

CLUSTERPRO MC

ApplicationMonitor 1.2 for Linux

ユーザーズガイド

© 2014(Apr) NEC Corporation

- ☐ 製品の概要と特長
- ☐ 製品のソフトウェア構成
- ☐ 製品の設定
- ☐ 製品の動作
- ☐ 製品の運用
- ☐ コマンドリファレンス

はしがき

本書は、CLUSTERPRO MC ApplicationMonitor 1.2 for Linux(以後 ApplicationMonitor と記載します)の機能と操作方法について記載したものです。

(1) 本書は、以下のオペレーティングシステムに対応します。

- ・ Red Hat Enterprise Linux 5.7 ~ 5.10 , 6.1 ~ 6.5
- ・ Oracle Linux 6.1 ~ 6.5

ただし、OS がバージョンアップする際に、本書の内容が変更される場合があります。

(2) 本書で説明しているすべての機能は、プログラムプロダクトであり、次の表のプロダクト型番およびプロダクト名に対応します。

プロダクト型番	プロダクト名	プロダクトリリース
UL4437-302	CLUSTERPRO MC ApplicationMonitor 1.2 for Linux (1CPU ライセンス)	1.2
UL4437-312	CLUSTERPRO MC ApplicationMonitor 1.2 for Linux (1CPU ライセンス)(他社機版)	1.2
UL4437-322	CLUSTERPRO MC ApplicationMonitor 1.2 for Linux VM (1 ノードライセンス)	1.2
UL4437-332	CLUSTERPRO MC ApplicationMonitor 1.2 for Linux VM (1 ノードライセンス)(他社機版)	1.2
UL4437-301	CLUSTERPRO MC ApplicationMonitor CD 1.2	1.2

(3) 本書は、次の表現を使用します。

- ・ 大かっこ [] 省略可能なパラメータを表します。
- ・ 山かっこ < > 任意の文字列を指定することを表します。

(4) 商標および登録商標

- ✓ Linux は、Linus Torvalds 氏の米国およびその他の国における、登録商標または商標です。
- ✓ CLUSTERPRO は、日本電気株式会社の登録商標です。
- ✓ ORACLE は、米国 Oracle Corporation の登録商標です。
- ✓ その他、本書に登場する会社名および商品名は各社の商標または登録商標です。
- ✓ なお、本書では®、TM マークを明記しておりません。

目次

1. 製品の概要と特長	1
1.1. 概要.....	1
1.2. 特長.....	3
1.2.1. インスタンス障害の監視と障害時アクション.....	4
1.2.2. リスナー障害の監視と障害時アクション	4
1.2.3. データベース領域の容量不足の監視と障害時アクション	4
1.2.4. ローカルディスク領域の容量不足の監視と障害時アクション.....	5
1.2.5. Oracle Clusterware / Oracle Restart 無応答障害の監視と障害時アクション	5
1.2.6. Oracle Data Guard Broker 構成の REDO ログ転送状況によるスタンバイ・インスタンスの監視と障害時アクション.....	5
1.2.7. Oracle Data Guard Broker 構成の REDO ログ適用状況によるスタンバイ・インスタンスの監視と障害時アクション.....	5
1.2.8. Oracle Data Guard Broker 構成の Oracle Data Guard Broker プロセスの状態によるスタンバイ・インスタンスの監視と障害時アクション.....	6
1.2.9. Oracle のシステムステート・ダンプの自動採取とシェルスクリプトの実行.....	6
1.2.10. Oracle Clusterware / Oracle Restart によるインスタンス / リスナー再起動の待ち合わせ	6
1.2.11. クラスタウェアへの障害通知を障害種別ごとに指定	6
1.2.12. 障害検出時の ApplicationMonitor 停止の設定	6
1.2.13. Oracle Data Guard Broker によるファスト・スタート・フェイルオーバー(自動フェイルオーバー)の待ち合わせ.....	7
1.2.14. Oracle Automatic Storage Management (ASM) インスタンス停止に伴うデータベース再起動の待ち合わせ	7
1.2.15. 統計情報の採取.....	7
1.2.16. ApplicationMonitor の一括設定、一括変更	7
1.2.17. 監視、採取の一時停止と再開.....	8
1.2.18. 各モニタ状態、各モニタ設定の表示と最大監視応答時間の表示	8
1.2.19. 設定ファイルのテンプレート作成機能.....	8
1.2.20. 複数インスタンス監視.....	9
1.2.21. 監視モニタ、採取モニタの単独起動	9
1.2.22. Oracle エラーのフィルタリング設定	9
1.2.23. Oracle のディザスタ・リカバリ構成のバックアップサイトの監視	9
2. 製品のソフトウェア構成	11
2.1. 監視プロセス.....	14

2.1.1.	モニタ制御デーモン (oramond)	14
2.1.2.	インスタンス監視モニタ (instmond).....	15
2.1.3.	リスナー監視モニタ (lsnrmond)	16
2.1.4.	表領域監視モニタ (tschkmond).....	16
2.1.5.	ディスク領域監視モニタ (fdsmond).....	17
2.1.6.	スタンバイデータベース監視モニタ (stbdbmond)	17
2.1.7.	アラートログ監視モニタ (altmond).....	18
2.1.8.	CRS 監視モニタ (crsmond).....	18
2.2.	採取プロセス.....	19
2.2.1.	統計情報採取モニタ (statsmond).....	19
2.2.2.	システムステートダンプ採取デーモン (ssdump).....	19
2.3.	管理インタフェース	20
2.3.1.	管理コマンド (oraadmin).....	20
2.3.2.	構成情報管理サーバ (oraconfd)	20
2.4.	設定ファイル	21
2.4.1.	設定ファイル (oramond.conf)	21
2.4.2.	構成情報ファイル (oramond<構成番号>.bin)	21
2.4.3.	CRS 監視設定ファイル (crsmond.conf).....	21
2.4.4.	フィルタリング設定ファイル (inst.conf、lsnr.conf、tschk.conf、fds.conf、 stbdb.conf、stats.conf、ssdump.conf).....	21
2.4.5.	RENS 連 携 用 辞 書 フ ァ イ ル (haam.dict 、 haam_stat.dict 、 haam_stat_map.dict).....	21
3.	製品の設定	23
3.1.	インスタンス監視表の作成	24
3.1.1.	インスタンス監視表の作成方法 (Oracle 11g (11.1.0、11.2.0) の場合 / Oracle 12c のマルチテナント構成以外の場合)	24
3.1.2.	インスタンス監視表の作成方法 (Oracle 12c のマルチテナント構成の場合)	28
3.2.	設定ファイル (oramond.conf) の作成と適用	36
3.2.1.	設定ファイルの記述例	37
3.2.2.	コメントの記述形式	44
3.2.3.	パラメータの記述形式	44
3.2.4.	ステートメントの記述形式	45
3.2.5.	ステートメントの階層	64
3.2.6.	パラメータとステートメントの記述順	66
3.2.7.	パラメーター一覧	68
3.2.8.	ログレベルの設定.....	102
3.2.9.	ノード名の設定	103

3.2.10.	スタンバイ・インスタンス監視の設定 (Oracle Data Guard Broker 構成のみ)	103
3.2.11.	停止ノード選択ポリシーの設定 (RAC 構成のみ)	104
3.2.12.	障害ノード上の ApplicationMonitor の停止動作の設定	104
3.2.13.	Oracle 構成プロセス停止の設定	104
3.2.14.	Oracle DBA 監査 (audit) 機能対応の設定	105
3.2.15.	Oracle パスワードの扱い	105
3.2.16.	パスワード・ファイル認証での接続	106
3.2.17.	監視モニタ制御の設定	106
3.2.18.	Oracle ハング検知機能を利用した監視 (RAC 構成のみ)	106
3.2.19.	統計情報採取の設定	107
3.2.20.	ASM インスタンス停止に伴うデータベース再起動の待ち合わせの設定	109
3.2.21.	インスタンス監視表のスキーマの設定 (Oracle 12c のみ)	109
3.2.22.	指定表監視時の障害の設定 (Oracle 12c のマルチテナント構成のみ)	109
3.2.23.	インスタンス / リスナー障害時の CLUSTERPRO への障害通知の設定	110
3.2.24.	リスナー再起動方法の設定	110
3.2.25.	リスナーの繰り返し障害の監視	111
3.3.	Oracle エラーのフィルタリング設定	112
3.3.1.	フィルタリング設定ファイルの記述形式	113
3.3.2.	フィルタリングのデフォルト設定	115
3.4.	情報採取スクリプト (oradiag) の設定	117
3.5.	リスナー監視モニタの単独起動	118
3.6.	表領域監視モニタの単独起動	118
3.7.	ディスク領域監視モニタの単独起動	119
3.8.	統計情報採取モニタの単独起動	119
3.9.	CRS 監視設定ファイル (crsmond.conf) の作成	120
3.9.1.	CRS 監視設定ファイルの記述例	120
3.9.2.	コメントの記述形式	121
3.9.3.	パラメータの記述形式	121
3.9.4.	パラメーター一覧	122
3.9.5.	crsd/ohasd の状態表示用コマンドの設定	123
4.	製品の動作	125
4.1.	ApplicationMonitor の起動と停止	125
4.2.	インスタンス監視時の動作	127
4.3.	リスナー監視時の動作	137
4.4.	データベース領域の容量監視時の動作	149
4.5.	ローカルディスク領域の容量監視時の動作	150
4.6.	スタンバイ・インスタンス監視時の動作	151
4.7.	統計情報採取時の動作	161

4.8.	システムステート・ダンプ採取時の動作	162
4.9.	シェルスクリプトの実行時の動作	164
4.10.	Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの動作	165
4.11.	Oracle Data Guard Broker による自動フェイルオーバー時の監視動作	169
4.12.	ASM インスタンス停止に伴うデータベース再起動時の監視動作	171
4.13.	Oracle Clusterware / Oracle Restart 監視時の動作	173
5.	製品の運用	183
5.1.	事前準備	184
5.1.1.	Oracle ハング検知機能の停止 (RAC 構成のみ)	186
5.1.2.	Oracle Clusterware / Oracle Restart の監視対象設定	187
5.2.	ApplicationMonitor の設定	190
5.3.	ApplicationMonitor の設定確認	195
5.4.	ApplicationMonitor の設定配布	197
5.5.	ApplicationMonitor の CLUSTERPRO 連携設定	199
5.5.1.	EXEC リソースの追加	199
5.5.2.	PID モニタリソースの作成	203
5.5.3.	外部連携モニタリソースの作成	204
5.5.4.	クラスタ構成情報の反映	205
5.6.	ApplicationMonitor の起動	206
5.7.	ApplicationMonitor 動作中の状態確認	207
5.8.	ApplicationMonitor の設定パラメータの調整	209
5.9.	ApplicationMonitor 動作中の設定パラメータの変更	211
5.10.	ApplicationMonitor の一時停止と再開	213
5.10.1.	監視および採取の一時停止	213
5.10.2.	監視および採取の再開	216
5.11.	フェイルオーバーグループ停止原因の確認	219
6.	コマンドリファレンス	221

1. 製品の概要と特長

1.1. 概要

現用待機構成の場合:

ApplicationMonitor は、Oracle Database (11.1.0、11.2.0、12.1.0) (以後 Oracle と記載します)のインスタンス障害、リスナー障害、データベース領域の容量不足、ローカルディスク領域の容量不足を監視します。また、Oracle 11g R2 および Oracle 12c 環境の Oracle Restart、Oracle Data Guard Broker 構成のスタンバイ・インスタンスを監視します。

CLUSTERPRO と組み合わせて使用することにより、アプリケーションレベルでより高度な可用性を実現できます。さらに、サーバ管理基盤(Resource Event Notification Service、以後 RENS と記載します)と連携することで、RENS で行われていたハードウェア監視機能などと統一されたインタフェースでのログ監視(障害情報管理)やクラスタ連携が行えます。

