタマイズ用キー] 127

webserver.sessionManager.<コンテキストルート名>.maxActiveSessions [J2EE サーバのカスタマイズ用キー] 127

webserver.sessionManager.<コンテキストルート名>.properties.<プロパティ> [J2EE サーバのカスタマイズ用キー] 128

webserver.sessionManager.backgroundProcessor.delay [J2EE サーバのカスタマイズ用キー] 128

webserver.sessionManager.backgroundProcessor.threads [J2EE サーバのカスタマイズ用キー] 128

webserver.shutdown.port [Management Server 環境設定ファイルのキー] 423

webserver.static_content.cache.enabled [J2EE サーバのカスタマイズ用キー] 128

webserver.static_content.cache.filesize.threshold [J2EE サーバのカスタマイズ用キー] 129

webserver.static_content.cache.size [J2EE サーバのカスタマイズ用キー] 129

webserver.static_content.encoding.extension [J2EE サーバのカスタマイズ用キー] 129

webserver.work.clean [J2EE サーバのカスタマイズ用キー] 130

webserver.work.directory [J2EE サーバのカスタマイズ用キー] 130

webserver.xml.validate [J2EE サーバのカスタマイズ用キー] 131

webserver.xml.validate [J2EE サーバ用オプション] 24

Web アプリケーションで使用するファイルの一覧 549

Web アプリケーション用 DD (web.xml) 781

Web サーバの J2EE サーバとの連携方法を設定するパラメタ 259

Web サーバ用定義を設定するパラメタ 261

Web システムの属性定義 233, 245

い

インタフェースバージョン定義ファイル 485

う

運用監視エージェントプロパティファイル 411

運用監視エージェントプロパティを設定するパラメタ 332

運用管理エージェント自動起動用設定ファイル 405

運用管理エージェント設定ファイル 408

運用管理エージェントプロパティファイル 389
運用管理エージェント用オプション定義ファイル 406

か

拡張 MIB オブジェクト定義ファイル 705
拡張子と MIME タイプの対応づけ 781
各論理サーバ共通で指定できるパラメタ 256
仮想化システムで使用するファイル 468
仮想化システムで使用するファイルの一覧 469
仮想サーバグループ情報ファイル 492
仮想サーバマネージャ側の負荷分散機接続設定プロパティファイル 470
仮想サーバマネージャのクライアント共通設定プロパティファイル 488
仮想サーバマネージャのクライアント設定プロパティファイル 489
仮想サーバマネージャプロパティファイル 486
簡易構築定義ファイル 186
簡易構築定義ファイルおよび構成変更定義ファイルに指定できるタグ 233
監査ログ定義ファイル 503
監査ログの設定で使用するファイル 501
管理者設定用プロパティファイル 490

く

クライアント共通設定プロパティファイル 179
クライアント設定プロパティファイル 176
グローバルトランザクション 60

こ

構成変更定義ファイル 222
コンパイル対象外リストファイル 550

さ

サーバ管理コマンドで使用するファイル 353
サーバ管理コマンドのカスタマイズ用キー 361
サーバ管理コマンドの予約済みキー 170, 361
サーバ管理コマンド用システムプロパティファイル 360
サーバ設定プロパティファイル 175

サーバ通信エージェント用オプション定義ファイル 473

サービスユニットの定義 240, 248

サービスユニット・ホスト追加用の構成変更定義ファイル 224

作業ディレクトリ 26

し

実行結果リストファイル 551

使用するサーバと指定するパラメタの参照先の対応 277

す

スケールアウト用ホスト定義ファイル 230

て

ティア側の負荷分散機接続設定プロパティファイル 483

ティア別プロパティファイル 480

は

バッチアプリケーション用オプション定義ファイル 165

バッチアプリケーション用ユーザプロパティファイル 170

バッチアプリケーションを実行するシステムで指定できるタグ 245

バッチアプリケーションを実行するシステムで使用するファイルの一覧 174

バッチサーバで使用するファイル 140

バッチサーバの JavaVM のシステムプロパティに設定できるパラメタ 329

バッチサーバの拡張パラメタ 330

バッチサーバのカスタマイズ用キー 147

バッチサーバ用オプション定義ファイル 142

バッチサーバ用オプション定義を設定するパラメタ 327

バッチサーバ用セキュリティポリシーファイル 159

バッチサーバ用ユーザプロパティファイル 146

バッチサーバ用ユーザプロパティを設定するパラメタ 318

ふ

負荷分散機定義プロパティファイル 183

負荷分散機の定義 234

物理ティアの定義 237, 246

ほ

保護区リストファイル 138, 163

ホストの定義 244, 251

ら

ライトトランザクション 60

り

リバースプロキシを設定するパラメタ 274

ろ

ログの運用で使用するファイル 495

論理 CTM で指定できるパラメタ 345

論理 CTM ドメインマネジャで指定できるパラメタ
343

論理 J2EE サーバで指定できるパラメタ 277

論理 Web サーバで指定できるパラメタ 259

論理サーバ参照定義ファイル 227

論理サーバのパラメタ変更用の構成変更定義ファイル
223

論理スマートエージェントで指定できるパラメタ 350

論理ネーミングサービスで指定できるパラメタ 352

論理パフォーマンストレーサで指定できるパラメタ
341

論理ユーザサーバで指定できるパラメタ 351