

高信頼化システム監視機能 HA モニタ kit for Oracle

解説・手引・操作書

3000-9-135-30

前書き

■ 対象製品

P-F852C-E1111 HA モニタ kit for Oracle 01-04 (適用 OS : Red Hat(R) Enterprise Linux(R) server 6(x86), Red Hat(R) Enterprise Linux(R) server 6(x86_64), Red Hat(R) Enterprise Linux(R) server 7(x86_64), Red Hat(R) Enterprise Linux(R) server 8(x86_64))

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

■ 輸出時の注意

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

なお、不明な場合は、弊社担当営業にお問い合わせください。

■ 商標類

HITACHI, HA モニタは、株式会社日立製作所の商標または登録商標です。

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

Intel は、アメリカ合衆国および / またはその他の国における Intel Corporation またはその子会社の商標です。

Linux は、Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。

Oracle と Java は、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。

Red Hat is a registered trademark of Red Hat, Inc. in the United States and other countries.

Red Hat 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。

Red Hat Enterprise Linux is a registered trademark of Red Hat, Inc. in the United States and other countries.

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。

UNIX は、The Open Group の商標です。

その他記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

■ 発行

2020 年 10 月 3000-9-135-30

■ 著作権

All Rights Reserved. Copyright (C) 2011, 2020, Hitachi, Ltd.

変更内容

変更内容 (3000-9-135-30) HA モニタ kit for Oracle 01-04

追加・変更内容	変更箇所
各パラメタは、実際の環境に合わせて設定することを記載した。	3.2.7(2), 3.3.1, 3.3.2, 3.3.3
Oracle Database 19c をサポートした。	-

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

このマニュアルは、HA モニタ kit for Oracle の機能、設定方法、および運用方法について説明したものです。このマニュアルを読むことで、HA モニタのユーザが、HA モニタ kit for Oracle を使用したシステム構成を設計・構築・運用し、Oracle を使用したシステムを監視できるようになることを目的としています。

■ 対象読者

システム管理者、およびオペレータの方を対象としています。システム管理者は、システムの導入、設計、構築をすることを想定しています。オペレータは、構築されたシステムで日々の運用をすることを想定しています。

また、次の知識をお持ちの方を前提としています。

- ハードウェアの知識
- OS の知識
- HA モニタの知識
- Oracle の知識

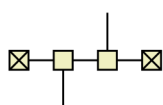
■ 図中で使用している記号

このマニュアルの図中で使用している記号を、次のように定義します。

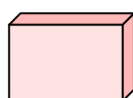
● ネットワーク



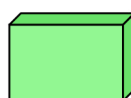
● バス型の LAN



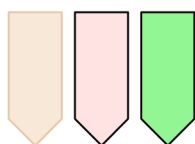
● 現用系・実行系のプログラム



● 予備系・待機系のプログラム



● 処理の流れ



目次

前書き	2
変更内容	3
はじめに	4

1	HA モニタ kit の概要	9
1.1	HA モニタ kit の目的	10
1.2	HA モニタ kit を適用できる構成	11
1.2.1	Oracle のシステム構成	11
1.2.2	Oracle の構成	13
1.2.3	系切り替え構成	13
1.3	HA モニタ kit の動作環境	14
1.3.1	必要なハードウェア	14
1.3.2	必要なソフトウェア	15
1.4	HA モニタとの関係	17
2	システムの管理	18
2.1	サーバの管理	19
2.1.1	障害検出の仕組み	19
2.1.2	検出できる障害の種類	21
2.2	処理の流れ	23
2.2.1	サーバの起動処理の流れ	23
2.2.2	サーバの起動失敗処理の流れ	26
2.2.3	サーバの停止処理の流れ	33
2.2.4	サーバの監視処理の流れ	36
2.2.5	サーバ障害時の系切り替え処理の流れ	38
3	システムの設計と構築	43
3.1	システムの設計	44
3.1.1	Oracle の構成と系切り替え単位の検討	44
3.1.2	リソース構成の検討	46
3.2	システムの構築	47
3.2.1	構築の流れ	47
3.2.2	ディレクトリ構成	49
3.2.3	構築に必要な設定項目の対応と設定の概要	50
3.2.4	Oracle のインストール	52

3.2.5	Oracle のセットアップ・データベースの構築	52
3.2.6	Oracle が使用する共有リソースの設定	54
3.2.7	HA モニタ kit の設定	54
3.2.8	HA モニタ kit コマンドの編集	59
3.2.9	定義ファイルの作成 (サーバ)	65
3.2.10	障害情報を収集するための設定	67
3.2.11	構築したシステムの動作確認	67
3.3	環境設定例	69
3.3.1	CDB (シングルテナント) 構成で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例	69
3.3.2	シングル構成 (HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合) で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例	74
3.3.3	シングル構成 (HA モニタのサーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合) で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例	79
4	サーバの起動・停止・監視コマンドの設定	84
4.1	サーバの起動・停止・監視コマンドの概要	85
4.2	サーバの起動コマンドの設定	86
4.2.1	環境変数	87
4.2.2	設定方法	87
4.2.3	サーバの起動コマンドのカスタマイズ	88
4.3	サーバの停止コマンドの設定	89
4.3.1	環境変数	90
4.3.2	設定方法	91
4.3.3	サーバの停止コマンドのカスタマイズ	91
4.4	サーバの監視コマンドの設定 (ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合)	93
4.4.1	サーバの監視コマンド	93
4.5	サーバの監視コマンドの設定 (patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合)	98
4.5.1	サーバの監視コマンド	98
4.5.2	サーバの監視コマンド実行シェル	101
4.6	サーバの起動・停止・監視コマンドの設定に使用するコマンド	103
	コマンドの説明で使用する見出し	103
	文法記述記号	104
	haorainsbgn (Oracle インスタンスの起動)	104
	haorainsend (Oracle インスタンスの停止)	105
	haorainsptl (Oracle インスタンスの状態確認)	106
	haorainsendsub (Oracle インスタンスの停止)	107
	haorainsptlsub (Oracle インスタンスの状態確認)	108
	haoralsnrbgn (Oracle リスナーの起動)	108

haoralsnrend (Oracle リスナーの停止)	109
haoralsnrptl (Oracle リスナーの状態確認)	110
haoralsnrendsub (Oracle リスナーの停止)	111
haoralsnrptlsub (Oracle リスナーの状態確認)	112
haoraenvcheck (制御用定義ファイルのチェック)	112

5 システムの運用 114

5.1	運用の流れ	115
5.2	Oracle インスタンス・Oracle リスナーの起動・停止・監視	116
5.2.1	Oracle インスタンスまたは Oracle リスナーの起動	116
5.2.2	Oracle インスタンスまたは Oracle リスナーの停止	116
5.2.3	Oracle インスタンスまたは Oracle リスナーの監視	116
5.3	Oracle インスタンス・Oracle リスナーの HA モニタを介さない起動・停止	118
5.3.1	Oracle インスタンス・Oracle リスナーを HA モニタを介さないで起動する	118
5.3.2	Oracle インスタンス・Oracle リスナーを HA モニタを介さないで停止する	119
5.4	サーバログの確認	120
5.4.1	ファイルの種類	120
5.4.2	ファイルサイズ	120
5.4.3	出力例	121
5.4.4	メッセージの言語環境	122
5.5	障害発生による系切り替え時の運用	123
5.5.1	障害情報を収集する	123
5.6	障害への対処	124
5.7	システムの変更	125
5.7.1	Oracle をアップグレードする	125
5.7.2	Oracle インスタンス・Oracle リスナーを追加する	125
5.7.3	Oracle インスタンス・Oracle リスナーを削除する	126
5.7.4	Oracle インスタンス・Oracle リスナーに関する設定を変更する	126
5.7.5	CDB (シングルテナント) 構成で PDB を unplug および plug する	127

6 メッセージ 130

6.1	メッセージの形式	131
6.1.1	メッセージの出力形式	131
6.1.2	メッセージの記述形式	131
6.2	メッセージの詳細	133

付録 145

付録 A	各バージョンの変更内容	146
付録 A.1	HA モニタ kit for Oracle 01-03 の変更内容 (3000-9-135-20)	146
付録 A.2	HA モニタ kit for Oracle 01-01 の変更内容 (3000-9-135-10)	146

付録 B	このマニュアルの参考情報	148
付録 B.1	関連マニュアル	148
付録 B.2	このマニュアルでの表記	148
付録 B.3	英略語	148
付録 B.4	KB (キロバイト) などの単位表記について	149

索引 150

1

HA モニタ kit の概要

この章では、HA モニタ kit の目的、HA モニタ kit を適用できる構成、HA モニタ kit の動作環境、および HA モニタとの関係について説明します。なお、このマニュアルでは、「HA モニタ kit for Oracle」を、「HA モニタ kit」と表記します。

1.2 HA モニタ kit を適用できる構成

HA モニタでは、システムの障害に備えて二つのシステムを用意し、一つで業務処理を実行させ、もう一つを待機させておきます。用意したこれらのシステムを系といいます。また、システム（系）で業務を実行するプログラムのことをサーバといいます。HA モニタは、このサーバ単位で系切り替えをします。

ここでは、HA モニタ kit を適用するに当たって、Oracle のシステム構成、Oracle の構成、および系切り替え構成について説明します。

1.2.1 Oracle のシステム構成

HA モニタ kit を適用して系切り替え構成にできる構成とできない構成を、次に示します。

HA モニタ kit を適用できる構成

Non-CDB（シングル）構成および CDB（シングルテナント）構成の Oracle と、災害からデータベースを保護できるディザスタリカバリ構成のプライマリ・データベースに、HA モニタ kit は適用できます。それぞれの構成の例を以降の図に示します。

図 1-2 HA モニタ kit を適用できる構成（Non-CDB（シングル）構成の Oracle）

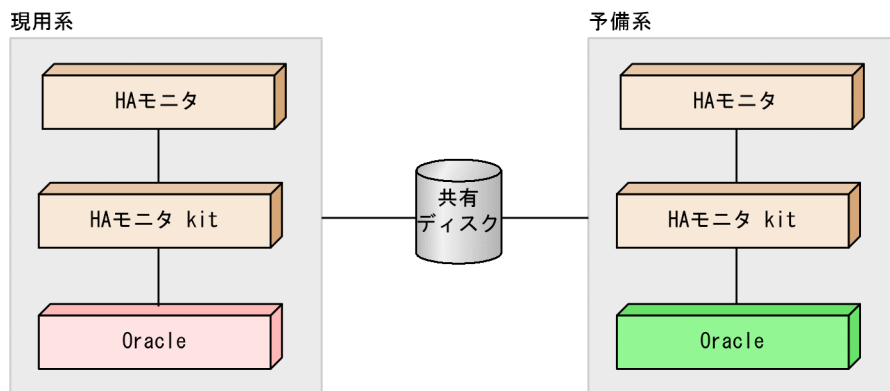


図 1-3 HA モニタ kit を適用できる構成（CDB（シングルテナント）構成の Oracle）

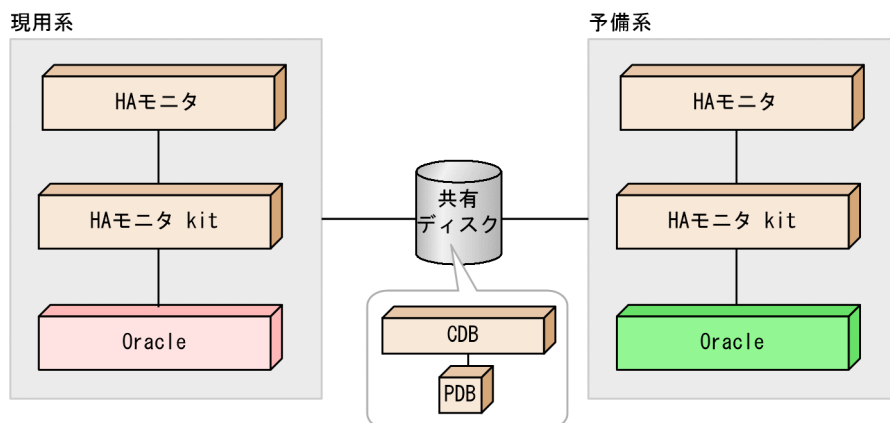
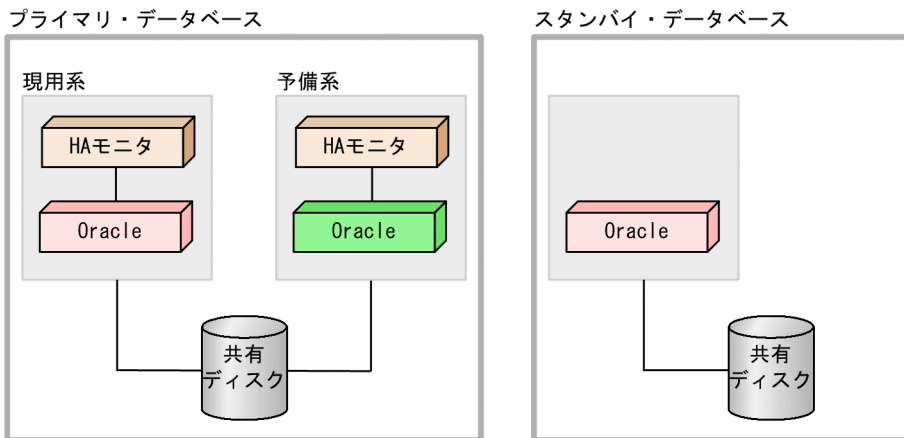


図 1-4 HA モニタ kit を適用できる構成 (ディザスタリカバリ構成のプライマリ・データベース)



HA モニタ kit を適用できない構成

Oracle RAC を使用した RAC 構成および CDB (マルチテナント) 構成には、HA モニタ kit を適用できません。それぞれの構成の例を次に示します。

図 1-5 HA モニタ kit を適用できない構成 (RAC 構成)

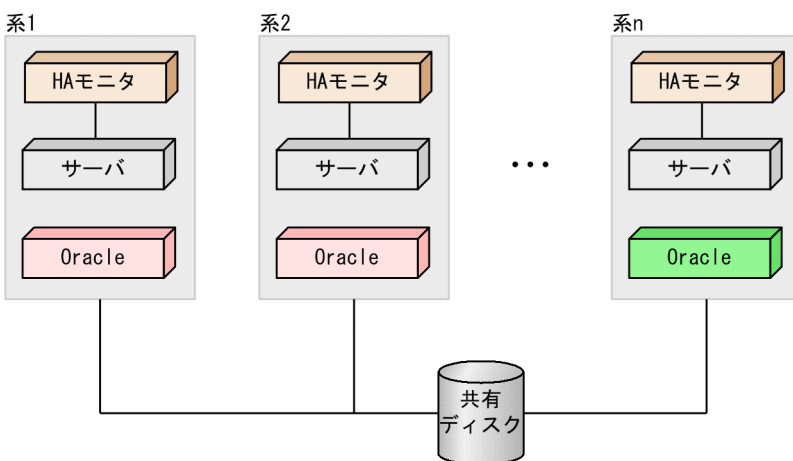
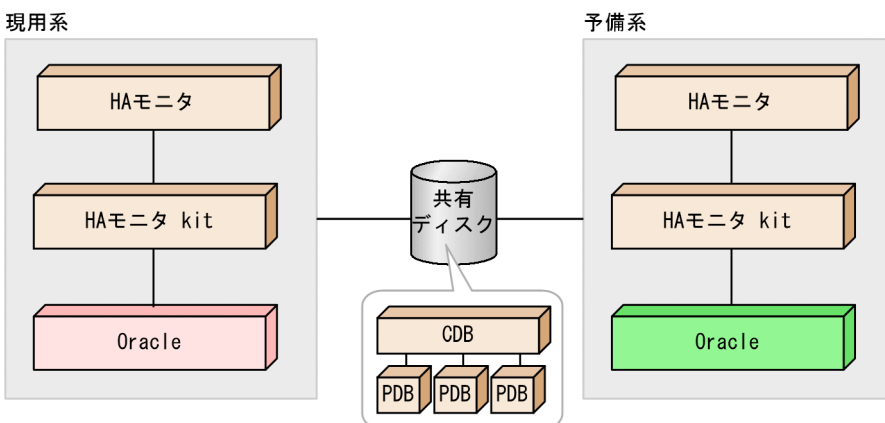


図 1-6 HA モニタ kit を適用できない構成 (CDB (マルチテナント) 構成)

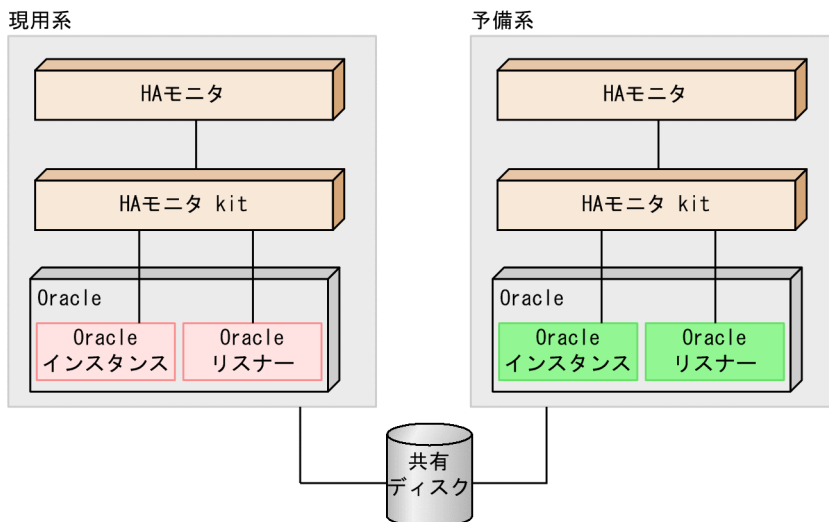


1.2.2 Oracle の構成

HA モニタ kit を適用したシステムでは、Oracle インスタンスと Oracle リスナーを系切り替えの対象とし、系切り替えの単位であるサーバとして扱います。Oracle インスタンスは、クライアントが業務を続行するために必要な要素としてデータベースに関連づけられる、共用メモリとプロセスの組み合わせです。Oracle リスナーは、クライアントの要求を Oracle インスタンスに中継する要素です。Oracle インスタンスとそのインスタンスの管理をする Oracle リスナーは、同一の系で稼働している必要があります。

Oracle の構成を次の図に示します。

図 1-7 Oracle の構成



HA モニタ kit では、Oracle インスタンスと Oracle リスナーを個別のサーバとして管理したり、複数個をまとめて一つのサーバとして管理したりすることもできます。詳細については、「[3.1.1 Oracle の構成と系切り替え単位の検討](#)」を参照してください。

1.2.3 系切り替え構成

HA モニタ kit は、HA モニタで対応している系切り替え構成で使用できます。ご使用の HA モニタのバージョンに応じて、使用できる系切り替え構成を確認してください。HA モニタの系切り替え構成については、マニュアル「HA モニタ Linux(R)編」の「系切り替え構成」の説明を参照してください。

なお、HA モニタの系切り替え構成では、サーバの運用方法にはサーバモードとモニタモードがあります。HA モニタ kit を適用する構成では、サーバ (Oracle) はモニタモードのサーバとして運用されます。

1.3 HA モニタ kit の動作環境

HA モニタ kit を使用したシステムを動作させるために必要なハードウェア、およびソフトウェアについて説明します。HA モニタを使用した一般的なシステムと共通の内容については、マニュアル「HA モニタ Linux(R)編」の「HA モニタの動作環境」の説明を参照してください。

1.3.1 必要なハードウェア

HA モニタ kit を使用したシステムを動作させるために必要なハードウェア構成は、HA モニタに依存します。詳細については、マニュアル「HA モニタ Linux(R)編」の「必要なハードウェア」の説明を参照してください。

ここでは、HA モニタ kit を使用する上で留意する必要があるハードウェアの構成要素について説明します。

(1) 共有ディスク

共有ディスクは、サーバが実行系と待機系とで共有するデータを格納するために使用します。Oracle では、データベースを構成するための次のファイルを格納します。

- サーバー・パラメータファイル
- 制御ファイル
- データファイル
- REDO ログファイル
- アーカイブログファイル

また、フラッシュリカバリ領域には、共有ディスク上のパスを指定してください。

Oracle を共有ディスクにインストールすると、ソフトウェアを実行系と待機系とで共有することもできます。また、Oracle はローカルディスクにインストールしてもかまいません。

(2) ローカルディスク

系間で共有しない、系固有のディスク装置として使用します。Oracle や HA モニタ kit をインストールします。Oracle は共有ディスクにインストールしてもかまいません。

Oracle のインストールについては、「[3.2.4 Oracle のインストール](#)」を参照してください。

(3) LAN

クライアントからデータベースに接続するために使用します。

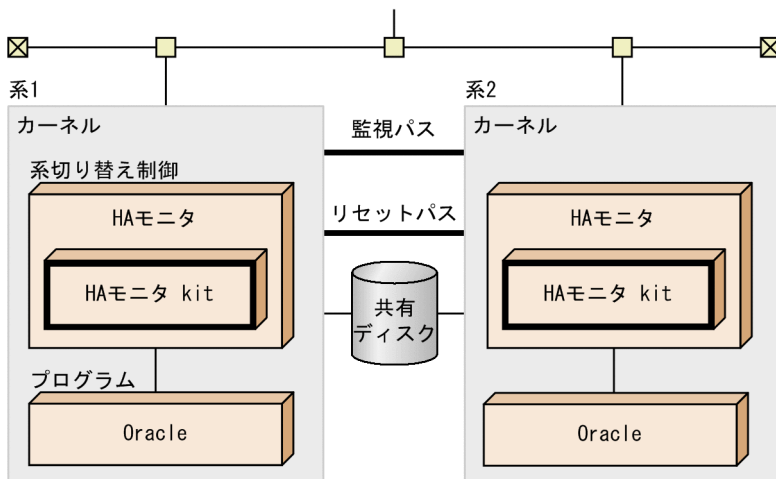
1.3.2 必要なソフトウェア

ここでは、HA モニタ kit を使用する上で留意する必要があるソフトウェアの構成と、注意事項について説明します。

(1) ソフトウェア構成

ソフトウェア構成を次の図に示します。

図 1-8 ソフトウェア構成



(a) カーネル

Linux(x86)

システムの基盤となるオペレーティングシステムです。

(b) 系切り替え制御

HA モニタ

系切り替えを制御します。

HA モニタ kit

HA モニタを前提として動作する、HA モニタのオプション製品です。

(c) プログラム

Oracle

リレーショナルデータベース管理システムです。HA モニタを使用したシステムでは、モニタモードのサーバとして管理されます。

(2) ソフトウェアに関する注意事項

ソフトウェアに関する注意事項を次に示します。

- ファイルシステムに ASM（自動ストレージ管理）を使用する場合は、HA モニタ kit を使用したクラスタ環境は構築できません。
- 次のオプション製品、および機能は使用できません。
 - Oracle Clusterware
 - Oracle Data Guard Broker
 - Oracle RAC
 - Oracle Restart

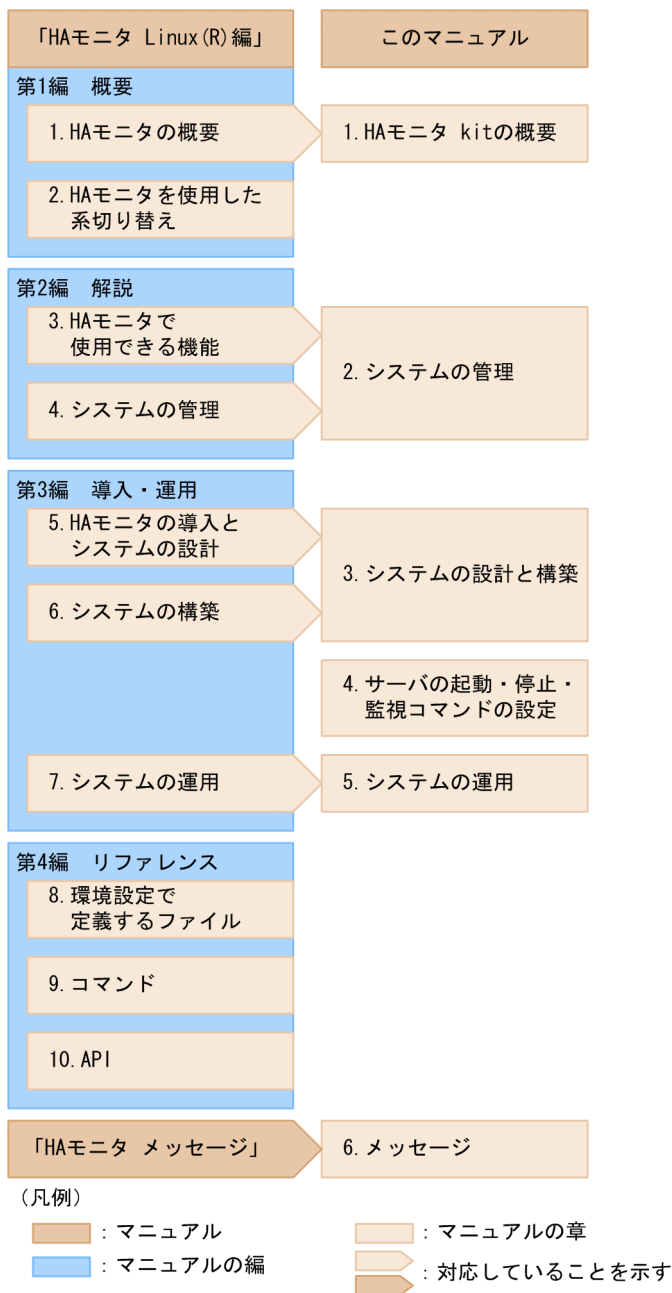
1.4 HA モニタとの関係

HA モニタ kit は、HA モニタのオプション製品です。このマニュアルでは、HA モニタ kit に特化した内容だけを記載しています。このため、マニュアル「HA モニタ Linux(R)編」を参照し、HA モニタ kit 固有の部分についてだけこのマニュアルを参照するようにしてください。

また、メッセージについては、HA モニタ kit 固有のメッセージはこのマニュアルを、HA モニタ固有のメッセージはマニュアル「HA モニタ メッセージ」を参照してください。

このマニュアルと HA モニタシリーズのマニュアルの対応を、次の図に示します。

図 1-9 このマニュアルと HA モニタシリーズのマニュアルの対応



2

システムの管理

この章では、HA モニタ kit を使用したシステムで、HA モニタがサーバをどのように管理しているかについて説明します。HA モニタ kit を使用したシステムの動作の詳細についても説明しています。

なお、系や共有リソースの管理については、マニュアル「HA モニタ Linux(R)編」の「システムの管理」の説明を参照してください。

2.1 サーバの管理

HA モニタ kit を使用することで、HA モニタは稼働中のサーバ（Oracle インスタンスまたは Oracle リスナー）を監視し、Oracle の障害を検出できます。ここでは、障害検出の仕組みや検出できる障害の種類について説明します。

2.1.1 障害検出の仕組み

HA モニタ kit を使用したシステムでは、サーバの監視コマンドで実行サーバ（Oracle インスタンスまたは Oracle リスナー）を監視します。

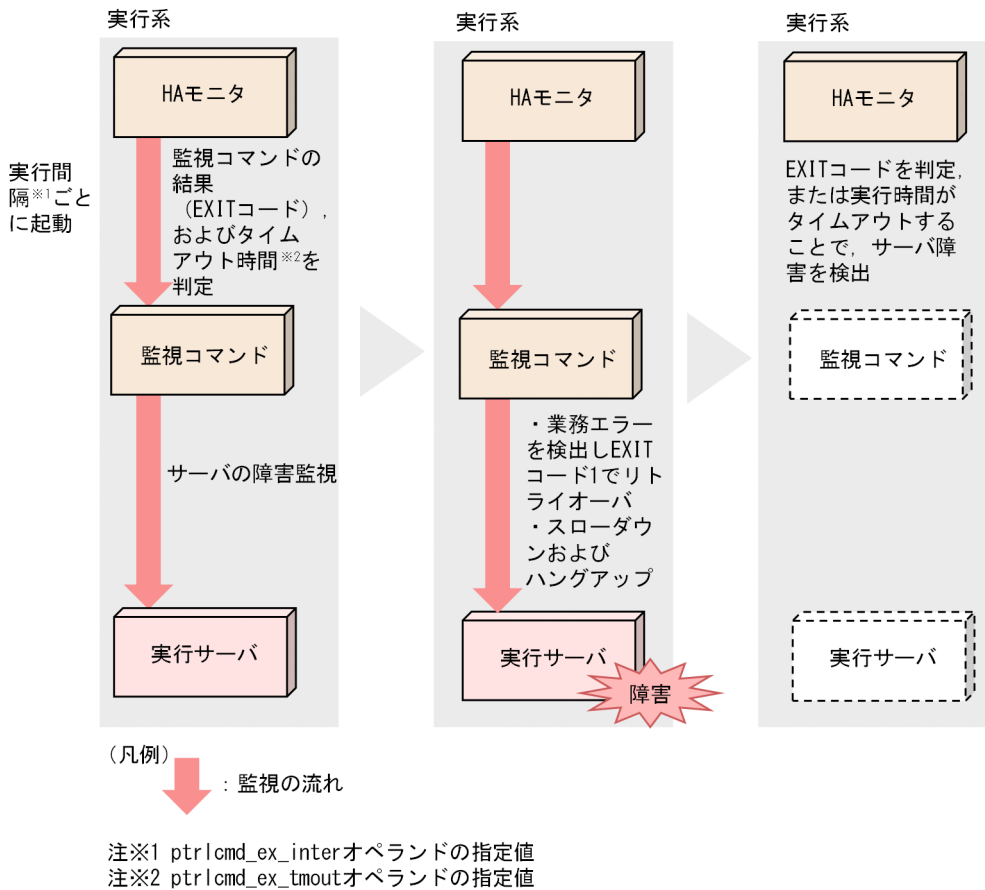
(1) サーバ対応の環境設定の `ptrlcmd_ex` オペランドでサーバの監視コマンドを指定する場合

指定された実行間隔（サーバ対応の環境設定の `ptrlcmd_ex_inter` オペランドの指定値）ごとにサーバの監視コマンドを実行し、サーバの監視コマンドの終了結果（EXIT コード）を HA モニタが判定します。

HA モニタは、サーバの監視コマンドの終了結果（EXIT コード）に応じた動作をします。詳細は、マニュアル「HA モニタ Linux(R)編」を参照してください。また、サーバ対応の環境設定の `ptrlcmd_ex_tmout` オペランドが指定されている場合、サーバの監視コマンド実行時、指定したタイムアウト時間内にコマンドが終了しないと、スローダウンとして検知します。この場合は、サーバ障害が発生したと判断します。障害を検出すると、設定に従って、HA モニタが系切り替え、または再起動をして業務を続行します。

サーバの監視コマンドによる障害検出の流れ（サーバ対応の環境設定の `ptrlcmd_ex` オペランドでサーバの監視コマンドを指定する場合）を、次の図に示します。

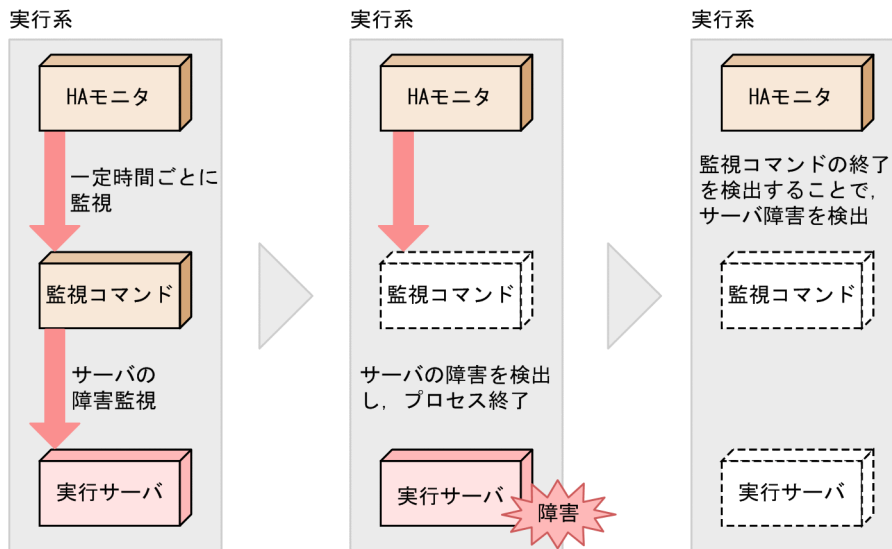
図 2-1 サーバの監視コマンドによる障害検出の流れ（サーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合）



(2) サーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンドを指定する場合

サーバの監視コマンドのプロセスを HA モニタが監視して障害を検出します。障害を検出すると、設定に従って、HA モニタが系切り替え、または再起動をして業務を続行します。サーバの監視コマンドによる障害検出の流れ（サーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンドを指定する場合）を、次の図に示します。

図 2-2 サーバの監視コマンドによる障害検出の流れ（サーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンドを指定する場合）



(凡例) ↓ : 監視の流れ

HA モニタ kit が提供するサーバの監視コマンドでは、Oracle インスタンスの場合、状態やスローダウン、プロセスの生存を監視します。Oracle リスナーの場合は、スローダウンやプロセスの生存を監視します。監視方法は、それぞれの制御用定義ファイルに設定します。制御用定義ファイルの設定については、「3.2.7(2) 制御用定義ファイルの設定」を参照してください。