RENS との連携機能は、Express 5800/A1000 シリーズのみ対応しています。

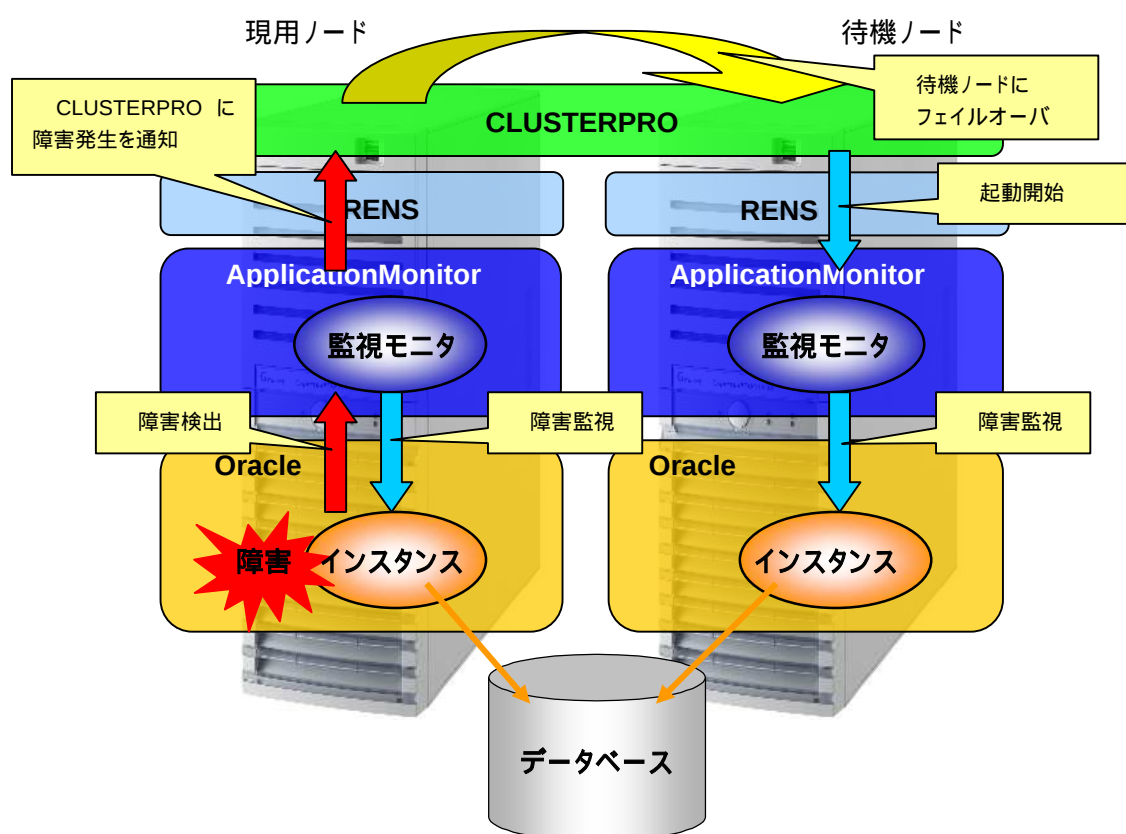


図 1-1 概要(現用待機構成)

RAC 構成の場合:

ApplicationMonitor は、Oracle Real Application Clusters (11.1.0、11.2.0、12.1.0) (以後 RAC と記載します)のインスタンス障害、リスナー障害、データベース領域の容量不足、ローカルディスク領域の容量不足を監視します。また、Oracle Clusterware、Oracle Data Guard Broker 構成のスタンバイ・インスタンスを監視します。

CLUSTERPRO と組み合わせて使用することにより、アプリケーションレベルでより高度な可用性を実現できます。さらに、RENS と連携することで、RENS で行われていたハードウェア監視機能などと統一されたインタフェースでのログ監視(障害情報管理)やクラスタ連携が行えます。

RENS との連携機能は、Express5800/A1000 シリーズのみ対応しています。

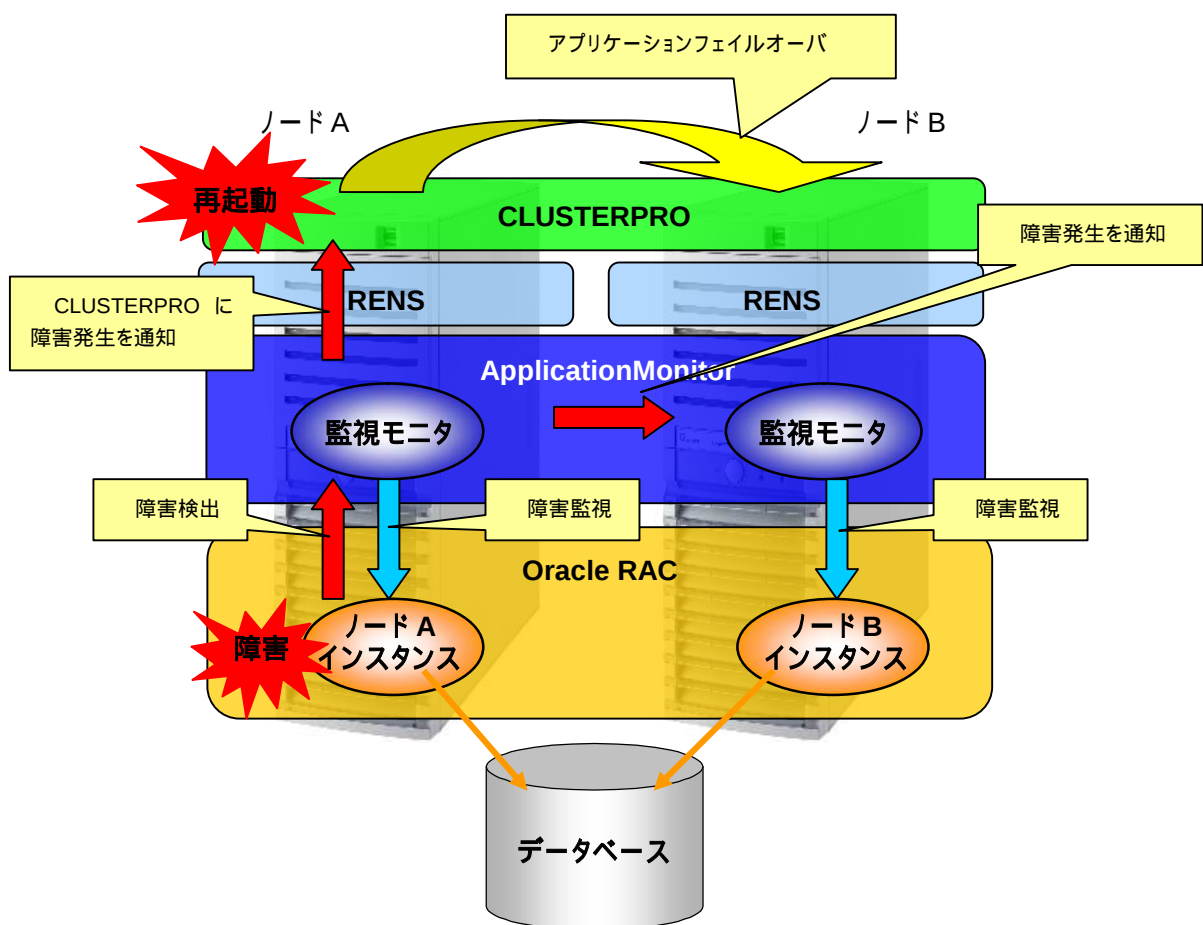


図 1-2 概要(RAC 構成)

1.2. 特長

ApplicationMonitor は、以下の特長・機能を持ちます。

◆ 監視機能

- ・ インスタンス障害の監視
- ・ リスナー障害の監視
- ・ データベース領域の容量不足の監視
- ・ ローカルディスク領域の容量不足の監視
- ・ Oracle Clusterware / Oracle Restart 無応答障害の監視
- ・ Oracle Data Guard Broker 構成の REDO ログ転送状況によるスタンバイ・インスタンスの監視
- ・ Oracle Data Guard Broker 構成の REDO ログ適用状況によるスタンバイ・インスタンスの監視
- ・ Oracle Data Guard Broker 構成の Oracle Data Guard Broker プロセスの状態によるスタンバイ・インスタンスの監視

◆ 障害時アクション

- ・ 障害検出時に CLUSTERPRO へ障害を通知
- ・ 障害検出時に Oracle のシステムステート・ダンプを自動採取
- ・ 障害検出時に任意のシェルスクリプトを実行
- ・ リスナー障害検出時にリスナーを再起動
- ・ データベース領域の容量不足検出時にコマンドを実行
- ・ ローカルディスク領域の容量不足検出時にコマンドを実行
- ・ 障害検出時に Oracle Clusterware / Oracle Restart によるインスタンス / リスナーの再起動を待ち合わせることが可能
- ・ 障害種別ごとにクラスタウェアへの障害通知を指定することが可能
- ・ Oracle Data Guard Broker によるファスト・スタート・フェイルオーバー(自動フェイルオーバー)を待ち合わせることが可能
- ・ Oracle Automatic Storage Management (ASM) インスタンス停止に伴うデータベース再起動を待ち合わせることが可能

◆ 設定・運用・メンテナンス時の支援機能

- ・ 障害検出時の ApplicationMonitor 停止の設定
- ・ 統計情報の採取
- ・ ApplicationMonitor の一括設定、一括変更
- ・ 監視、採取の一時停止と再開

1 製品の概要と特長

- ・ 各モニタ状態、各モニタ設定の表示
 - ・ 最大監視応答時間(ターンアラウンドタイム)の表示
 - ・ 設定ファイルのテンプレート作成機能
- ◆ さまざまな運用形態に対応した構成
- ・ 同一ノード上で複数のインスタンスを監視
 - ・ リスナー障害、データベース領域の容量不足、ローカルディスク領域の容量不足の単独監視
 - ・ 統計情報の単独採取
 - ・ Oracle エラーのフィルタリング設定
 - ・ Oracle の災害対策(ディザスタ・リカバリ)構成のバックアップサイトの監視

特長・機能について説明します。

1.2.1. インスタンス障害の監視と障害時アクション

ApplicationMonitor は、一定時間ごとにデータベースの参照 / 更新などを行い、Oracle バックグラウンドプロセス単位でインスタンス障害を監視します。また、障害監視の対象となる表を指定すると、特定の表で発生するインスタンス障害を監視します。

障害検出時、設定した回数のインスタンスを再確認します。設定回数内にデータベースの参照 / 更新が成功しない場合には、最終的にインスタンスの障害と判断します。このとき、ApplicationMonitor は、自身を停止することで CLUSTERPRO へ障害を通知します。RENS と連携している場合は、プロセス停止時に RENS を経由し、CLUSTERPRO へ障害を通知します。

1.2.2. リスナー障害の監視と障害時アクション

ApplicationMonitor は、一定時間ごとに指定したリスナー名に対するネット・サービスの有無を監視することにより、リスナー障害を監視します。

障害検出時、設定した回数のリスナーを再起動します。設定回数内にリスナーが回復しない場合には、最終的にリスナーの障害と判断します。このとき、インスタンス障害と同様(設定により変更可)に CLUSTERPRO へ障害を通知します。RENS と連携している場合は、プロセス停止時に RENS を経由し、CLUSTERPRO へ障害を通知します。

1.2.3. データベース領域の容量不足の監視と障害時アクション

ApplicationMonitor は、一定時間ごとにデータベース領域の最大使用可能エクステントサイズと空き容量率を監視します。設定したしきい値を下回った場合、任意のコマンドを実行します。

1.2.4. ローカルディスク領域の容量不足の監視と障害時アクション

ApplicationMonitor は、一定時間ごとに Oracle アーカイブログ出力先などのローカルディスク領域の空き容量率を監視します。設定したしきい値を下回った場合、任意のコマンドを実行します。

1.2.5. Oracle Clusterware / Oracle Restart 無応答障害の監視と障害時アクション

ApplicationMonitor は、一定時間ごとに指定されたコマンドにより、Oracle Clusterware / Oracle Restart の状態を監視します。

障害検出時、設定した回数の Oracle Clusterware / Oracle Restart の状態を再確認します。設定回数内に状態が回復しない場合には、任意のコマンドを実行します。RENS と連携している場合は、プロセス停止時に RENS を経由し、CLUSTERPRO へ障害を通知します。

1.2.6. Oracle Data Guard Broker 構成の REDO ログ転送状況によるスタンバイ・インスタンスの監視と障害時アクション

ApplicationMonitor は、SYSTEM 表領域上の表の更新または一定時間ごとに REDO ログの転送状況からスタンバイ・インスタンスの状態を監視します。

障害検出時、設定した回数の REDO ログの転送状況を再確認します。設定回数内に状態が回復しない場合には、任意のコマンドを実行します。

注意: カスケード・スタンバイ構成、Active Data Guard 遠隔同期構成の場合、または Oracle Data Guard Broker 構成でない場合は、監視できません。

1.2.7. Oracle Data Guard Broker 構成の REDO ログ適用状況によるスタンバイ・インスタンスの監視と障害時アクション

ApplicationMonitor は、SYSTEM 表領域上の表の更新または一定時間ごとに REDO ログの適用状況からスタンバイ・インスタンスの状態を監視します。

障害検出時、任意のコマンドを実行します。

注意: カスケード・スタンバイ構成、Active Data Guard 遠隔同期構成の場合、または Oracle Data Guard Broker 構成でない場合は、監視できません。

1.2.8. Oracle Data Guard Broker 構成の Oracle Data Guard Broker プロセスの状態によるスタンバイ・インスタンスの監視と障害時アクション

ApplicationMonitor は、一定時間ごとに Oracle Data Guard Broker プロセスの状態からスタンバイ・インスタンスの状態を監視します。

障害検出時、設定した回数の Oracle Data Guard Broker プロセスの状態を再確認します。設定回数内に状態が回復しない場合には、任意のコマンドを実行します。

注意: カスケード・スタンバイ構成、Active Data Guard 遠隔同期構成の場合、または Oracle Data Guard Broker 構成でない場合は、監視できません。

1.2.9. Oracle のシステムステート・ダンプの自動採取とシェルスクリプトの実行

ApplicationMonitor は、障害検出時に Oracle のシステムステート・ダンプを自動で採取します。また、シェルスクリプトの実行を指定できます。ApplicationMonitor には、情報採取スクリプトが標準で添付されていますが、任意のシェルスクリプトを実行することも可能です。

1.2.10. Oracle Clusterware / Oracle Restart によるインスタンス / リスナー再起動の待ち合わせ

ApplicationMonitor では、障害検出後の動作として Oracle の環境により、Oracle Clusterware / Oracle Restart によるインスタンス / リスナー再起動を "待ち合わせる" または "待ち合わせない" の選択ができます。

1.2.11. クラスタウェアへの障害通知を障害種別ごとに指定

インスタンス監視およびリスナー監視は、それぞれ障害種別ごとに対象ノード上のクラスタウェアへ "障害を通知する" または "障害を通知しない" の選択ができます。ApplicationMonitor は、クラスタウェアへ障害を通知する場合、障害が発生したノード上の ApplicationMonitor 自身を停止します。

Oracle Clusterware / Oracle Restart の監視モジュールも同様の選択が可能です。

1.2.12. 障害検出時の ApplicationMonitor 停止の設定

ApplicationMonitor では、障害が発生したノード上の ApplicationMonitor 自身に対して、"常に停止"、"停止しない"、"生存ノード数が 1 のとき、停止しない" の 3 種類の選択ができます。また、複数ノードで同時に障害を検出したときに、"障害検出順" または "優先ノード順" の停止順序を選択できます。

1.2.13. Oracle Data Guard Broker によるファスト・スタート・フェイルオーバー(自動フェイルオーバー)の待ち合わせ

ApplicationMonitor では、Oracle Data Guard Broker によるファスト・スタート・フェイルオーバー(以後 自動フェイルオーバーと記載します)を検出し、自動フェイルオーバー処理を待ち合わせるすることができます。

注意: Oracle Data Guard で手動フェイルオーバーおよびスイッチオーバーを実行した場合は、フェイルオーバーを検出できません。

1.2.14. Oracle Automatic Storage Management (ASM) インスタンス停止に伴うデータベース再起動の待ち合わせ

ApplicationMonitor では、Oracle Automatic Storage Management(以後 ASM と記載します)インスタンス停止に伴うデータベースの再起動を "待ち合わせる" または "待ち合わせない" の選択ができます。

1.2.15. 統計情報の採取

ApplicationMonitor は、一定時間ごとにインスタンス監視およびリスナー監視の監視応答時間(以後 監視 TAT と記載します)情報、データベース容量不足の監視情報、システム・グローバル領域(以後 SGA と記載します)の情報を採取します。

対象ノード上の統計情報を "採取する" または "採取しない" の選択ができます。

1.2.16. ApplicationMonitor の一括設定、一括変更

ApplicationMonitor は、複数ノード構成のいずれかのノードで 1 つの設定ファイルを作成することで、一括設定もしくは一括変更が可能です。

注意: Oracle Clusterware / Oracle Restart を監視する場合は、監視するそれぞれのノードで CRS 監視設定ファイルを設定してください。

1.2.17. 監視、採取の一時停止と再開

ApplicationMonitor の suspend (一時停止) 機能を用いることにより、CLUSTERPRO のモニタリソースを停止することなく、Oracle のメンテナンス (データベースのバックアップ、保守など) を行えます。また、メンテナンス後に ApplicationMonitor の resume (再開) 機能を用いることにより、監視および採取を再開できます。

注意: ApplicationMonitor のモニタは、起動時に Oracle ライブラリをリンクしています。suspend 機能にて一時停止を行った場合、Oracle データベースに対するアクセスは停止しますが、モニタプロセス自体は動作しているため、Oracle パッチ適用の際は、ApplicationMonitor を停止してください。

注意: Oracle Data Guard で手動フェイルオーバーまたはスイッチオーバーを実施する場合は、実施前に ApplicationMonitor の suspend 機能を使用して監視および採取を一時停止してください。

フェイルオーバーまたはスイッチオーバーが完了後、apply 機能を使用して ApplicationMonitor を再構成し、resume 機能を使用して監視および採取を再開してください。

1.2.18. 各モニタ状態、各モニタ設定の表示と最大監視応答時間の表示

ApplicationMonitor の管理コマンドを用いることにより、各監視モニタおよび採取モニタの状態や現在の設定値を表示します。これにより、ApplicationMonitor の状態を把握できます。また、各監視モニタの最大監視応答時間の表示により、ApplicationMonitor が監視対象のストールと認識するまでの時間の調整目安となり、障害検出からフェイルオーバーさせるまでの時間短縮につながります。

1.2.19. 設定ファイルのテンプレート作成機能

ApplicationMonitor は、Oracle の環境変数を自動で収集し、ApplicationMonitor の設定ファイルのテンプレートを作成できます。

作成した設定ファイルのテンプレートを用いることで、簡易設定ファイルが容易に作成可能です。

設定ファイルのテンプレート作成機能の詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 1.2 for Linux 設定ファイルテンプレート作成コマンド ユーザーズガイド』を参照してください。

1.2.20. 複数インスタンス監視

ApplicationMonitor は、同一ノード上に複数のインスタンスが存在するような構成に対応します。複数のインスタンスのグループ(現用待機構成もしくは RAC 構成で 1 つのグループ)ごとに一意の構成番号を割り振り、ApplicationMonitor を CLUSTERPRO のモニタリソースの監視対象として複数登録することで実現します。

1.2.21. 監視モニタ、採取モニタの単独起動

リスナー監視モニタ、表領域監視モニタ、ディスク領域監視モニタおよび統計情報採取モニタは、それぞれ単独で起動できます。これにより、インスタンス監視とは別に CLUSTERPRO のモニタリソースの監視対象として登録でき、より柔軟な構成が可能です。

1.2.22. Oracle エラーのフィルタリング設定

ApplicationMonitor は、監視中に発生する Oracle エラーごとに、障害として扱う(エラー) / 扱わない(許可エラー)を指定できます。これにより、CLUSTERPRO のフェイルオーバーグループの切り替えを行う / 行わないを運用形態に合わせて指定できます。

1.2.23. Oracle のディザスタ・リカバリ構成のバックアップサイトの監視

ApplicationMonitor は、Oracle Data Guard による災害対策(以後 ディザスタ・リカバリ と記載します)構成に対応します。プライマリ・データベース(本番データベース)の監視に加え、スタンバイ・データベースの監視が可能です。

スタンバイ・データベースでは、Oracle の起動状態に合わせた監視を行います。

1 製品の概要と特長

2. 製品のソフトウェア構成

ApplicationMonitor は、以下の監視プロセス、採取プロセス、管理インタフェース、設定ファイルから構成します。

◆ 監視プロセス

- ・ モニタ制御デーモン (oramond)
- ・ インスタンス監視モニタ (instmond) 1
- ・ リスナー監視モニタ (lsnrmond)
- ・ 表領域監視モニタ (tschkmond) 1
- ・ ディスク領域監視モニタ (fdsmond) 1
- ・ スタンバイデータベース監視モニタ (stbdbmond) 1、2
- ・ アラートログ監視モニタ (altmond) 1
- ・ CRS 監視モニタ (crsmond) 3

◆ 採取プロセス

- ・ 統計情報採取モニタ (statsmond) 1
- ・ システムステートダンプ採取デーモン (ssdump) 1

◆ 管理インタフェース

- ・ 管理コマンド (oraadmin)
- ・ 構成情報管理サーバ (oraconfd)

◆ 設定ファイル

- ・ 設定ファイル (oramond.conf)
- ・ 構成情報ファイル (oramond<構成番号>.bin) 4
- ・ CRS 監視設定ファイル (crsmond.conf) 3
- ・ フィルタリング設定ファイル (inst.conf、lsnr.conf、tschk.conf、fds.conf、stbdb.conf、stats.conf、ssdump.conf)
- ・ RENS 連携用辞書ファイル (haam.dict、haam_stat.dict、haam_stat_map.dict)

- 1 監視対象とする Oracle のバージョンにより、実際に動作するプロセス名は、"_11G"、"_11GR2" または "_12C" が末尾に付きますが、本文中では省略しています。
- 2 Oracle Data Guard Broker 構成が対象です。
- 3 RAC 環境、Oracle Restart 環境が対象です。
- 4 構成番号は、設定ファイル (oramond.conf) の COMPONENT_ID パラメータの値を自動で付加します。COMPONENT_ID パラメータを省略した場合は、0 となります。

2 製品のソフトウェア構成

ApplicationMonitor のソフトウェア構成を CLUSTERPRO、RENS や Oracle(RAC) との関連性と合わせて示すと図 2-1 のようになります。