2.1.2 検出できる障害の種類

ここでは、HA モニタ kit を使用したシステムで検出できる障害の種類について説明します。

(1) HA モニタ kit が検出する障害

HA モニタ kit は、次の表に示す Oracle の障害を検出できます。

表 2-1 HA モニタ kit が検出する障害

検出する障害	説明
Oracle インスタンスの障害	次の状態を Oracle インスタンスの障害として検出します。 <ul style="list-style-type: none"> • Oracle インスタンスのプロセスが存在しない • SQL によるデータベースへの問い合わせ異常
Oracle インスタンスのスローダウン	SQL による問い合わせ応答が一定時間ない状態を、Oracle インスタンスのスローダウンとして検出します*。
Oracle リスナーの障害	Oracle リスナーのプロセスが存在しない状態を、Oracle リスナーの障害として検出します。

検出する障害	説明
Oracle リスナーのスローダウン	Oracle リスナーのプロセスが一定時間応答しない状態を、Oracle リスナーのスローダウンとして検出します※。

注※ HA モニタ kit が問い合わせを開始してからの時間で検出します。

(2) HA モニタが検出する障害

次の表に示す Oracle の障害は、HA モニタ kit ではなく HA モニタが検出します。

表 2-2 HA モニタが検出する障害

検出する障害	説明
ネットワーク障害	HA モニタの LAN の監視機能を用いて検出します。LAN の監視機能については、マニュアル「HA モニタ Linux(R)編」の「LAN の監視および障害時の自動系切り替え」の説明を参照してください。
系障害	HA モニタが alive メッセージの途絶によって検出します。系障害の検出については、マニュアル「HA モニタ Linux(R)編」の「系障害の検出」を参照してください。

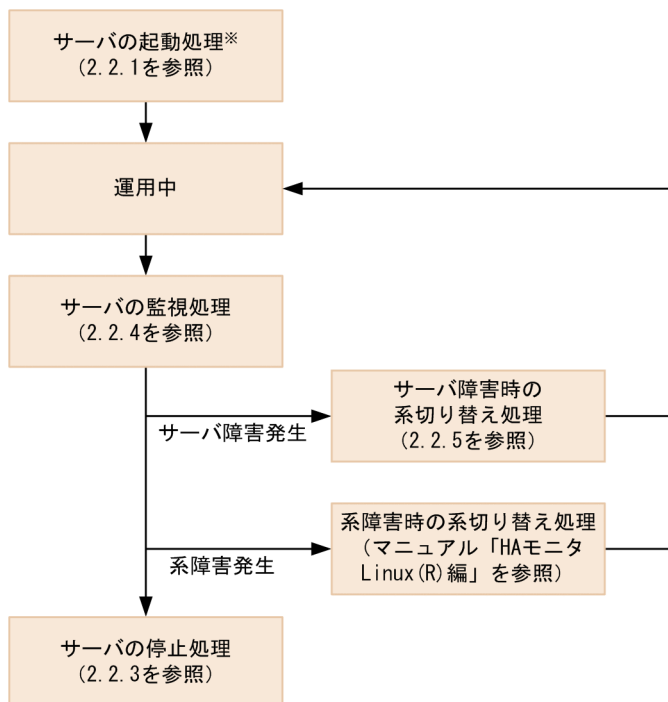
2.2 処理の流れ

ここでは、HA モニタおよび HA モニタ kit がする処理の流れについて、すべての共有リソースを含めて説明します。

この節で説明する処理の流れでは、HA モニタおよび HA モニタ kit が出力するメッセージ ID、ならびに発行するイベント ID を記載しているため、どのタイミングでメッセージ ID やイベントが発行されるかを調べるときにお読みください。また、ユーザコマンドを実行するタイミング、およびユーザコマンドに渡される引数も記載しています。ユーザコマンド作成時の参考にしてください。

系切り替え構成での運用の流れと、処理の流れで説明する内容との対応を次の図に示します。

図 2-3 系切り替え構成での運用の流れと、処理の流れで説明する内容との対応



注※ サーバの起動に失敗した場合の処理の流れについては、「2.2.2 サーバの起動失敗処理の流れ」を参照してください。

サーバの起動・停止・監視方法については、「5.2 Oracle インスタンス・Oracle リスナーの起動・停止・監視」を参照してください。

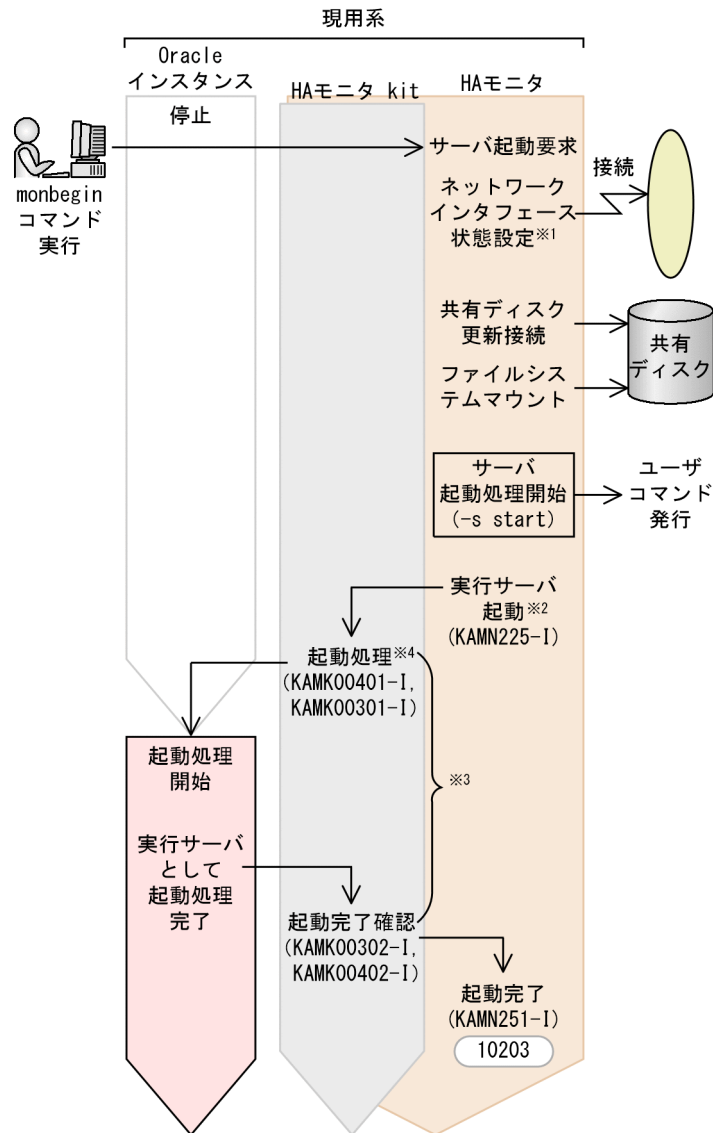
2.2.1 サーバの起動処理の流れ

サーバ (Oracle インスタンスまたは Oracle リスナー) の起動時に、HA モニタおよび HA モニタ kit がする処理の流れについて説明します。なお、この項で示す図は、一つのサーバに Oracle インスタンスまたは Oracle リスナーだけを対応させた場合の例です。

(1) Oracle インスタンスの起動処理の流れ

Oracle インスタンスの起動時に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-4 Oracle インスタンスの起動処理の流れ



- (凡例) (KAMNnnn-n) : HAモニタから出力されるメッセージID
 (KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID
 (nn...nn) : ユーザコマンドに渡されるパラメタ
 nnnnn : 発行されるJP1のイベントID (000nnnnn)

注※1 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

注※2 HA モニタが、サーバの起動コマンドを実行します。

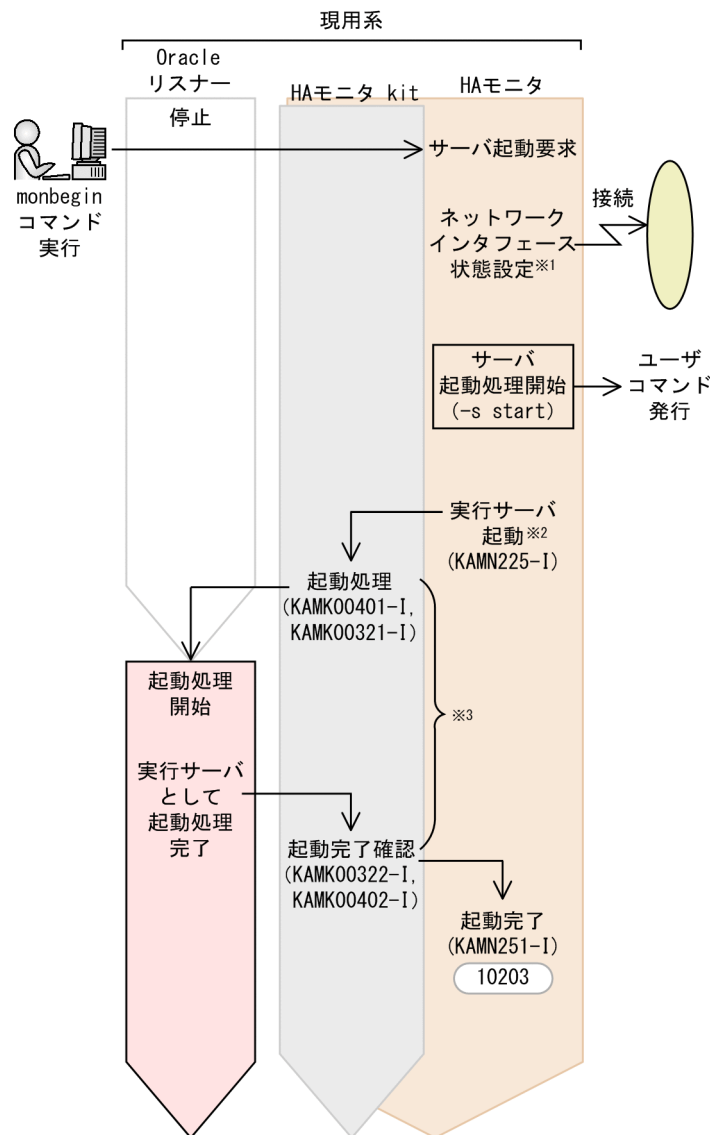
注※3 一つのサーバで複数の Oracle インスタンスを制御する場合、この部分の処理の流れを、制御する Oracle インスタンスの数だけ繰り返します。

注※4 CDB (シングルテナント) 構成の場合、モニタモードのサーバ起動コマンド (monbegin コマンド) でインスタンスを起動すると、PDB の自動起動の設定に関係なく、PDB も自動で起動します。

(2) Oracle リスナーの起動処理の流れ

Oracle リスナーの起動時に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-5 Oracle リスナーの起動処理の流れ



- (凡例) (KAMNnnn-n) : HA モニタから出力されるメッセージ ID
 (KAMKnnnnn-n) : HA モニタ kit から出力されるメッセージ ID
 (nn...nn) : ユーザコマンドに渡されるパラメタ
 (nnnnn) : 発行されるJPIのイベントID (000nnnnn)

注※1 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

注※2 HA モニタが、サーバの起動コマンドを実行します。

注※3 一つのサーバで複数の Oracle リスナーを制御する場合、この部分の処理の流れを、制御する Oracle リスナーの数だけ繰り返します。

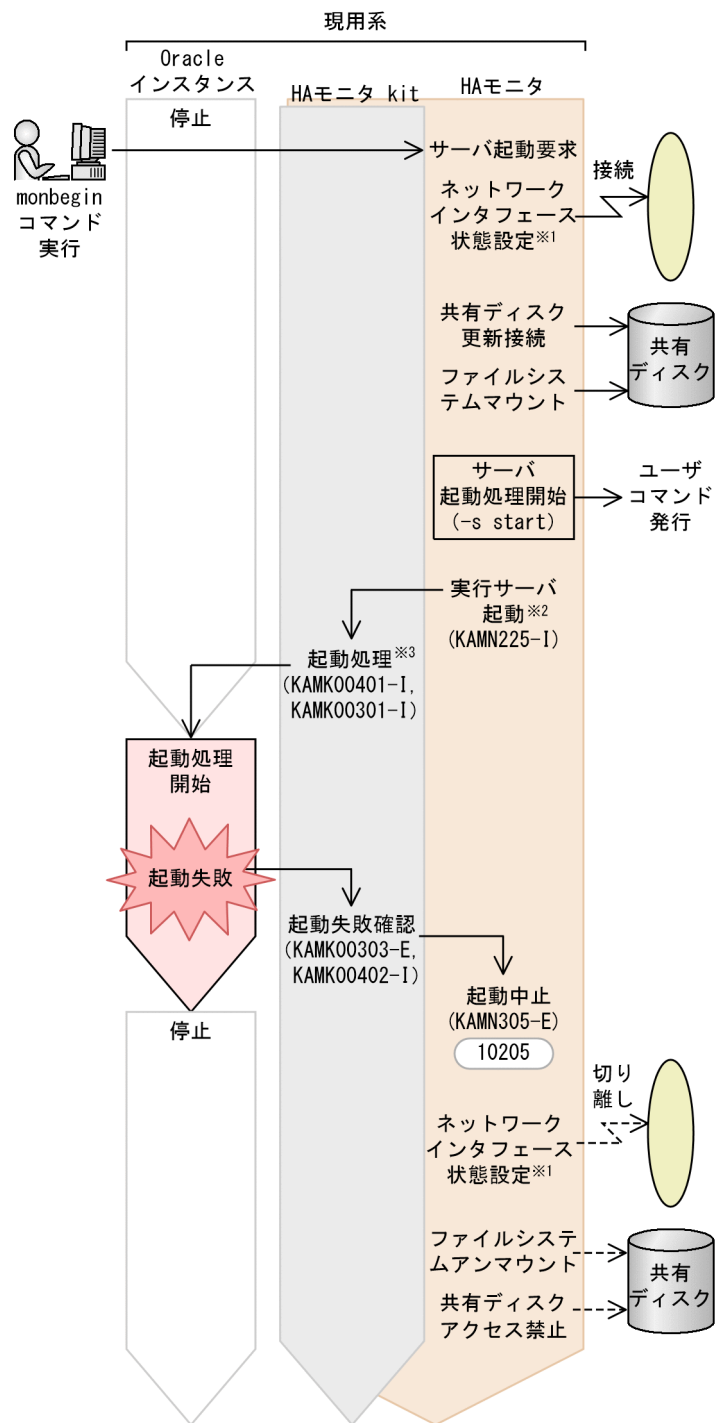
2.2.2 サーバの起動失敗処理の流れ

サーバ (Oracle インスタンスまたは Oracle リスナー) の起動失敗時に、HA モニタおよび HA モニタ kit がする処理の流れについて説明します。なお、この項で示す図は、一つのサーバに Oracle インスタンスまたは Oracle リスナーだけを対応させた場合の例です。

(1) Oracle インスタンスの起動失敗処理の流れ

Oracle インスタンスの起動失敗時に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-6 Oracle インスタンスの起動失敗処理の流れ



- (凡例) (KAMNnnn-n) : HA モニタから出力されるメッセージID
 (KAMKnnnnn-n) : HA モニタ kit から出力されるメッセージID
 (nn...nn) : ユーザコマンドに渡されるパラメタ
 nnnnn : 発行されるJP1のイベントID (000nnnnn)

注※1 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

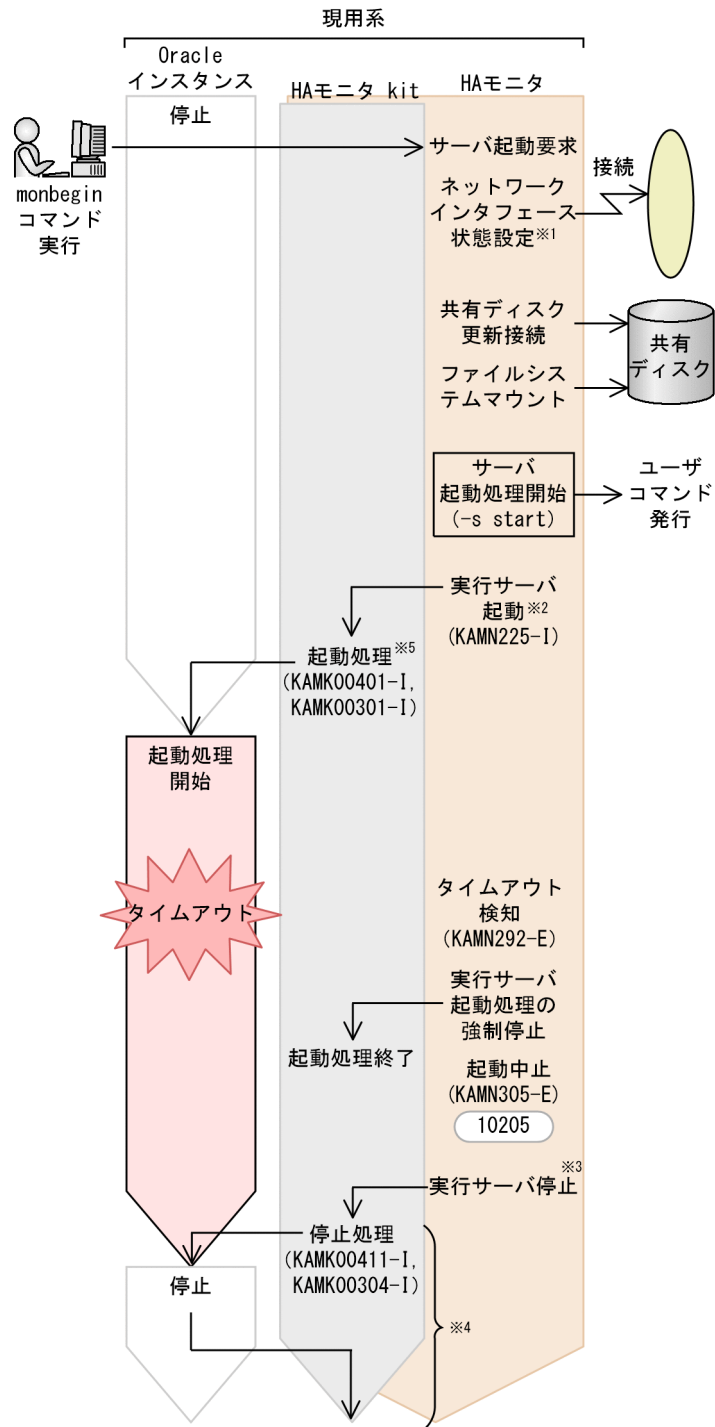
注※2 HA モニタが、サーバの起動コマンドを実行します。

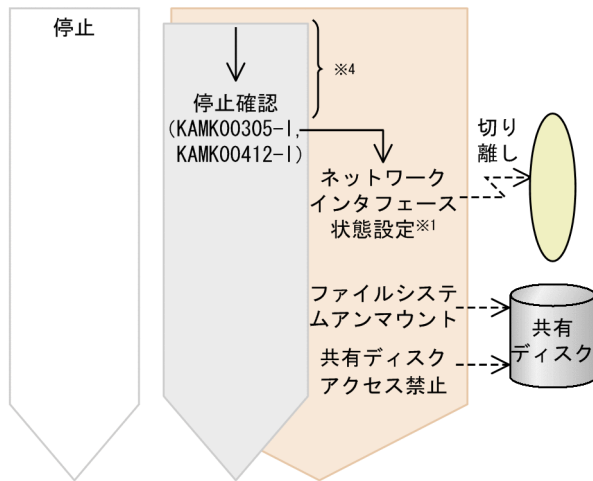
注※3 CDB (シングルテナント) 構成の場合、モニタモードのサーバ起動コマンド (monbegin コマンド) でインスタンスを起動すると、PDB の自動起動の設定に関係なく、PDB も自動で起動します。

(2) Oracle インスタンスの起動タイムアウト時の処理の流れ

Oracle インスタンスが起動時にタイムアウトした場合に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-7 Oracle インスタンスの起動タイムアウト時の処理の流れ





- (凡例) (KAMNnnn-n) : HAモニタから出力されるメッセージID
 (KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID
 (nn...nn) : ユーザコマンドに渡されるパラメタ
 nnnnn : 発行されるJP1のイベントID (000nnnnn)

注※1 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

注※2 HA モニタが、サーバの起動コマンドを実行します。

注※3 HA モニタが、サーバの停止コマンドを実行します。ただし、HA モニタの環境設定の termcmd_at_abort オペランドに use を指定したときに行います。

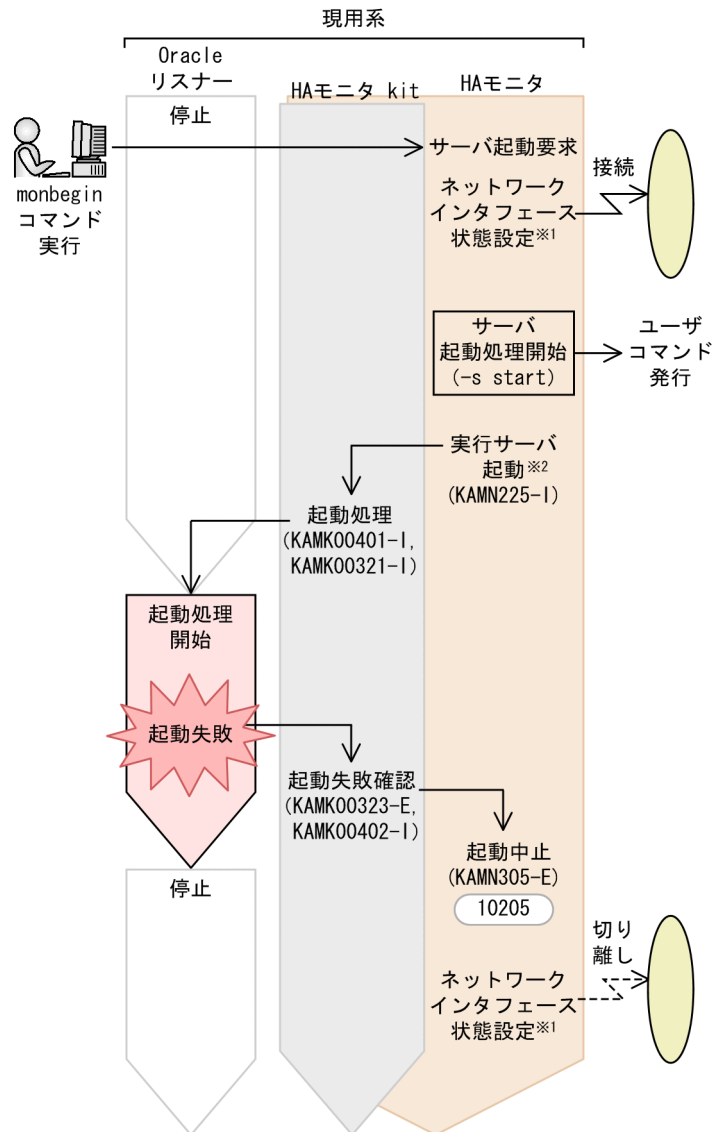
注※4 一つのサーバで複数の Oracle インスタンスを制御する場合、この部分の処理の流れを、制御する Oracle インスタンスの数だけ繰り返します。

注※5 CDB (シングルテナント) 構成の場合、モニタモードのサーバ起動コマンド (monbegin コマンド) でインスタンスを起動すると、PDB の自動起動の設定に関係なく、PDB も自動で起動します。

(3) Oracle リスナーの起動失敗処理の流れ

Oracle リスナーの起動失敗時に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-8 Oracle リスナーの起動失敗処理の流れ



- (凡例) (KAMNnnn-n) : HA モニタから出力されるメッセージID
 (KAMKnnnnn-n) : HA モニタ kit から出力されるメッセージID
 (nn...nn) : ユーザコマンドに渡されるパラメタ
 nnnnn : 発行されるJP1のイベントID (000nnnnn)

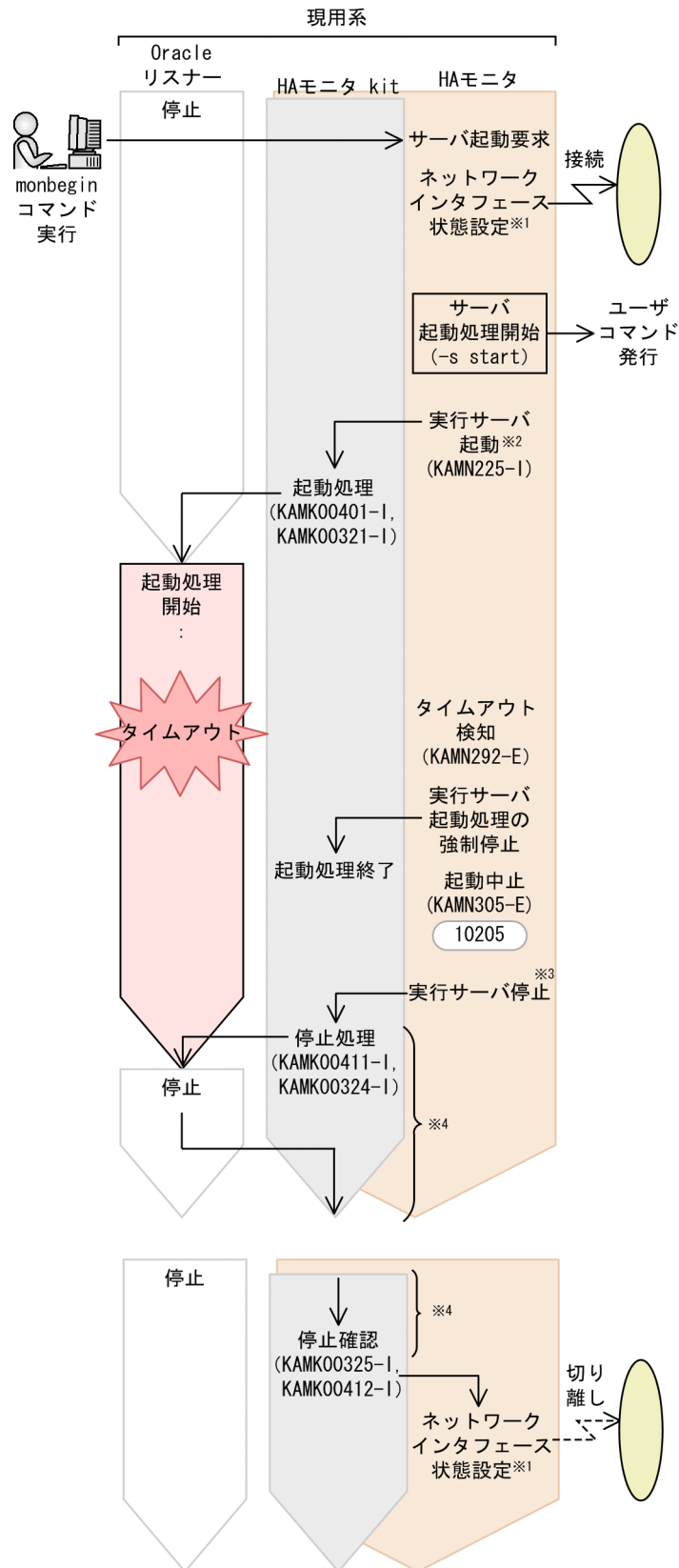
注※1 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

注※2 HA モニタが、サーバの起動コマンドを実行します。

(4) Oracle リスナーの起動タイムアウト時の処理の流れ

Oracle リスナーが起動時にタイムアウトした場合に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-9 Oracle リスナーの起動タイムアウト時の処理の流れ



- (凡例) (KAMNnnn-n) : HAモニタから出力されるメッセージID
(KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID
(nn...nn) : ユーザコマンドに渡されるパラメタ
nnnnn : 発行されるJP1のイベントID (000nnnn)

注※1 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

注※2 HA モニタが、サーバの起動コマンドを実行します。

注※3 HA モニタが、サーバの停止コマンドを実行します。ただし、HA モニタの環境設定の `termcmd_at_abort` オペランドに `use` を指定したときに行います。

注※4 一つのサーバで複数の Oracle リスナーを制御する場合、この部分の処理の流れを、制御する Oracle リスナーの数だけ繰り返します。

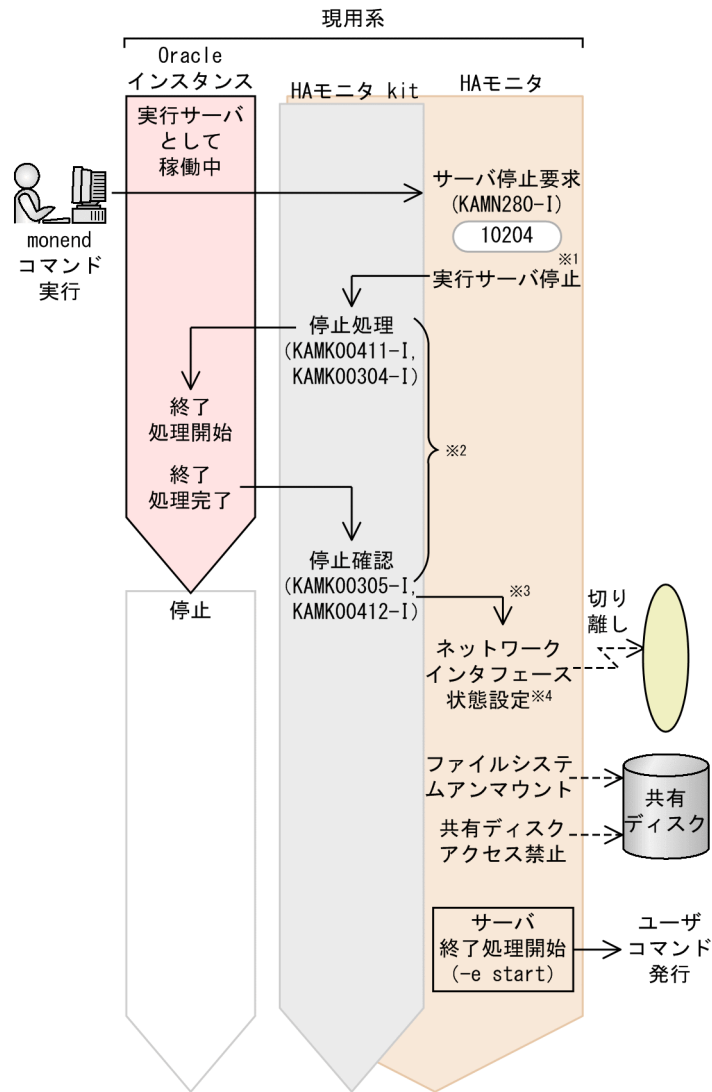
2.2.3 サーバの停止処理の流れ

サーバ (Oracle インスタンスまたは Oracle リスナー) の停止時に、HA モニタおよび HA モニタ kit がする処理の流れについて説明します。なお、この項で示す図は、一つのサーバに Oracle インスタンスまたは Oracle リスナーだけを対応させた場合の例です。

(1) Oracle インスタンスの停止処理の流れ

Oracle インスタンスの停止時に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-10 Oracle インスタンスの停止処理の流れ



- (凡例) (KAMNnnn-n) : HAモニタから出力されるメッセージID
 (KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID
 (nn...nn) : ユーザコマンドに渡されるパラメタ
 nnnnn : 発行されるJP1のイベントID (000nnnnn)

注※1 HA モニタが、サーバの停止コマンドを実行します。

注※2 一つのサーバで複数の Oracle インスタンスを制御する場合、この部分の処理の流れを、制御する Oracle インスタンスの数だけ繰り返します。

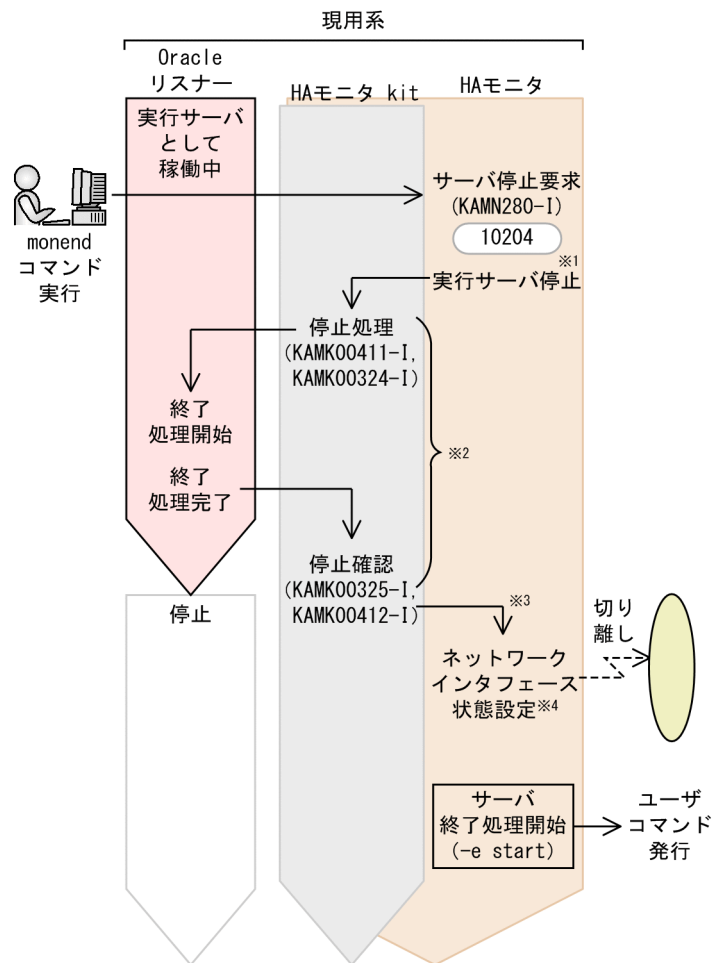
注※3 HA モニタは、サーバの停止コマンドの完了を待ちます。

注※4 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

(2) Oracle リスナーの停止処理の流れ

Oracle リスナーの停止時に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-11 Oracle リスナーの停止処理の流れ