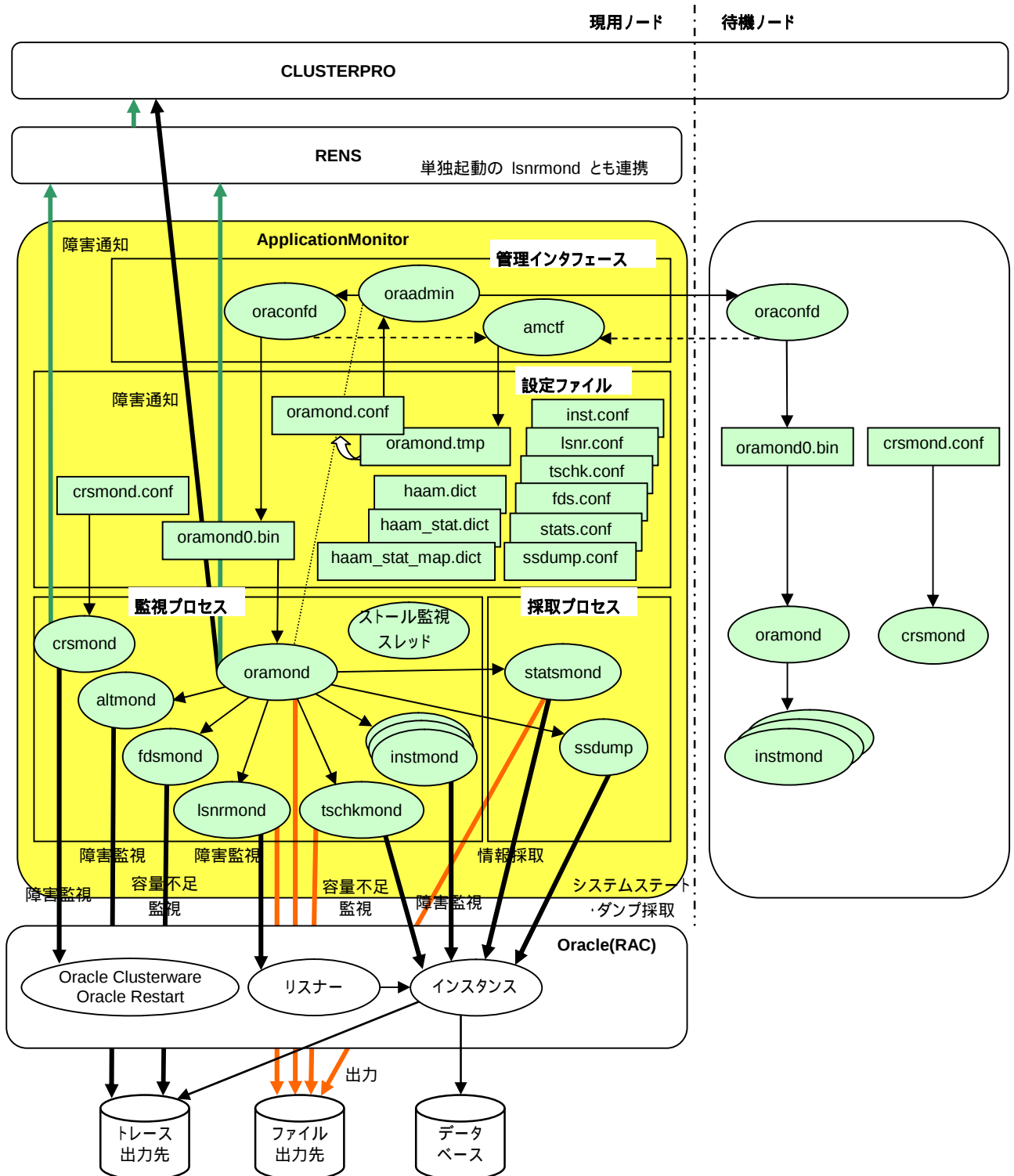


図 2-1 ソフトウェア構成

Oracle Data Guard Broker 構成のスタンバイ・インスタンス監視との関連性を示すと図 2-2 のようになります。

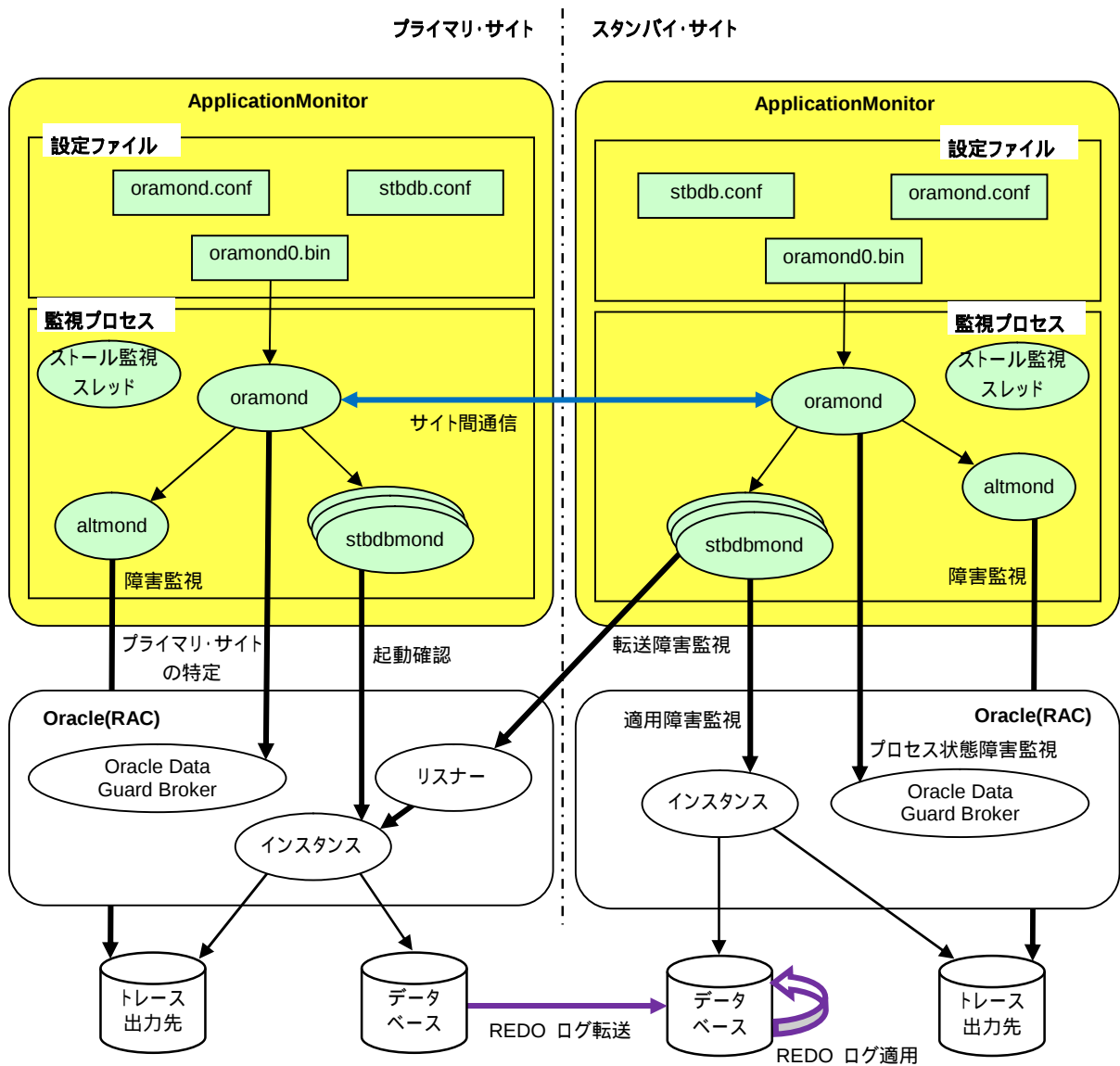


図 2-2 ソフトウェア構成 (Oracle Data Guard Broker 構成のスタンバイ・インスタンス監視との関連性)

2.1. 監視プロセス

2.1.1. モニタ制御デーモン (oramond)

モニタ制御デーモンは、ApplicationMonitor の中心となるデーモンです。

インスタンス監視モニタ (instmond)、リスナー監視モニタ (lsnrmond)、表領域監視モニタ (tschkmond)、ディスク領域監視モニタ (fdsmond)、スタンバイデータベース監視モニタ (stbldbmond)、アラートログ監視モニタ (altmond)、統計情報採取モニタ (statsmond)、システムステートダンプ採取デーモン (ssdump) の起動と管理を行います。また、インスタンス監視モニタの監視 TAT 情報の採取およびファイルへの出力を行います。

現用待機構成時のモニタ制御デーモンは、インスタンス監視モニタ、リスナー監視モニタと通信を行うことにより、Oracle の状態を把握します。障害が発生した場合は、システムステートダンプ採取デーモンへ Oracle のシステムステート・ダンプの採取を要求、同時にシェルスクリプトを実行します。なお、監視モニタとの通信が切断された場合は、監視モニタを再起動し、Oracle の監視を継続します。

RAC 構成時のモニタ制御デーモンは、各ノード上で動作し、それぞれのノードで Oracle の状態を把握します。また、ノード間通信を行うことで各ノード上のモニタ制御デーモン同士が情報を交換し、お互いの Oracle の状態を把握します。1 つまたは複数のノードで RAC に障害が発生した場合は、すべてのノード上でシステムステートダンプ採取デーモンへ Oracle のシステムステート・ダンプの採取を要求し、同時にシェルスクリプトを実行します。

システムステート・ダンプの採取およびシェルスクリプト実行後は、障害が発生したノードのうち、モニタ制御デーモン自身を停止するノードを選択します。モニタ制御デーモン自身を停止することで CLUSTERPRO へ障害を通知します。RENS 連携時には、RENS に対しても DOWN 通知を行い、RENS 経由でも CLUSTERPRO へ障害を通知します。

なお、Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境では、Oracle Clusterware / Oracle Restart での再起動処理を有効にしたままでも、Oracle Clusterware / Oracle Restart による Oracle 再起動動作の結果も含め、最終的なモニタ制御デーモン停止を判断できます。

モニタ制御デーモンには、自身のストールを監視するストール監視スレッドが存在します。モニタ制御デーモンと監視スレッドは、共有領域を使い、時刻情報を更新 / 参照することでストールを監視します。モニタ制御デーモンがストールして時刻情報が更新されない場合、監視用スレッドは、モニタ制御デーモンのストールを検出し、異常終了します。

スタンバイ・インスタンス監視時のモニタ制御デーモンは、ノード間通信を行うことで各ノード上

のモニタ制御デーモン同士が情報を交換し、お互いの Oracle の状態を把握します。また、SYSTEM 表領域上の表の更新または一定時間ごとに Oracle の Data Guard コマンドを用いて、Oracle Data Guard Broker プロセスの状態を監視します。Oracle Data Guard Broker プロセスに障害が発生した場合は、指定されたシェルスクリプトを実行します。

Oracle Data Guard Broker のプロセスの状態監視は、スタンバイ・サイトのデータベースがフィジカル・スタンバイ・データベースの場合、起動確認のみ行います。

注意: Oracle Clusterware / Oracle Restart によるリソース再起動中に、起動中リソースを強制的に停止 (SIGKILL 送信など) すると、再起動の上限 (RESTART_ATTEMPTS) に達していなくても、それ以降の Oracle Clusterware / Oracle Restart は、リソースを再起動しなくなります。

注意: モニタ制御デーモン動作中にシステム時間を変更した場合、正常な監視が行われない可能性があります。システム時間を変更する場合は、suspend 機能の監視スレッド停止オプションを付与し、監視を一時停止した上で行ってください。

suspend 機能の監視スレッド停止オプションの詳細は、「6 コマンドリファレンス」を参照してください。

また、監視スレッドを停止した場合でも、大幅な未来または過去時間への時刻補正は、動作保証対象外となります。

注意: Oracle Data Guard Broker プロセスの状態監視は、カスケード・スタンバイ構成、Active Data Guard 遠隔同期構成の場合、または Oracle Data Guard Broker 構成でない場合は、監視できません。

2.1.2. インスタンス監視モニタ (instmond)

インスタンス監視モニタは、インスタンス障害を監視するデーモンです。

インスタンスへ接続し、一定時間ごとに Oracle バックグラウンドプロセス単位で V\$ 表の参照、SYSTEM 表領域上の表の更新などを行います。また、ユーザー指定表を参照します。

Oracle 12c のマルチテナント構成の場合は、指定されたインスタンスのマルチテナント・コンテナ・データベース (以後 CDB と記載します) またはプラガブル・データベース (以後 PDB と記載します) へ接続し、一定時間ごとに Oracle バックグラウンドプロセス単位で V\$ 表の参照、SYSTEM 表領域上の表の更新などを行います。

インスタンス障害検出時は、モニタ制御デーモン (oramond) に障害を通知します。

なお、データベースが読取り専用アクセスでオープンされている場合およびデータベースがマウント状態の場合は、インスタンスに対する起動確認のみ行います。

2.1.3. リスナー監視モニタ (lsnrmond)

リスナー監視モニタは、リスナー障害を監視するデーモンです。

一定時間ごとに Oracle の tnsping コマンドを用いてネット・サービスの有無を確認します。また、監視 TAT 情報の採取およびファイルへの出力を行います。

リスナー障害検出時は、リスナーの再起動を行います。リスナーの再起動に失敗した場合は、モニタ制御デーモン (oramond) に障害を通知します。

なお、Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境では、Oracle Clusterware / Oracle Restart での再起動処理を有効にしたままでも、Oracle Clusterware / Oracle Restart による Oracle 再起動動作の結果も含め、最終的なリスナー監視モニタ停止を判断できます。

注意： Oracle Clusterware / Oracle Restart によるリソース再起動中に、起動中リソースを強制的に停止 (SIGKILL 送信など) すると、再起動の上限 (RESTART_ATTEMPTS) に達していなくても、それ以降の Oracle Clusterware / Oracle Restart は、リソースを再起動しなくなります。

2.1.4. 表領域監視モニタ (tschkmond)

表領域監視モニタは、データベース領域の容量不足を監視するデーモンです。

一定時間ごとにインスタンスへ接続し、表領域の最大使用可能エクステントサイズと空き容量率を監視します。また、表領域情報の採取およびファイルへの出力を行います。

Oracle 12c のマルチテナント構成の場合は、指定されたインスタンスの CDB または PDB へ一定時間ごとに接続し、表領域の最大使用可能エクステントサイズと空き容量率を監視します。

設定されたしきい値を下回った場合、設定ファイル (oramond.conf) に設定されたコマンドを実行します。

なお、データベースがマウント状態の場合は、インスタンスに対する起動確認のみ行います。

注意： ローカル管理されている一時表領域の監視はできません。ローカル管理されている一時表領域の監視は、Oracle 独自の機能 (Enterprise Manager など) を使用してください。

2.1.5. ディスク領域監視モニタ (fdsmond)

ディスク領域監視モニタは、ローカルディスク領域の容量不足を監視するデーモンです。

一定時間ごとにインスタンスへ接続し、設定されているユーザー・トレース出力先、バックグラウンド・プロセス・トレース出力先、コアファイル出力先および監査証跡ファイル出力先を取得し、該当ディスクの空き容量率を監視します。また、アーカイブログモードで運用されている場合は、アーカイブログファイル出力先の空き容量率も監視します。

設定されたしきい値を下回った場合、設定ファイル (oramond.conf) に設定されたコマンドを実行します。

なお、データベースがマウント状態の際、任意の Oracle ユーザーによる監視を行う(設定ファイルに ORACLE_USER / ORACLE_PASS パラメータが指定されている)場合は、インスタンスに対する起動確認のみ行います。

2.1.6. スタンバイデータベース監視モニタ (stbdbmond)

スタンバイデータベース監視モニタは、REDO ログの転送、適用状況からスタンバイ・インスタンスを監視するデーモンです。

REDO ログの転送状況の監視は、SYSTEM 表領域上の表の更新または一定時間ごとにプライマリ・サイトのリスナーを経由してプライマリ・インスタンスへ接続し、スタンバイ・サイトへの REDO ログ転送の状況を監視します。

REDO ログの適用状況の監視は、SYSTEM 表領域上の表の更新または一定時間ごとにスタンバイ・インスタンスへ接続し、スタンバイ・サイトに適用された REDO ログの状況を監視します。

Oracle 12c のマルチテナント構成の場合、SYSTEM 表領域上の表の更新または一定時間ごとに、指定されたスタンバイ・サイトのスタンバイ・インスタンスの CDB または PDB へ接続し、スタンバイ・サイトに適用された REDO ログの状況を監視します。

障害検出時は、再確認を行い、設定ファイル (oramond.conf) に設定されたコマンドを実行します。

なお、REDO ログの転送監視は、スタンバイ・サイトのデータベースがフィジカル・スタンバイ・データベースまたはスタンバイ・サイトのデータ保護モードが最大パフォーマンスモードの場合、起動確認のみ行います。

REDO ログの適用監視は、スタンバイ・サイトのデータベースがフィジカル・スタンバイ・データベースまたはスナップショット・スタンバイ・データベースの場合、起動確認のみ行います。

注意: カスケード・スタンバイ構成、Active Data Guard 遠隔同期構成の場合、または Oracle Data Guard Broker 構成でない場合は、監視できません。

2.1.7. アラートログ監視モニタ (altmond)

アラートログ監視モニタは、Oracle が監視可能な状態か判断するデーモンです。

Oracle の起動時またはノード構成変更時など、一時的に Oracle にアクセスできない時間帯が発生します。この時間帯に監視を中断するように Oracle のログから判断し、有効な監視開始タイミングを決定します。

2.1.8. CRS 監視モニタ (crsmond)

CRS 監視モニタは、Oracle 11g を監視対象とする場合にクラスタ・レディ・サービス(CRS)・デーモン(以後 crsd と記載します)、Oracle 11g R2 以降を監視対象とする場合に Oracle 高可用性サービス・デーモン(以後 ohasd と記載します)の無応答障害を監視するデーモンです。

CRS 監視設定ファイル (crsmond.conf) に指定されたコマンドを一定時間ごとに実行することにより、crsd/ohasd の正常動作を確認します。

crsd/ohasd の障害検出時は、syslog にメッセージを出力します。crsd/ohasd プロセスが存在した場合、プロセスを強制終了することにより、init(8) による crsd/ohasd の再起動を促します。また、CRS 監視設定ファイルに指定されたコマンドを実行します。

注意: CRS 監視モニタは、モニタ制御デーモン (oramond) により制御されません。Oracle Clusterware / Oracle Restart を監視する場合、設定ファイル (oramond.conf) ではなく、CRS 監視設定ファイルの設定を行い、別途 CRS 監視モニタを起動してください。

2.2. 採取プロセス

2.2.1. 統計情報採取モニタ (statsmond)

統計情報採取モニタは、統計情報を採取するデーモンです。

一定時間ごとにインスタンスへ接続し、SGA 情報を採取します。採取した情報は、ファイルに出力します。

Oracle 12c のマルチテナント構成の場合は、指定されたインスタンスの CDB または PDB へ一定時間ごとに接続し、SGA 情報を採取します。

なお、データベースがマウント状態の際、任意の Oracle ユーザーによる採取を行う(設定ファイル (oramond.conf) に ORACLE_USER / ORACLE_PASS パラメータが指定されている)場合、またはインスタンスの PDB へ接続した際に PDB がマウント状態だった場合は、インスタンスに対する起動確認のみ行います。

2.2.2. システムステートダンプ採取デーモン (ssdump)

システムステートダンプ採取デーモンは、モニタ制御デーモン (oramond) の要求により、Oracle のシステムステート・ダンプを採取するデーモンです。

常にインスタンスとの接続状態を保ち、障害による接続エラーの場合にも、Oracle のシステムステート・ダンプを採取できます。

Oracle 12c のマルチテナント構成の場合は、指定されたインスタンスの CDB または PDB へ接続し、Oracle のシステムステート・ダンプを採取します。

なお、データベースがマウント状態の際、任意の Oracle ユーザーによる採取を行う(設定ファイル (oramond.conf) に ORACLE_USER / ORACLE_PASS パラメータが指定されている場合)は、インスタンスに対する起動確認のみ行います。

2.3. 管理インタフェース

2.3.1. 管理コマンド (oraadmin)

管理コマンドは、ApplicationMonitor の設定、状態表示などを行うコマンドです。以下のリクエスト(コマンド)がサポートされています。

リクエスト名	機能
show	監視モニタおよび採取モニタの状態を表示します。
suspend	監視モニタおよび採取モニタを一時停止します。
resume	監視モニタおよび採取モニタを再開します。
check	設定ファイル (oramond.conf) の書式および設定内容が有効か確認します。ノード上で Oracle が動作中の場合、テーブル名、表領域名、ユーザー名、パスワードについても確認します。
apply	設定内容を適用します。また、設定ファイル (oramond.conf) から構成情報ファイル (oramond<構成番号>.bin) を生成し、各ノードに配布します。
delete	構成情報ファイル (oramond<構成番号>.bin) を削除します。
view	構成情報ファイル (oramond<構成番号>.bin) の内容を表示します。

管理コマンドの詳細は、「6 コマンドリファレンス」を参照してください。

2.3.2. 構成情報管理サーバ (oraconfd)

構成情報管理サーバは、各ノード上の構成情報ファイル (oramond<構成番号>.bin) を管理するデーモンです。

管理コマンド (oraadmin) からのリクエストにしたがい、構成情報ファイルの設定内容の確認、ファイルの作成、ファイルの削除を行います。

2.4. 設定ファイル

2.4.1. 設定ファイル (oramond.conf)

設定ファイルは、ApplicationMonitor の動作に必要な環境、構成など各種パラメータを定義するためのテキストファイルです。マニュアル上では、設定ファイル (oramond.conf) と記載していますが、任意の名称でファイルを作成することも可能です。

設定ファイルの詳細は、「3.2 設定ファイル (oramond.conf) の作成と適用」を参照してください。

2.4.2. 構成情報ファイル (oramond<構成番号>.bin)

構成情報ファイルは、設定ファイル (oramond.conf) から管理コマンド (oraadmin) および構成情報管理サーバ (oraconfd) によって生成されたバイナリファイルです。ApplicationMonitor が動作する各ノード上に配布されます。構成番号は、設定ファイルの COMPONENT_ID パラメータの番号が付加されます。

監視プロセスおよび採取プロセスは、この構成情報ファイルを元に動作します。

2.4.3. CRS 監視設定ファイル (crsmond.conf)

CRS 監視設定ファイルは、CRS 監視モニタの動作に必要な環境、構成など各種パラメータを定義するためのテキストファイルです。

CRS 監視設定ファイルの詳細は、「3.9 CRS 監視設定ファイル (crsmond.conf) の作成」を参照してください。

2.4.4. フィルタリング設定ファイル (inst.conf、lsnr.conf、tschk.conf、fds.conf、stbdb.conf、stats.conf、ssdump.conf)

フィルタリング設定ファイルは、各モニタの監視時または採取時に検出する Oracle エラーを、エラーとして扱うか、またはエラーとして扱わない(許可エラー)かを設定するためのテキストファイルです。

フィルタリング設定ファイルの詳細は、「3.3 Oracle エラーのフィルタリング設定」を参照してください。

2.4.5. RENS 連携用辞書ファイル (haam.dict、haam_stat.dict、haam_stat_map.dict)

haam.dict (イベント辞書ファイル)、haam_stat.dict (ステータス値定義ファイル)、haam_stat_map.dict (ステータス値マッピング定義ファイル) は、RENS 連携時に使用されるファイルです。

このファイルの設定変更は、必要ありません。

3. 製品の設定

ApplicationMonitor を利用するためには、以下の設定が必要です。

- ◆ インスタンス監視表の作成
- ◆ 設定ファイル (oramond.conf) の作成と適用
- ◆ Oracle エラーのフィルタリング設定
- ◆ 情報採取スクリプト (oradiag) の設定
- ◆ リスナー監視モニタの単独起動
- ◆ 表領域監視モニタの単独起動
- ◆ ディスク領域監視モニタの単独起動
- ◆ 統計情報採取モニタの単独起動
- ◆ CRS 監視設定ファイル (crsmond.conf) の作成

これらの設定方法について説明します。

3.1. インスタンス監視表の作成

ApplicationMonitor のインスタンス監視を利用するためには、監視対象となるインスタンスの共有データベース上にインスタンス監視を行うための表(以後 インスタンス監視表 と記載します)を作成する必要があります。

ApplicationMonitor に添付されているシェルスクリプトを実行することで作成できますので、**運用開始前にインスタンス監視表を必ず作成してください。**

なお、ご使用の構成によって作成方法が異なります。以下の表とご使用の構成を照らし合わせ、それぞれの方法で作成してください。

Oracle のバージョン	監視対象の Oracle の構成	インスタンス監視表の作成方法
Oracle 11g (11.1.0、11.2.0)	すべての構成	3.1.1 章参照
Oracle 12c	マルチテナント構成	3.1.2 章参照
Oracle 12c	マルチテナント構成以外	3.1.1 章参照

3.1.1. インスタンス監視表の作成方法 (Oracle 11g (11.1.0、11.2.0) の場合 / Oracle 12c のマルチテナント構成以外の場合)

Oracle 12c のマルチテナント構成時以外のインスタンス監視表の作成方法は、以下のとおりです。

1 ビューの作成

監視を行うためのビューを作成します。

Oracle が起動した状態で、以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh create xview
```

2 監視用プロファイルの作成

SYS ユーザーで監視を行う場合は、実行する必要はありません。

SYS ユーザー以外のユーザーで監視を行う場合、Oracle 11g より、パスワードの有効期限がデフォルトで 180 日に設定されます。パスワード有効期限切れにより、接続不可となる状況を防ぐため、パスワードの有効期限が無期限となるプロファイルを作成します。

Oracle が起動した状態で、以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh create profile
```

3 監視用 Oracle 接続ユーザーの作成

SYS ユーザーで監視を行う場合は、実行する必要はありません。

SYS ユーザー以外のユーザーで監視を行う場合、監視用の Oracle 接続ユーザーを作成してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで Oracle に接続してください。

【注意】2 で作成した監視用プロファイルを指定してください。

```
$ sqlplus "/ as sysdba"
SQL> CREATE USER <Oracle 接続ユーザー>
      IDENTIFIED BY <Oracle 接続パスワード>
      DEFAULT TABLESPACE SYSTEM
      QUOTA UNLIMITED ON SYSTEM
      PROFILE HAAM_PROFILE;
```

Oracle 接続ユーザーの確認方法および出力例は、以下のとおりです。

```
$ sqlplus "/ as sysdba"
SQL> SELECT USERNAME, DEFAULT_TABLESPACE, PROFILE FROM DBA_USERS;
```

USERNAME	DEFAULT_TABLESPACE	PROFILE
SYS	SYSTEM	DEFAULT
SYSTEM	SYSTEM	DEFAULT
SCOTT	SYSTEM	DEFAULT
HAAM	SYSTEM	HAAM_PROFILE

4 監視表の作成

監視表を作成します。

Oracle が起動した状態で、以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

【注意】Oracle 接続ユーザーを指定する場合、大文字 / 小文字に注意してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh create table <XXX>[ <YYY>]
```

XXX: 監視対象ノード数を指定します。省略はできません。

YYY: 3 で作成した Oracle 接続ユーザーを指定します。

SYS ユーザーで監視を行う場合は、省略できます。

【注意】Oracle Data Guard Broker 構成で、プライマリ・サイトとスタンバイ・サイトのノード数が異なる場合、以下の (例) のように全サイトの最大ノード数分の監視表を作成してください。

(例) プライマリ・サイトが 2 ノード、スタンバイ・サイトが 3 ノードの場合、3 ノード分の監視表を作成する。

【注意】監視対象ノード数または Oracle 接続ユーザーを変更する場合は、監視表を再作成してください。

5 作成した監視表の確認

監視表作成時、確認処理が自動的に実施されます。

個別に確認を行う場合は、以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

【注意】Oracle 接続ユーザーを指定する場合、大文字 / 小文字に注意してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh verify <XXX>[ <YYY>]
```

XXX: 監視対象ノード数を指定します。省略はできません。

YYY: 3 で作成した Oracle 接続ユーザーを指定します。

SYS ユーザーで監視を行う場合は、省略できます。

5.1 TABLE オブジェクトの確認

監視対象ノード数分のレコードが検索され、TABLESPACE_NAME 列が SYSTEM であることを確認してください。

出力例は、以下のとおりです。

TABLE_NAME	TABLESPACE_NAME
-----	-----
HAAM_TIME_CHK0	SYSTEM
HAAM_TIME_CHK1	SYSTEM

5.2 SEQUENCE オブジェクトの確認

監視対象ノード数分のレコードが検索され、CYCLE_FLAG 列が Y であることを確認してください。また、CACHE_SIZE 列が 0 であること、ならびに FNO 列および BNO 列が作成したすべての SEQUENCE で同一であることを確認してください。