- (凡例) (KAMNnnn-n) : HAモニタから出力されるメッセージID
(KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID
(nn...nn) : ユーザコマンドに渡されるパラメタ
(nnnnn) : 発行されるJP1のイベントID (000nnnnn)

注※1 HA モニタが、サーバの停止コマンドを実行します。

注※2 一つのサーバで複数の Oracle リスナーを制御する場合、この部分の処理の流れを、制御する Oracle リスナーの数だけ繰り返します。

注※3 HA モニタは、サーバの停止コマンドの完了を待ちます。

注※4 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

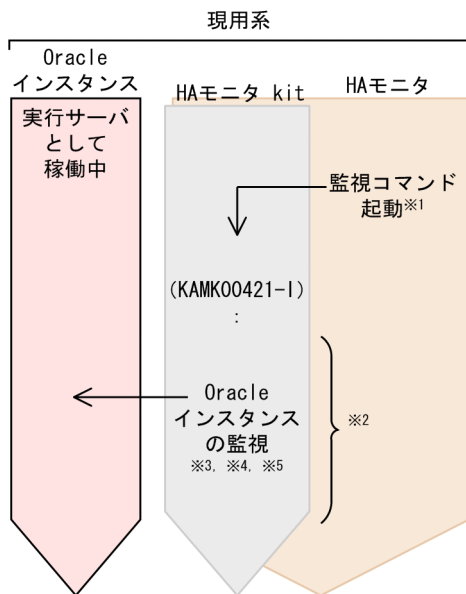
2.2.4 サーバの監視処理の流れ

サーバ（Oracle インスタンスまたは Oracle リスナー）の監視時に、HA モニタおよび HA モニタ kit がする処理の流れについて説明します。なお、この項で示す図は、一つのサーバに Oracle インスタンスまたは Oracle リスナーだけに対応させた場合の例です。

(1) Oracle インスタンスの監視処理の流れ

Oracle インスタンスの監視時に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-12 Oracle インスタンスの監視処理の流れ



(凡例) (KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID

注※1 HA モニタが、サーバの監視コマンドを実行します。

サーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合
実行サーバの起動完了時に、サーバの監視コマンドを起動します。

サーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合
実行サーバの起動完了後、サーバ対応の環境設定の ptrlcmd_ex_inter オペランドに指定した監視間隔
ごとに、サーバの監視コマンドを起動します。

注※2 次の監視間隔で、この部分の処理を繰り返します。

サーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合
サーバの監視コマンドの PATROL_INTERVAL 環境変数に指定した監視間隔

サーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合
サーバ対応の環境設定の ptrlcmd_ex_inter オペランドに指定した監視間隔

注※3 HA モニタ kit は、SQL による問い合わせ応答（データベースへのアクセス、および I/O 確認）を行います。応答が一定時間ない場合は、スローダウンと見なします。なお、スローダウンと見なすまでの時間は、Oracle インスタンス制御用定義ファイルの PATROL に指定します。

注※4 HA モニタ kit は、Oracle インスタンスの次のプロセスが存在するかどうかを確認します。

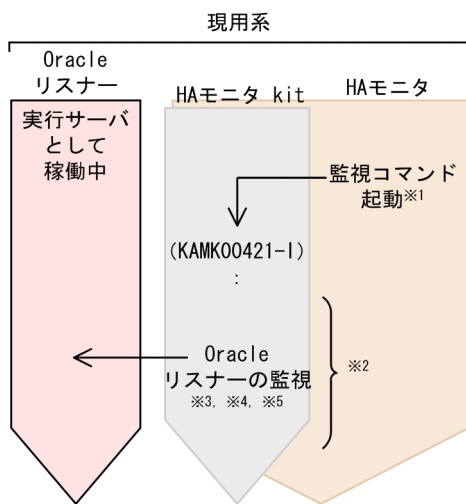
CKPT, DBW0, LGWR, PMON, SMON

注※5 一つのサーバで複数の Oracle インスタンスを制御する場合、この部分の処理の流れを、制御する Oracle インスタンスの数だけ繰り返します。

(2) Oracle リスナーの監視処理の流れ

Oracle リスナーの監視時に HA モニタおよび HA モニタ kit がする処理の流れを次の図に示します。

図 2-13 Oracle リスナーの監視処理の流れ



(凡例) (KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID

注※1 HA モニタが、サーバの監視コマンドを実行します。

注※2 サーバの監視コマンドの PATROL_INTERVAL 環境変数に指定した監視間隔で、この部分の処理を繰り返します。

注※3 HA モニタ kit は、リスナー制御ユーティリティのコマンドを実行し、問い合わせ応答を行います。応答が一定時間ない場合は、スローダウンと見なします。なお、スローダウンと見なすまでの時間は、Oracle リスナー制御用定義ファイルの PATROL に指定します。

注※4 HA モニタ kit は、Oracle リスナーの TNSLSNR プロセスが存在するかどうかを確認します。

注※5 一つのサーバで複数の Oracle リスナーを制御する場合、この部分の処理の流れを、制御する Oracle リスナーの数だけ繰り返します。

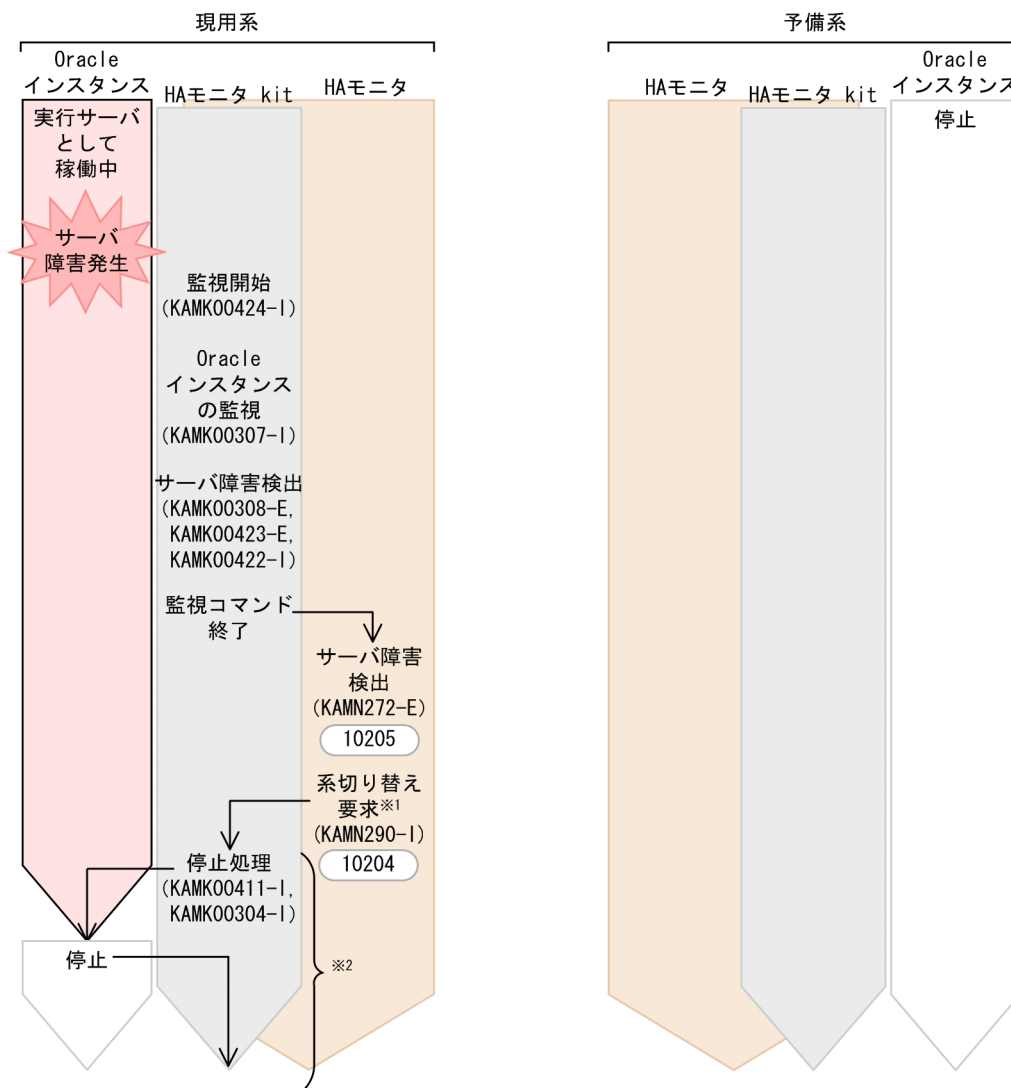
2.2.5 サーバ障害時の系切り替え処理の流れ

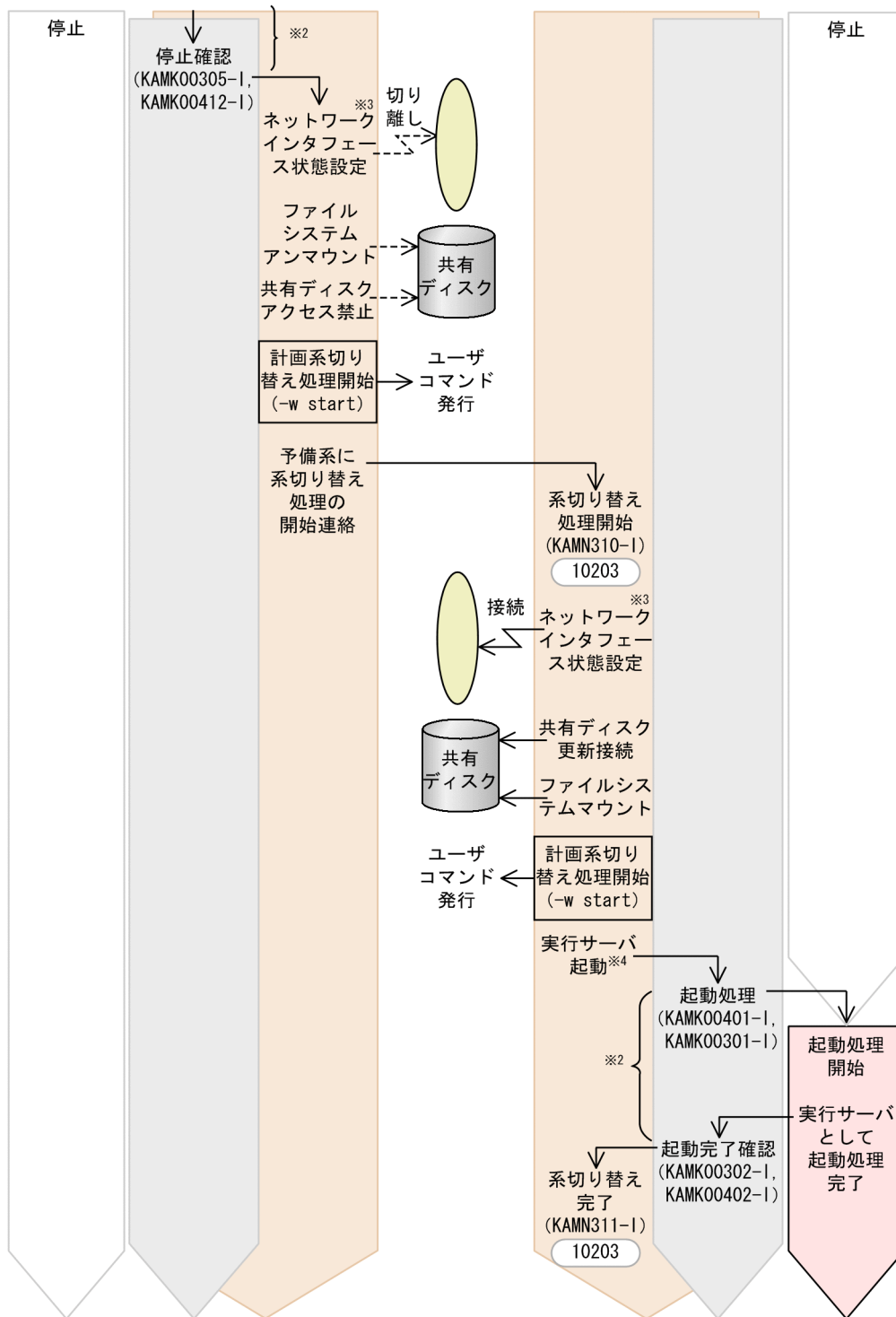
サーバ（Oracle インスタンスまたは Oracle リスナー）の障害が発生した場合に、HA モニタおよび HA モニタ kit がする処理の流れについて説明します。なお、この項で示す図は、一つのサーバに Oracle インスタンスまたは Oracle リスナーだけを対応させた場合の例です。

(1) Oracle インスタンス障害時の系切り替え処理の流れ

Oracle インスタンス障害時に HA モニタおよび HA モニタ kit がする系切り替え処理の流れを次の図に示します。

図 2-14 Oracle インスタンス障害時の系切り替え処理の流れ





- (凡例) (KAMNnnn-n) : HAモニタから出力されるメッセージID
 (KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID
 (nn...nn) : ユーザコマンドに渡されるパラメタ
 nnnnn : 発行されるJP1のイベントID (000nnnnn)

注※1 HA モニタが、サーバの停止コマンドを実行します。

注※2 一つのサーバで複数の Oracle インスタンスを制御する場合、この部分の処理の流れを、制御する Oracle インスタンスの数だけ繰り返します。

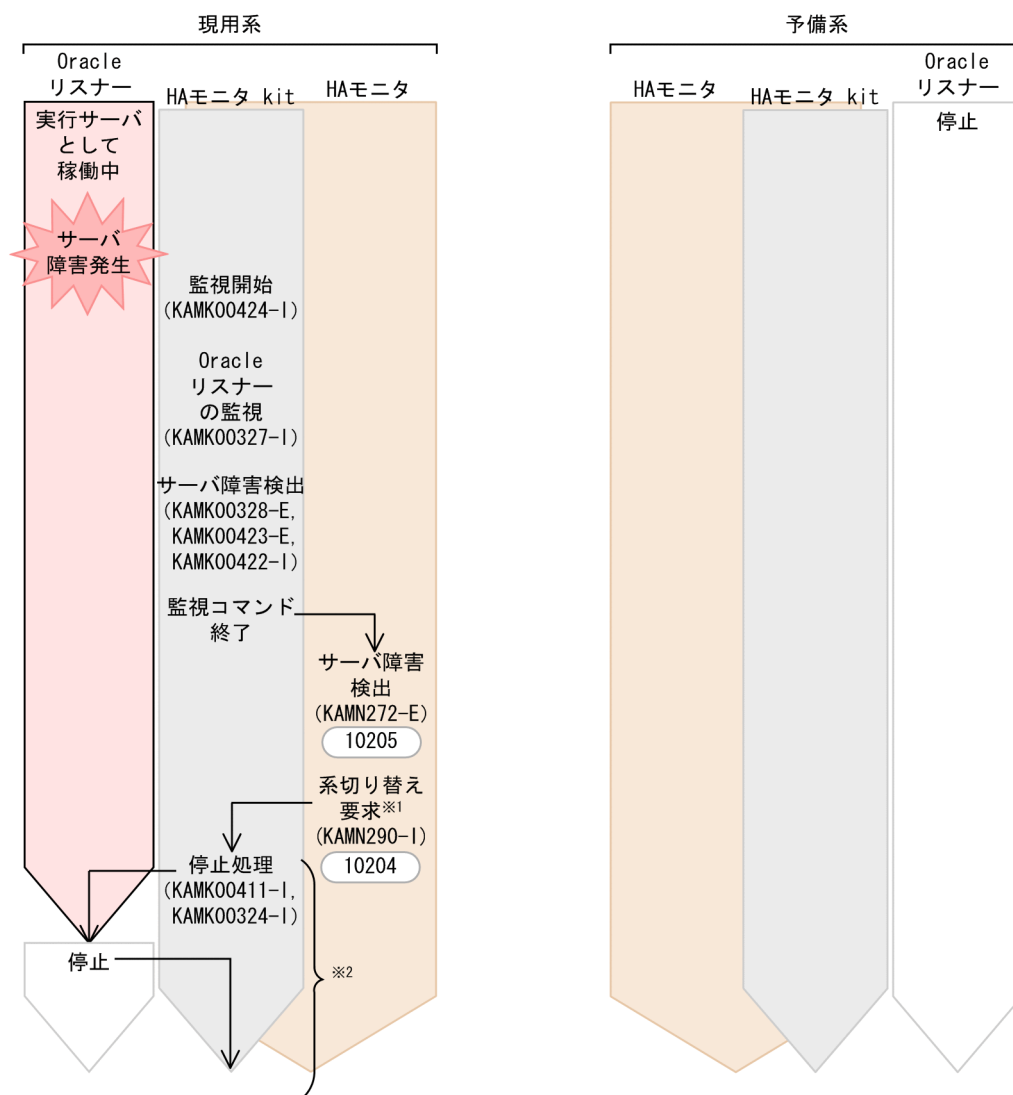
注※3 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

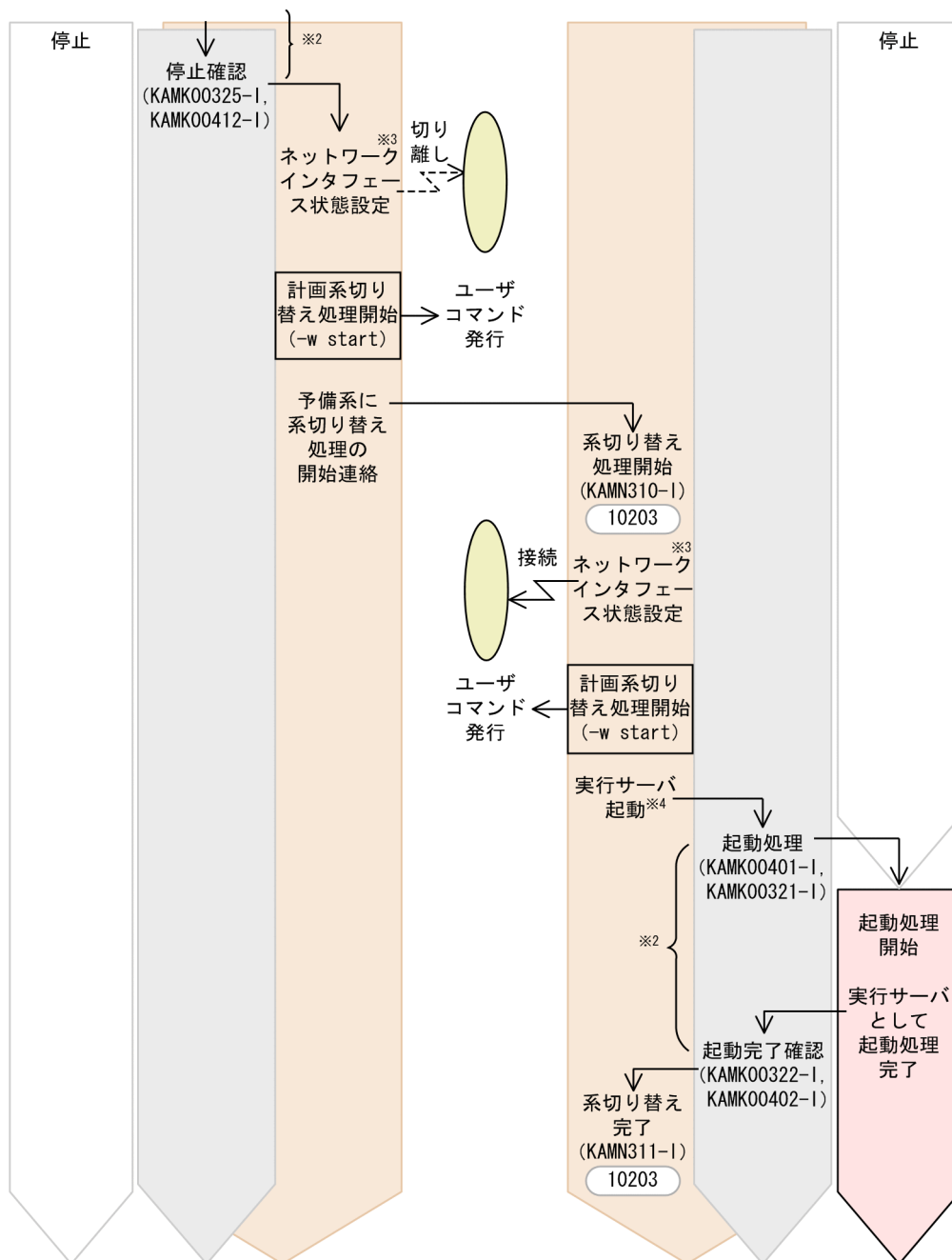
注※4 HA モニタが、サーバの起動コマンドを実行します。

(2) Oracle リスナー障害時の系切り替え処理の流れ

Oracle リスナー障害時に HA モニタおよび HA モニタ kit がする系切り替え処理の流れを次の図に示します。

図 2-15 Oracle リスナー障害時の系切り替え処理の流れ





- (凡例) (KAMNnnn-n) : HAモニタから出力されるメッセージID
 (KAMKnnnnn-n) : HAモニタ kitから出力されるメッセージID
 (nn...nn) : ユーザコマンドに渡されるパラメタ
 nnnnn : 発行されるJP1のイベントID (000nnnnn)

注※1 HA モニタが、サーバの停止コマンドを実行します。

注※2 一つのサーバで複数の Oracle リスナーを制御する場合、この部分の処理の流れを、制御する Oracle リスナーの数だけ繰り返します。

注※3 サーバのリソースに、クライアント接続用の IP アドレスを含める場合に実施します。サーバに必要なリソースについては、「3.1.2 リソース構成の検討」および「3.2.6 Oracle が使用する共有リソースの設定」を参照してください。

注※4 HA モニタが、サーバの起動コマンドを実行します。

3

システムの設計と構築

この章では、システムの設計と構築、および環境設定例について説明します。

システムの設計では、HA モニタ kit を業務システムに導入するに当たり、検討することについて説明します。

システムの構築では、HA モニタ kit を組み込んだシステムの構築手順、HA モニタ kit のディレクトリ構成、および構築に必要な設定項目の対応と設定の概要について説明します。

3.1 システムの設計

HA モニタ kit を使用した系切り替え構成を構築する前に、HA モニタのシステムの設計が必要です。HA モニタ kit を使用しない HA モニタのシステムと共通の部分については、マニュアル「HA モニタ Linux(R) 編」の「HA モニタの導入とシステムの設計」の説明を参照してください。

ここでは、Oracle をモニタモードのサーバとして管理するために必要な検討事項について説明します。

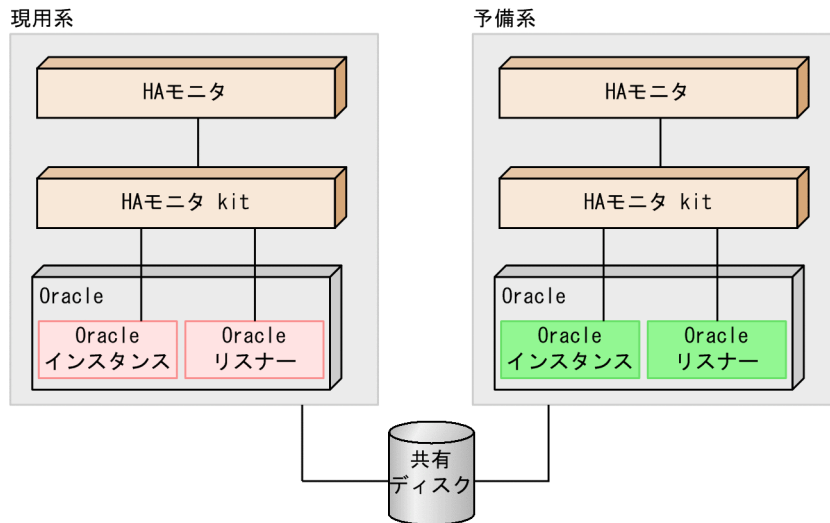
3.1.1 Oracle の構成と系切り替え単位の検討

HA モニタは、サーバ単位で系切り替えをします。HA モニタ kit の制御対象となる Oracle インスタンスおよび Oracle リスナーを、系切り替えの単位であるサーバとどのように対応させるかを検討する必要があります。検討時は、次のことを考慮してください。

- サーバは、運用および系切り替えをする単位で一つずつ設定してください。
複数の Oracle インスタンスおよび Oracle リスナーをまとめて運用・系切り替えしたい場合は、一つのサーバにまとめることもできます。
- 複数の Oracle インスタンスおよび Oracle リスナーを、まとめて運用・系切り替えしたい場合で、かつ次に示す場合は、複数のサーバを設定し、それらのサーバをグループ化してください。
 - 起動・停止などの運用を個別に実施したい（一括で起動・停止しない）場合
運用する単位でサーバを分け、グループ化します。
 - Oracle ユーザを分けたい場合
同じ Oracle ユーザごとにサーバを分け、グループ化します。
 - 監視対象となる Oracle インスタンスおよび Oracle リスナーそれぞれで、監視間隔に異なる値を設定したい場合
監視間隔が異なるごとにサーバを分け、グループ化します。
 - Oracle インスタンスや Oracle リスナーごとに障害検知時の動作を分けるなど、HA モニタのサーバ対応の環境設定を分けたい場合
サーバ対応の環境設定単位でサーバを分け、グループ化します。

次に、代表的な系切り替え構成の例を示します。

図 3-1 代表的な系切り替え構成の例



この例の現用系の場合の、Oracle インスタンス・Oracle リスナーとサーバの対応の例を、以降に示します。

図 3-2 一つのサーバに複数の Oracle インスタンス・Oracle リスナーを対応させる例

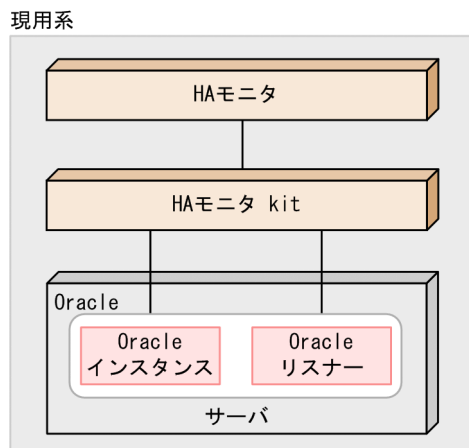
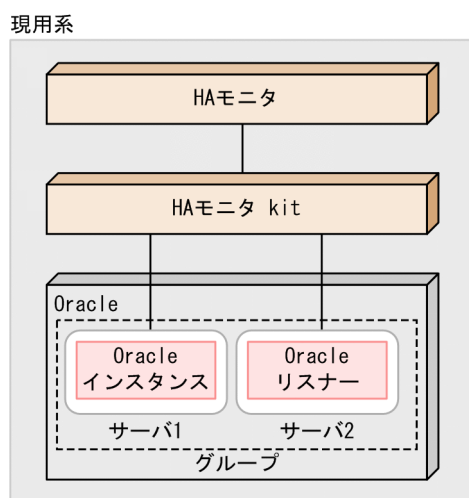


図 3-3 Oracle インスタンス・Oracle リスナーごとにサーバを対応させ、かつグループ化する例



3.1.2 リソース構成の検討

ここでは、リソースに関する検討事項について説明します。

(1) リソースとサーバとの対応

Oracle の系切り替え構成で、HA モニタが切り替える必要があるリソースと、サーバとの対応について次に示します。

- 共有ディスク
共有ディスクは、共有ディスクを使用する Oracle インスタンスおよび Oracle リスナーに対応する、サーバに定義します。
- クライアント接続用の IP アドレス (エイリアス IP アドレス)
クライアントがデータベースに接続するためのエイリアス IP アドレスは、Oracle リスナーに対応するサーバの、LAN の状態設定ファイルに定義します。Oracle リスナーを使用しない場合は、クライアントが接続する Oracle インスタンスに対応するサーバの、LAN の状態設定ファイルに定義します。

HA モニタで制御できるリソースについては、マニュアル「HA モニタ Linux(R)編」の「リソース構成の検討」の説明を参照してください。

(2) リソースサーバの検討

複数のサーバで一つのリソースを共有する構成では、リソースサーバの使用を検討します。

リソースサーバについては、マニュアル「HA モニタ Linux(R)編」の「リソースサーバを使用する場合の構成」の説明を参照してください。

3.2 システムの構築

ここでは、システムの構築について説明します。システムの構築は、スーパーユーザの権限で行ってください。系を追加する場合の手順など、HA モニタ kit を使用しない HA モニタのシステムと共通の部分については、マニュアル「HA モニタ Linux(R)編」の「システムの構築」の説明を参照してください。

なお、ここでは、システム構築の前提となるハードウェアはすでに構成されているものとして説明します。

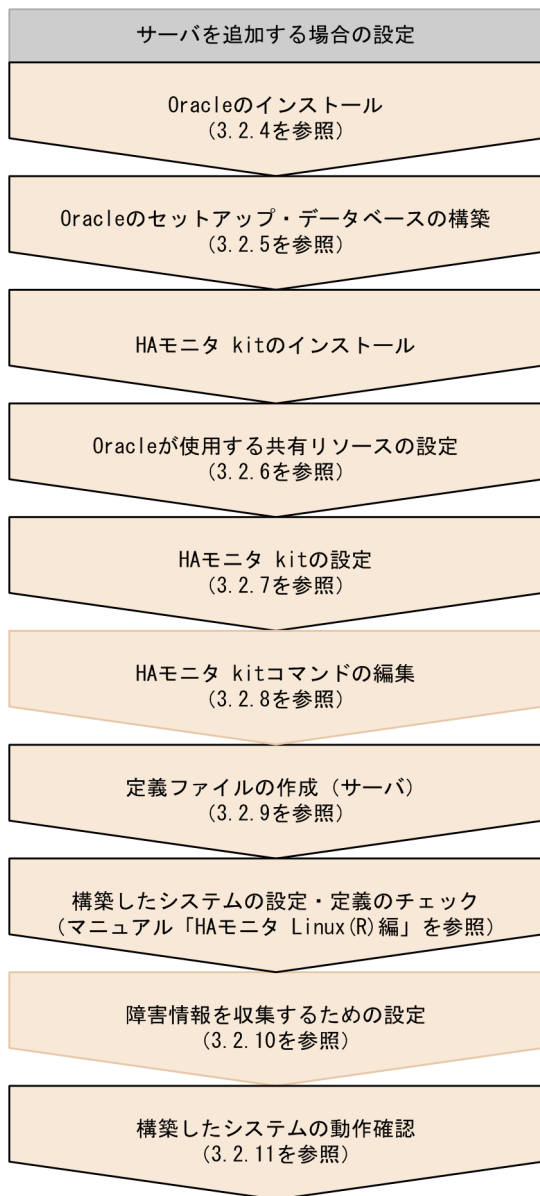
3.2.1 構築の流れ

ここでは、HA モニタ kit を使用して、Oracle を HA モニタのサーバとして設定する場合の構築の流れについて説明します。マニュアル「HA モニタ Linux(R)編」の「構築の流れ」の説明中の、「サーバを追加する場合の設定」を、ここに記載されている構築の流れに置き換えて読んでください。



ここで説明する作業は、マニュアル「HA モニタ Linux(R)編」の「構築の流れ」の説明の、「系を追加する場合」の設定が完了したあとに、システム管理者がスーパーユーザの権限で実施します。

システムの構築の流れを、次の図に示します。

図 3-4 システム構築の流れ



(凡例)

 : 必須  : 任意

システム構築の目的と、システム構築の流れで必要な手順との対応について説明します。

- 初めて HA モニタと HA モニタ kit を導入してシステムを構築する場合
このマニュアルとマニュアル「HA モニタ Linux(R)編」を併読し、図中に示すすべての項目を確認・設定します。
- 系を追加する場合
構築済みの系切り替え構成に系を追加する場合、マニュアル「HA モニタ Linux(R)編」の「構築の流れ」の説明を参照してください。HA モニタは、各系に一つ必要です。
- サーバを追加する場合

構築済みの系切り替え構成にサーバを追加する場合、この図中に示す「サーバを追加する場合の設定」をします。

構築済みのシステムで系やサーバを追加する手順については、マニュアル「HA モニタ Linux(R)編」の「系を追加する」または「サーバを追加する」の説明を参照してください。

また、各手順の注意事項について説明します。

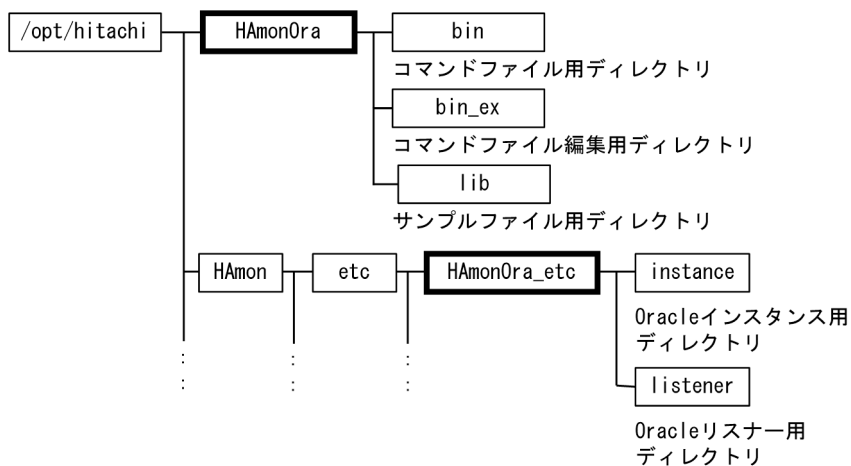
- 「Oracle のインストール」は、HA モニタが制御する Oracle がインストールされていない場合にだけ実施します。
- 「Oracle のセットアップ・データベースの構築」は、HA モニタが制御する Oracle インスタンスおよび Oracle リスナーの数だけ実施します。
- 「HA モニタ kit のインストール」は、初めて HA モニタ kit を導入する場合に、系切り替え構成内の各系で実施します。