出力例は、以下のとおりです。

OBJECT_NAME	CYCLE_FLAG	CACHE_SIZE	FNO	BNO
-----	-----	-----	-----	-----
HAAM_TEST_SEQ0	Y	0	8	1083
HAAM_TEST_SEQ1	Y	0	8	1083

5.3 監視表の再作成

監視表が正しく作成されていない場合は、以下のシェルスクリプトを実行して監視表を削除したのち、4 を実行して監視表を再作成してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

【注意】Oracle 接続ユーザーを指定する場合、大文字 / 小文字に注意してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh drop table [<YYY>]
```

YYY :3 で作成した Oracle 接続ユーザーを指定します。

SYS ユーザーで監視を行う場合は、省略できます。

6 権限の付与

SYS ユーザーで監視を行う場合は、実行する必要はありません。

SYS ユーザー以外のユーザーで監視を行う場合、3 で作成した Oracle 接続ユーザーに、監視に必要な権限を付与します。

以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

【注意】Oracle 接続ユーザーを指定する場合、大文字 / 小文字に注意してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh grant <YYY>
```

YYY:3 で作成した Oracle 接続ユーザー名を指定します。省略はできません。

【注意】Oracle 接続ユーザーを変更する場合は、権限を付与し直してください。

3.1.2. インスタンス監視表の作成方法 (Oracle 12c のマルチテナント構成の場合)

Oracle 12c のマルチテナント構成時の場合、インスタンス監視表の作成パターンは、以下のイメージ図のように 6 つあります。

	インスタンス監視表を作成するデータベース	監視用 Oracle ユーザー	インスタンス監視表を所有するユーザー
パターン 1	CDB	SYS ユーザー	SYS ユーザー
パターン 2	CDB	共通ユーザー	SYS ユーザー
パターン 3	PDB	SYS ユーザー	SYS ユーザー
パターン 4	PDB	SYS ユーザー	ローカル・ユーザー
パターン 5	PDB	共通ユーザー	SYS ユーザー
パターン 6	PDB	共通ユーザー	ローカル・ユーザー

共通ユーザーとは、C## または c## で始まる名前のすべての PDB に共通するデータベース・ユーザーを指し、ローカル・ユーザーとは、単一の PDB にのみ存在するデータベース・ユーザーを指します。

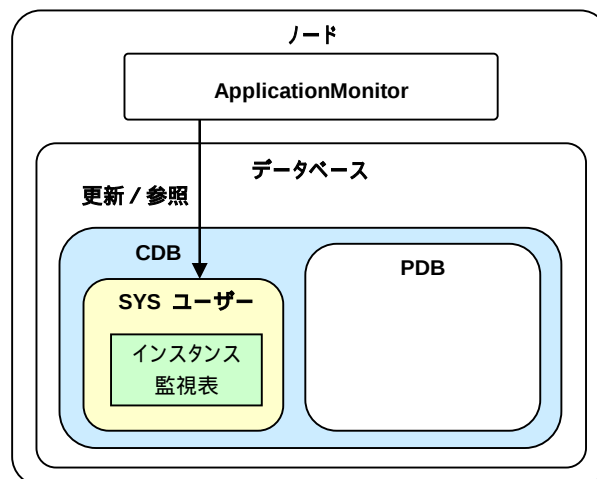


図 3-1 インスタンス監視表の作成 (パターン 1)

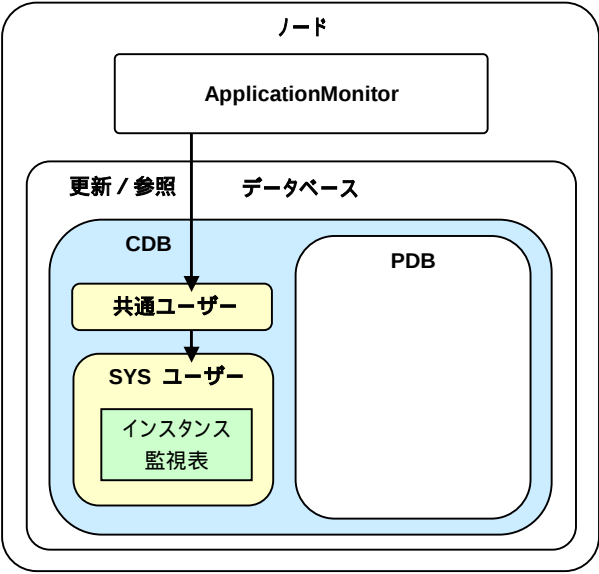


図 3-2 インスタンス監視表の作成 (パターン 2)

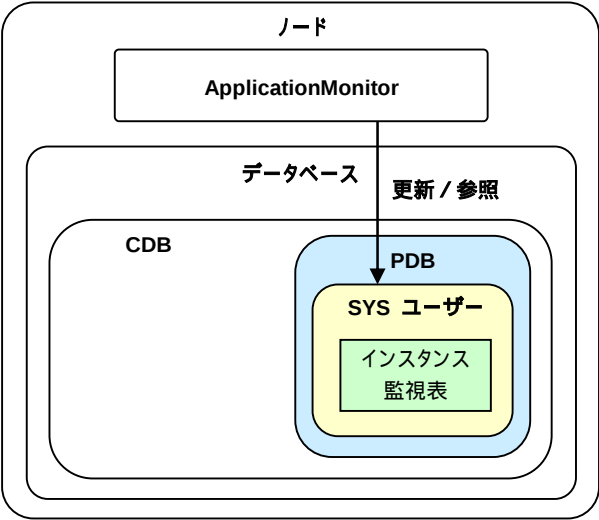


図 3-3 インスタンス監視表の作成 (パターン 3)

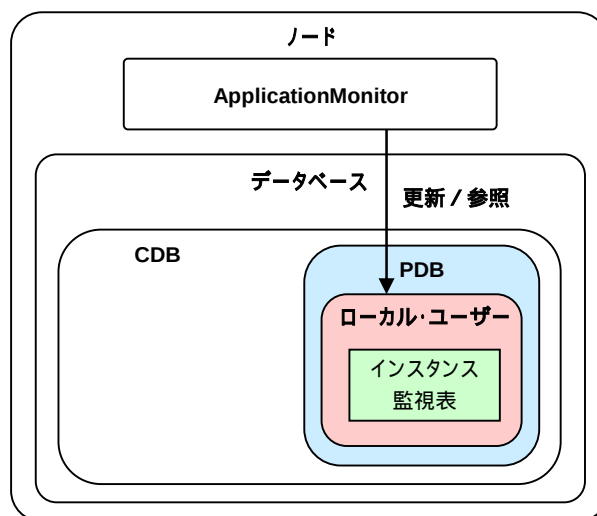


図 3-4 インスタンス監視表の作成 (パターン 4)

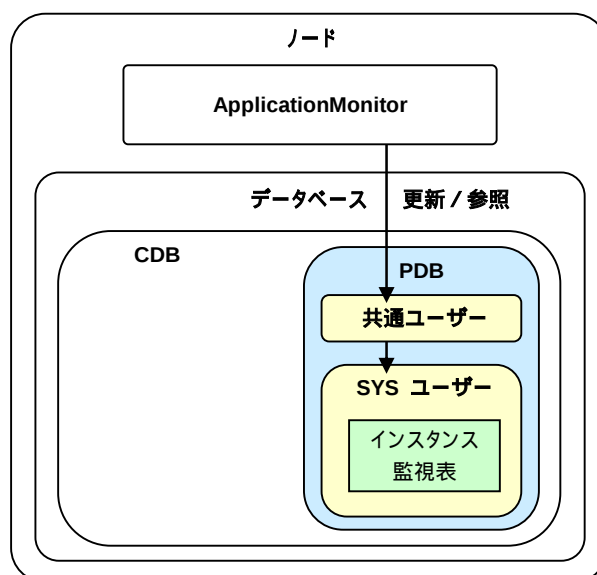


図 3-5 インスタンス監視表の作成 (パターン 5)

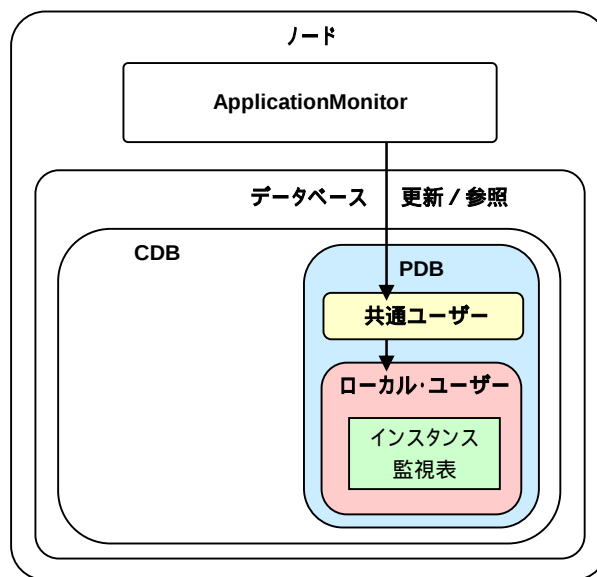


図 3-6 インスタンス監視表の作成 (パターン 6)

Oracle 12c のマルチテナント構成時のインスタンス監視表の作成方法は、以下のとおりです。

ローカル・ユーザーにインスタンス監視表を作成する場合、事前にローカル・ユーザーを PDB 上に作成してください。

ローカル・ユーザーの詳細は、Oracle のマニュアル『Oracle Database セキュリティ・ガイド』を参照してください。

1 ビューの作成

監視を行うためのビューを CDB 上に作成します。

Oracle が起動した状態で、以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh create xview
```

PDB 上にインスタンス監視表を作成する場合は、ビューを PDB 上にも作成します。

上記に引き続き、Oracle が起動した状態で、以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh create xview <ZZZ>
```

ZZZ: 接続先データベース名 (PDB 名) を指定します。省略はできません。

【注意】PDB 名を変更する場合は、ビューを再作成してください。

2 監視用プロファイルの作成

SYS ユーザーで監視を行う場合は、実行する必要はありません。

SYS ユーザー以外のユーザーで監視を行う場合、Oracle 11g より、パスワードの有効期限がデフォルトで 180 日に設定されます。パスワード有効期限切れにより、接続不可となる状況を防ぐため、パスワードの有効期限が無期限となるプロファイルを作成します。

Oracle が起動した状態で、以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh create profile
```

3 監視用 Oracle 接続ユーザー (共通ユーザー) の作成

SYS ユーザーで監視を行う場合は、実行する必要はありません。

SYS ユーザー以外のユーザーで監視を行う場合、監視用の Oracle 接続ユーザー (共通ユーザー) を作成してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで Oracle に接続してください。

【注意】2 で作成した監視用プロファイルを指定してください。

【注意】Oracle 接続ユーザー (共通ユーザー) を作成する場合、ユーザー名が C## または c## で始まる必要があります。

```
$ sqlplus "/ as sysdba"
SQL> CREATE USER <Oracle 接続ユーザー (共通ユーザー)>
IDENTIFIED BY <Oracle 接続パスワード>
DEFAULT TABLESPACE SYSTEM
QUOTA UNLIMITED ON SYSTEM
PROFILE C##HAAM_PROFILE;
```

Oracle 接続ユーザー (共通ユーザー) の確認方法および出力例は、以下のとおりです。

```
$ sqlplus "/ as sysdba"
SQL> SELECT USERNAME, DEFAULT_TABLESPACE, PROFILE FROM DBA_USERS;
```

USERNAME	DEFAULT_TABLESPACE	PROFILE
SYS	SYSTEM	DEFAULT
SYSTEM	SYSTEM	DEFAULT
SCOTT	SYSTEM	DEFAULT
C##HAAM	SYSTEM	C##HAAM_PROFILE

4 監視表の作成

監視表を作成します。

Oracle が起動した状態で、以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

【注意】監視表を所有するユーザー名を指定する場合、大文字 / 小文字に注意してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh create table <XXX>[ <YYY>[ <ZZZ>]]
```

XXX: 監視対象ノード数を指定します。省略はできません。

YYY: 監視表を所有するユーザー名(以後 スキーマ と記載します)を指定します。

監視表を CDB 上に作成する場合は、省略できます。

監視表を PDB 上に作成する場合は、SYS またはローカル・ユーザーを指定してください。

ZZZ: 接続先データベース名を指定します。

監視表を CDB 上に作成する場合は、省略できます。

監視表を PDB 上に作成する場合は、PDB 名を指定してください。

【注意】Oracle Data Guard Broker 構成で、プライマリ・サイトとスタンバイ・サイトのノード数が異なる場合、以下の(例)のように全サイトの最大ノード数分の監視表を作成してください。

(例) プライマリ・サイトが 2 ノード、スタンバイ・サイトが 3 ノードの場合、3 ノード分の監視表を作成する。

【注意】監視対象ノード数、監視表のスキーマまたは PDB 名を変更する場合は、監視表を再作成してください。

5 作成した監視表の確認

監視表作成時、確認処理が自動的に実施されます。

個別に確認を行う場合は、以下のシェルスクリプトを実行してください。

[注意] OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

[注意] 監視表のスキーマを指定する場合、大文字 / 小文字に注意してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh verify <XXX>[ <YYY>[ <ZZZ>]]
```

XXX: 監視対象ノード数を指定します。省略はできません。

YYY: 監視表のスキーマを指定します。

監視表を CDB 上に作成した場合は、省略できます。

監視表を PDB 上に作成した場合は、SYS またはローカル・ユーザーを指定してください。

ZZZ: 接続先データベース名を指定します。

監視表を CDB 上に作成した場合は、省略できます。

監視表を PDB 上に作成した場合は、PDB 名を指定してください。

5.1 TABLE オブジェクトの確認

監視対象ノード数分のレコードが検索され、TABLESPACE_NAME 列が SYSTEM であることを確認してください。

出力例は、以下のとおりです。

TABLE_NAME	TABLESPACE_NAME
-----	-----
HAAM_TIME_CHK0	SYSTEM
HAAM_TIME_CHK1	SYSTEM

5.2 SEQUENCE オブジェクトの確認

監視対象ノード数分のレコードが検索され、CYCLE_FLAG 列が Y であることを確認してください。また、CACHE_SIZE 列が 0 であること、ならびに FNO 列および BNO 列が作成したすべての SEQUENCE で同一であることを確認してください。

出力例は、以下のとおりです。

OBJECT_NAME	CYCLE_FLAG	CACHE_SIZE	FNO	BNO
-----	-----	-----	-----	-----
HAAM_TEST_SEQ0	Y	0	8	1083
HAAM_TEST_SEQ1	Y	0	8	1083

5.3 監視表の再作成

監視表が正しく作成されていない場合は、以下のシェルスクリプトを実行して監視表を削除したのち、4 を実行して監視表を再作成してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

【注意】監視表のスキーマを指定する場合、大文字 / 小文字に注意してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh drop table[ <YYY>[ <ZZZ>]]
```

YYY : 監視表のスキーマを指定します。

監視表を CDB 上に作成した場合は、省略できます。

監視表を PDB 上に作成した場合は、SYS またはローカル・ユーザーを指定してください。

ZZZ : 接続先データベース名を指定します。

監視表を CDB 上に作成した場合は、省略できます。

監視表を PDB 上に作成した場合は、PDB 名を指定してください。

6 権限の付与

SYS ユーザーで監視を行う場合は、実行する必要はありません。

SYS ユーザー以外のユーザーで監視を行う場合、3 で作成した Oracle 接続ユーザー (共通ユーザー) に、監視に必要な権限を付与します。

以下のシェルスクリプトを実行してください。

【注意】OSDBA(dba) グループに所属する Oracle ユーザーで実行してください。

【注意】Oracle 接続ユーザー (共通ユーザー) を指定する場合、ユーザー名が C## または c## で始まる必要があります。

【注意】監視表のスキーマを指定する場合、大文字 / 小文字に注意してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh grant <XXX>[ <YYY>[ <ZZZ>]]
```

XXX : 3 で作成した Oracle 接続ユーザー (共通ユーザー) を指定します。省略はできません。

YYY : 監視表のスキーマを指定します。

監視表を CDB 上に作成した場合は、省略できます。

監視表を PDB 上に作成した場合は、SYS またはローカル・ユーザーを指定してください。

ZZZ : 接続先データベース名を指定します。

監視表を CDB 上に作成した場合は、省略できます。

監視表を PDB 上に作成した場合は、PDB 名を指定してください。

【注意】Oracle 接続ユーザーを変更する場合は、権限を付与し直してください。

3.2. 設定ファイル (oramond.conf) の作成と適用

ApplicationMonitor を利用するためには、設定ファイルを作成する必要があります。設定ファイルには、監視対象とするインスタンスが動作するノード名 (NODE_NAME)、Oracle に関する情報 (ORACLE_SID, ORACLE_BASE, ORACLE_HOME, ORA_NLS10, SHLIB_PATH, NLS_LANG)、監視モニタの動作設定などを定義します。

設定ファイルのサンプルが /etc/opt/HA/AM/conf/sample ディレクトリにインストールされます。サンプルを参照して、システムの環境に応じた設定を行ってください。

ApplicationMonitor の設定ファイルに記述する ORACLE_SID パラメータ、ORACLE_BASE パラメータ、ORACLE_HOME パラメータ、ORA_NLS パラメータ (ORA_NLS10 に相当)、SHLIB_PATH パラメータおよび NLS_LANG パラメータは、Oracle の環境に依存しますので、環境を確認した上で必ず指定してください。

設定ファイルを作成後、管理コマンド (oraadmin) の apply リクエストにより、設定内容を適用します。Oracle が動作している状態で設定内容を適用してください。

設定ファイルは、構成情報ファイル (oramond<構成番号>.bin) に変換され、各ノード上の構成情報管理サーバ (oraconfd) に送信されます。このとき、各ノード上で構成情報管理サーバが起動されていることを ps(1) コマンドなどを用いて確認してください。

管理コマンドによって生成可能な構成情報ファイルの上限サイズは、31744 バイトです。文字列パラメータ値の長さによって異なりますが、1 つのステートメントにつき、約 100 ~ 300 バイトのサイズを必要とします。

管理コマンドの詳細は、「6 コマンドリファレンス」を参照してください。

注意: 構成情報ファイルの上限サイズを超えた場合、適用時にエラーになります。

注意: 作成した設定ファイルは、任意のディレクトリに置くことができますが、/etc/opt/HA/AM/conf/sample ディレクトリ内に置くと、アップデート時などに設定ファイルの上書きや削除が行われる可能性があります。

作成した設定ファイルは、/etc/opt/HA/AM/conf ディレクトリに置くことを推奨します。

3.2.1. 設定ファイルの記述例

設定ファイル (oramond.conf) は、コメント、パラメータ、ステートメントによって構成されたテキストファイルです。

現用ノードと待機ノードで、Oracle のインスタンス監視、リスナー監視を同一の設定内容にして監視をする場合は、以下のように記述します。

```
# ノード名
NODE_NAME = node1
NODE_NAME = node2

# 構成番号(0-9)
COMPONENT_ID = 0

# モニタタイプ 12C ( 11G or 11GR2 or 12C )
MONTYPE = 12C

#
# Oracle 環境変数
#
ORACLE_BASE = /u01/app/oracle
ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1
ORA_NLS = /u01/app/oracle/product/12.1.0/dbhome_1/nls/data
SHLIB_PATH = /u01/app/oracle/product/12.1.0/dbhome_1/lib
NLS_LANG = japanese_japan.ja16euc

#
# 現用ノード設定
#
Node node1 {
    ORACLE_SID = sid1
    WAIT_ASM_RESTART = YES

    InstanceMonitor {
        OWNER_NAME = SYS
        Table USER_TABLE {
            TABLE_USER = scott
            TABLE_PASSWORD = "tiger"
        }
    }

    ListenerMonitor LISTENER {
        NET_SERVICE_NAME = LSNR
    }
}

#
# 待機ノード設定
#
Node node2 {
    ORACLE_SID = sid1
    WAIT_ASM_RESTART = YES

    InstanceMonitor {
        OWNER_NAME = SYS
        Table USER_TABLE {
            TABLE_USER = scott
            TABLE_PASSWORD = "tiger"
        }
    }

    ListenerMonitor LISTENER {
        NET_SERVICE_NAME = LSNR
    }
}
```

3 製品の設定

2 ノードで構成されたクラスタ環境で、RAC のインスタンス監視、リスナー監視、データベース容量不足の監視、ローカルディスク容量不足の監視、統計情報の採取および Oracle のシステムステート・ダンプの採取を行う場合は、以下のように記述します。

デフォルト設定値の記述を省略することで、簡易設定を行うことも可能です。

```
# ノード名
NODE_NAME = node1
NODE_NAME = node2

# 構成番号(0-9)
COMPONENT_ID = 1

# モニタタイプ 12C ( 11G or 11GR2 or 12C )
MONTYPE = 12C

#
# Oracle 環境変数
#
ORACLE_BASE = /u01/app/oracle
ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1
ORA_NLS      = /u01/app/oracle/product/12.1.0/dbhome_1/nls/data
SHLIB_PATH   = /u01/app/oracle/product/12.1.0/dbhome_1/lib
NLS_LANG     = japanese_japan.ja16euc

GRID_BASE = /u01/app/grid
GRID_HOME = /u01/app/12.1.0/grid
GRID_USER = grid

# 障害時動作
HALT_CONTROL_POLICY = CONFIGURED
HALT_CONTROL_OPTION = ALWAYS
HALT_CONTROL_METHOD = NONE

#
# インスタンス監視設定
#
InstanceMonitor {
    LOG_LEVEL          = 2
    MONITOR_USER       = oracle
    BOOT_TIMEOUT       = 10
    POLL_INTERVAL      = 90
    POLL_TIMEOUT       = 120
    POLL_DOWN_RETRY    = 0
    POLL_STALL_RETRY   = 0
    REFORM_TIMEOUT     = 30
}

#
# リスナー監視設定
#
ListenerMonitor {
    LOG_LEVEL          = 2
    MONITOR_USER       = oracle
    BOOT_TIMEOUT       = 10
    MONITOR_CONTROL    = YES
    TNSPING            = /u01/app/oracle/product/11.2.0/dbhome_1/bin/tnsping
    LSNRCTL            = /etc/opt/HA/AM/bin/lsnr_control.sh
    POLL_INTERVAL      = 60
    POLL_TIMEOUT       = 90
    SERVICE_DOWN       = NO
    RESTART_COUNT       = 5
    RESTART_INTERVAL   = 3
}

#
# データベース容量不足監視設定
#
TableSpaceMonitor {
    LOG_LEVEL          = 2
    MONITOR_USER       = oracle
    BOOT_TIMEOUT       = 10
    MONITOR_CONTROL    = YES
    POLL_INTERVAL      = 3600
    POLL_TIMEOUT       = 60
    COMMAND_NAME       =
}

#
```

```

# ローカルディスク容量不足監視設定
#
DiskSpaceMonitor {
    LOG_LEVEL          = 2
    MONITOR_USER       = oracle
    BOOT_TIMEOUT       = 10
    MONITOR_CONTROL    = YES
    POLL_INTERVAL      = 3600
    POLL_TIMEOUT       = 60
    FREE_SPACE         = 10
    Directory ARCHIVE {
        FREE_SPACE = 20
    }
}

#
# 統計情報採取設定
#
StatsMonitor {
    LOG_LEVEL          = 2
    MONITOR_USER       = oracle
    BOOT_TIMEOUT       = 10
    MONITOR_CONTROL    = YES
}

#
# Oracle のシステムステート・ダンプ採取設定
#
SystemStateDaemon {
    LOG_LEVEL          = 2
    MONITOR_USER       = oracle
    BOOT_TIMEOUT       = 10
    GET_DUMP           = YES
    POLL_INTERVAL      = 60
    POLL_TIMEOUT       = 60
    DUMP_INTERVAL      = 30
    DUMP_COUNT         = 3
    DUMP_TIMEOUT       = 90
}

#
# ノード設定
#
Node node1 {
    ORACLE_SID = sid1
    PDB_NAME   = PDB1

    InstanceMonitor {
        GET_STATS          = YES
        OWNER_NAME         = SYS
        UNOPEN_PDB_DOWN    = YES
        Table USER_TABLE1 {
            TABLE_USER     = scott
            TABLE_PASSWORD = "tiger"
        }
    }

    ListenerMonitor LISTENER1 {
        NET_SERVICE_NAME = LSNR1
        GET_STATS         = YES
    }

    ListenerMonitor LISTENER2 {
        MONITOR_CONTROL = NO
        NET_SERVICE_NAME = LSNR2
        GET_STATS         = YES
    }

    TableSpaceMonitor TSCHKMOND1 {
        TableSpace TABLESPACE1 {
            USEABLE_EXTENT_BLOCK = 0
            FREE_SPACE           = 20
            GET_STATS             = YES
        }
    }