3.2.2 ディレクトリ構成

HA モニタ kit をインストールすると、/opt/hitachi の下に HA モニタ kit のディレクトリ HAmonOra が作成されます。また、HA モニタをインストールした際に作成される/opt/hitachi/HAmom/etc の下に、HA モニタ kit のディレクトリ HAmonOra_etc が作成されます。

HA モニタ kit をインストールしたあとのディレクトリ構成を、次の図に示します。

図 3-5 ディレクトリ構成



注意事項

- コマンドファイル用ディレクトリ (/opt/hitachi/HAmomOra/bin) にあるファイルは編集しないでください。運用上、編集する必要がある場合は、該当するファイルをコマンドファイル編集用ディレクトリ (/opt/hitachi/HAmomOra/bin_ex) にコピーしてから、コマンドファイ

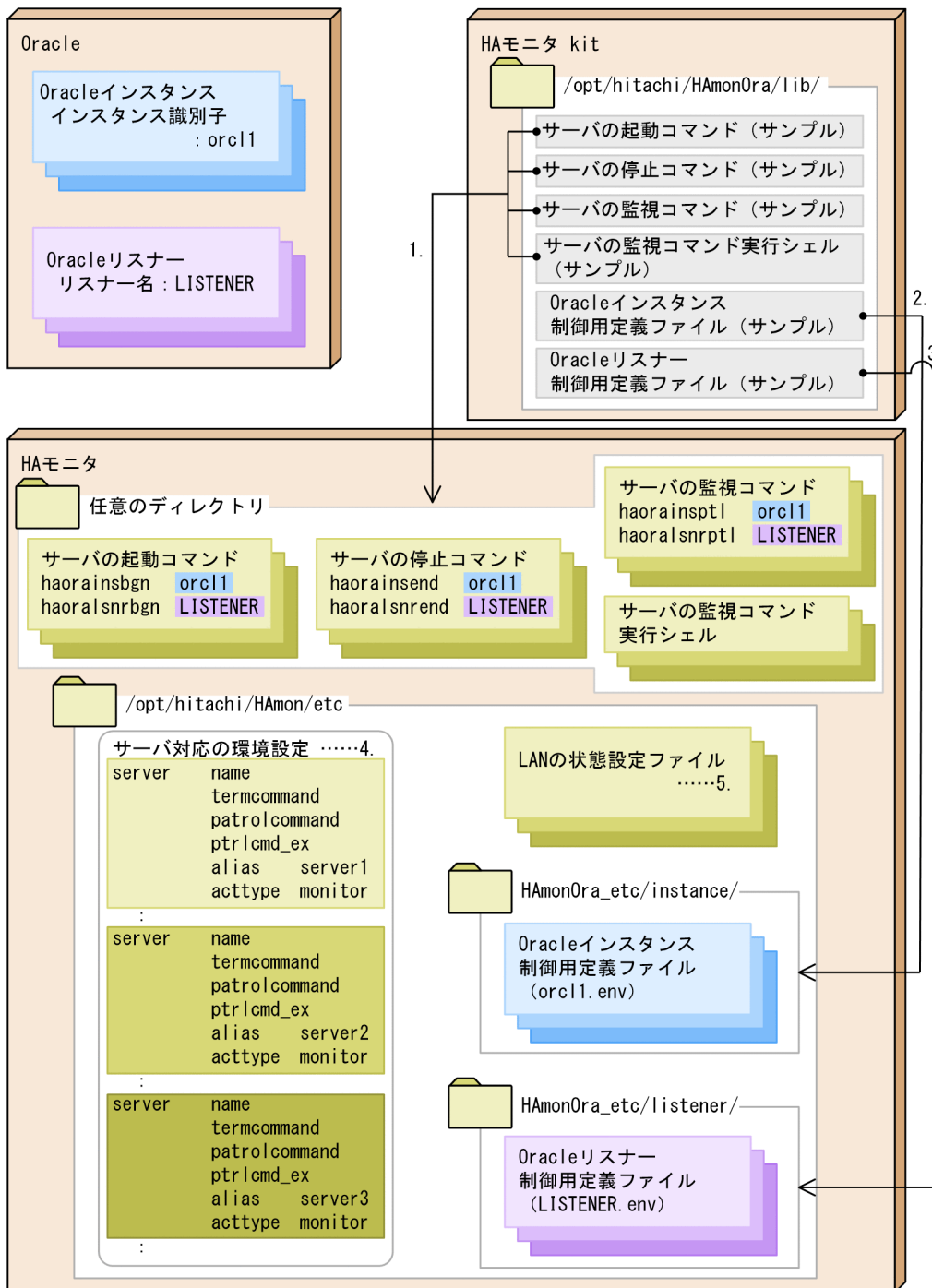
ル編集用ディレクトリのファイルを編集してください。詳細は、「[3.2.8 HA モニタ kit コマンドの編集](#)」を参照してください。


- 上記の図に示すディレクトリ、およびインストール時にそれらのディレクトリ上に作成されるファイルのアクセス権を変更しないでください。

3.2.3 構築に必要な設定項目の対応と設定の概要

ここでは、HA モニタ kit を使用したシステムを設定するに当たり、理解しておきたい設定項目の対応と設定の概要について説明します。システム内の設定の関連を、次の図に示します。

図 3-6 構築に必要な設定項目の対応と設定の概要



(凡例)
 → : ファイルをコピーして設定する。  : フォルダ

次に、図に示した設定の対応について説明します。番号は、図中の番号と対応しています。

1. サーバの起動・停止・監視コマンド、およびサーバの監視コマンド実行シェルのサンプルを、サーバの数だけコピーして設定します。
 詳細については、「[3.2.7\(1\) サーバの起動・停止・監視コマンドの設定](#)」を参照してください。
2. Oracle インスタンス制御用定義ファイルのサンプルを、Oracle インスタンスの数だけコピーして設定します。

詳細については、「[3.2.7\(2\) 制御用定義ファイルの設定](#)」を参照してください。

- Oracle リスナー制御用定義ファイルのサンプルを、Oracle リスナーの数だけコピーして設定します。詳細については、「[3.2.7\(2\) 制御用定義ファイルの設定](#)」を参照してください。
- サーバ対応の環境設定 (servers) に、「[3.1.1 Oracle の構成と系切り替え単位の検討](#)」で検討したサーバの数だけ、サーバの定義をします。定義ファイルの作成については、「[3.2.9 定義ファイルの作成 \(サーバ\)](#)」を参照してください。
- LAN の状態設定ファイルは、クライアント接続用の IP アドレスを使用するサーバの分だけ作成します。クライアント接続用の IP アドレスを使用するサーバは、「[3.1.2 リソース構成の検討](#)」を参照して決定してください。また、設定方法については、「[3.2.6 Oracle が使用する共有リソースの設定](#)」を参照してください。

3.2.4 Oracle のインストール

Oracle のインストール手順については、Oracle のマニュアルを参照してください。ここでは、Oracle のインストール時の注意事項について説明します。

- Oracle をローカルディスクにインストールする場合は、現用系と予備系で、次の設定値を同一にしてください。
 - Oracle ユーザのユーザ名 (通常は oracle) とユーザ ID, グループ名とグループ ID
 - Oracle ホームディレクトリ
- Oracle を共有ディスクにインストールする場合は、インストールに使用しなかった系で、インストールに使用した系と次の設定値を同一にしてください。
 - Oracle ユーザのユーザ名 (通常は oracle) とユーザ ID, グループ名とグループ ID
 - カーネルパラメタなどの OS の設定
- インストールの実行中にデータベースを作成する場合は、事前に「[3.2.5\(1\) データベースの構築](#)」を参照してください。
- Oracle Enterprise Manager Database Control を系切り替え環境 (クラスタ環境) で使用する場合は、注意事項があります。詳細については、Oracle のマニュアルを参照してください。
- Oracle ユーザのデータベースへの接続の認証方式は、OS 認証である必要があります。OS 認証の詳細については、Oracle のマニュアルを参照してください。
- Oracle ユーザには、sysdba 権限が必要です。Oracle ユーザに、sysdba 権限を付与してください。

3.2.5 Oracle のセットアップ・データベースの構築

Oracle のインストールが完了したら、Oracle のセットアップ・データベースの構築を実施します。Oracle のセットアップ・データベースの構築手順については、Oracle のマニュアルを参照してください。

ここでは、データベースの構築時、および Oracle リスナーの構築時の注意事項について説明します。

(1) データベースの構築

ここでは、系切り替え構成に対応したデータベースを構築するために必要な設定について説明します。データベースの構築時には、次の項目を参照して設定してください。

- データベースの構築は、Oracle インスタンスを稼働させる系のどれか一つから実施してください。
- データベースは、HA モニタが制御できるファイルシステムに作成してください。
- データベースを構成する次のファイルは、共有ディスク上に格納してください。
 - サーバー・パラメータファイル
 - 制御ファイル
 - データファイル
 - REDO ログファイル
 - アーカイブログファイル

サーバー・パラメータファイル (SPFILE) がローカルディスクに作成されている場合は、ファイルを共有ディスクに移動してから、次のどちらかを実施してください。

- 初期化パラメータファイル (PFILE) に移動先のサーバー・パラメータファイルのパスを定義する。
- 移動元のディレクトリに、移動先のサーバー・パラメータファイルのシンボリックリンクを作成し、移動前のパスから参照できるようにする。
- フラッシュリカバリ領域には、共有ディスク上のパスを指定してください。

(2) Oracle リスナーの構築

Oracle リスナーの構築は、現用系・予備系の全系で実施します。

クライアント接続用の IP アドレス (エイリアス IP アドレス) を使用するため、Oracle リスナー構築後に次の設定をしてください。

1. /etc/hosts ファイルに使用する、エイリアス IP アドレスとホスト名を登録します。全系で同じ名称にしてください。
2. Oracle のローカル・ネーミング・パラメータのファイル (tnsnames.ora) のネット・サービス名のホストを、手順 1. で登録したエイリアス IP アドレスのホスト名に変更します。

設定例を次に示します。

/etc/hosts ファイルの例

```
192.168.0.11 server1 # alias IP address
```

```
ORCL1 =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = server1)(PORT = 1521))  
    (CONNECT_DATA =  
      (SERVER = SHARED)  
      (SERVICE_NAME = orcl1)  
    )  
  )  
)
```

参考

クライアントでは、手順 2. で修正した tnsnames.ora を使用して、登録したホスト名で接続してください。

```
# sqlplus user/passwd@server1/ORCL1
```

3.2.6 Oracle が使用する共有リソースの設定

Oracle のセットアップ・データベースの構築が完了したら、Oracle が使用する共有リソースを設定します。次の項目を設定してください。

共有ディスクの設定

共有ディスクを HA モニタが制御するための設定をします。詳細については、マニュアル「HA モニタ Linux(R)編」の「共有ディスクの設定」の説明を参照してください。

LAN の状態設定ファイルの設定

クライアント接続用の IP アドレスを HA モニタで切り替えができるように、IP アドレスを使用するサーバごとに LAN の状態設定ファイルを作成します。詳細については、マニュアル「HA モニタ Linux(R)編」の「LAN の状態設定ファイルの設定」を参照してください。

3.2.7 HA モニタ kit の設定

Oracle が使用する共有リソースの設定が完了したら、HA モニタ kit の設定をします。HA モニタ kit の設定には、サーバの起動・停止・監視コマンドの設定、および制御用定義ファイルの設定があります。

(1) サーバの起動・停止・監視コマンドの設定

HA モニタ kit では、サーバの起動・停止・監視コマンドのサンプルファイルを提供しています。サーバの数だけ、これらのファイルを任意のディレクトリにコピーし、環境に合わせて設定します。HA モニタの環境設定用ディレクトリ (/opt/hitachi/HAMon/etc) 下にコピーすることを推奨します。

HA モニタ kit が提供するサーバの起動・停止コマンドのサンプルファイルの格納ディレクトリを、次の表に示します。

表 3-1 サーバの起動・停止コマンドのサンプルファイルの格納ディレクトリ

コマンド	格納ディレクトリ
サーバの起動コマンド	/opt/hitachi/HAMonOra/lib/actcommand
サーバの停止コマンド	/opt/hitachi/HAMonOra/lib/termcommand

HA モニタ kit が提供するサーバの監視コマンドのサンプルファイルの格納ディレクトリを、次の表に示します。格納ディレクトリは Oracle のデータベース構成によって異なります。

表 3-2 サーバの監視コマンドのサンプルファイルの格納ディレクトリ

Oracle のデータベース構成	コマンド		格納ディレクトリ
CDB (シングルテナント) 構成	HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合	サーバの監視コマンド	/opt/hitachi/HAMonOra/lib/ptrlcmd_ex
シングル構成	HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合 (推奨)	サーバの監視コマンド	/opt/hitachi/HAMonOra/lib/ptrlcmd_ex
		サーバの監視コマンド	/opt/hitachi/HAMonOra/lib/patrolcommand
	HA モニタのサーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合	サーバの監視コマンド実行シェル	/opt/hitachi/HAMonOra/lib/patrol.sh

コピーしたファイルの名称は、コピー元のファイルと同一にする必要はありません。構築するサーバが複数存在し、単純にコピーするとファイル名が重複する場合は、コピー先のディレクトリを変更するか、重複しないようにファイル名を変更してください。

サーバの起動・停止・監視コマンド、およびサーバの監視コマンド実行シェルは、次の表に示すユーザで読み込みと実行を行います。アクセス権をサンプルファイルから変更する場合は、読み込みと実行を行うユーザにアクセス権を付与してください。アクセス権がない場合、各コマンドの実行に失敗します。

表 3-3 サーバの起動・停止・監視コマンド，およびサーバの監視コマンド実行シェルの読み込みと実行を行うユーザ

ファイル名		読み込みと実行を行うユーザ
サーバの起動コマンド		スーパーユーザ
サーバの停止コマンド		スーパーユーザ
HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合（推奨）	サーバの監視コマンド	スーパーユーザ
HA モニタのサーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合	サーバの監視コマンド	サーバの監視コマンド実行シェルの ORACLE_USER 環境変数に指定する Oracle ユーザ
	サーバの監視コマンド実行シェル	スーパーユーザ

それぞれのコマンドの詳細な設定方法については、「4. サーバの起動・停止・監視コマンドの設定」を参照してください。

(2) 制御用定義ファイルの設定

HA モニタ kit が制御する Oracle インスタンスまたは Oracle リスナーごとに，制御用定義ファイルを設定する必要があります。HA モニタ kit が提供する制御用定義ファイルのサンプルファイルを，Oracle のデータベース構成に合わせてコピーしてリネームし，必要に応じて，ファイル内のパラメタの値の変更や，任意の環境変数の設定などを行ってください。また，監視する Oracle インスタンスまたは Oracle リスナーが複数存在する場合は，Oracle インスタンスまたは Oracle リスナーの数だけ制御用定義ファイルを設定してください。

制御用定義ファイルのコピー元ファイルとコピー先ディレクトリを，次の表に示します。

表 3-4 制御用定義ファイルのコピー元ファイルとコピー先ディレクトリ

制御用定義ファイル	コピー元ファイル	コピー先ディレクトリ
Oracle インスタンス制御用定義ファイル	<ul style="list-style-type: none"> シングル構成の場合 /opt/hitachi/HAmonOra/lib/orcl1.env CDB（シングルテナント）構成の場合 /opt/hitachi/HAmonOra/lib/orcl1_cdb.env 	/opt/hitachi/HAmon/etc/HAmonOra_etc/instance
Oracle リスナー制御用定義ファイル	/opt/hitachi/HAmonOra/lib/LISTENER.env	/opt/hitachi/HAmon/etc/HAmonOra_etc/listener

それぞれのファイルは，次のようにリネームしてください。ファイル名の太字の部分は，それぞれ該当する値に置き換えてください。

- Oracle インスタンス制御用定義ファイル：**インスタンス識別子**.env

- Oracle リスナー制御用定義ファイル：リスナー名.env

コピーする際は、コピー先の設定済みのファイルを誤って上書きして消さないように注意してください。また、コピーしたファイルのアクセス権は、コピー元ファイルのアクセス権と同じにして、そのあとはアクセス権を変更しないでください。

以降、ファイルの詳細について説明します。

記述形式

パラメタを「パラメタ名=値」の形式で1行に一つずつ記述します。パラメタ、イコール (=)、および値の前後に空白を記述しないでください。なお、パラメタの記述は順不同です。

任意の環境変数を設定したい場合は、「環境変数名=値」の形式で1行に一つずつ記述します。

パラメタの詳細

指定できるパラメタの詳細について、次に示します。

表 3-5 制御用定義ファイルに指定できるパラメタの詳細

パラメタ名	説明
ORACLE_SID	Oracle インスタンス制御用定義ファイルに指定します。制御対象の Oracle インスタンスの、ORACLE_SID 環境変数の値（インスタンス識別子）を指定します。 ORACLE_SID 環境変数の詳細については、Oracle のマニュアルを参照してください。
LISTENER_NAME	Oracle リスナー制御用定義ファイルに指定します。制御対象の Oracle リスナーのリスナー名を指定します。リスナー名の詳細については、Oracle のマニュアルを参照してください。
ORACLE_BASE	制御対象の Oracle インスタンスまたは Oracle リスナーの ORACLE_BASE 環境変数の値（Oracle ベースディレクトリ）を指定します。 ORACLE_BASE 環境変数の詳細については、Oracle のマニュアルを参照してください。
ORACLE_HOME	制御対象の Oracle インスタンスまたは Oracle リスナーの ORACLE_HOME 環境変数の値（Oracle ホームディレクトリ）を指定します。 ORACLE_HOME 環境変数の詳細については、Oracle のマニュアルを参照してください。
ORACLE_USER	Oracle ユーザのアカウント名を指定します。Oracle ユーザの詳細については、Oracle のマニュアルを参照してください。
DB_NOACCESS	Oracle インスタンス制御用定義ファイルに指定します。SQL を使用して、Oracle インスタンスにアクセスするかどうかを指定します。指定できる値は次のとおりです。 <ul style="list-style-type: none"> • yes：SQL で Oracle インスタンスにアクセスしません。Oracle インスタンスの状態やスローダウンの監視はしないで、プロセスの生存監視だけをします。 • no：SQL で Oracle インスタンスにアクセスして、Oracle インスタンスの状態やスローダウンの監視、およびプロセスの生存監視をします。 パラメタを省略した場合は、no が仮定されます。次のような運用をする場合は、yes を指定してください。 <ul style="list-style-type: none"> • HA モニタ kit がデータベースにアクセスするのを禁止する運用にしたい場合

パラメタ名	説明
	<ul style="list-style-type: none"> SQL の ALTER SYSTEM SUSPEND 文で、データベースへのアクセスを禁止する運用にしたい場合※ Oracle インスタンスの状態を、"OPEN"以外の状態にする運用にしたい場合※ 注※ ただし、次の条件をすべて満たす場合は、パラメタに no (または省略) を指定できます。 <ul style="list-style-type: none"> HA モニタのバージョンが 01-69 以降 HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定している HA モニタ・サーバ稼働中の設定変更コマンド (monchange コマンド) で監視を一時停止したあとに、SQL の ALTER SYSTEM SUSPEND 文でデータベースへのアクセスを禁止する、または Oracle インスタンスの状態を"OPEN"以外の状態にするなどの運用である
LISTENER_NOACCESS	<p>Oracle リスナー制御用定義ファイルの場合に指定します。リスナー制御ユーティリティを使用して、Oracle リスナーにアクセスするかどうかを指定します。指定できる値は次のとおりです。</p> <ul style="list-style-type: none"> yes: リスナー制御ユーティリティを使用しません。Oracle リスナーのスローダウンの監視はしないで、プロセスの生存監視だけをします。 no: リスナー制御ユーティリティを使用して、Oracle リスナーのスローダウンの監視、およびプロセスの生存監視をします。 <p>パラメタを省略した場合は、no が仮定されます。</p>
PATROL	<p>Oracle インスタンスまたは Oracle リスナーにアクセスしてから、スローダウンと見なすまでの時間 (秒) を指定します。指定できる値は、60~3600 です。</p> <p>DB_NOACCESS または LISTENER_NOACCESS に yes を指定した場合は、このパラメタに指定した値は無視されます。</p>
ABORT_LIMIT	<p>通常停止ができない場合に、強制停止に移行するまでの待ち時間 (秒) を、10~3600 の範囲で指定します。</p>
PATROL_TABLE	<p>監視する表領域を指定します。通常は、指定する必要はありません。このパラメタを省略した場合は、dual が仮定されます。</p> <p>CDB に yes, または、DB_NOACCESS に yes を指定した場合、このパラメタに指定した値は無視されます。</p> <ul style="list-style-type: none"> dual: dual 表を監視します。 dba: HA モニタ kit の 01-02 以前のバージョンと同じ表領域 (ディクショナリ表 (dba_data_files)) を監視します。
CDB	<p>Oracle のデータベース構成が、CDB (シングルテナント) 構成かどうかを指定します。指定できる値は次のとおりです。</p> <ul style="list-style-type: none"> yse: CDB (シングルテナント) 構成の場合に指定します。 no: シングル構成の場合に指定します。 <p>パラメタを省略した場合は、no が仮定されます。</p>
PDB_NAME	<p>Oracle のデータベース構成が、CDB (シングルテナント) 構成の場合、このパラメタに起動・監視する PDB 名を指定します。CDB (シングルテナント) 構成の場合、必ず指定してください。指定できる PDB 名は一つです。</p> <p>CDB に no を指定した場合は、このパラメタは無視されます。</p>

記述例

制御用定義ファイルの記述例を示します。この記述例は、サンプルファイルの内容と同様です。各パラメータは、実際の環境に合わせて設定してください。

Oracle インスタンス制御用定義ファイル (/opt/hitachi/HAMon/etc/HAMonOra_etc/instance/orcl1.env) の場合

- シングル構成

```
ORACLE_SID=orcl1
ORACLE_BASE=/home/oracle
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1
ORACLE_USER=oracle
PATROL=120
ABORT_LIMIT=180
```

- CDB (シングルテナント) 構成

```
ORACLE_SID=orcl1
ORACLE_BASE=/home/oracle
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1
ORACLE_USER=oracle
PATROL=120
ABORT_LIMIT=180
CDB=yes
PDB_NAME=orcl_pdb
```

Oracle リスナー制御用定義ファイル (/opt/hitachi/HAMon/etc/HAMonOra_etc/listener/LISTENER.env) の場合

```
LISTENER_NAME=LISTENER
ORACLE_BASE=/home/oracle
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1
ORACLE_USER=oracle
PATROL=120
ABORT_LIMIT=180
```

注意事項

- Oracle インスタンスまたは Oracle リスナーの稼働中に、定義済みの制御用定義ファイルを変更しないでください。
- 定義の指定方法を誤ると、HA モニタ kit のコマンドが異常終了し、Oracle インスタンスまたは Oracle リスナーの制御に失敗します。この結果、サーバの起動や停止に失敗したり、Oracle インスタンスまたは Oracle リスナーの異常を HA モニタが検出したりします。

3.2.8 HA モニタ kit コマンドの編集

SQL*Plus で使用するコマンド、およびリスナー制御ユーティリティのコマンドなどを編集します。HA モニタ kit コマンドは、通常は編集する必要はありません。

参考

例えば、Oracle の設定で SYS の監査ログ取得が有効化されている場合、SYS ユーザで Oracle に接続すると、監査ログが監視ごとに出力され、大量のログファイルが生成されます。これを回避するために、Oracle インスタンスの監視時に接続する Oracle ユーザを変更したい場合、HA モニタ kit コマンドの編集を検討してください。詳細については、「(2)編集例」の「Oracle インスタンスの監視時に接続する Oracle ユーザを変更する場合」を参照してください。

HA モニタ kit コマンドを編集する場合は、次に示すコマンドの構造、および編集例を参考にして、編集手順に従って編集してください。

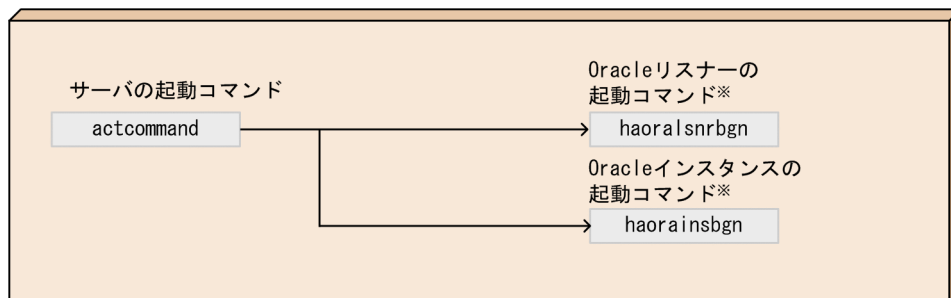
HA モニタ kit のコマンドの構造

HA モニタ kit のコマンドの構造を次に示します。

• サーバの起動コマンドの構造

サーバの起動コマンドと、Oracle インスタンスおよび Oracle リスナーの起動コマンドの関係を次の図に示します。

図 3-7 サーバの起動コマンドと、Oracle インスタンスおよび Oracle リスナーの起動コマンドの関係



(凡例)

—————> : コマンドの呼び出し
[] : コマンド

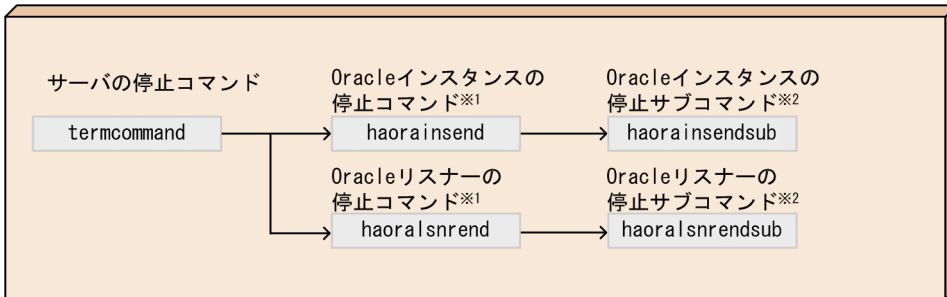
注※

制御用定義ファイルチェックコマンドhaoraenvcheckを呼び出して定義チェックを実行します。

• サーバの停止コマンドの構造

サーバの停止コマンドと、Oracle インスタンスおよび Oracle リスナーの停止コマンドの関係を次の図に示します。

図 3-8 サーバの停止コマンドと、Oracle インスタンスおよび Oracle リスナーの停止コマンドの関係



(凡例)

—————> : コマンドの呼び出し

□ : コマンド

注※1

制御用定義ファイルチェックコマンドhaoraenvcheckを呼び出して定義チェックを実行します。

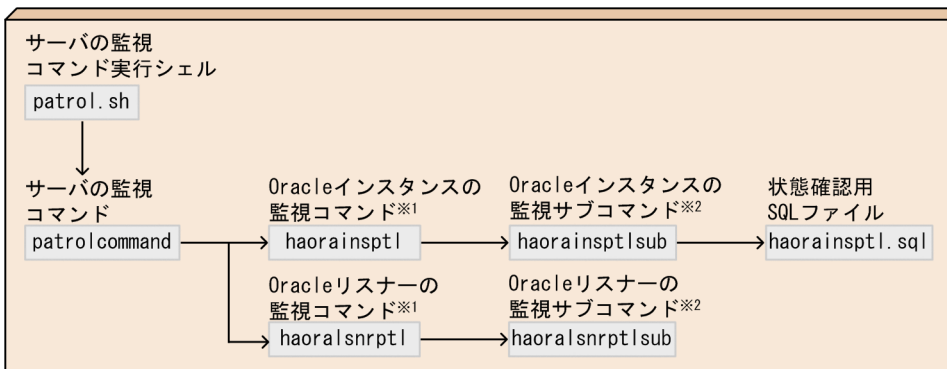
注※2

コマンドから指定した「SQL*Plusのコマンド」や「リスナー制御ユーティリティコマンド」を実行します。

• サーバの監視コマンドの構造

サーバの監視コマンドと、Oracle インスタンスおよび Oracle リスナーの監視コマンドの関係を次の図に示します。

図 3-9 サーバの監視コマンドと、Oracle インスタンスおよび Oracle リスナーの監視コマンドの関係 (サーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合)



(凡例)

—————> : コマンドの呼び出し

□ : コマンド

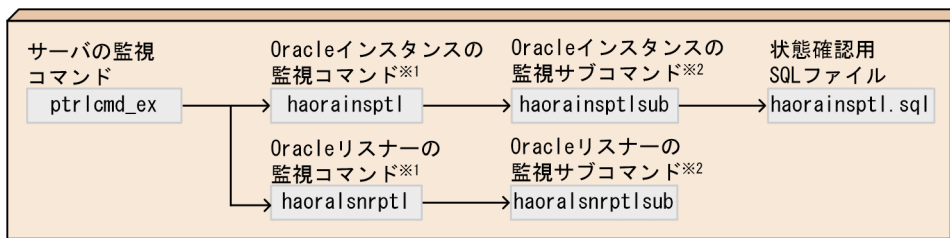
注※1

制御用定義ファイルチェックコマンドhaoraenvcheckを呼び出して定義チェックを実行します。

注※2

コマンドから指定した「SQL*Plusのコマンド」や「リスナー制御ユーティリティコマンド」を実行します。インスタンスの監視サブコマンドの場合、状態確認のためのSQL文を発行するために、状態確認用SQLファイルを実行します。

図 3-10 サーバの監視コマンドと、Oracle インスタンスおよび Oracle リスナーの監視コマンドの関係（サーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合）



(凡例) : コマンドの呼び出し
 : コマンド

注※1

制御用定義ファイルチェックコマンドhaoraenvcheckを呼び出して定義チェックを実行します。

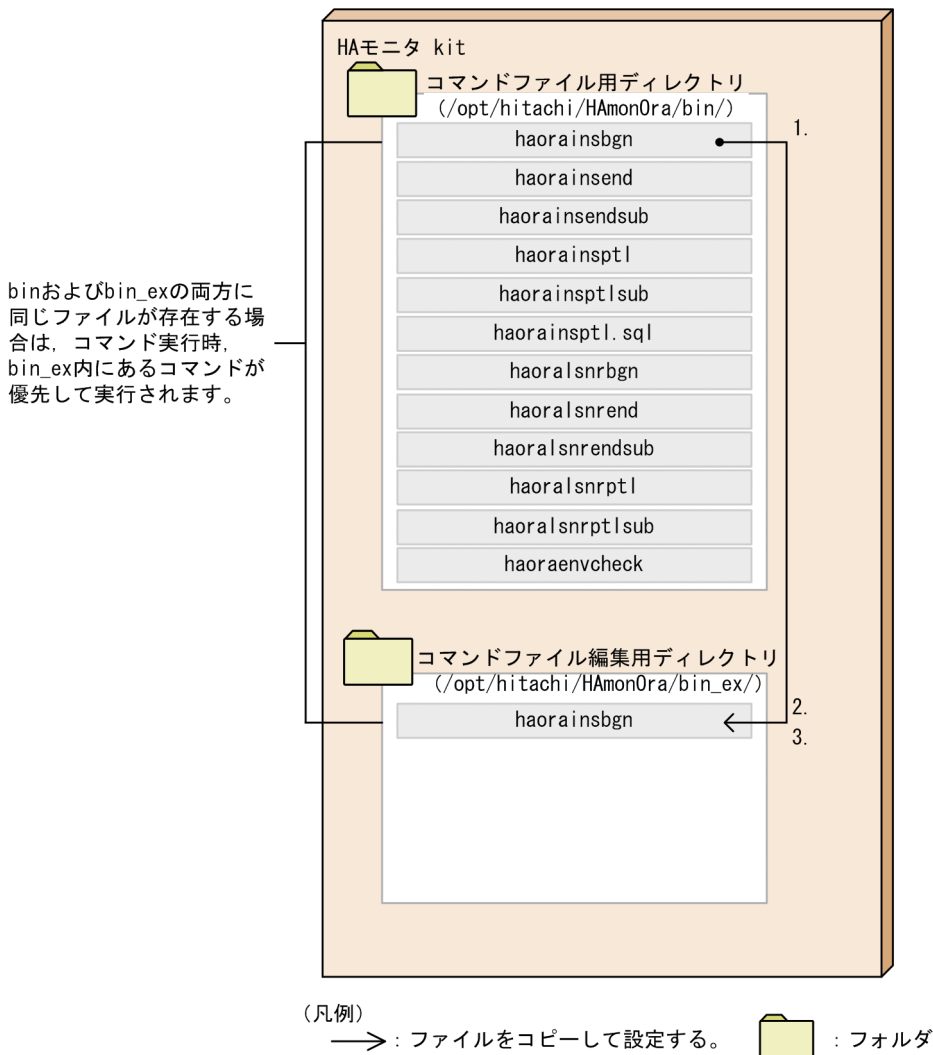
注※2

コマンドから指定した「SQL*Plusのコマンド」や「リスナー制御ユーティリティコマンド」を実行します。インスタンスの監視サブコマンドの場合、状態確認のためのSQL文を発行するために、状態確認用SQLファイルを実行します。

(1) 編集手順

HA モニタ kit コマンドを編集する手順を次に示します。手順 3 は、必要に応じて実施してください。

図 3-11 HA モニタ kit コマンドを編集する手順



1. 編集したいファイルを、cp -p コマンドでコマンドファイル用ディレクトリからコマンドファイル編集用ディレクトリにコピーします。

コマンドファイル用ディレクトリ

/opt/hitachi/HAMonOra/bin

コマンドファイル編集用ディレクトリ

/opt/hitachi/HAMonOra/bin_ex

2. コマンドファイル編集用ディレクトリのファイルを編集します。

ここでは、haorainsbgn ファイルを編集しています。

詳細については、「(2)編集例」を参照してください。

3. HA モニタ kit 01-00 からバージョンアップしてから編集する場合だけこの手順を実施します。バージョンアップ後の HA モニタ kit のサーバの起動・停止・監視コマンドのサンプルファイルを使用して、サーバの起動・停止・監視コマンドを再設定します。

詳細は、「3.2.7(1) サーバの起動・停止・監視コマンドの設定」を参照してください。

注意事項

編集後に HA モニタ kit 01-01 以降からバージョンアップした場合は、コマンドファイル編集用ディレクトリ (/opt/hitachi/HAMonOra/bin_ex) に新しいバージョンのファイルがインストールされません。そのため、コマンドファイル用ディレクトリ (/opt/hitachi/HAMonOra/bin) にある新しいバージョンのファイルを、編集手順に従って再度編集しなければならない場合があります。編集が必要なファイルについては、リリースノートの「使用上の注意事項」を参照してください。

(2) 編集例

任意の Oracle ユーザで Oracle に接続したい場合の、HA モニタ kit コマンドの編集例を次に示します。

• Oracle インスタンスの起動および停止時に接続する Oracle ユーザを変更する場合

編集内容に従って編集してください。次の編集内容は、パスワード認証した SYS ユーザで Oracle インスタンスを起動および停止する場合の例です。

対象ファイル

haorainsbgn, および haorainsendsub

編集内容

編集前

```
connect △/△as△sysdba
```

編集後

```
connect △sys/パスワード△as△sysdba
```

(凡例) △：半角スペース

ファイル中の該当箇所はすべて同様に編集してください。

注意事項

- Oracle ユーザには sysdba 権限が付与されている必要があります。
- パスワード認証とする場合、パスワード有効期限を無期限にしてください。

• Oracle インスタンスの監視時に接続する Oracle ユーザを変更する場合

編集内容に従って編集してください。ただし、DB_NOACCESS に yes を指定した場合、Oracle インスタンスの監視時に Oracle へ接続しないため、編集する必要はありません。

対象ファイル

haorainsptlsub, および haorainsptl.sql

編集内容 1 (パスワード認証した SYSTEM ユーザで Oracle インスタンスを監視する場合)

編集前

```
connect △/△as△sysdba
```


編集後

```
connect△system/パスワード
```

(凡例) △：半角スペース

ファイル中の該当箇所はすべて同様に編集してください。

注意事項

パスワード認証とする場合、パスワード有効期限を無期限にしてください。

編集内容 2 (OS 認証した一般 DB ユーザで Oracle インスタンスを監視する場合)

編集前

```
connect△/△as△sysdba
```

編集後

```
connect△/
```

(凡例) △：半角スペース

ファイル中の該当箇所はすべて同様に編集してください。

注意事項

- ・サーバの監視コマンド実行シェル「patrol.sh」の ORACLE_USER 環境変数に、新たに作成した OS ユーザのアカウント名を指定してください。
- ・OS 認証した一般 DB ユーザを作成してください。
- ・一般 DB ユーザに DBA ロールおよび CONNECT ロールを付与してください。

3.2.9 定義ファイルの作成 (サーバ)

HA モニタ kit の設定が完了したら、「3.1.1 Oracle の構成と系切り替え単位の検討」で決定したサーバごとに、HA モニタのサーバ対応の環境設定を作成します。作成方法については、マニュアル「HA モニタ Linux(R)編」の「サーバ対応の環境設定 (servers)」の説明を参照してください。

サーバ対応の環境設定のオペランドで注意が必要なものについてだけ、次の表に示します。

表 3-6 注意が必要なサーバ対応の環境設定のオペランド

オペランド名	注意事項
name	「4.2 サーバの起動コマンドの設定」で設定した、サーバの起動コマンドの絶対パスを指定します。
acttype	monitor を指定します。
termcommand	「4.3 サーバの停止コマンドの設定」で設定した、サーバの停止コマンドの絶対パスを指定します。

オペランド名	注意事項
patrolcommand	Oracle のデータベース構成が、シングル構成で、サーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合に指定します。 [4.5.2 サーバの監視コマンド実行シェル] で設定した、サーバの監視コマンド実行シェルの絶対パスを指定します。
ptrlcmd_ex	Oracle のデータベース構成が次の場合に指定します。 <ul style="list-style-type: none"> • CDB (シングルテナント) 構成の場合 • シングル構成で、サーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合 HA モニタの機能によって、サーバ稼働中に Oracle インスタンスおよび Oracle リスナーの監視を一時停止および再開できるため、このオペランドを指定することを推奨します。 [3.2.7(1) サーバの起動・停止・監視コマンドの設定] で設定した、サーバの監視コマンドの絶対パスを指定します。
ptrlcmd_ex_inter	サーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合に指定します。5 秒を指定してください。
ptrlcmd_ex_retry	サーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合に指定します。ptrlcmd_ex オペランドに指定したサーバの監視コマンドで監視処理に失敗した場合に、監視をリトライする回数を指定します。 サーバの監視コマンドが一時的に失敗する場合を考慮して、2 回リトライすることを推奨します。
ptrlcmd_ex_tmout	サーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合に指定します。 Oracle インスタンスおよび Oracle リスナーのスローダウンの検知については、PATROL で監視しますが、ptrlcmd_ex_tmout オペランドではサーバの監視コマンドの無応答を監視します。監視がいつまでも無応答とならないようにするため、ptrlcmd_ex_tmout オペランドの指定を推奨します。ただし、監視時間が小さいと誤検知につながりますので、十分大きな値を指定してください。 目安として、サーバの監視コマンドの開始から終了までの時間 (監視する Oracle インスタンスおよび Oracle リスナーの PATROL 値の総和) の 2 倍以上の値を指定してください。 (例) Oracle インスタンス数が 1 で PATROL が 120 秒、Oracle リスナー数が 1 で PATROL が 120 秒である場合、480 以上を指定します ((120 秒+120 秒) × 2=480 秒)。
servexec_retry	0 を指定することを推奨します。 ただし、サーバ障害を検出した際に再起動をしたい場合は、指定を省略するか、または再起動回数を指定してください。
waitserv_exec	yes を指定します。
group	[3.1.1 Oracle の構成と系切り替え単位の検討] で検討したとおり、サーバを分けてグループ化する場合に指定します。同じグループのサーバには、同じグループ名を指定します。

オペランド名	注意事項
start_timeout	このオペランドを指定する場合は、HA モニタの環境設定の termcmd_at_abort オペランドに yes を指定し、タイムアウト時にサーバの停止コマンドを実行して、再起動できるようにします。
stop_timeout	指定しないでください。stop_timeout オペランドの指定による停止処理のタイムアウトでは、Oracle インスタンスおよび Oracle リスナーの強制停止が実行されない場合があります。

3.2.10 障害情報を収集するための設定

Oracle を制御するサーバに UAP を追加する構成では、UAP のスローダウンが発生した場合にシェルを実行させて原因を調査するための情報を取得できます。

詳細については、マニュアル「HA モニタ Linux(R)編」の「障害情報を収集するための設定」を参照してください。

3.2.11 構築したシステムの動作確認

システムの構築が完了して設定と定義のチェックに問題がなかったら、構築システムの動作確認をしてください。ここでは、HA モニタ kit を使用したシステムで、特に確認しておきたい点についてだけ説明します。システムの動作確認の流れや詳細については、マニュアル「HA モニタ Linux(R)編」の「構築したシステムの動作確認」の説明を参照してください。

(1) サーバの動作確認

HA モニタ kit を使用する上で必要な、サーバの起動・停止確認について説明します。

サーバの起動確認

HA モニタ kit のサーバログを基に、次のことを確認します。

- Oracle インスタンスに対応するサーバの場合は、該当する Oracle インスタンスに対してメッセージ KAMK00302-I が出力されている。
- Oracle リスナーに対応するサーバの場合は、該当する Oracle リスナーに対してメッセージ KAMK00322-I が出力されている。

メッセージが出力されていない場合、サーバの起動コマンドに設定した環境変数に誤りがあるおそれがあります。サーバの起動コマンドの設定については「[4.2 サーバの起動コマンドの設定](#)」を参照してください。

サーバの停止確認

HA モニタ kit のサーバログを基に、次のことを確認します。

- Oracle インスタンスに対応するサーバの場合は、該当する Oracle インスタンスに対してメッセージ KAMK00305-I が出力されている。
- Oracle リスナーに対応するサーバの場合は、該当する Oracle リスナーに対してメッセージ KAMK00325-I が出力されている。

メッセージが出力されていない場合、サーバの停止コマンドに設定した環境変数に誤りがあるおそれがあります。サーバの停止コマンドの設定については「[4.3 サーバの停止コマンドの設定](#)」を参照してください。

サーバの監視確認

HA モニタ kit のサーバログを基に、次のことを確認します。

- Oracle インスタンスに対応するサーバ、Oracle リスナーに対応するサーバのどちらの場合もメッセージ KAMK00421-I が出力されている。

メッセージが出力されていない場合、サーバの監視コマンドまたはサーバの監視コマンド実行シェルに設定した環境変数に誤りがあるおそれがあります。サーバの監視コマンドおよびサーバの監視コマンド実行シェルの設定については「[4.5 サーバの監視コマンドの設定 \(patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合\)](#)」を参照してください。

(2) 系切り替えのテスト

HA モニタ kit を使用する上で必要な、系切り替えのテストについて説明します。

計画系切り替えの確認／サーバ障害時の自動系切り替えの確認

系切り替えが成功したあとに切り替え先の系で、HA モニタ kit のサーバログを基に、次のことを確認します。

- Oracle インスタンスに対応するサーバの場合は、該当する Oracle インスタンスに対してメッセージ KAMK00302-I が出力されている。
- Oracle リスナーに対応するサーバの場合は、該当する Oracle リスナーに対してメッセージ KAMK00322-I が出力されている。

メッセージが出力されていない場合、サーバの起動コマンドに設定した環境変数に誤りがあるおそれがあります。サーバの起動コマンドの設定については「[4.2 サーバの起動コマンドの設定](#)」を参照してください。

3.3 環境設定例

HA モニタ kit を使用したシステムの環境設定例を示します。ここでは、Oracle のシステム構成別に環境設定の例を示します。

- CDB (シングルテナント) 構成で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成
- シングル構成 (HA モニタのサーバ対応の環境設定の `ptrlcmd_ex` オペランドでサーバの監視コマンドを指定する場合) で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成
- シングル構成 (HA モニタのサーバ対応の環境設定の `patrolcommand` オペランドでサーバの監視コマンド実行シェルを指定する場合) で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成

3.3.1 CDB (シングルテナント) 構成で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例

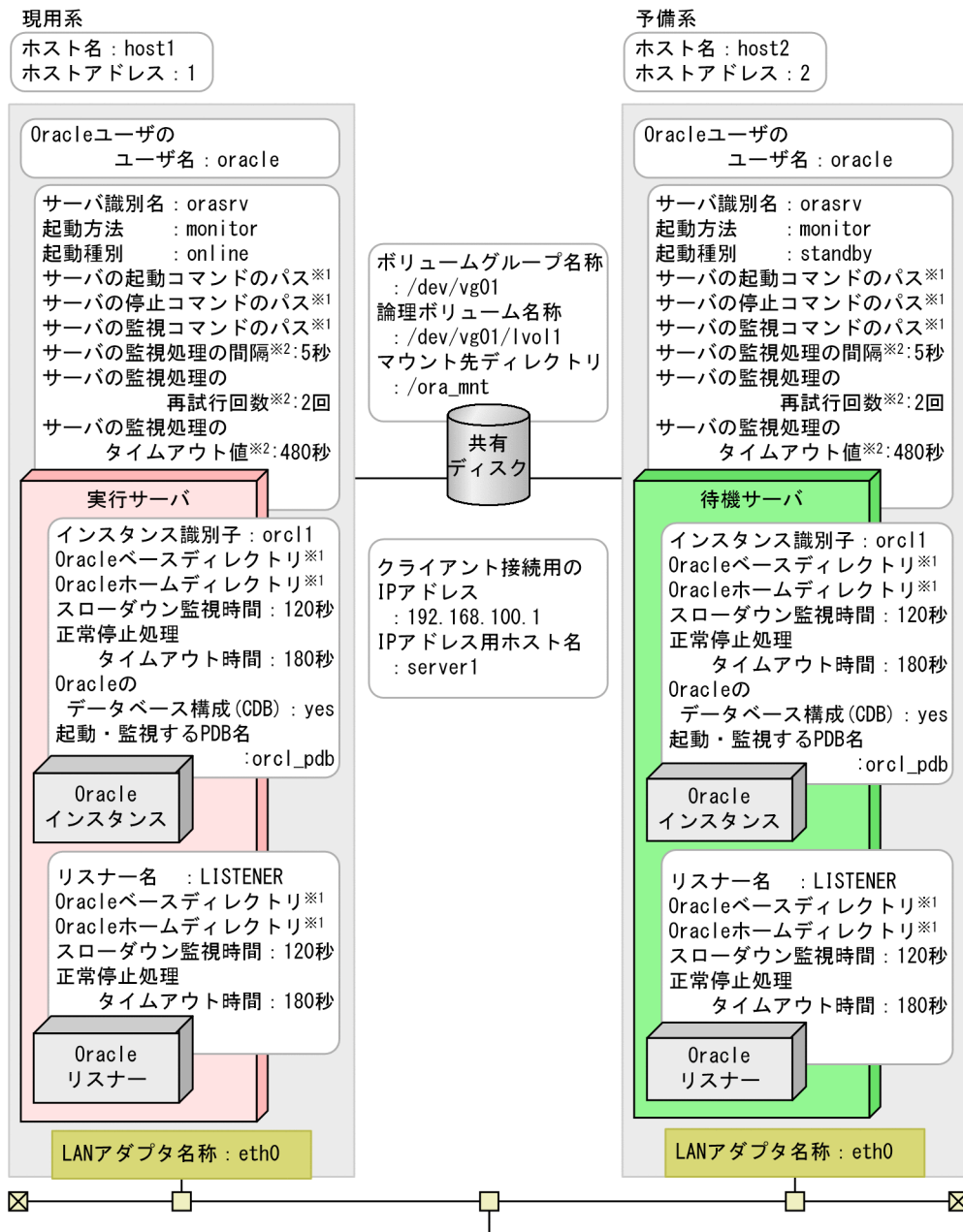
CDB (シングルテナント) 構成で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成の例を、次の図に示します。

参考

LAN の状態設定ファイル (サーバ識別名.up およびサーバ識別名.down) については、マニュアル「HA モニタ Linux(R)編」の「LAN の状態設定ファイルの設定」の説明を参照して作成してください。構成例では、次のようになります。

- サーバ識別名.up (定義ファイル: /opt/hitachi/HAMon/etc/orasrv.up)
- サーバ識別名.down (定義ファイル: /opt/hitachi/HAMon/etc/orasrv.down)

図 3-12 CDB (シングルテナント) 構成で, Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例



注※1 環境設定例中で使用するディレクトリのパスを次に示します。

- サーバの起動コマンド: /opt/hitachi/HAMon/etc/orasrv/actcommand_orasrv
 - サーバの停止コマンド: /opt/hitachi/HAMon/etc/orasrv/termcommand_orasrv
 - サーバの監視コマンド: /opt/hitachi/HAMon/etc/orasrv/ptlrcmd_ex_orasrv
 - Oracle ベースディレクトリ: /home/oracle
 - Oracle ホームディレクトリ: /home/oracle/product/18.0.0/dbhome_1
- 各パラメタは, 実際の環境に合わせて設定してください。

注※2 サーバの監視コマンドを指定します。

- サーバの監視コマンドが監視処理を実施する間隔です (ptrlcmd_ex_inter)。
- サーバの監視コマンドを再試行する回数です (ptrlcmd_ex_retry)。
- サーバの監視コマンドがタイムアウトする時間です (ptrlcmd_ex_tmout)。

このシステム構成での環境設定例を、次に示します。各パラメタは、実際の環境に合わせて設定してください。

/etc/hosts ファイル

```
192.168.100.1 server1 # alias IP address
```

Oracle のローカル・ネーミング・パラメータのファイル (定義ファイル: tnsnames.ora)

```
ORCL1 =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = server1)(PORT = 1521))  
    (CONNECT_DATA =  
      (SERVER = SHARED)  
      (SERVICE_NAME = orcl1)  
    )  
  )
```

Oracle インスタンス制御用定義ファイル (定義ファイル: /opt/hitachi/HAmon/etc/HAmonOra_etc/instance/orcl1.env)

```
ORACLE_SID=orcl1  
ORACLE_BASE=/home/oracle  
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1  
ORACLE_USER=oracle  
PATROL=120  
ABORT_LIMIT=180  
CDB=yes  
PDB_NAME=orcl_pdb
```

Oracle リスナー制御用定義ファイル (定義ファイル: /opt/hitachi/HAmon/etc/HAmonOra_etc/listener/LISTENER.env)

```
LISTENER_NAME=LISTENER  
ORACLE_BASE=/home/oracle  
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1  
ORACLE_USER=oracle  
PATROL=120  
ABORT_LIMIT=180
```

サーバ対応の環境設定 (定義ファイル: /opt/hitachi/HAmon/etc/servers)

```
server name /opt/hitachi/HAmon/etc/orasrv/actcommand_orasrv,  
alias orasrv,  
acttype monitor,  
initial online※,  
termcommand /opt/hitachi/HAmon/etc/orasrv/termcommand_orasrv,  
ptrlcmd_ex /opt/hitachi/HAmon/etc/orasrv/ptrlcmd_ex_orasrv,
```



```

ptrlcmd_ex_inter 5,
ptrlcmd_ex_retry 2,
ptrlcmd_ex_tmout 480,
disk             /dev/vg01,
fs_name          /dev/vg01/lvol1,
fs_mount_dir    /ora_mnt,
lan_updown       use,
waitserv_exec    yes,
servexec_retry  0,
start_timeout    180;

```

注※ 予備系の場合は、この値を「standby」にしてください。

サーバの起動コマンド

```

#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
act_proc(){
    RT_CD=0
    # Listener
    haoralsnrbgn LISTENER

    # Instance
    haorainsbgn orcl1

    return ${RT_CD}
}
#####
: (省略)

```

サーバの停止コマンド

```

#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
term_proc(){
    RT_CD=0
    case "$1" in
        "-e" ) # normal end

            # Instance
            haorainsend orcl1
            if [ "$?" != "0" ]
            then
                RT_CD=1
            fi

            # Listener
            haoralsnrend LISTENER
            if [ "$?" != "0" ]
            then
                RT_CD=2
            fi
        ;;
    esac
}

```



```

"-w" )                                # monswap
    # Instance
    haorainsend orcl1
    if [ "$?" != "0" ]
    then
        RT_CD=1
    fi

    # Listener
    haoralsnrend LISTENER
    if [ "$?" != "0" ]
    then
        RT_CD=2
    fi
    ;;
"-c" )                                # down
    # Instance
    haorainsend orcl1

    # Listener
    haoralsnrend LISTENER
    ;;
esac
return ${RT_CD}
}
#####
: (省略)

```

サーバの監視コマンド

```

#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
patrol_proc(){
    RT_CD=0
    RC=0

    # Instance
    haorainsptl orcl1
    RC=$?
    if [ "$RC" != "0" ]
    then
        if [ "$RC" = "125" ]
        then
            RT_CD=10
        else
            RT_CD=1
        fi
    fi

    # Listener
    haoralsnrptl LISTENER
    RC=$?
    if [ "$RC" != "0" ]
    then
        if [ "$RC" = "125" ]

```

```

        then
            RT_CD=11
        else
            if [ "$RT_CD" -lt "10" ]
            then
                RT_CD=2
            fi
        fi
    fi

    return ${RT_CD}
}
#####
: (省略)

```

3.3.2 シングル構成（HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合）で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例

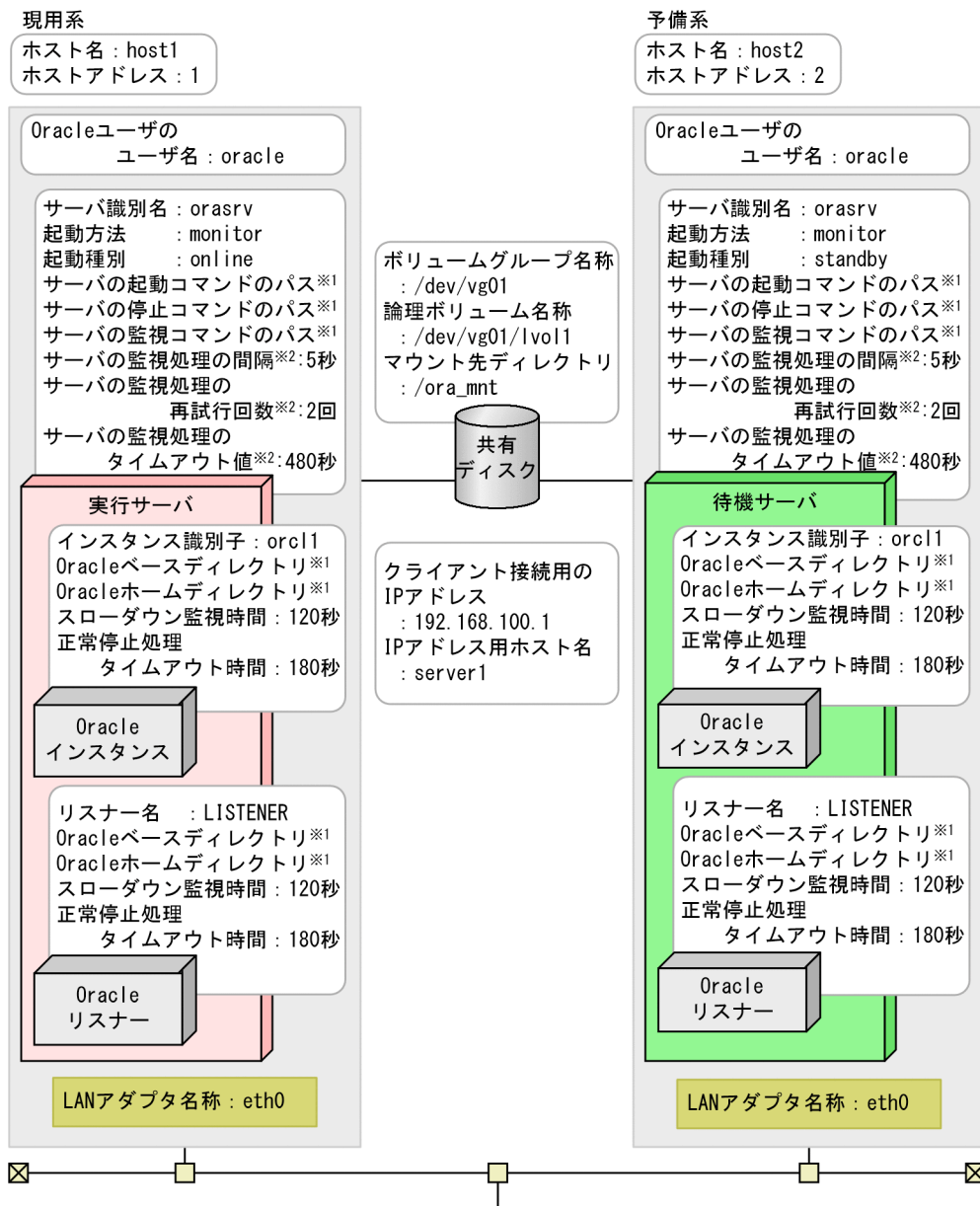
シングル構成（HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合）で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成の例を、次に示します。

参考

LAN の状態設定ファイル（サーバ識別名.up およびサーバ識別名.down）については、マニュアル「HA モニタ Linux(R)編」の「LAN の状態設定ファイルの設定」の説明を参照して作成してください。構成例では、次のようになります。

- サーバ識別名.up（定義ファイル：/opt/hitachi/HAMon/etc/orasrv.up）
- サーバ識別名.down（定義ファイル：/opt/hitachi/HAMon/etc/orasrv.down）

図 3-13 シングル構成 (HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合) で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例



注※1 環境設定例中で使用するディレクトリのパスを次に示します。

- サーバの起動コマンド : /opt/hitachi/HAMon/etc/orasrv/actcommand_orasrv
 - サーバの停止コマンド : /opt/hitachi/HAMon/etc/orasrv/termcommand_orasrv
 - サーバの監視コマンド : /opt/hitachi/HAMon/etc/orasrv/ptrlcmd_ex_orasrv
 - Oracle ベースディレクトリ : /home/oracle
 - Oracle ホームディレクトリ : /home/oracle/product/18.0.0/dbhome_1
- 各パラメタは、実際の環境に合わせて設定してください。

注※2 サーバの監視コマンドを指定します。

- サーバの監視コマンドが監視処理を実施する間隔です (ptrlcmd_ex_inter)。
- サーバの監視コマンドを再試行する回数です (ptrlcmd_ex_retry)。
- サーバの監視コマンドがタイムアウトする時間です (ptrlcmd_ex_tmout)。

このシステム構成での環境設定例を、次に示します。各パラメタは、実際の環境に合わせて設定してください。

/etc/hosts ファイル

```
192.168.100.1 server1 # alias IP address
```

Oracle のローカル・ネーミング・パラメータのファイル (定義ファイル: tnsnames.ora)

```
ORCL1 =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = server1)(PORT = 1521))  
    (CONNECT_DATA =  
      (SERVER = SHARED)  
      (SERVICE_NAME = orcl1)  
    )  
  )
```

Oracle インスタンス制御用定義ファイル (定義ファイル: /opt/hitachi/HAMon/etc/HAMonOra_etc/instance/orcl1.env)

```
ORACLE_SID=orcl1  
ORACLE_BASE=/home/oracle  
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1  
ORACLE_USER=oracle  
PATROL=120  
ABORT_LIMIT=180
```

Oracle リスナー制御用定義ファイル (定義ファイル: /opt/hitachi/HAMon/etc/HAMonOra_etc/listener/LISTENER.env)

```
LISTENER_NAME=LISTENER  
ORACLE_BASE=/home/oracle  
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1  
ORACLE_USER=oracle  
PATROL=120  
ABORT_LIMIT=180
```

サーバ対応の環境設定 (定義ファイル: /opt/hitachi/HAMon/etc/servers)

```
server name          /opt/hitachi/HAMon/etc/orasrv/actcommand_orasrv,  
alias                orasrv,  
acttype              monitor,  
initial              online※,  
termcommand          /opt/hitachi/HAMon/etc/orasrv/termcommand_orasrv,  
ptrlcmd_ex           /opt/hitachi/HAMon/etc/orasrv/ptrlcmd_ex_orasrv,  
ptrlcmd_ex_inter 5,  
ptrlcmd_ex_retry 2,
```

```

ptrlcmd_ex_tmout 480,
disk              /dev/vg01,
fs_name           /dev/vg01/lvol1,
fs_mount_dir     /ora_mnt,
lan_updown       use,
waitserv_exec    yes,
servexec_retry   0,
start_timeout    180;

```

注※ 予備系の場合は、この値を「standby」にしてください。

サーバの起動コマンド

```

#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
act_proc(){
    RT_CD=0
    # Listener
    haoralsnrbgn LISTENER

    # Instance
    haorainsbgn orcl1

    return ${RT_CD}
}
#####
: (省略)

```

サーバの停止コマンド

```

#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
term_proc(){
    RT_CD=0
    case "$1" in
        "-e" )                # normal end
            # Instance
            haorainsend orcl1
            if [ "$?" != "0" ]
            then
                RT_CD=1
            fi

            # Listener
            haoralsnrend LISTENER
            if [ "$?" != "0" ]
            then
                RT_CD=2
            fi
            ;;
        "-w" )                # monswap
            # Instance

```

```

        haorainsend orcl1
        if [ "$?" != "0" ]
        then
            RT_CD=1
        fi

        # Listener
        haoralsnrend LISTENER
        if [ "$?" != "0" ]
        then
            RT_CD=2
        fi
        ;;
    "-c" )                # down
        # Instance
        haorainsend orcl1

        # Listener
        haoralsnrend LISTENER
        ;;
    esac
    return ${RT_CD}
}
#####
: (省略)

```

サーバの監視コマンド

```

#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
patrol_proc(){
    RT_CD=0
    RC=0

    # Instance
    haorainsptl orcl1
    RC=$?
    if [ "$RC" != "0" ]
    then
        if [ "$RC" = "125" ]
        then
            RT_CD=10
        else
            RT_CD=1
        fi
    fi

    # Listener
    haoralsnrptl LISTENER
    RC=$?
    if [ "$RC" != "0" ]
    then
        if [ "$RC" = "125" ]
        then
            RT_CD=11

```

```

        else
            if [ "$RT_CD" -lt "10" ]
            then
                RT_CD=2
            fi
        fi
    fi

    return ${RT_CD}
}
#####
: (省略)

```

3.3.3 シングル構成 (HA モニタのサーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合) で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例

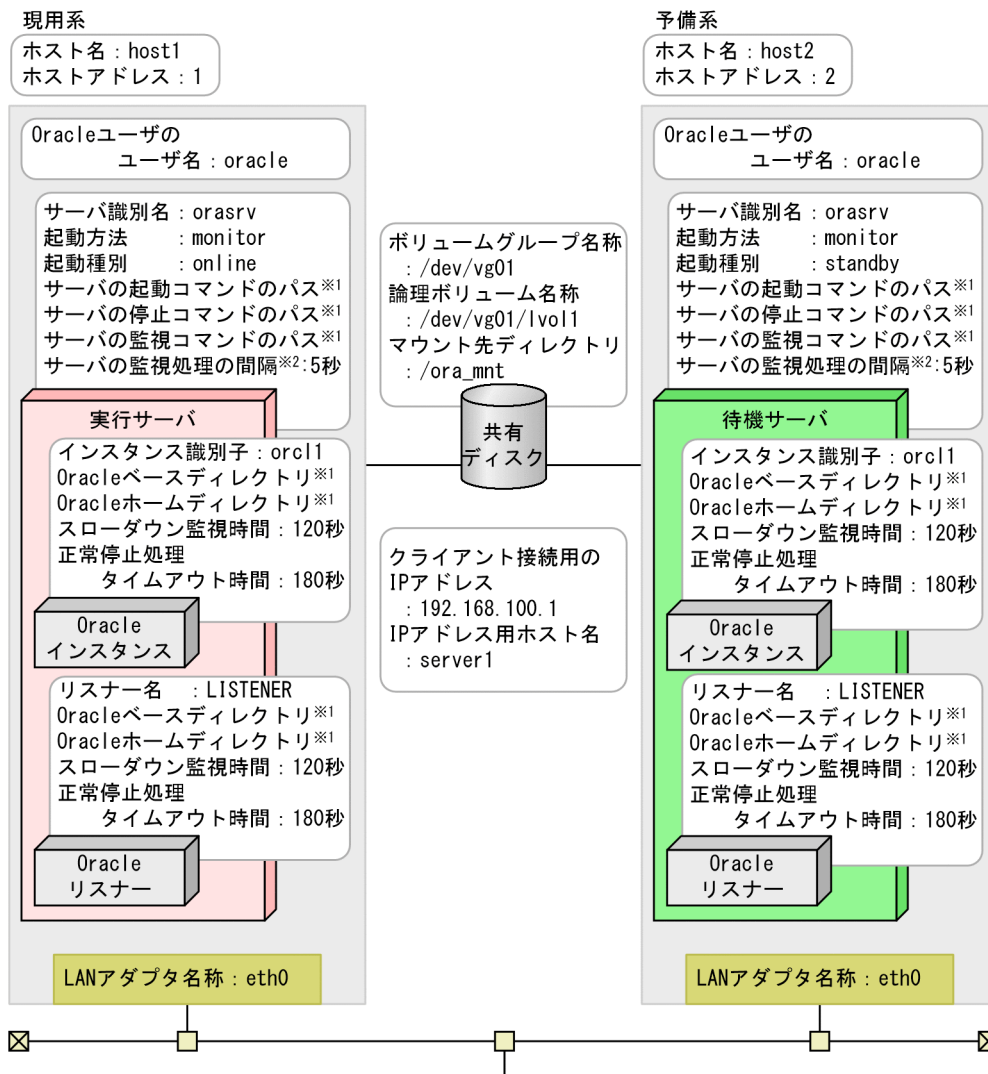
シングル構成 (HA モニタのサーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合) で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成の例を、次に示します。

参考

LAN の状態設定ファイル (サーバ識別名.up およびサーバ識別名.down) については、マニュアル「HA モニタ Linux(R)編」の「LAN の状態設定ファイルの設定」の説明を参照して作成してください。構成例では、次のようになります。

- サーバ識別名.up (定義ファイル: /opt/hitachi/HAMon/etc/orasrv.up)
- サーバ識別名.down (定義ファイル: /opt/hitachi/HAMon/etc/orasrv.down)

図 3-14 シングル構成 (HA モニタのサーバ対応の環境設定の patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合) で、Oracle インスタンスと Oracle リスナーを一つのサーバで監視する場合のシステム構成例



注※1 環境設定例中で使用するディレクトリのパスを次に示します。

- サーバの起動コマンド : /opt/hitachi/HAMon/etc/orasrv/actcommand_orasrv
 - サーバの停止コマンド : /opt/hitachi/HAMon/etc/orasrv/termcommand_orasrv
 - サーバの監視コマンド実行シェル : /opt/hitachi/HAMon/etc/orasrv/patrol_orasrv.sh
 - Oracle ベースディレクトリ : /home/oracle
 - Oracle ホームディレクトリ : /home/oracle/product/18.0.0/dbhome_1
- 各パラメタは、実際の環境に合わせて設定してください。

注※2 サーバの監視コマンドが監視処理を実施する間隔です。

このシステム構成での環境設定例を、次に示します。各パラメタは、実際の環境に合わせて設定してください。

/etc/hosts ファイル

```
192.168.100.1 server1 # alias IP address
```

Oracle のローカル・ネーミング・パラメータのファイル (定義ファイル: tnsnames.ora)

```
ORCL1 =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = server1)(PORT = 1521))  
    (CONNECT_DATA =  
      (SERVER = SHARED)  
      (SERVICE_NAME = orcl1)  
    )  
  )  
)
```

Oracle インスタンス制御用定義ファイル (定義ファイル: /opt/hitachi/HAmom/etc/HAmomOra_etc/instance/orcl1.env)

```
ORACLE_SID=orcl1  
ORACLE_BASE=/home/oracle  
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1  
ORACLE_USER=oracle  
PATROL=120  
ABORT_LIMIT=180
```

Oracle リスナー制御用定義ファイル (定義ファイル: /opt/hitachi/HAmom/etc/HAmomOra_etc/listener/LISTENER.env)

```
LISTENER_NAME=LISTENER  
ORACLE_BASE=/home/oracle  
ORACLE_HOME=/home/oracle/product/18.0.0/dbhome_1  
ORACLE_USER=oracle  
PATROL=120  
ABORT_LIMIT=180
```

サーバ対応の環境設定 (定義ファイル: /opt/hitachi/HAmom/etc/servers)

```
/* サーバ対応の環境設定 */  
server name      /opt/hitachi/HAmom/etc/orasrv/actcommand_orasrv,  
alias            orasrv,  
acttype          monitor,  
initial          online※,  
termcommand      /opt/hitachi/HAmom/etc/orasrv/termcommand_orasrv,  
patrolcommand    /opt/hitachi/HAmom/etc/orasrv/patrol_orasrv.sh,  
disk             /dev/vg01,  
fs_name          /dev/vg01/lvol1,  
fs_mount_dir     /ora_mnt,  
lan_updown       use,  
waitserv_exec    yes,  
servexec_retry   0,  
start_timeout    180;
```

注※ 予備系の場合は、この値を「standby」にしてください。

サーバの起動コマンド

```
#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
act_proc(){
    RT_CD=0
    # Listener
    haoralsnrbgn LISTENER

    # Instance
    haorainsbgn orcl1

    return ${RT_CD}
}
#####
: (省略)
```

サーバの停止コマンド

```
#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
term_proc(){
    RT_CD=0
    case "$1" in
        "-e" ) # normal end
            # Instance
            haorainsend orcl1
            if [ "$?" != "0" ]
            then
                RT_CD=1
            fi

            # Listener
            haoralsnrend LISTENER
            if [ "$?" != "0" ]
            then
                RT_CD=2
            fi
            ;;
        "-w" ) # monswap
            # Instance
            haorainsend orcl1
            if [ "$?" != "0" ]
            then
                RT_CD=1
            fi

            # Listener
            haoralsnrend LISTENER
            if [ "$?" != "0" ]
            then
                RT_CD=2
            fi
        ;;
    esac
}
```

```

        fi
        ;;
    "-c" )                # down
        # Instance
        haorainsend orcl1

        # Listener
        haoralsnrend LISTENER
        ;;
    esac
    return ${RT_CD}
}
#####
: (省略)

```

サーバの監視コマンド

```

#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
PATROL_INTERVAL=5
#####
patrol_proc(){
    RT_CD=0
    # Instance
    haorainsptl orcl1
    if [ "$?" != "0" ]
    then
        RT_CD=1
    fi

    # Listener
    haoralsnrptl LISTENER
    if [ "$?" != "0" ]
    then
        RT_CD=2
    fi

    return 0
}
#####
: (省略)

```

サーバの監視コマンド実行シェル

```

#!/bin/sh
#####
ORACLE_USER=oracle
PATROL_SHELL=/opt/hitachi/HAMon/etc/patrolcommand_orasrv
#####
: (省略)

```

4

サーバの起動・停止・監視コマンドの設定

この章では、サーバの起動・停止・監視コマンドの設定、および使用するコマンドについて説明します。

4.1 サーバの起動・停止・監視コマンドの概要

サーバの起動・停止・監視コマンドは、プログラムを起動・停止・監視するための処理を記述したシェルです。

HA モニタ kit では、サーバの起動・停止・監視コマンドのサンプルファイルを提供しています。サンプルファイルをコピーし、この章で説明する内容に従ってサーバの起動・停止・監視コマンドを設定し、必要に応じてカスタマイズしてください。

サーバの起動・停止・監視コマンドの詳細については、マニュアル「HA モニタ Linux(R)編」の「サーバの起動・停止・監視コマンドの作成」の説明を参照してください。

なお、サーバの起動・停止・監視コマンドが出力するメッセージ（サーバの起動・停止・監視コマンドの延長で実行される Oracle のコマンドの、標準出力・標準エラー出力も含まれます）は、サーバログに出力されます。サーバログについては、「[5.4 サーバログの確認](#)」を参照してください。

4.2 サーバの起動コマンドの設定

HA モニタ kit は、サーバの起動コマンドのサンプルファイルを提供しています。サンプルファイルをコピーしてここで説明する内容に従って設定し、必要に応じてカスタマイズしてください。サンプルファイルの格納場所やコピー先については、「3.2.7(1) サーバの起動・停止・監視コマンドの設定」を参照してください。

サーバの起動コマンドでは、起動処理が記述された `act_proc()` シェル関数を実行し、Oracle リスナーと Oracle インスタンスを起動します。

サーバの起動コマンドのサンプルファイルの内容を次の図に示します。

図 4-1 サーバの起動コマンドのサンプルファイルの内容

```
#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
act_proc() {
    RT_CD=0
    # Listener
    haoralsnrbgn LISTENER
    # Instance
    haorainsbgn orcl1
    return ${RT_CD}
}
#####
:
```

図中の 1.~5. について説明します。

1. プログラムの起動に必要な動作環境を設定する環境変数です。

ここで設定する内容については、「4.2.1 環境変数」を参照してください。

2. Oracle リスナーの起動処理です。

`haoralsnrbgn` コマンドを実行して、リスナー名で指定した Oracle リスナーを起動する処理が記述されています。ここで設定する内容については、「4.2.2 設定方法」を参照してください。

`haoralsnrbgn` コマンドについては、「4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド」の「`haoralsnrbgn (Oracle リスナーの起動)`」を参照してください。

3. Oracle インスタンスの起動処理です。

`haorainsbgn` コマンドを実行して、インスタンス識別子で指定した Oracle インスタンスを起動する処理が記述されています。ここで設定する内容については、「4.2.2 設定方法」を参照してください。

`haorainsbgn` コマンドについては、「4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド」の「`haorainsbgn (Oracle インスタンスの起動)`」を参照してください。

4. サーバの起動コマンドの戻り値です。

ここで設定する内容については、「4.2.2 設定方法」を参照してください。

5. act_proc()シェル関数の処理です。

ユーザ独自の処理を追加したい場合、この act_proc()シェル関数の中に追加します。ここで設定する内容については、「4.2.3 サーバの起動コマンドのカスタマイズ」を参照してください。

4.2.1 環境変数

図 4-1 の 1.の環境変数について説明します。

- LOGNAME

サーバログのファイル名を指定します。起動・停止・監視コマンドの LOGNAME の値は、すべて同じ値を指定してください。また、サーバごとにユニークな値を指定してください。そのため LOGNAME に指定する値は、サーバのサーバ識別名と合わせることをお勧めします。提供するサンプルファイルでは、サーバ識別名として「orasrv」が仮定されています。

サーバログの出力先については、「5.4 サーバログの確認」を参照してください。

- LOGSIZE

サーバログのファイルサイズ（単位：バイト）を整数で指定します。デフォルトでは「1048576」が仮定されています。通常はデフォルトのファイルサイズのまま運用します。出力内容が多く、ログ情報が不足する場合にだけ拡張してください。サーバログのファイルサイズの算出方法については、「5.4.2 ファイルサイズ」を参照してください。

なお、LOGNAME 環境変数に同じ値を指定したサーバでは、サーバの起動・停止・監視コマンドの LOGSIZE 環境変数にも同じ値を指定してください。

4.2.2 設定方法

図 4-1 の 2.~4.の起動処理の設定方法について説明します。

- 図中の 2. (Oracle リスナーの起動処理) の設定

- 提供するサンプルファイルでは、サーバに対応する Oracle リスナー（リスナー名「LISTENER」）を一つ起動する設定になっています。サーバに対応する Oracle リスナーがない場合は、図中の 2. の記述をコメントアウト、または削除してください。一つのサーバで複数の Oracle リスナーを起動したい場合は、図中の 2.の記述を追加してください。

- 提供するサンプルファイルでは、リスナー名として「LISTENER」が仮定されています。下線部の引数「LISTENER」を、起動したい Oracle リスナーのリスナー名に変更してください。

- 図中の 3. (Oracle インスタンスの起動処理) の設定

- 提供するサンプルファイルでは、サーバに対応する Oracle インスタンス（インスタンス識別子「orcl1」）を一つ起動する設定になっています。サーバに対応する Oracle インスタンスがない場合

は、図中の 3.の記述をコメントアウト、または削除してください。一つのサーバで複数の Oracle インスタンスを起動したい場合は、図中の 3.の記述を追加してください。

- 提供するサンプルファイルでは、インスタンス識別子として「orcl1」が仮定されています。下線部の引数「orcl1」を、起動したい Oracle インスタンスのインスタンス識別子に変更してください。
- 図中の 4. (サーバの起動コマンドの戻り値) の設定
 - 提供するサンプルファイルでは、Oracle インスタンスまたは Oracle リスナーの起動に失敗した場合に、サーバの起動を中止しない設定になっています。Oracle インスタンスまたは Oracle リスナーの起動に失敗した場合は、サーバの起動完了後に監視処理で障害を検出します。

4.2.3 サーバの起動コマンドのカスタマイズ

サーバの起動コマンドの中に、業務に必要な、Oracle 以外のアプリケーションの処理やユーザ独自の処理などを追加できます。カスタマイズをする場合には、次の注意事項に従ってください。

- 処理を追加する前に、マニュアル「HA モニタ Linux(R)編」の「サーバの起動コマンドの作成」の説明を参照してください。
- 処理は、サーバの起動コマンド内の `act_proc()` シェル関数の中に追加してください。`act_proc()` シェル関数とシェル変数以外は、編集しないでください。
ここに記述した処理はスーパーユーザの権限で実行されます。権限を変更した場合は、スーパーユーザの権限に戻してください。
- `act_proc()` シェル関数の中では、`exit` コマンドを使用しないでください。処理を終了させる場合は、`return` コマンドを使用してください。
- `act_proc()` シェル関数のリターンコードがサーバの起動コマンドの戻り値となります。サーバの起動を中止したい場合は、`act_proc()` シェル関数を 0 以外でリターンさせてください。
- 実行したコマンドが常駐プロセスになるなど、サーバの起動コマンドに制御が戻らない処理は追加しないでください。
- `act_proc()` シェル関数が出力するメッセージの出力先については、「[5.4 サーバログの確認](#)」を参照してください。
- `act_proc()` シェル関数の中で実行するコマンドや入出力するファイルは、絶対パスで指定してください。ここでいうコマンドとは、HA モニタ kit が提供するコマンド以外のものを指します。
- 次のシェル変数名を変更しないでください。
 - LOGNAME
 - LOGSIZE
- シェル変数を追加する場合は、コマンド内を検索し、すでに使用されていないことを確認してから追加してください。

4.3 サーバの停止コマンドの設定

HA モニタ kit は、サーバの停止コマンドのサンプルファイルを提供しています。サンプルファイルをコピーしてここで説明する内容に従って設定し、必要に応じてカスタマイズしてください。サンプルファイルの格納場所やコピー先については、「3.2.7(1) サーバの起動・停止・監視コマンドの設定」を参照してください。

サーバの停止コマンドのサンプルファイルは、停止処理が記述された `term_proc()` シェル関数を実行し、Oracle インスタンスと Oracle リスナーを停止します。

サーバの停止コマンドのサンプルファイルの内容を次の図に示します。

図 4-2 サーバの停止コマンドのサンプルファイルの内容

```
#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
term_proc() {
    RT_CD=0
    case "$1" in
        "-e" )                # normal end
            # Instance
            haorainsend orc11
            if [ "$?" != "0" ]
            then
                RT_CD=1
            fi
            # Listener
            haoralnsrend LISTENER
            if [ "$?" != "0" ]
            then
                RT_CD=2
            fi
            ;;
        "-w" )                # monswap
            # Instance
            haorainsend orc11
            if [ "$?" != "0" ]
            then
                RT_CD=1
            fi
            # Listener
            haoralnsrend LISTENER
            if [ "$?" != "0" ]
            then
                RT_CD=2
            fi
            ;;
    esac
}
```

The diagram illustrates the structure of the `term_proc()` function. Annotations are as follows:

- 1. Points to the header section: `LOGNAME=orasrv` and `LOGSIZE=1048576`.
- 2. Points to the Instance stopping logic: `haorainsend orc11` and the subsequent `if` statement.
- 3. Points to the Listener stopping logic: `haoralnsrend LISTENER` and the subsequent `if` statement.
- 4. Points to the `monswap` section, which includes the Instance and Listener stopping logic.

```

"-c" ) # down
        # Instance
        haorainsend orcl1 ]-2.
        # Listener
        haoralsnrend LISTENER ]-3.
        ;; ]-4.
    esac ]-5.
    return ${RT_CD}
}
#####
:
] これ以降は
  編集できません

```

図中の 1.~5.について説明します。

1. プログラムの停止に必要な動作環境を設定する環境変数です。

ここで設定する内容については、「[4.3.1 環境変数](#)」を参照してください。

2. Oracle インスタンスの停止処理です。

haorainsend コマンドを実行して、インスタンス識別子で指定した Oracle インスタンスを停止する処理が記述されています。ここで設定する内容については、「[4.3.2 設定方法](#)」を参照してください。

haorainsend コマンドについては、「[4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド](#)」の「[haorainsend \(Oracle インスタンスの停止\)](#)」を参照してください。

3. Oracle リスナーの停止処理です。

haoralsnrend コマンドを実行して、リスナー名で指定した Oracle リスナーを停止する処理が記述されています。ここで設定する内容については、「[4.3.2 設定方法](#)」を参照してください。

haoralsnrend コマンドについては、「[4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド](#)」の「[haoralsnrend \(Oracle リスナーの停止\)](#)」を参照してください。

4. term_proc()シェル関数の処理です。

ユーザ独自の処理を追加したい場合、この term_proc()シェル関数の中に追加します。ここで設定する内容については、「[4.3.3 サーバの停止コマンドのカスタマイズ](#)」を参照してください。

5. サーバの停止コマンドの戻り値です。

4.3.1 環境変数

図 4-2 の 1.の環境変数について説明します。

- LOGNAME

サーバログのファイル名を指定します。起動・停止・監視コマンドの LOGNAME の値は、すべて同じ値を指定してください。また、サーバごとにユニークな値を指定してください。そのため LOGNAME に指定する値は、サーバのサーバ識別名と合わせることをお勧めします。提供するサンプルファイルでは、サーバ識別名として「orasrv」が仮定されています。

サーバログの出力先については、「[5.4 サーバログの確認](#)」を参照してください。

- LOGSIZE

サーバログのファイルサイズ（単位：バイト）を整数で指定します。デフォルトでは「1048576」が仮定されています。通常はデフォルトのファイルサイズのまま運用します。出力内容が多く、ログ情報が不足する場合にだけ拡張してください。サーバログのファイルサイズの算出方法については、「[5.4.2 ファイルサイズ](#)」を参照してください。

なお、LOGNAME 環境変数に同じ値を指定したサーバでは、サーバの起動・停止・監視コマンドの LOGSIZE 環境変数に同じ値を指定してください。

4.3.2 設定方法

図 4-2 の 2. と 3. の停止処理の設定方法について説明します。

- 図中の 2. (Oracle インスタンスの停止処理) の設定
 - 提供するサンプルファイルでは、サーバに対応する Oracle インスタンス（インスタンス識別子「orcl1」）を一つ停止する設定になっています。サーバに対応する Oracle インスタンスがない場合は、図中の 2. の記述をコメントアウト、または削除してください。一つのサーバで複数の Oracle インスタンスを停止したい場合は、図中の 2. の記述を追加してください。
 - 提供するサンプルファイルでは、インスタンス識別子として「orcl1」が仮定されています。下線部の引数「orcl1」を、停止したい Oracle インスタンスのインスタンス識別子に変更してください。
- 図中の 3. (Oracle リスナーの停止処理) の設定
 - 提供するサンプルファイルでは、サーバに対応する Oracle リスナー（リスナー名「LISTENER」）を一つ停止する設定になっています。サーバに対応する Oracle リスナーがない場合は、図中の 3. の記述をコメントアウト、または削除してください。一つのサーバで複数の Oracle リスナーを停止したい場合は、図中の 3. の記述を追加してください。
 - 提供するサンプルファイルでは、リスナー名として「LISTENER」が仮定されています。下線部の引数「LISTENER」を、停止したい Oracle リスナーのリスナー名に変更してください。

なお、Oracle インスタンスおよび Oracle リスナーの停止順序を変更したい場合は、ファイル内の停止処理の順番を変更してください。

4.3.3 サーバの停止コマンドのカスタマイズ

サーバの停止コマンドの中に、業務に必要な、Oracle 以外のアプリケーションの処理やユーザ独自の処理などを追加できます。カスタマイズをする場合には、次の注意事項に従ってください。

- 処理を追加する前に、マニュアル「HA モニタ Linux(R)編」の「サーバの停止コマンドの作成」の説明を参照してください。

- 処理は、サーバの停止コマンド内の `term_proc()` シェル関数の中に追加してください。 `term_proc()` シェル関数とシェル変数以外は、編集しないでください。
ここに記述した処理はスーパーユーザの権限で実行されます。権限を変更した場合は、スーパーユーザの権限に戻してください。
- `term_proc()` シェル関数の中では、 `exit` コマンドを使用しないでください。処理を終了させる場合は、 `return` コマンドを使用してください。
- `term_proc()` シェル関数のリターンコードがサーバの停止コマンドの戻り値となります。
- HA モニタがサーバの停止コマンドに渡す引数は、 `term_proc()` シェル関数の第 1 引数 (`$1`) を参照することで確認できます。
- `term_proc()` シェル関数が出力するメッセージの出力先については、「[5.4 サーバログの確認](#)」を参照してください。
- `term_proc()` シェル関数の中で実行するコマンドや入出力するファイルは、絶対パスで指定してください。ここでいうコマンドとは、HA モニタ kit が提供するコマンド以外のものを指します。
- 次のシェル変数名を変更しないでください。
 - LOGNAME
 - LOGSIZE
- シェル変数を追加する場合は、コマンド内を検索し、すでに使用されていないことを確認してから追加してください。

4.4 サーバの監視コマンドの設定 (ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合)

ここでは、ptrlcmd_ex オペランド (HA モニタのサーバ対応の環境設定) でサーバの監視コマンドを指定する場合の、サーバの監視コマンドについて説明します。

4.4.1 サーバの監視コマンド

HA モニタ kit は、サーバの監視コマンドのサンプルファイルを提供しています。サンプルファイルをコピーしてここで説明する内容に従って設定し、必要に応じてカスタマイズしてください。サンプルファイルの格納場所やコピー先については、「[3.2.7\(1\) サーバの起動・停止・監視コマンドの設定](#)」を参照してください。

サーバの監視コマンドのサンプルファイルは、監視処理が記述された patrol_proc() シェル関数を、HA モニタのサーバ対応の環境設定の ptrlcmd_ex_inter オペランドの間隔で実行し、Oracle インスタンスと Oracle リスナーを監視します。

サーバの監視コマンドのサンプルファイルの内容を次の図に示します。

図 4-3 サーバの監視コマンドのサンプルファイルの内容

```
#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
#####
patrol_proc() {
    RT_CD=0
    RC=0

    # Instance
    haorainsptl orcl1
    RC=$?
    if [ "$RC" != "0" ]
    then
        if [ "$RC" = "125" ]
        then
            RT_CD=10
        else
            RT_CD=1
        fi
    fi

    # Listener
    haoralnsrptl LISTENER
    RC=$?
    if [ "$RC" != "0" ]
    then
        if [ "$RC" = "125" ]
        then
            RT_CD=11
        else
            if [ "$RT_CD" -lt "10" ]
            then
                RT_CD=2
            fi
        fi
    fi

    return ${RT_CD}
}
#####
:
}
これ以降は
編集できません
```

1. プログラムの監視に必要な動作環境を設定する環境変数です。

ここで設定する内容については、「(1) 環境変数」を参照してください。

2. Oracle インスタンスの監視処理です。

haorainsptl コマンドを実行して、インスタンス識別子で指定した Oracle インスタンスを監視する処理が記述されています。ここで設定する内容については、「(2) 設定方法」を参照してください。

haorainsptl コマンドについては、「4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド」の「haorainsptl (Oracle インスタンスの状態確認)」を参照してください。

3. Oracle リスナーの監視処理です。

haoralnsrptl コマンドを実行して、リスナー名で指定した Oracle リスナーを監視する処理が記述されています。ここで設定する内容については、「(2) 設定方法」を参照してください。

haoralnsrptl コマンドについては、「4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド」の「haoralnsrptl (Oracle リスナーの状態確認)」を参照してください。

4. patrol_proc()シェル関数の処理です。

ユーザ独自の処理を追加したい場合、この patrol_proc()シェル関数の中に追加します。ここで設定する内容については、「(3) サーバの監視コマンドのカスタマイズ」を参照してください。

(1) 環境変数

図 4-3 の 1.の環境変数について説明します。

- LOGNAME

サーバログのファイル名を指定します。起動・停止・監視コマンドの LOGNAME の値は、すべて同じ値を指定してください。また、サーバごとにユニークな値を指定してください。そのため LOGNAME に指定する値は、サーバのサーバ識別名と合わせることをお勧めします。提供するサンプルファイルでは、サーバ識別名として「orasrv」が仮定されています。

サーバログの出力先については、「5.4 サーバログの確認」を参照してください。

- LOGSIZE

サーバログのファイルサイズ（単位：バイト）を整数で指定します。デフォルトでは「1048576」が仮定されています。通常はデフォルトのファイルサイズのまま運用します。出力内容が多く、ログ情報が不足する場合にだけ拡張してください。サーバログのファイルサイズの算出方法については、「5.4.2 ファイルサイズ」を参照してください。

なお、LOGNAME 環境変数に同じ値を指定したサーバでは、サーバの起動・停止・監視コマンドの LOGSIZE 環境変数に同じ値を指定してください。

(2) 設定方法

図 4-3 の 2.と 3.の監視処理の設定方法について説明します。

- 図中の 2. (Oracle インスタンスの監視処理) の設定

- 提供するサンプルファイルでは、サーバに対応する Oracle インスタンス（インスタンス識別子「orcl1」）を一つ監視する設定になっています。サーバに対応する Oracle インスタンスがない場合は、図中の 2.の記述をコメントアウト、または削除してください。一つのサーバで複数の Oracle インスタンスを監視したい場合は、図中の 2.の記述を追加してください。

監視処理の結果、スローダウンを検知した場合は、サーバの監視コマンドの戻り値に 10 を返します。これによって HA モニタはサーバ障害と判断します。

監視処理が失敗した場合は、サーバの監視コマンドの戻り値に 1 を返します。これによって HA モニタは監視処理をリトライします。

- 提供するサンプルファイルでは、インスタンス識別子として「orcl1」が仮定されています。下線部の引数「orcl1」を、監視したい Oracle インスタンスのインスタンス識別子に変更してください。

- 図中の 3. (Oracle リスナーの監視処理) の設定

- 提供するサンプルファイルでは、サーバに対応する Oracle リスナー（リスナー名「LISTENER」）を一つ監視する設定になっています。サーバに対応する Oracle リスナーがない場合は、図中の 3.

の記述をコメントアウト、または削除してください。一つのサーバで複数の Oracle リスナーを監視したい場合は、図中の 3. の記述を追加してください。

監視処理の結果、スローダウンを検知した場合は、サーバの監視コマンドの戻り値に 11 を返します。これによって HA モニタはサーバ障害と判断します。

監視処理が失敗した場合は、サーバの監視コマンドの戻り値に 2 を返します。これによって HA モニタは監視処理をリトライします。ただし、図中の 2. (Oracle インスタンスの監視処理) で、サーバの監視コマンドの戻り値が 10 以上の場合は、Oracle リスナーの監視処理が失敗しても戻り値を変更しません (戻り値 2 を返しません)。

- 提供するサンプルファイルでは、リスナー名として「LISTENER」が仮定されています。下線部の引数「LISTENER」を、監視したい Oracle リスナーのリスナー名に変更してください。

なお、Oracle インスタンスおよび Oracle リスナーの監視順序を変更したい場合は、ファイル内の監視処理の順番を変更してください。

(3) サーバの監視コマンドのカスタマイズ

サーバの監視コマンドの中に、業務に必要な、Oracle 以外のアプリケーションの処理やユーザ独自の処理などを追加できます。カスタマイズをする場合には、次の注意事項に従ってください。

- 処理を追加する前に、マニュアル「HA モニタ Linux(R)編」の「サーバの監視コマンドの作成」の説明を参照してください。
- 処理は、サーバの監視コマンド内の patrol_proc() シェル関数の中に追加してください。patrol_proc() シェル関数とシェル変数以外は、編集しないでください。
ここに記述した処理はスーパーユーザの権限で実行されます。権限を変更した場合は、スーパーユーザの権限に戻してください。
- patrol_proc() シェル関数の中では、exit コマンドを使用しないでください。処理を終了させる場合は、return コマンドを使用してください。
- patrol_proc() シェル関数のリターンコードが 0 以外の場合、patrol_proc() シェル関数のリターンコードがサーバの監視コマンドの終了結果 (EXIT コード) となります。HA モニタは、サーバの監視コマンドの終了結果 (EXIT コード) に応じた動作をします。詳細は、マニュアル「HA モニタ Linux(R)編」を参照してください。
- patrol_proc() シェル関数の中では、無限ループなどの終了しない処理を作り込まないでください。
- patrol_proc() シェル関数が出力するメッセージの出力先については、「5.4 サーバログの確認」を参照してください。
- patrol_proc() シェル関数の中で実行するコマンドや入出力するファイルは、絶対パスで指定してください。ここでいうコマンドとは、HA モニタ kit が提供するコマンド以外のものを指します。
- 次のシェル変数名を変更しないでください。
 - LOGNAME
 - LOGSIZE

- シェル変数を追加する場合は、コマンド内を検索し、すでに使用されていないことを確認してから追加してください。

4.5 サーバの監視コマンドの設定 (patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合)

ここでは、patrolcommand オペランド (HA モニタのサーバ対応の環境設定) でサーバの監視コマンド実行シェルを指定する場合の、サーバの監視コマンドおよびサーバの監視コマンド実行シェルについて説明します。

4.5.1 サーバの監視コマンド

HA モニタ kit は、サーバの監視コマンドのサンプルファイルを提供しています。サンプルファイルをコピーしてここで説明する内容に従って設定し、必要に応じてカスタマイズしてください。サンプルファイルの格納場所やコピー先については、「3.2.7(1) サーバの起動・停止・監視コマンドの設定」を参照してください。

サーバの監視コマンドのサンプルファイルは、監視処理が記述された patrol_proc() シェル関数をユーザが設定した間隔で実行し、Oracle インスタンスと Oracle リスナーを監視します。

サーバの監視コマンドのサンプルファイルの内容を次の図に示します。

図 4-4 サーバの監視コマンドのサンプルファイルの内容

```
#!/bin/sh
#####
LOGNAME=orasrv
LOGSIZE=1048576
PATROL_INTERVAL=5
#####
patrol_proc() {
    RT_CD=0

    # Instance
    haorainsptl orcl1
    if [ "$?" != "0" ]
    then
        RT_CD=1
    fi

    # Listener
    haoralnsrptl LISTENER
    if [ "$?" != "0" ]
    then
        RT_CD=2
    fi

    return ${RT_CD}
}
#####
:
] これ以降は
  編集できません
```

図中の 1.~4.について説明します。

1. プログラムの監視に必要な動作環境を設定する環境変数です。

ここで設定する内容については、「(1) 環境変数」を参照してください。

2. Oracle インスタンスの監視処理です。

haorainsptl コマンドを実行して、インスタンス識別子で指定した Oracle インスタンスを監視する処理が記述されています。ここで設定する内容については、「(2) 設定方法」を参照してください。

haorainsptl コマンドについては、「4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド」の「haorainsptl (Oracle インスタンスの状態確認)」を参照してください。

3. Oracle リスナーの監視処理です。

haoralnsrptl コマンドを実行して、リスナー名で指定した Oracle リスナーを監視する処理が記述されています。ここで設定する内容については、「(2) 設定方法」を参照してください。

haoralnsrptl コマンドについては、「4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド」の「haoralnsrptl (Oracle リスナーの状態確認)」を参照してください。

4. patrol_proc()シェル関数の処理です。

ユーザ独自の処理を追加したい場合、この patrol_proc()シェル関数の中に追加します。ここで設定する内容については、「(3) サーバの監視コマンドのカスタマイズ」を参照してください。

(1) 環境変数

図 4-4 の 1.の環境変数について説明します。

- LOGNAME

サーバログのファイル名を指定します。起動・停止・監視コマンドの LOGNAME の値は、すべて同じ値を指定してください。また、サーバごとにユニークな値を指定してください。そのため LOGNAME に指定する値は、サーバのサーバ識別名と合わせることをお勧めします。提供するサンプルファイルでは、サーバ識別名として「orasrv」が仮定されています。

サーバログの出力先については、「5.4 サーバログの確認」を参照してください。

- LOGSIZE

サーバログのファイルサイズ (単位: バイト) を整数で指定します。デフォルトでは「1048576」が仮定されています。通常はデフォルトのファイルサイズのまま運用します。出力内容が多く、ログ情報が不足する場合にだけ拡張してください。サーバログのファイルサイズの算出方法については、「5.4.2 ファイルサイズ」を参照してください。

なお、LOGNAME 環境変数に同じ値を指定したサーバでは、サーバの起動・停止・監視コマンドの LOGSIZE 環境変数に同じ値を指定してください。

- PATROL_INTERVAL

監視処理の実行間隔 (監視処理が終了してから次の監視処理を開始するまでの間隔) を、1~3600 (単位: 秒) の範囲で指定します。デフォルトでは「5」が仮定されています。通常はデフォルトの実行間隔から変更する必要はありません。

(2) 設定方法

図 4-4 の 2.と 3.の監視処理の設定方法について説明します。

- 図中の 2. (Oracle インスタンスの監視処理) の設定
 - 提供するサンプルファイルでは、サーバに対応する Oracle インスタンス (インスタンス識別子「orcl1」) を一つ監視する設定になっています。サーバに対応する Oracle インスタンスがない場合は、図中の 2. の記述をコメントアウト、または削除してください。一つのサーバで複数の Oracle インスタンスを監視したい場合は、図中の 2. の記述を追加してください。
 - 提供するサンプルファイルでは、インスタンス識別子として「orcl1」が仮定されています。下線部の引数「orcl1」を、監視したい Oracle インスタンスのインスタンス識別子に変更してください。
- 図中の 3. (Oracle リスナーの監視処理) の設定
 - 提供するサンプルファイルでは、サーバに対応する Oracle リスナー (リスナー名「LISTENER」) を一つ監視する設定になっています。サーバに対応する Oracle リスナーがない場合は、図中の 3. の記述をコメントアウト、または削除してください。一つのサーバで複数の Oracle リスナーを監視したい場合は、図中の 3. の記述を追加してください。
 - 提供するサンプルファイルでは、リスナー名として「LISTENER」が仮定されています。下線部の引数「LISTENER」を、監視したい Oracle リスナーのリスナー名に変更してください。

なお、Oracle インスタンスおよび Oracle リスナーの監視順序を変更したい場合は、ファイル内の監視処理の順番を変更してください。

(3) サーバの監視コマンドのカスタマイズ

サーバの監視コマンドの中に、業務に必要な、Oracle 以外のアプリケーションの処理やユーザ独自の処理などを追加できます。カスタマイズをする場合には、次の注意事項に従ってください。

- 処理を追加する前に、マニュアル「HA モニタ Linux(R)編」の「サーバの監視コマンドの作成」の説明を参照してください。
- 処理は、サーバの監視コマンド内の patrol_proc() シェル関数の中に追加してください。patrol_proc() シェル関数とシェル変数以外は、編集しないでください。
ここに記述した処理は Oracle ユーザの権限で実行されます。権限を変更した場合は、Oracle ユーザの権限に戻してください。
- patrol_proc() シェル関数の中では、exit コマンドを使用しないでください。処理を終了させる場合は、return コマンドを使用してください。
- patrol_proc() シェル関数のリターンコードが 0 以外の場合、サーバの監視コマンドは終了し、HA モニタがサーバ障害を検出します。この際、patrol_proc() シェル関数のリターンコードがサーバの監視コマンドの戻り値となります。
- patrol_proc() シェル関数の中では、無限ループなどの終了しない処理を作り込まないでください。
- patrol_proc() シェル関数が出力するメッセージの出力先については、「[5.4 サーバログの確認](#)」を参照してください。
- patrol_proc() シェル関数の中で実行するコマンドや入出力するファイルは、絶対パスで指定してください。ここでいうコマンドとは、HA モニタ kit が提供するコマンド以外のものを指します。

- 次のシェル変数名を変更しないでください。
 - LOGNAME
 - LOGSIZE
 - PATROL_INTERVAL
- シェル変数を追加する場合は、コマンド内を検索し、すでに使用されていないことを確認してから追加してください。

4.5.2 サーバの監視コマンド実行シェル

サーバの監視コマンド実行シェルは、サーバの監視コマンドを実行するために HA モニタ kit がサンプルファイルを提供しているシェルです。サンプルファイルをコピーしてここで説明する内容に従って設定してください。サンプルファイルの格納場所やコピー先については、「3.2.7(1) サーバの起動・停止・監視コマンドの設定」を参照してください。

サーバの監視コマンド実行シェルのサンプルファイルの内容を次の図に示します。

図 4-5 サーバの監視コマンド実行シェルのサンプルファイルの内容

```

#!/bin/sh
#####
ORACLE_USER=oracle
PATROL_SHELL=/opt/hitachi/HMon/etc/patrolcommand } 1.
#####
: } これ以降は
} 編集できません

```

図中の 1. について説明します。

1. プログラムの監視に必要な動作環境を設定する環境変数です。
ここで設定する内容については、「(1) 環境変数」を参照してください。

(1) 環境変数

図 4-5 の 1. の環境変数について説明します。

- ORACLE_USER
Oracle ユーザのアカウント名を指定します。
- PATROL_SHELL
「4.5.1 サーバの監視コマンド」で設定した、サーバの監視コマンドを絶対パスで指定します。

(2) 注意事項

- 次のシェル変数名を変更しないでください。
 - ORACLE_USER

- PATROL_SHELL

4.6 サーバの起動・停止・監視コマンドの設定に使用するコマンド

サーバの起動・停止・監視コマンドの設定に使用するコマンドの一覧を次の表に示します。

表 4-1 サーバの起動・停止・監視コマンドの設定に使用するコマンドの一覧

コマンド名	機能	関連するサブコマンド※
haorainsbgn	Oracle インスタンスを起動します。	haoraenvcheck
haorainsend	Oracle インスタンスを停止します。	haoraenvcheck haorainsendsub
haorainsptl	Oracle インスタンスの状態を確認します。	haoraenvcheck haorainsptlsub
haoralsnrbgn	Oracle リスナーを起動します。	haoraenvcheck
haoralsnrend	Oracle リスナーを停止します。	haoraenvcheck haoralsnrendsub
haoralsnrptl	Oracle リスナーの状態を確認します。	haoraenvcheck haoralsnrptlsub

注※

サブコマンドは、コマンド名に記載しているコマンドから呼び出されます。そのため、サーバの起動・停止・監視コマンドに設定する必要はありません。

コマンドの説明で使用する見出し

コマンドの説明で使用する、各見出しについて説明します。なお、各コマンドの説明では、次の項目のうち必要な項目についてだけ説明しています。

形式

コマンドの記述形式について説明しています。

実行できるユーザ

コマンドを実行する権限を持つユーザを示します。

機能

コマンドの機能について説明します。

オプション

オプションの指定内容や、オプションを指定するときの注意事項などを説明します。

戻り値

コマンドの戻り値について説明します。

注意事項

コマンドを実行するときの注意事項を説明します。

文法記述記号

コマンドの説明で使用する文法記述記号を説明します。各文法記述記号の意味を、次の表に示します。

表 4-2 コマンドの文法記述記号一覧

文法記述記号	意味
{ }	この記号で囲まれている項目の中から、一つを選択して指定することを示します。複数の項目が記述されている場合に、そのうちの一つを選択して指定します。 (例) haorainsendsub インスタンス識別子 { -n -a } haorainsendsub サブコマンドでは、-n または -a のどちらかを選択して指定することを示します。
	{ } で囲まれている複数の項目間の区切りを示します。 (例) haoralsnrendsub リスナー名 { -n -a } haoralsnrendsub サブコマンドでは、-n または -a のどちらかを選択して指定することを示します。

haorainsbgn (Oracle インスタンスの起動)

形式

```
haorainsbgn インスタンス識別子
```

実行できるユーザ

スーパーユーザ、Oracle ユーザ

機能

HA モニタ kit で監視する Oracle インスタンスを起動します。このコマンドは、サーバの起動コマンドの中でだけ使用できます。サーバの起動コマンドの記述内容については、「[4.2 サーバの起動コマンドの設定](#)」を参照してください。

このコマンドで起動できるのは、HA モニタ kit で監視するための設定が完了した Oracle インスタンスだけです。

オプション

- インスタンス識別子

Oracle インスタンス制御用定義ファイルの ORACLE_SID に定義済みの、起動したい Oracle インスタンスのインスタンス識別子を指定します。

戻り値

値	意味
0	正常終了 (Oracle インスタンスの起動に成功しました)
0 以外	異常終了

haorainsend (Oracle インスタンスの停止)

形式

```
haorainsend インスタンス識別子
```

実行できるユーザ

スーパーユーザ, Oracle ユーザ

機能

HA モニタ kit で監視中の Oracle インスタンスを停止します。このコマンドは、サーバの停止コマンドの中でだけ使用できます。サーバの停止コマンドの記述内容については、「[4.3 サーバの停止コマンドの設定](#)」を参照してください。

このコマンドで停止できるのは、HA モニタ kit で監視中の Oracle インスタンスだけです。

Oracle インスタンスを正常に停止できない場合は、Oracle インスタンス制御用定義ファイルの ABORT_LIMIT に設定した待ち時間の経過後に、Oracle インスタンスを強制停止します。

オプション

- インスタンス識別子

Oracle インスタンス制御用定義ファイルの ORACLE_SID に定義済みの、停止したい Oracle インスタンスのインスタンス識別子を指定します。

戻り値

値	意味
0	正常終了 (Oracle インスタンスの停止に成功しました)

値	意味
0 以外	異常終了

注意事項

Oracle インスタンスを強制停止できない場合、Oracle インスタンス制御用定義ファイルの ABORT_LIMIT で設定した待ち時間の経過後に強制停止の処理を中止します。このコマンドの実行から終了までに、最大で設定した待ち時間の 2 倍の時間が掛かることがあります。

haorainsptl (Oracle インスタンスの状態確認)

形式

```
haorainsptl インスタンス識別子
```

実行できるユーザ

スーパーユーザ, Oracle ユーザ

機能

Oracle インスタンスのプロセスの存在を確認します。Oracle インスタンス制御用定義ファイルの DB_NOACCESS を省略するか、no を指定した場合は、加えて Oracle インスタンスの状態やスローダウンを確認します。

このコマンドは、サーバの監視コマンドの中でだけ使用できます。サーバの監視コマンドの記述内容については、「4.4 サーバの監視コマンドの設定 (ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合)」または「4.5 サーバの監視コマンドの設定 (patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合)」を参照してください。

このコマンドで状態を確認できるのは、HA モニタ kit で監視するための設定が完了した Oracle インスタンスだけです。

オプション

- インスタンス識別子

Oracle インスタンス制御用定義ファイルの ORACLE_SID に定義済みの、状態を確認したい Oracle インスタンスのインスタンス識別子を指定します。

戻り値

値	意味
0	正常終了 (確認の結果、Oracle インスタンスの異常を検知しませんでした)

値	意味
1	正常終了（確認の結果、Oracle インスタンスの異常を検知しました）
125	正常終了（確認の結果、Oracle インスタンスのスローダウンを検知しました。この結果は、HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドでサーバの監視コマンドを指定している場合に出力されます）
上記以外	異常終了（確認処理中に異常が発生、または確認処理が異常終了しました）

haorainsendsub (Oracle インスタンスの停止)

形式

```
haorainsendsub インスタンス識別子 { -n | -a }
```

実行できるユーザ

スーパーユーザ、Oracle ユーザ

機能

HA モニタ kit で監視中の Oracle インスタンスを停止します。このコマンドは、haorainsend (Oracle インスタンスの停止) から呼び出されるサブコマンドです。

オプション

- インスタンス識別子
Oracle インスタンス制御用定義ファイルの ORACLE_SID に定義済みの、停止したい Oracle インスタンスのインスタンス識別子を指定します。
- -n
Oracle インスタンスを正常停止させます。
- -a
Oracle インスタンスを強制停止させます。

戻り値

値	意味
0	正常終了 (Oracle インスタンスの停止に成功しました)
0 以外	異常終了

haorainsptlsub (Oracle インスタンスの状態確認)

形式

```
haorainsptlsub インスタンス識別子 { -slowdown | -status }
```

実行できるユーザ

スーパーユーザ, Oracle ユーザ

機能

Oracle インスタンスの状態, およびスローダウンを確認するための SQL を実行します。このコマンドは, haorainsptl (Oracle インスタンスの状態確認) から呼び出されるサブコマンドです。

オプション

- インスタンス識別子
Oracle インスタンス制御用定義ファイルの ORACLE_SID に定義済みの, 状態を確認したい Oracle インスタンスのインスタンス識別子を指定します。
- -slowdown
スローダウン監視用の SQL を実行します。
- -status
状態確認用の SQL を実行します。

戻り値

値	意味
0	正常終了 (Oracle インスタンスの状態確認に成功しました)
0 以外	異常終了

haoralnrbgn (Oracle リスナーの起動)

形式

```
haoralnrbgn リスナー名
```

実行できるユーザ

スーパーユーザ, Oracle ユーザ

機能

HA モニタ kit で監視する Oracle リスナーを起動します。このコマンドは、サーバの起動コマンドの中でだけ使用できます。サーバの起動コマンドの記述内容については、「[4.2 サーバの起動コマンドの設定](#)」を参照してください。

このコマンドで起動できるのは、HA モニタ kit で監視するための設定が完了した Oracle リスナーだけです。

オプション

- リスナー名

Oracle リスナー制御用定義ファイルの LISTENER_NAME に定義済みの、起動したい Oracle リスナーのリスナー名を指定します。

戻り値

値	意味
0	正常終了 (Oracle リスナーの起動に成功しました)
0 以外	異常終了

haoralsnrend (Oracle リスナーの停止)

形式

```
haoralsnrend リスナー名
```

実行できるユーザ

スーパーユーザ, Oracle ユーザ

機能

HA モニタ kit で監視中の Oracle リスナーを停止します。このコマンドは、サーバの停止コマンドの中でだけ使用できます。サーバの停止コマンドの記述内容については、「[4.3 サーバの停止コマンドの設定](#)」を参照してください。

このコマンドで停止できるのは、HA モニタ kit で監視中の Oracle リスナーだけです。

オプション

- リスナー名

Oracle リスナー制御用定義ファイルの LISTENER_NAME に定義済みの、停止したい Oracle リスナーのリスナー名を指定します。

戻り値

値	意味
0	正常終了 (Oracle リスナーの停止に成功しました)
0 以外	異常終了

haoralsnrptl (Oracle リスナーの状態確認)

形式

```
haoralsnrptl リスナー名
```

実行できるユーザ

スーパーユーザ, Oracle ユーザ

機能

Oracle リスナーのプロセスの存在を確認します。Oracle リスナー制御用定義ファイルの LISTENER_NOACCESS を省略するか、no を指定した場合は、加えてスローダウンを確認します。

このコマンドは、サーバの監視コマンドの中でだけ使用できます。サーバの監視コマンドの記述内容については、「[4.4 サーバの監視コマンドの設定 \(ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合\)](#)」または「[4.5 サーバの監視コマンドの設定 \(patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合\)](#)」を参照してください。

このコマンドで状態を確認できるのは、HA モニタ kit で監視するための設定が完了した Oracle リスナーだけです。

オプション

- リスナー名

Oracle リスナー制御用定義ファイルの LISTENER_NAME に定義済みの、状態を確認したい Oracle リスナーのリスナー名を指定します。

戻り値

値	意味
0	正常終了 (確認の結果, Oracle リスナーの異常を検知しませんでした)
1	正常終了 (確認の結果, Oracle リスナーの異常を検知しました)

値	意味
125	正常終了（確認の結果、Oracle リスナーのスローダウンを検知しました。この結果は、HA モニタのサーバ対応の環境設定の <code>ptlcmd_ex</code> オペランドでサーバの監視コマンドを指定している場合に出力されます）
上記以外	異常終了（確認処理中に異常が発生、または確認処理が異常終了しました）

haoralsnrendsub（Oracle リスナーの停止）

形式

```
haoralsnrendsub リスナー名 { -n | -a }
```

実行できるユーザ

スーパーユーザ、Oracle ユーザ

機能

HA モニタ kit で監視中の Oracle リスナーを停止します。このコマンドは、`haoralsnrend`（Oracle リスナーの停止）から呼び出されるサブコマンドです。

オプション

- リスナー名
Oracle リスナー制御用定義ファイルの `LISTENER_NAME` に定義済みの、停止したい Oracle リスナーのリスナー名を指定します。
- n
Oracle リスナーを正常停止させます。
- a
Oracle リスナーを強制停止させます。

戻り値

値	意味
0	正常終了（Oracle リスナーの停止に成功しました）
0 以外	異常終了

haoralsnrptlsub (Oracle リスナーの状態確認)

形式

```
haoralsnrptlsub リスナー名
```

実行できるユーザ

スーパーユーザ, Oracle ユーザ

機能

Oracle リスナーのスローダウンを確認するための SQL を実行します。このコマンドは、haoralsnrptl (Oracle リスナーの状態確認) から呼び出されるサブコマンドです。

オプション

- リスナー名

Oracle リスナー制御用定義ファイルの LISTENER_NAME に定義済みの、状態を確認したい Oracle リスナーのリスナー名を指定します。

戻り値

値	意味
0	正常終了 (Oracle リスナーの状態確認に成功しました)
0 以外	異常終了

haoraenvcheck (制御用定義ファイルのチェック)

形式

```
haoraenvcheck 制御用定義ファイル名 { -instance | -listener }
```

実行できるユーザ

スーパーユーザ, Oracle ユーザ

機能

制御用定義ファイルの定義内容が規則に合っているかどうかをチェックするサブコマンドです。次のコマンドの中でだけ使用できます。

- haorainsbgn

- haoralsnrbgn
- haorainsend
- haoralsnrend
- haorainsptl
- haoralsnrptl

オプション

- 制御用定義ファイル名
制御用定義ファイル名を指定します。
- -instance
Oracle インスタンス制御用定義ファイルの定義をチェックをする場合に指定します。
- -listener
Oracle リスナー制御用定義ファイルの定義をチェックをする場合に指定します。

戻り値

値	意味
0	正常終了（制御用定義ファイルの定義チェックに成功しました）
0 以外	異常終了

5

システムの運用

この章では、構築したシステムの運用に関する内容のうち、Oracle インスタンス・Oracle リスナーの起動・停止方法、サーバログの確認、障害が発生した場合のオペレータの操作、およびシステムを変更する方法などについて説明します。

このマニュアルに記載がないシステムの運用に関する内容については、マニュアル「HA モニタ Linux(R)編」の「システムの運用」を参照してください。

5.1 運用の流れ

HA モニタ kit を使用した系切り替え構成の Oracle インスタンスおよび Oracle リスナーは、HA モニタのモニタモードのサーバとして運用します。HA モニタ kit を使用した系切り替え構成での運用の流れは、HA モニタを使用した一般的な系切り替え構成の場合と同じです。構築したシステムの運用の流れについては、マニュアル「HA モニタ Linux(R)編」の「運用の流れ」の説明を参照してください。

注意事項

- Oracle インスタンス制御用定義ファイルの DB_NOACCESS を省略しているか、または no を指定している Oracle インスタンスでは、稼働中に SQL の ALTER SYSTEM SUSPEND 文を実行してデータベースの I/O を停止しないでください。稼働中に ALTER SYSTEM SUSPEND 文を実行すると、HA モニタ kit が Oracle インスタンスの障害を検出するおそれがあります。
- サンプルファイルの停止コマンド・監視コマンド、およびそれらをコピー・編集して作成したコマンドを、手動で実行しないでください。/opt/hitachi/HAMon/spool/配下のログファイル (oralog) が一般ユーザの書き込み権限なしで作成されてしまうおそれがあります。その場合、サーバ起動時にサーバの監視コマンドがそのログファイルにアクセスできないため、異常を検知します。もしそれらのコマンドを手動で実行してしまった場合は、ログファイルに一般ユーザの書き込み権限を付与してください。
- HA モニタ kit が提供しているコマンド内では、ヒアドキュメントを利用しているため、/tmp ディレクトリに一時ファイルが出力されます。そのため、もし/tmp 領域に空きがないと、一時ファイルの出力エラーとなります。この場合、サーバの監視コマンドが異常を検知して、HA モニタはサーバの系切り替えをします。これを防ぐために、サーバ稼働中は/tmp 領域が満杯にならない運用としてください。

5.2 Oracle インスタンス・Oracle リスナーの起動・停止・監視

ここでは、HA モニタ kit を使用したシステムの Oracle インスタンスまたは Oracle リスナーの起動・停止・監視について説明します。Oracle インスタンスおよび Oracle リスナーは、モニタモードのサーバとして管理されます。このため、Oracle インスタンスまたは Oracle リスナーの起動・停止・監視は、HA モニタのコマンドで実施します。

なお、HA モニタのサーバとして稼働中の Oracle インスタンスまたは Oracle リスナーは、必ず HA モニタのコマンドで起動・停止してください。HA モニタ kit を使用した系切り替え構成の Oracle インスタンスまたは Oracle リスナーを HA モニタのコマンド以外で起動・停止すると、HA モニタが障害を検出してしまう場合があります。

5.2.1 Oracle インスタンスまたは Oracle リスナーの起動

Oracle インスタンスまたは Oracle リスナーを起動するには、モニタモードのサーバ起動コマンド (monbegin コマンド) を実行します。モニタモードのサーバ起動コマンド (monbegin コマンド) は、HA モニタの運用コマンドです。なお、CDB (シングルテナント) 構成の場合、monbegin コマンドで Oracle インスタンスを起動すると、PDB の自動起動設定に関係なく、PDB も自動で起動します。

このコマンドについては、マニュアル「HA モニタ Linux(R)編」の「monbegin (モニタモードのサーバの起動)」の説明を参照してください。

5.2.2 Oracle インスタンスまたは Oracle リスナーの停止

Oracle インスタンスまたは Oracle リスナーを停止するには、モニタモードのサーバ停止コマンド (monend コマンド) を実行します。モニタモードのサーバ停止コマンド (monend コマンド) は、HA モニタの運用コマンドです。HA モニタ kit は、サーバの停止処理中に Oracle インスタンスまたは Oracle リスナーの停止に失敗した場合、またはユーザが指定した時間を経過しても Oracle インスタンスまたは Oracle リスナーが停止しない場合、Oracle インスタンスまたは Oracle リスナーを強制停止します。

このコマンドについては、マニュアル「HA モニタ Linux(R)編」の「monend (モニタモードのサーバの停止)」の説明を参照してください。

5.2.3 Oracle インスタンスまたは Oracle リスナーの監視

Oracle インスタンスまたは Oracle リスナーの起動が完了すると、HA モニタはサーバの監視コマンドを実行し、サーバを監視します。

サーバの監視コマンドでは、次の方法でサーバを監視しています。

サーバが Oracle インスタンスの場合

- Oracle インスタンスのプロセスの生存監視
- SQL による Oracle インスタンスの状態問い合わせ
- Oracle インスタンスのスローダウン監視

サーバが Oracle リスナーの場合

- Oracle リスナーのプロセスの生存監視
- リスナー制御ユーティリティを使用した、応答の有無によるスローダウン監視

サーバの監視コマンドの概要や動作の詳細については、マニュアル「HA モニタ Linux(R)編」の「モニタモードのサーバの監視」、および「サーバの監視コマンドの制御」の説明を参照してください。

サーバ (Oracle インスタンスおよび Oracle リスナー) の監視の一時停止・再開をする場合

次の条件を両方とも満たす場合は、HA モニタ・サーバ稼働中の設定変更コマンド (monchange コマンド) で、サーバの監視の一時停止・再開ができます。

- HA モニタのバージョンが 01-69 以降
- HA モニタのサーバ対応の環境設定の ptrlcmd_ex オペランドで監視コマンドを指定している

サーバの監視を一時停止する手順

HA モニタ・サーバ稼働中の設定変更コマンド (monchange コマンド) で、サーバの監視を一時停止します。なお、サーバの監視を一時停止している間に、系切り替えをしないようにするためには、あらかじめ、待機サーバ停止コマンド (monsbystp コマンド) を実行して待機サーバを停止してください。

サーバの監視を再開する手順

HA モニタ・サーバ稼働中の設定変更コマンド (monchange コマンド) で、サーバの監視を再開します。監視再開後、系切り替え可能状態でなければ、待機系で、モニタモードのサーバの起動コマンド (monbegin コマンド) を実行して系切り替え可能状態にしてください。

5.3 Oracle インスタンス・Oracle リスナーの HA モニタを介さない起動・停止

HA モニタ kit はデータベースをマウントし、オープンした状態で Oracle インスタンスを起動します。このため、Oracle インスタンスに対して次に示す運用をする場合は、HA モニタを介さないで起動・停止する必要があります。

- データベースをマウントしないで Oracle インスタンスを起動する場合
- データベースをオープンしないで Oracle インスタンスを起動する場合
- Oracle インスタンスへのアクセスを制限して起動する場合
- Oracle インスタンスを UPGRADE モードとして起動する場合
- Oracle インスタンスを強制的に起動する場合
- Oracle インスタンスを起動してメディアリカバリをする場合
- Oracle Data Guard を使用したディザスタリカバリ構成で、スイッチオーバーを実施する場合
- Oracle Data Guard を使用したディザスタリカバリ構成で、プライマリ・データベースのスイッチオーバー後にスタンバイ・データベースとして運用する場合

ここでは、Oracle インスタンスと Oracle リスナーを、HA モニタを介さないで起動・停止する方法を説明します。

5.3.1 Oracle インスタンス・Oracle リスナーを HA モニタを介さないで起動する

HA モニタを介さないで Oracle インスタンスまたは Oracle リスナーを起動する場合、HA モニタおよび HA モニタ kit がサーバの起動時に実施しているすべての制御をユーザが行います。

HA モニタを介さないでサーバを起動した場合、HA モニタによる監視（系切り替え）はできません。また、HA モニタを介さないで起動した Oracle インスタンスおよび Oracle リスナーの稼働中は、それらに対応するサーバを起動しないでください。

HA モニタを介さないで Oracle インスタンスまたは Oracle リスナーを起動する手順を、次に示します。

1. Oracle インスタンスまたは Oracle リスナーに対応するサーバが起動している場合は、HA モニタのサーバの停止コマンド（monend コマンド）でサーバを停止する。
2. Oracle インスタンスまたは Oracle リスナーが使用するリソース（共有ディスクやクライアント接続用の IP アドレス）を手動で接続する。
3. Oracle インスタンスまたは Oracle リスナーを起動する。
起動方法、および起動に必要なオプションについては、Oracle のマニュアルを参照してください。

5.3.2 Oracle インスタンス・Oracle リスナーを HA モニタを介さないで停止する

HA モニタを介さないで起動した Oracle インスタンスまたは Oracle リスナーを停止する場合、HA モニタおよび HA モニタ kit がサーバの停止時に実施しているすべての制御をユーザが行います。

HA モニタを介さないで Oracle インスタンスまたは Oracle リスナーを停止する手順を、次に示します。

1. Oracle インスタンスまたは Oracle リスナーを停止する。

停止方法、および停止に必要なオプションについては、Oracle のマニュアルを参照してください。

2. Oracle インスタンスまたは Oracle リスナーが使用するリソース（共有ディスクやクライアント接続用の IP アドレス）を手動で切断する。

切断しないと、HA モニタを介した運用に戻したときに、現用・待機の両系から共有ディスクが接続されデータベースを破壊するなどのおそれがあります。必ず切断してください。

5.4 サーバログの確認

HA モニタ kit は、サーバの起動・停止・監視コマンドが出力するメッセージをサーバログに出力します。サーバの起動・停止・監視コマンドの中で実行する Oracle のコマンドなどの標準出力・標準エラー出力も、サーバログに出力されます。サーバログを確認することで、サーバの起動・停止・監視コマンドの処理の結果を調査できます。システムの運用中は、監視処理の結果によって、サーバログの内容は常に更新されます。

注意事項

サーバの稼働中にサーバログを確認する場合は参照だけとし、保存しないでください。参照中に出力されたログ情報が、失われるおそれがあります。

ここでは、サーバログの詳細について説明します。

5.4.1 ファイルの種類

サーバログの一覧を、次の表に示します。ファイル名の「サーバ識別名」の部分には、HA モニタ kit が提供するサーバの起動・停止・監視コマンドの、LOGNAME 環境変数の値が適用されます。

表 5-1 サーバログの一覧

ファイル名	説明
/opt/hitachi/HAMon/spool/サーバ識別名.oralog	サーバログのファイルです。サーバの起動・停止・監視コマンドが出力するメッセージの内容が出力されます。
/opt/hitachi/HAMon/spool/サーバ識別名.oralog_old	1 世代前のサーバログのファイルです。次に示すタイミングでサーバログのファイルが指定サイズを超えた場合に、サーバログのファイル名の末尾に"_old"を付けてリネームされ、作成されます。 <ul style="list-style-type: none">サーバの起動・停止・監視コマンドの実行時サーバの監視コマンド内で監視処理を実行する直前
/opt/hitachi/HAMon/spool/サーバ識別名.oralog_err	直近のエラー発生時のサーバログのファイルです。サーバの起動・停止・監視コマンドが 0 以外の戻り値で終了する場合に、サーバログのファイル名の末尾に"_err"を付けてリネームされ、作成されます。障害調査をする場合は、このファイルを参照してください。

5.4.2 ファイルサイズ

サーバログのファイルサイズは、サーバの起動・停止・監視コマンドの LOGSIZE 環境変数で変更できます。デフォルトでは、1,048,576 バイト（1 メガバイト）です。通常はデフォルトのファイルサイズのまま運用します。出力内容が多く、ログ情報が不足する場合にだけ拡張してください。

ファイルサイズを変更する場合は、次の計算式 1～計算式 3 の結果のうち、最も大きな値よりも大きい値に設定してください。計算式の単位はバイトです。

計算式 1

240+ (サーバ内のOracleインスタンスおよびOracleリスナーの数) ×160
+ (OracleなどHAモニタ kit以外の製品がサーバの起動処理で出力するメッセージの合計サイズ)

計算式 2

60+ (サーバ内のOracleインスタンスおよびOracleリスナーの数) ×170
+ (OracleなどHAモニタ kit以外の製品が1回の監視処理で出力するメッセージの合計サイズ)

計算式 3

240+ (サーバ内のOracleインスタンスおよびOracleリスナーの数) ×160
+ (OracleなどHAモニタ kit以外の製品がサーバの停止処理で出力するメッセージの合計サイズ)

サーバの起動・停止・監視コマンドの LOGSIZE 環境変数については、「4. サーバの起動・停止・監視コマンドの設定」を参照してください。

なお、一定期間分のログを保持するように見積もりたい場合や、ファイルが保持できるログの合計サイズを計算したい場合は、サーバの起動から停止までに出力されるメッセージの合計サイズを、次に示す計算式で求めてください。

なお、次の計算式中の「計算式 1」「計算式 2」「計算式 3」は、サーバログのファイルサイズの見積もりに使用した計算式の結果のことです。

計算式1+計算式2×(サーバの稼働時間(単位:秒)÷サーバの監視コマンドの監視処理の間隔(単位:秒))+計算式3

5.4.3 出力例

サーバログの出力例を次の図に示します。ここで示す例は、Oracle インスタンスの起動に失敗した場合の出力内容です。

```
2011/07/07 11:24:15 KAMK00401-I ***** The actcommand start. *****
2011/07/07 11:24:15 KAMK00301-I Oracle instance orcl1 will now start.
SQL*Plus: Release 10.2.0.1.0 - Production on 木 7月 7 11:24:15 2011

Copyright (c) 1982, 2005, Oracle. All rights reserved.

SQL> アイドル・インスタンスに接続しました。
SQL> ORA-01081: cannot start already-running ORACLE - shut it down first
SQL> 切断しました。

2011/07/07 11:24:16 KAMK00303-E An attempt to start Oracle instance orcl1 has failed.
2011/07/07 11:24:16 KAMK00402-I ***** The actcommand end. (1) *****
```

} 1.
} 2.
} 1.

次に、図中の番号について説明します。

1. HA モニタ kit のコマンドが出力するメッセージです。メッセージの出力日時や ID などが出力されます。出力例では、メッセージ KAMK00401-I などが出力されています。HA モニタ kit が出力するメッセージの詳細については、「6.2 メッセージの詳細」を参照してください。
2. HA モニタ kit が Oracle の操作をした結果、Oracle が出力したメッセージや、ユーザがサーバの起動・停止・監視コマンドに追加した処理によって出力されたメッセージです。この行に出力されるメッセージについては、Oracle のマニュアルなどを参照してください。

5.4.4 メッセージの言語環境

サーバログに出力される Oracle のメッセージは、HA モニタの言語環境と対応した言語で出力されます。メッセージの言語環境の対応を、次の表に示します。

表 5-2 メッセージの言語環境の対応

HA モニタの言語環境 (LANG)	Oracle の言語環境 (NLS_LANG)	メッセージの言語
ja_JP.UTF-8	JAPANESE_JAPAN.AL32UTF8	日本語 (UTF-8)
ja_JP.eucJP	JAPANESE_JAPAN.JA16EUC	日本語 (EUC)
ja_JP.SJIS	JAPANESE_JAPAN.JA16SJIS	日本語 (S-JIS)
その他	設定されない	英語

5.5 障害発生による系切り替え時の運用

ここでは、HA モニタ kit を使用したシステムで障害が発生した場合に収集する障害情報について説明します。障害を検出したあとのオペレータの対処については、マニュアル「HA モニタ Linux(R)編」の「障害発生による系切り替え時の運用」の説明を参照してください。

5.5.1 障害情報を収集する

障害が発生した場合、HA モニタ kit の障害情報を収集します。HA モニタ kit の障害情報は、HA モニタのトラブルシューティング情報収集コマンド (monts コマンド) で HA モニタの障害情報と一緒に収集されます。HA モニタの障害情報の収集については、マニュアル「HA モニタ Linux(R)編」の「障害情報を収集する」の説明を参照してください。

HA モニタ kit には、次の障害情報があります。

- サーバログ
詳細については、「[5.4 サーバログの確認](#)」を参照してください。
- 制御用定義ファイル
詳細については、「[3.2.7\(2\) 制御用定義ファイルの設定](#)」を参照してください。
- サーバの起動・停止・監視コマンド
/opt/hitachi/HAmom/etc に、サーバの起動・停止・監視コマンドがある場合にだけ収集されます。詳細については、「[3.2.7\(1\) サーバの起動・停止・監視コマンドの設定](#)」を参照してください。

5.6 障害への対処

障害の発生時に出力されるメッセージを基に、オペレータは障害に対処してください。また、必要に応じて、HA モニタ kit の障害情報を収集してください。HA モニタ kit の障害情報については、「[5.5.1 障害情報を収集する](#)」を参照してください。

HA モニタ kit では、次の障害が発生した場合は、サーバログの内容を確認して対処します。サーバログには、サーバの起動・停止・監視コマンドが出力するメッセージの内容が出力されます。

- Oracle インスタンスの起動失敗
- Oracle インスタンスの再起動失敗
- Oracle インスタンスの停止失敗
- Oracle インスタンスの障害
- Oracle リスナーの起動失敗
- Oracle リスナーの再起動失敗
- Oracle リスナーの停止失敗
- Oracle リスナーの障害

サーバログについては「[5.4 サーバログの確認](#)」を、メッセージの対処については「[6. メッセージ](#)」またはマニュアル「[HA モニタ メッセージ](#)」を参照してください。また、HA モニタの系切り替えシステムで発生する代表的な障害の例については、マニュアル「[HA モニタ Linux\(R\)編](#)」の「[障害への対処](#)」の説明を参照してください。

5.7 システムの変更

ここでは、システムを変更したときに必要な設定、および注意事項について説明します。

5.7.1 Oracle をアップグレードする

Oracle をアップグレードする場合、サーバを設定し直す必要があります。Oracle をアップグレードするときに必要なサーバの設定手順について、次に示します。

1. モニタモードのサーバの停止コマンド (monend コマンド) を実行して、サーバを停止する。
2. Oracle が使用する共有ディスクを、手動で参照+更新接続する。
共有ディスクの接続方法については、OS のマニュアルを参照してください。
3. Oracle をアップグレードする。
アップグレードの方法については、Oracle のマニュアルを参照してください。
4. Oracle インスタンス・Oracle リスナーの制御用定義ファイルを修正する。
Oracle をアップグレードすると、Oracle ホームディレクトリが変更されます。制御用定義ファイルの ORACLE_HOME を、アップグレード後の値に修正してください。
5. 共有ディスクを手動で切り離す。
共有ディスクの切り離し方法については、OS のマニュアルを参照してください。
6. モニタモードのサーバの起動コマンド (monbegin コマンド) でサーバを起動する。

5.7.2 Oracle インスタンス・Oracle リスナーを追加する

ここでは、Oracle インスタンスまたは Oracle リスナーを、HA モニタ kit の制御対象に追加する方法について、構築済みのサーバに追加する場合と新しいサーバとして追加する場合とに分けて説明します。

(1) 構築済みのサーバに追加する

構築済みのサーバに、Oracle インスタンスまたは Oracle リスナーを追加する手順を、次に示します。

1. モニタモードのサーバの停止コマンド (monend コマンド) を実行して、サーバを停止する。
Oracle インスタンスまたは Oracle リスナーを追加したいサーバを停止しないまま、サーバの停止コマンドに設定を追加して設定内容を誤ると、Oracle インスタンスおよび Oracle リスナーを停止できなくなるおそれがあります。設定の追加は、必ずサーバを停止してから実施してください。
2. データベースを作成する。
追加するデータベースを、Oracle のマニュアルを参照して作成してください。

3. 追加する Oracle インスタンス・Oracle リスナーの制御用定義ファイルを作成する。

制御用定義ファイルについては、「[3.2.7\(2\) 制御用定義ファイルの設定](#)」を参照してください。

4. サーバの起動・停止・監視コマンドに設定を追加する。

追加したい Oracle インスタンスまたは Oracle リスナー用の処理を、サーバの起動・停止・監視コマンドに追加します。サーバの起動・停止・監視コマンドの設定については、「[4. サーバの起動・停止・監視コマンドの設定](#)」を参照してください。

5. モニタモードのサーバの起動コマンド (monbegin コマンド) でサーバを起動する。

(2) 新しいサーバとして追加する

新しいサーバとして Oracle インスタンスまたは Oracle リスナーを追加する方法は、「[3.2 システムの構築](#)」を参照してください。Oracle インスタンスおよび Oracle リスナーを設定するサーバの起動・停止・監視コマンドについては、「[4. サーバの起動・停止・監視コマンドの設定](#)」を参照してください。

5.7.3 Oracle インスタンス・Oracle リスナーを削除する

Oracle インスタンスまたは Oracle リスナーを、HA モニタ kit の制御対象から削除する手順を、次に示します。

1. モニタモードのサーバの停止コマンド (monend コマンド) を実行して、サーバを停止する。

Oracle インスタンスまたは Oracle リスナーを削除したいサーバを停止しないまま、誤ってサーバの停止コマンドの設定を削除すると、Oracle インスタンスおよび Oracle リスナーを停止できなくなるおそれがあります。設定の削除は、必ずサーバを停止してから実施してください。

2. サーバの起動・停止・監視コマンドから設定を削除する。

削除したい Oracle インスタンスまたは Oracle リスナー用の処理を、サーバの起動・停止・監視コマンドから削除、またはコメントアウトします。サーバの起動・停止・監視コマンドの設定については、「[4. サーバの起動・停止・監視コマンドの設定](#)」を参照してください。

3. Oracle インスタンス・Oracle リスナーの制御用定義ファイルを削除する (任意)。

定義ファイルの削除は任意です。データベースを削除する場合は削除します。データベースを残す場合は将来再利用することを目的として、定義ファイルを残しておくことができます。

4. モニタモードのサーバの起動コマンド (monbegin コマンド) でサーバを起動する。

5.7.4 Oracle インスタンス・Oracle リスナーに関する設定を変更する

構築済みの Oracle インスタンスまたは Oracle リスナーに関するサーバの起動・停止・監視コマンド、および制御用定義ファイルを変更する方法を、次に示します。

1. モニタモードのサーバの停止コマンド (monend コマンド) を実行して、サーバを停止する。

サーバを停止しないまま設定を変更すると、設定誤りによって Oracle インスタンスおよび Oracle リスナーを停止できなくなったり、設定内容と動作中のサーバの監視コマンドの動作が一致しなくなり、サーバの監視コマンドの状態を把握できなくなったりするおそれがあります。設定の変更は、必ずサーバを停止してから実施してください。

2. サーバの起動・停止・監視コマンドや制御用定義ファイルなどの設定を変更する。

サーバの起動・停止・監視コマンドについては、「4. サーバの起動・停止・監視コマンドの設定」を参照してください。

制御用定義ファイルについては、「3.2.7(2) 制御用定義ファイルの設定」を参照してください。

3. モニタモードのサーバの起動コマンド (monbegin コマンド) でサーバを起動する。

5.7.5 CDB (シングルテナント) 構成で PDB を unplug および plug する

HA モニタ kit が Oracle を監視中に PDB を unplug すると、Oracle のスローダウンを検知します。

PDB を unplug する場合は、次のどちらかの運用をしてください。

- 実行サーバを停止したあと、HA モニタを介さないで Oracle を起動してから、PDB を unplug する。
- HA モニタのコマンドで Oracle の監視を一時停止してから、PDB を unplug する。

それぞれの運用で PDB を unplug および plug する手順を次に示します。

(1) 実行サーバを停止したあと、HA モニタを介さないで Oracle を起動してから、PDB を unplug および plug する

1. モニタモードのサーバの停止コマンド (monend コマンド) を実行して、実行サーバを停止する。

このコマンドについては、マニュアル「HA モニタ Linux(R)編」の「monend (モニタモードのサーバの停止)」の説明を参照してください。

2. Oracle インスタンスを、HA モニタを介さないで起動する。

詳細は、「5.3.1 Oracle インスタンス・Oracle リスナーを HA モニタを介さないで起動する」を参照してください。

3. PDB が OPEN している場合は、PDB を CLOSE する。

PDB を CLOSE する方法については Oracle のドキュメントを参照してください。

4. PDB を unplug する。

PDB を unplug する方法については Oracle のドキュメントを参照してください。

5. PDB を plug する。

運用に合わせて、必要となる PDB を plug してください。PDB を plug する方法については Oracle のドキュメントを参照してください。

6. Oracle インスタンス制御用定義ファイルの PDB_NAME を変更する。

Oracle インスタンス制御用定義ファイルの PDB_NAME の指定を、系切り替え構成内のすべての系で、plug した PDB 名に変更してください。

PDB_NAME に指定した PDB 名を変更しない場合、および PDB_NAME に誤った PDB 名を指定した場合は、Oracle インスタンス起動時に、HA モニタ kit が Oracle インスタンスの異常を検出します。

7. Oracle インスタンスを、HA モニタを介さないで停止する。

詳細は、「[5.3.2 Oracle インスタンス・Oracle リスナーを HA モニタを介さないで停止する](#)」を参照してください。

8. モニタモードのサーバの起動コマンド (monbegin コマンド) でサーバを起動する。

(2) HA モニタのコマンドで Oracle の監視を一時停止してから、PDB を unplug および plug する

1. HA モニタ・サーバ稼働中の設定変更コマンド (monchange コマンド) で Oracle インスタンスの監視を一時停止する。

DB_NOACCESS に yes を指定している場合、この操作は不要です。

このコマンドについては、マニュアル「[HA モニタ Linux\(R\)編](#)」の、HA モニタ・サーバ稼働中の設定変更コマンド (monchange コマンド) の説明を参照してください。

2. PDB を CLOSE する。

PDB を CLOSE する方法については Oracle のドキュメントを参照してください。

3. PDB を unplug する。

PDB を unplug する方法については Oracle のドキュメントを参照してください。

4. PDB を plug する。

運用に合わせて、必要となる PDB を plug してください。PDB を plug する方法については Oracle のドキュメントを参照してください。

5. Oracle インスタンス制御用定義ファイルの PDB_NAME を変更する。

DB_NOACCESS に yes を指定している場合、この操作は不要です。

Oracle インスタンス制御用定義ファイルの PDB_NAME の指定を、系切り替え構成内のすべての系で、plug した PDB 名に変更してください。

PDB_NAME に指定した PDB 名を変更しない場合、および PDB_NAME に誤った PDB 名を指定した場合は、Oracle インスタンス監視の再開時に、HA モニタ kit が Oracle インスタンスの異常を検出します。

6. PDB を READ WRITE で OPEN する。

PDB を OPEN する方法については Oracle のドキュメントを参照してください。

7. HA モニタ・サーバ稼働中の設定変更コマンド (monchange コマンド) で Oracle インスタンスの監視を再開する。

このコマンドについては、マニュアル「HA モニタ Linux(R)編」の「monchange (HA モニタ・サーバ稼働中の設定変更)」の説明を参照してください。

6

メッセージ

この章では、HA モニタ kit が出力するメッセージについて説明します。

6.1 メッセージの形式

メッセージの形式について説明します。

6.1.1 メッセージの出力形式

出力されるメッセージの形式を次に示します。

```
KAMKnnnnn-i xx....xx
```

- KAMKnnnnn-i：メッセージ ID
- xx....xx：メッセージテキスト

サーバログに出力されるメッセージには、メッセージ ID の前に、出力日時が"yyyy/mm/dd HH:MM:SS"の形式で付与されます。

6.1.2 メッセージの記述形式

(1) 記述形式

このマニュアルでのメッセージの記述形式を次に示します。

KAMKnnnnn-i (X)

```
Message text
```

メッセージの意味について説明します。

aa....aa：可変値について説明します。

S：

システムがメッセージを出力したあとにする、主な処理を示します。

O：

メッセージ確認後、オペレータがする処理を示します。

対策：

メッセージ確認時のシステム管理者の処置を示します。

なお、「O：」や「対策：」に記載されている「保守員に連絡してください。」とは、購入時の契約に基づいて、システム管理者が弊社問い合わせ窓口へ連絡することを示します。

(2) メッセージ ID の記号の説明

メッセージ ID の記号の意味を次に示します。

KAMK :

HA モニタ kit のメッセージであることを示します。

nnnnn :

メッセージの通し番号を示します。

i :

メッセージの種類を示します。

E :

- 各ライブラリ、コマンド、サーバの機能が働かない障害が発生したことを示しています。
- 定義誤り、コマンドのオペランド指定誤りによって、動作できないことを示しています。

W :

- コマンドの動作状況などについての警告を示しています。

I :

- E, および W に該当しないメッセージで、単純に動作を示しています。

(X) :

メッセージの出力先種別を示します。

(P) : サーバログファイル

(L) : メッセージログファイル (syslog)

一つのメッセージが、複数の出力先を持つ場合は、出力先種別を"+"でつないで記載します。

6.2 メッセージの詳細

HA モニタ kit のメッセージの詳細について説明します。

KAMK00021-E (P+L)

```
The aa....aa command contains syntax errors.
```

HA モニタ kit のコマンドの実行形式に誤りがあります。

aa....aa : コマンド名

S :

コマンド処理を中止します。

O :

コマンドを正しい形式で実行してください。

KAMK00120-E (P+L)

```
The file aa....aa contains errors. Parameter : bb....bb
```

aa....aa に示す制御用定義ファイル内の、bb....bb に示すパラメタの指定方法が誤っています。

aa....aa : ファイル名

bb....bb : パラメタ名

S :

処理を終了します。

対策 :

[[3.2.7\(2\) 制御用定義ファイルの設定](#)] を参照して、パラメタ名に示すパラメタの誤りを修正してください。

KAMK00130-E (P+L)

```
The file aa....aa does not exist.
```

aa....aa に示す制御用定義ファイルが存在しません。

aa....aa : ファイル名

S :

処理を終了します。

対策：

次を確認してください。

- コマンドの引数に指定した Oracle インスタンスのインスタンス識別子、または Oracle リスナーのリスナー名が正しいか。
- 制御用定義ファイル名が正しいか。
- 制御用定義ファイル、およびそれらを配置しているディレクトリに、一般ユーザの読み込み権限が付与されているか。

制御用定義ファイルが存在しない場合は、「3.2.7(2) 制御用定義ファイルの設定」を参照して作成してください。

KAMK00201-I (P)

```
Usage : haorainsbgn ORACLE_SID
```

Oracle インスタンスの起動コマンド (haorainsbgn コマンド) の実行形式が誤っているとき、メッセージ KAMK00021-E に続いて出力されます。

S:

処理を続行します。

KAMK00202-I (P)

```
Usage : haorainsend ORACLE_SID
```

Oracle インスタンスの停止コマンド (haorainsend コマンド) の実行形式が誤っているとき、メッセージ KAMK00021-E に続いて出力されます。

S:

処理を続行します。

KAMK00203-I (P)

```
Usage : haorainsptl ORACLE_SID
```

Oracle インスタンスの状態確認コマンド (haorainsptl コマンド) の実行形式が誤っているとき、メッセージ KAMK00021-E に続いて出力されます。

S:

処理を続行します。

KAMK00204-I (P)

```
Usage : haoralsnrbgn LISTENER_NAME
```

Oracle リスナーの起動コマンド (haoralsnrbgn コマンド) の実行形式が誤っているとき、メッセージ KAMK00021-E に続いて出力されます。

S:

処理を続行します。

KAMK00205-I (P)

```
Usage : haoralsnrend LISTENER_NAME
```

Oracle リスナーの停止コマンド (haoralsnrend コマンド) の実行形式が誤っているとき、メッセージ KAMK00021-E に続いて出力されます。

S:

処理を続行します。

KAMK00206-I (P)

```
Usage : haoralsnrptl LISTENER_NAME
```

Oracle リスナーの状態確認コマンド (haoralsnrptl コマンド) の実行形式が誤っているとき、メッセージ KAMK00021-E に続いて出力されます。

S:

処理を続行します。

KAMK00301-I (P)

```
Oracle instance aa....aa will now start.
```

aa....aa で示す Oracle インスタンスを起動します。

aa....aa : インスタンス識別子

S:

処理を続行します。

KAMK00302-I (P)

```
Oracle instance aa....aa has started.
```

aa....aa で示す Oracle インスタンスの起動が完了しました。

aa....aa : インスタンス識別子

S:

処理を続行します。

KAMK00303-E (P+L)

An attempt to start Oracle instance aa....aa has failed.

aa....aa で示す Oracle インスタンスの起動に失敗しました。

aa....aa：インスタンス識別子

S：

処理を終了します。

O：

サーバログに出力されている内容を基に原因を調査してください。

KAMK00304-I (P)

Oracle instance aa....aa will now stop.

aa....aa で示す Oracle インスタンスを停止します。

aa....aa：インスタンス識別子

S：

処理を続行します。

KAMK00305-I (P)

Oracle instance aa....aa has stopped.

aa....aa で示す Oracle インスタンスを停止しました。

aa....aa：インスタンス識別子

S：

処理を続行します。

KAMK00306-E (P+L)

An attempt to stop Oracle instance aa....aa has failed.

aa....aa で示す Oracle インスタンスの停止に失敗しました。

aa....aa：インスタンス識別子

S：

処理を終了します。

O:

サーバログに出力されている内容を基に原因を調査してください。

KAMK00308-E (P+L)

An error was detected in Oracle instance aa....aa.

Oracle インスタンスの状態確認によって、aa....aa で示す Oracle インスタンスの異常を検知しました。

aa....aa: インスタンス識別子

S:

処理を続行します。

対策:

Oracle のログを確認し、異常を検知した原因を取り除いてください。

KAMK00309-E (P+L)

A slowdown was detected in Oracle instance aa....aa.

Oracle インスタンスの状態確認によって、aa....aa で示す Oracle インスタンスのスローダウンを検知しました。

aa....aa: インスタンス識別子

S:

処理を続行します。

対策:

Oracle のログを確認し、スローダウンを検知した原因を取り除いてください。

KAMK00310-W (P)

Oracle instance aa....aa could not be terminated.

aa....aa で示す Oracle インスタンスを正常停止できませんでした。

aa....aa: インスタンス識別子

S:

処理を続行します。

KAMK00311-W (P)

The processing to stop Oracle instance aa....aa timed out.

aa....aa で示す Oracle インスタンスの正常停止処理がタイムアウトしたことによって、Oracle インスタンスを正常停止できませんでした。

aa....aa：インスタンス識別子

S：

処理を続行します。

KAMK00312-I (P)

```
Oracle instance aa....aa will be forcibly stopped.
```

aa....aa で示す Oracle インスタンスを強制停止します。

aa....aa：インスタンス識別子

S：

処理を続行します。

KAMK00321-I (P)

```
Oracle listener aa....aa will now start.
```

aa....aa で示す Oracle リスナーを起動します。

aa....aa：リスナー名

S：

処理を続行します。

KAMK00322-I (P)

```
Oracle listener aa....aa has started.
```

aa....aa で示す Oracle リスナーの起動が完了しました。

aa....aa：リスナー名

S：

処理を続行します。

KAMK00323-E (P+L)

```
An attempt to start Oracle listener aa....aa has failed.
```

aa....aa で示す Oracle リスナーの起動に失敗しました。

aa....aa：リスナー名

S：

処理を終了します。

O：

サーバログに出力されている内容を基に原因を調査してください。

KAMK00324-I (P)

```
Oracle listener aa....aa will now stop.
```

aa....aa で示す Oracle リスナーを停止します。

aa....aa：リスナー名

S：

処理を続行します。

KAMK00325-I (P)

```
Oracle listener aa....aa has stopped.
```

aa....aa で示す Oracle リスナーを停止しました。

aa....aa：リスナー名

S：

処理を続行します。

KAMK00326-E (P+L)

```
An attempt to stop Oracle listener aa....aa has failed.
```

aa....aa で示す Oracle リスナーの停止に失敗しました。

aa....aa：リスナー名

S：

処理を続行します。

O：

サーバログに出力されている内容を基に原因を調査してください。

KAMK00328-E (P+L)

```
An error was detected in Oracle listener aa....aa.
```

Oracle リスナーの状態確認によって、aa....aa で示す Oracle リスナーの異常を検知しました。

aa....aa：リスナー名

S：

処理を続行します。

対策：

Oracle のログを確認し、異常を検知した原因を取り除いてください。

KAMK00329-E (P+L)

```
A slowdown was detected in Oracle listener aa....aa.
```

Oracle リスナーの状態確認によって、aa....aa で示す Oracle リスナーのスローダウンを検知しました。

aa....aa：リスナー名

S：

処理を続行します。

対策：

Oracle のログを確認し、スローダウンを検知した原因を取り除いてください。

KAMK00330-W (P)

```
Oracle listener aa....aa could not be terminated.
```

aa....aa で示す Oracle リスナーを正常停止できませんでした。

aa....aa：リスナー名

S：

処理を続行します。

KAMK00331-W (P)

```
The processing to stop Oracle listener aa....aa timed out.
```

aa....aa で示す Oracle リスナーの正常停止処理がタイムアウトしたことによって、Oracle リスナーを正常停止できませんでした。

aa....aa：リスナー名

S：

処理を続行します。

KAMK00332-I (P)

```
Oracle listener aa....aa will be forcibly stopped.
```

aa....aa で示す Oracle リスナーを強制停止します。

aa....aa：リスナー名

S：

処理を続行します。

KAMK00356-E (P)

```
User name : aa....aa does not exist in system.
```

ユーザ名に示すユーザがシステムに存在しません。

aa....aa：ユーザ名

S：

処理を続行します。

対策：

制御用定義ファイルの、ORACLE_USER に設定した Oracle ユーザのアカウント名が正しいかどうかを確認してください。正しい場合は、システムに Oracle ユーザが存在することを確認してください。

KAMK00401-I (P)

```
***** The actcommand start. *****
```

サーバの起動コマンドが実行されたことを示します。

S：

処理を続行します。

KAMK00402-I (P)

```
***** The actcommand end. (aa....aa) *****
```

サーバの起動コマンドの実行が終了したことを示します。

aa....aa：サーバの起動コマンドの戻り値

S：

処理を続行します。

対策：

戻り値が 0 以外の場合は、サーバの起動コマンド内の設定内容とサーバログを調査してください。

KAMK00411-I (P)

```
***** The termcommand start. *****
```

サーバの停止コマンドが実行されたことを示します。

S:

処理を続行します。

KAMK00412-I (P)

```
***** The termcommand end. (aa....aa) *****
```

サーバの停止コマンドの実行が終了したことを示します。

aa....aa:サーバの停止コマンドの戻り値

S:

処理を続行します。

対策:

戻り値が0以外の場合は、サーバの停止コマンド内の設定内容とサーバログを調査してください。

KAMK00421-I (P)

```
***** The patrolcommand start. *****
```

サーバの監視コマンドが実行されたことを示します。

S:

処理を続行します。

KAMK00422-I (P)

```
***** The patrolcommand end. (aa....aa) *****
```

サーバの監視コマンドの実行が終了したことを示します。

aa....aa:サーバの監視コマンドの戻り値

S:

処理を続行します。

対策:

戻り値が0以外の場合は、サーバの監視コマンド内の設定内容とサーバログを調査してください。

KAMK00423-E (P+L)

An error was detected during monitoring process. The patrolcommand will now stop.

サーバの監視コマンド内の監視処理で異常を検知したため監視コマンドを終了します。

S:

処理を終了します。

対策:

サーバログに出力された内容を基に原因を調査してください。

なお、HA モニタ kit が提供しているコマンド内では、ヒアドキュメントを利用しているため、一時ファイルの出力先である/tmp 領域に空きがない場合、サーバの監視コマンドが異常を検知します。異常を検知した場合は、/tmp 領域に空きを作成し、かつ、サーバ稼働中に/tmp 領域が満杯にならないようにしてください。

また、サーバ起動後すぐにこのエラーメッセージが出力されて異常を検知する場合は、設定に誤りがあるおそれがあります。次に示す設定を確認してください。

- HA モニタから利用するシェルの改行コードが、OS に準拠したものになっているか。
- HA モニタから利用するシェルの構文、およびパスなどの設定値が正しいか。
- サーバの監視コマンドに、Oracle ユーザの読み込み権限と実行権限が付与されているか。
- サーバの監視コマンド実行シェルの PATROL_SHELL 環境変数に指定する、サーバの監視コマンドのパスが正しいか。
- 同じサーバの起動・停止・監視コマンドの LOGNAME 環境変数には、すべて同じ値を指定しているか。また、サーバごとにユニークな値を指定しているか。
- HA モニタのサーバ対応の環境設定 (servers) の waitserv_exec オペランドに「yes」を指定しているか。
- /opt/hitachi/HAMon/spool/配下のログファイル (oralog) に、一般ユーザの読み込み権限、および書き込み権限が付与されているか。
サーバの監視コマンド、およびサーバの停止コマンドを手動で実行した場合、一般ユーザの書き込み権限がないログファイルが生成されているおそれがあります。
- CDB (シングルテナント) 構成の場合、インスタンス制御用定義ファイルの CDB に yes が設定されているか。
- CDB (シングルテナント) 構成の場合、インスタンス制御用定義ファイルの PDB_NAME に設定した PDB 名が正しいか。

KAMK00601-E (P+L)

An error occurred in the aa.....aa system call. Error details : bb....bb Functional name : cc....cc

HA モニタ kit の関数内で発行したシステムコールでエラーが発生しました。詳細コードについては、対応する OS のマニュアルを参照してください。

aa....aa : 発行したシステムコール

bb....bb : 詳細コード (1~3桁)

cc....cc : エラーが発生した関数名

S :

処理を続行します。

O :

システム管理者に連絡してください。

対策 :

詳細コードを基に原因を調査してください。

付録

付録 A 各バージョンの変更内容

各バージョンの変更内容を示します。

付録 A.1 HA モニタ kit for Oracle 01-03 の変更内容 (3000-9-135-20)

HA モニタ kit for Oracle 01-03 の変更内容 (3000-9-135-20) を次の表に示します。

追加・変更内容
Oracle のデータベース構成として、CDB (シングルテナント) 構成に対応した。 これに伴い、次の制御用定義パラメタを追加した。 <ul style="list-style-type: none">• PATROL_TABLE• CDB• PDB_NAME
サーバの監視を一時停止できるようにした。 これに伴い、次のオペランドを追加した。 <ul style="list-style-type: none">• ptrlcmd_ex• ptrlcmd_ex_inter• ptrlcmd_ex_retry• ptrlcmd_ex_tmout これによって、本番運用中、監視対象のサーバプログラムをメンテナンスしたい場合などに監視状態を解除することができる。
HA モニタ kit が監視する表領域として dual 表を指定できるようにした。これに伴い、次の制御用定義パラメタを追加した。 <ul style="list-style-type: none">• PATROL_TABLE
次のメッセージを追加した。 <ul style="list-style-type: none">• KAMK00356-
次のメッセージを変更した。 <ul style="list-style-type: none">• KAMK00130-E• KAMK00423-E
前版の変更内容を付録に移動した。
Oracle Database 18c をサポートした。

付録 A.2 HA モニタ kit for Oracle 01-01 の変更内容 (3000-9-135-10)

HA モニタ kit for Oracle 01-01 の変更内容 (3000-9-135-10) を次の表に示します。

追加・変更内容
HA モニタおよび HA モニタ kit がする処理の流れについての説明を変更した。
HA モニタ kit のコマンドを編集できるようにした。これに伴い、次のサブコマンドを追加した。

追加・変更内容

- haorainsendsub (Oracle インスタンスの停止)
- haorainsptlsub (Oracle インスタンスの状態確認)
- haoralsnrendsub (Oracle リスナーの停止)
- haoralsnrptlsub (Oracle リスナーの状態確認)
- haoraenvcheck (制御用定義ファイルのチェック)

また、サーバの起動・停止・監視コマンドのサンプルファイルの内容を変更した。

制御用定義ファイルに、任意の環境変数を指定できるようにした。

制御用定義ファイルに指定できるパラメタに、ORACLE_BASE を追加した。

注意が必要なサーバ対応の環境設定のオペランドに servexec_retry を追加した。

Oracle Database 12c をサポートした。

付録 B このマニュアルの参考情報

このマニュアルを読むに当たっての参考情報を示します。

付録 B.1 関連マニュアル

関連マニュアルを次に示します。必要に応じてお読みください。

- 高信頼化システム監視機能 HA モニタ Linux(R)編 (3000-9-132), および高信頼化システム監視機能 HA モニタ Linux(R) (x86) 編 (3000-9-201)
このマニュアルでは, 「HA モニタ Linux(R)編」と略して記載します。
- 高信頼化システム監視機能 HA モニタ メッセージ (3000-9-203)
このマニュアルでは, 「HA モニタ メッセージ」と略して記載します。

付録 B.2 このマニュアルでの表記

このマニュアルでは, 製品名を次のように表記しています。

製品名		表記
Linux		Linux(R)
Linux	Linux (x86)	Red Hat(R) Enterprise Linux(R) server 6(x86)
		Red Hat(R) Enterprise Linux(R) server 6(x86_64)
		Red Hat(R) Enterprise Linux(R) server 7(x86_64)
		Red Hat(R) Enterprise Linux(R) server 8(x86_64)
Oracle		Oracle Database 10g
		Oracle Database 11g
		Oracle Database 12c
		Oracle Database 18c
		Oracle Database 19c
UNIX		UNIX

付録 B.3 英略語

このマニュアルで使用する英略語を次に示します。

英略語	英字での表記
ASM	Automatic Storage Management
CDB	Multitenant Container Database
DB	Database
EUC	Extended UNIX Code
FS	File System
HA	High Availability
I/O	Input/Output
LAN	Local Area Network
LVM	Logical Volume Manager
Non-CDB	Non-Multitenant Container Database
OS	Operating System
PDB	Pluggable Database
RAC	Real Application Clusters
UTF-8	8-bit UCS Transformation Format

付録 B.4 KB (キロバイト) などの単位表記について

1KB (キロバイト), 1MB (メガバイト), 1GB (ギガバイト), 1TB (テラバイト) はそれぞれ 1,024 バイト, 1,024² バイト, 1,024³ バイト, 1,024⁴ バイトです。

索引

A

ABORT_LIMIT 58
acttype 65

C

CDB 58
CDB(シングルテナント)構成 11
CDB(マルチテナント)構成 12

D

DB_NOACCESS 57

G

group 66

H

haoraenvcheck 112
haorainsbgn 104
haorainsend 105
haorainsendsub 107
haorainsptl 106
haorainsptlsub 108
haoralsnrbgn 108
haoralsnrend 109
haoralsnrendsub 111
haoralsnrptl 110
haoralsnrptlsub 112
HA モニタ kit 10
 構成 11
 設定 54
 必要なソフトウェア 15
 必要なハードウェア 14
 目的 10
HA モニタ kit が検出する障害 21
HA モニタが検出する障害 22

L

LAN 14
LISTENER_NAME 57
LISTENER_NOACCESS 58
LOGNAME [サーバの監視コマンド] 95, 99
LOGNAME [サーバの起動コマンド] 87
LOGNAME [サーバの停止コマンド] 90
LOGSIZE [サーバの監視コマンド] 95, 99
LOGSIZE [サーバの起動コマンド] 87
LOGSIZE [サーバの停止コマンド] 91

M

monbegin 116
monend 116
monts 123

N

name 65
Non-CDB (シングル) 構成 11

O

Oracle
 アップグレードする 125
 インストール 52
 構成 13
 構成と系切り替え単位の検討 44
 データベースの構築 53
ORACLE_BASE 57
ORACLE_HOME 57
ORACLE_SID 57
ORACLE_USER 57
ORACLE_USER [サーバの監視コマンド実行シェル]
 101
Oracle インスタンス 13
 HA モニタを介さないで起動する 118
 HA モニタを介さないで停止する 119
 監視 116

- 起動 116
- 削除する 126
- 制御用定義ファイル 56
- 追加する 125
- 停止 116
- Oracle インスタンスの起動 [コマンド] 104
- Oracle インスタンスの状態確認 [コマンド] 106
- Oracle インスタンスの状態確認 [サブコマンド] 108
- Oracle インスタンスの停止 [コマンド] 105
- Oracle インスタンスの停止 [サブコマンド] 107
- Oracle リスナー 13
 - HA モニタを介さないで起動する 118
 - HA モニタを介さないで停止する 119
 - 監視 116
 - 起動 116
 - 構築 53
 - 削除する 126
 - 制御用定義ファイル 56
 - 追加する 125
 - 停止 116
- Oracle リスナーの起動 [コマンド] 108
- Oracle リスナーの状態確認 [コマンド] 110
- Oracle リスナーの状態確認 [サブコマンド] 112
- Oracle リスナーの停止 [コマンド] 109
- Oracle リスナーの停止 [サブコマンド] 111

P

- PATROL 58
- PATROL_INTERVAL [サーバの監視コマンド] 99
- PATROL_SHELL [サーバの監視コマンド実行シェル] 101
- PATROL_TABLE 58
- patrolcommand 66
- PDB_NAME 58
- ptrlcmd_ex 66
- ptrlcmd_ex_inter 66
- ptrlcmd_ex_retry 66
- ptrlcmd_ex_tmout 66

R

- RAC 構成 12

S

- servexec_retry 66
- start_timeout 67
- stop_timeout 67

T

- termcommand 65

W

- waitserv_exec 66

い

- インスタンス識別子 57

か

- 環境設定例 69

き

- 共有ディスク 14
- 共有リソース
 - Oracle が使用する共有リソースの設定 54

け

- 系切り替え構成 13
- 系切り替えのテスト 68

こ

- 構築に必要な設定項目の対応と設定の概要 50

さ

- サーバ 11
 - 動作確認 67
- サーバの監視コマンド
 - カスタマイズ 96, 100
- サーバの監視コマンドのサンプルファイルの格納ディレクトリ 55

サーバの監視コマンドの設定 (patrolcommand オペランドでサーバの監視コマンド実行シェルを指定する場合) 98

サーバの監視コマンドの設定 (ptrlcmd_ex オペランドでサーバの監視コマンドを指定する場合) 93

サーバの起動コマンド

カスタマイズ 88

設定 86

サーバの起動・停止コマンドのサンプルファイルの格納ディレクトリ 55

サーバの停止コマンド

カスタマイズ 91

設定 89

サーバログの確認 120

し

システム構築の流れ 48

システムの動作確認 67

障害

Oracle インスタンスの障害 21

Oracle インスタンスのスローダウン 21

Oracle リスナーの障害 21

Oracle リスナーのスローダウン 22

系障害 22

種類 21

障害検出の仕組み 19

障害情報を収集する 123

ネットワーク障害 22

処理の流れ 23

Oracle インスタンス障害時の系切り替え 38

Oracle インスタンスの監視 36

Oracle インスタンスの起動 24

Oracle インスタンスの起動失敗 26

Oracle インスタンスの起動タイムアウト時 28

Oracle インスタンスの停止 33

Oracle リスナー障害時の系切り替え 40

Oracle リスナーの監視 37

Oracle リスナーの起動 25

Oracle リスナーの起動失敗 30

Oracle リスナーの起動タイムアウト時 31

Oracle リスナーの停止 35

シングル構成 11

シングルテナント構成 11

せ

制御用定義ファイルのコピー元ファイルとコピー先ディレクトリ 56

制御用定義ファイルの設定 56

制御用定義ファイルのチェック [サブコマンド] 112

て

定義ファイル 65

ディレクトリ構成 49

ま

マルチテナント構成 12

め

メッセージの言語環境の対応 122

り

リスナー名 57

リソース

構成の検討 46

サーバとの対応 46

リソースサーバ

検討 46

ろ

ローカルディスク 14