    StatsMonitor STATSMOND1 {
        Stats SGA_SPACE {
            POLL_INTERVAL = 3600
            POLL_TIMEOUT  = 60
            GET_STATS      = YES
        }
    }
}

```

3 製品の設定

```
#
# ノード設定
#
Node node2 {
    ORACLE_SID = sid2
    PDB_NAME   = PDB2

    InstanceMonitor {
        GET_STATS      = YES
        OWNER_NAME     = HAAM
        UNOPEN_PDB_DOWN = YES
        Table USER_TABLE2 {
            TABLE_USER      = scott
            TABLE_PASSWORD = "tiger"
        }
    }

    ListenerMonitor LISTENER3 {
        MONITOR_CONTROL = NO
        NET_SERVICE_NAME = LSNR3
        GET_STATS       = YES
    }

    TableSpaceMonitor TSCHKMOND2 {
        TableSpace TABLESPACE2 {
            USEABLE_EXTENT_BLOCK = 0
            FREE_SPACE           = 20
            GET_STATS             = YES
        }
    }

    StatsMonitor STATSMOND2 {
        Stats SGA_SPACE {
            POLL_INTERVAL = 3600
            POLL_TIMEOUT  = 60
            GET_STATS      = YES
        }
    }
}
```

2 サイトごとに 2 ノードで構成されたクラスタ環境で、RAC のスタンバイ・インスタンスの監視を行う場合は、以下のように記述します。

デフォルト設定値の記述を省略することで、簡易設定を行うことも可能です。

```
#
# プライマリ・サイトの設定
#

# ノード名
NODE_NAME = pri1
NODE_NAME = pri2

# 構成番号(0-9)
COMPONENT_ID = 0

# モニタタイプ 12C ( 11G or 11GR2 or 12C )
MONTYPE = 12C

#
# Oracle 環境変数
#
ORACLE_BASE = /u01/app/oracle
ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1
ORA_NLS = /u01/app/oracle/product/12.1.0/dbhome_1/nls/data
SHLIB_PATH = /u01/app/oracle/product/12.1.0/dbhome_1/lib
NLS_LANG = japanese_japan.ja16euc

# スタンバイ・インスタンス監視有無の設定
STANDBY_DB_CHECK = YES

# プライマリ・サイトの受信用ポート番号
INTERSITE_PORT = 25321

#
# サイト間通信の情報設定
#
DataGuardSite SiteStb {
    DB_UNIQUE_NAME = "db_stb"
    INTERSITE_PORT = 25322
    SITE_NODE = stb1
    SITE_NODE = stb2
}

#
# インスタンス監視設定
#
InstanceMonitor {
    OWNER_NAME = SYS
}

#
# スタンバイ・インスタンス監視時の設定
#
StandbyDatabaseMonitor {
    LOG_LEVEL = 2

    #
    # スタンバイ・インスタンス(REDO ログ転送状況監視時)の設定
    #
    Transport {
        POLL_INTERVAL = 60

        #
        # REDO ログ転送状況監視時の通信情報
        #
        Destination SiteStb {
            PRIMARY_NET_SERVICE_NAME = "dg_stb1"
            PRIMARY_NET_SERVICE_NAME = "dg_stb2"
            PRIMARY_ORACLE_USER = mcam
            PRIMARY_ORACLE_PASS = "mcam"
        }
    }

    #
    # スタンバイ・インスタンス(REDO ログ適用状況監視時)の設定
    #
    RedoApply {
        POLL_INTERVAL = 60
    }
}
```

3 製品の設定

```
#
# スタンバイ・インスタンス(Oracle Data Guard Broker プロセス状態監視時)の設定
#
OdgBroker {
    POLL_INTERVAL = 60
}

#
# ノード設定
#
Node pri1 {
    ORACLE_SID = clustdbpri1

    StandbyDatabaseMonitor {
        Transport {
            RETRY_COUNT = 0
        }
        RedoApply {
            RETRY_COUNT = 0
        }
        OdgBroker {
            RETRY_COUNT = 0
        }
    }
}

#
# ノード設定
#
Node pri2 {
    ORACLE_SID = clustdbpri2

    StandbyDatabaseMonitor {
        Transport {
            POLL_TIMEOUT = 90
        }
        RedoApply {
            POLL_TIMEOUT = 90
        }
        OdgBroker {
            POLL_TIMEOUT = 90
        }
    }
}
}
```

```
#
# スタンバイ・サイトの設定
#

# ノード名
NODE_NAME = stb1
NODE_NAME = stb2

# 構成番号(0-9)
COMPONENT_ID = 0

# モニタタイプ 12C ( 11G or 11GR2 or 12C )
MONTYPE = 12C

#
# Oracle 環境変数
#
ORACLE_BASE = /u01/app/oracle
ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1
ORA_NLS = /u01/app/oracle/product/12.1.0/dbhome_1/nls/data
SHLIB_PATH = /u01/app/oracle/product/12.1.0/dbhome_1/lib
NLS_LANG = japanese_japan.ja16euc

# スタンバイ・インスタンス監視有無の設定
STANDBY_DB_CHECK = YES

# スタンバイ・サイトの受信用ポート番号
INTERSITE_PORT = 25322

#
# サイト間通信の情報設定
#
DataGuardSite SitePri {
    DB_UNIQUE_NAME = "db_pri"
    INTERSITE_PORT = 25321
    SITE_NODE = pri1
    SITE_NODE = pri2
}
```

```

}

#
# インスタンス監視設定
#
InstanceMonitor {
    OWNER_NAME = SYS
}

#
# スタンバイ・インスタンス監視時の設定
#
StandbyDatabaseMonitor {
    LOG_LEVEL = 2

    #
    # スタンバイ・インスタンス (REDO ログ転送状況監視時) の設定
    #
    Transport {
        POLL_INTERVAL = 60

        #
        # REDO ログ転送状況監視時の通信情報
        #
        Destination SiteStb {
            PRIMARY_NET_SERVICE_NAME = "dg_pri1"
            PRIMARY_NET_SERVICE_NAME = "dg_pri2"
            PRIMARY_ORACLE_USER      = mcam
            PRIMARY_ORACLE_PASS      = "mcam"
        }
    }

    #
    # スタンバイ・インスタンス (REDO ログ適用状況監視時) の設定
    #
    RedoApply {
        POLL_INTERVAL = 60
    }

    #
    # スタンバイ・インスタンス (Oracle Data Guard Broker プロセス状態監視時) の設定
    #
    OdgBroker {
        POLL_INTERVAL = 60
    }
}

#
# ノード設定
#
Node pri1 {
    ORACLE_SID = clustdbstb1

    StandbyDatabaseMonitor {
        Transport {
            RETRY_COUNT = 0
        }
        RedoApply {
            RETRY_COUNT = 0
        }
        OdgBroker {
            RETRY_COUNT = 0
        }
    }
}

#
# ノード設定
#
Node pri2 {
    ORACLE_SID = clustdbstb2

    StandbyDatabaseMonitor {
        Transport {
            POLL_TIMEOUT = 90
        }
        RedoApply {
            POLL_TIMEOUT = 90
        }
        OdgBroker {
            POLL_TIMEOUT = 90
        }
    }
}

```

3.2.2. コメントの記述形式

コメントは、ダブルクォーテーション (") で囲まれていないシャープ記号 (#) で始まり、行の最後までとなります。

3.2.3. パラメータの記述形式

パラメータは、ApplicationMonitor の動作に必要な値を数値または文字列で指定します。パラメータ値にスペースを含める場合は、文字列全体をダブルクォーテーション (") で囲んでください。

1 つのパラメータを 1 行(1021 文字以内)に記述します。パラメータの記述形式は、以下のとおりです。

各種パラメータの詳細は、「3.2.7 パラメーター一覧」を参照してください。

記述形式:

パラメータ名 = パラメータ値[,パラメータ値]

注意: パラメータ値に全角文字、タブを使用することはできません。

また、文字列内にダブルクォーテーション (") および改行コードを使用することはできません。

3.2.4. ステートメントの記述形式

ステートメントは、以下の 18 種類に分類します。

- ◆ 全ノードステートメント
- ◆ DataGuardSite ステートメント 1
- ◆ Node ステートメント
- ◆ InstanceMonitor ステートメント
- ◆ ListenerMonitor ステートメント
- ◆ TableSpaceMonitor ステートメント
- ◆ DiskSpaceMonitor ステートメント
- ◆ StandbyDatabaseMonitor ステートメント 1
- ◆ StatsMonitor ステートメント
- ◆ SystemStateDaemon ステートメント
- ◆ Table ステートメント
- ◆ TableSpace ステートメント
- ◆ Directory ステートメント
- ◆ Transport ステートメント 1
- ◆ Destination ステートメント 1
- ◆ RedoApply ステートメント 2
- ◆ OdgBroker ステートメント 2
- ◆ Stats ステートメント

- 1 スタンバイ・インスタンスを監視する場合、記述が必要なステートメントです。
- 2 スタンバイ・インスタンスを監視する場合、記述を省略できるステートメントです。

各ステートメントの記述形式について説明します。

◆ 全ノードステートメント

全ノードステートメントには、ApplicationMonitor が動作する上で必要な情報を記述します。また、全ノードステートメントには、DataGuardSite ステートメント、InstanceMonitor ステートメント、ListenerMonitor ステートメント、TableSpaceMonitor ステートメント、DiskSpaceMonitor ステートメント、StandbyDatabaseMonitor ステートメント、StatsMonitor ステートメント、SystemStateDaemon ステートメントを記述できます。

全ノードステートメントに記述したパラメータおよびステートメントは、すべてのノードで共通の設定になります。

記述形式は、以下のとおりです。

記述形式:

パラメータ名 = パラメータ値
[DataGuardSite ステートメント]
[InstanceMonitor ステートメント]
[ListenerMonitor ステートメント]
[TableSpaceMonitor ステートメント]
[DiskSpaceMonitor ステートメント]
[StandbyDatabaseMonitor ステートメント]
[StatsMonitor ステートメント]
[SystemStateDaemon ステートメント]

全ノードステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の全ノードステートメントを参照してください。

◆ **DataGuardSite ステートメント**

DataGuardSite ステートメントを記述することで、スタンバイ・インスタンスを監視するために必要なサイト間で通信を行う情報をサイトごとに設定します。

記述形式は、以下のとおりです。

記述形式:

```
DataGuardSite サイト間通信 ID {
    [パラメータ名 = パラメータ値]
}
```

サイト間通信 ID は、各サイトを識別するための任意の文字列を記述します。

DataGuardSite ステートメントが複数ある場合は、指定した複数のサイトが通信を行います。

DataGuardSite ステートメントは、全ノードステートメントにのみ記述できます。

DataGuardSite ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の DataGuardSite ステートメントを参照してください。

注意: *サイト間通信 ID* は省略できません。

注意: 重複する *サイト間通信 ID* は指定できません。

注意: DataGuardSite ステートメントを省略した場合、スタンバイ・インスタンスの監視を行いません。

また、DataGuardSite ステートメントを記述しても、StandbyDatabaseMonitor ステートメントを省略した場合、スタンバイ・インスタンスの監視を行いません。

注意: DataGuardSite ステートメントに記述した *サイト間通信 ID* と Destination ステートメントに指定した *サイト間通信 ID* を一致させてください。

◆ Node ステートメント

Node ステートメントには、指定したノードごとに ApplicationMonitor が動作する上で必要な情報を記述します。また、Node ステートメントには、InstanceMonitor ステートメント、ListenerMonitor ステートメント、TableSpaceMonitor ステートメント、DiskSpaceMonitor ステートメント、StandbyDatabaseMonitor ステートメント、StatsMonitor ステートメント、SystemStateDaemon ステートメントを記述できます。

Node ステートメントに記述したパラメータは、該当するノード内で共通の設定になります。記述形式は、以下のとおりです。

記述形式:

```
Node ノード名 {  
    [パラメータ名 = パラメータ値]  
    [InstanceMonitor ステートメント]  
    [ListenerMonitor ステートメント]  
    [TableSpaceMonitor ステートメント]  
    [DiskSpaceMonitor ステートメント]  
    [StandbyDatabaseMonitor ステートメント]  
    [StatsMonitor ステートメント]  
    [SystemStateDaemon ステートメント]  
}
```

ノード名 は、NODE_NAME パラメータに指定したノード名を記述します。

Node ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の Node ステートメントを参照してください。

注意: ノード名 は省略できません。

◆ InstanceMonitor ステートメント

InstanceMonitor ステートメントを記述することで、インスタンス障害を監視するインスタンス監視モニタ (instmond) の動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
InstanceMonitor {
    [パラメータ名 = ノパラメータ値]
    [Table ステートメント]
}
```

InstanceMonitor ステートメントを全ノードステートメントに記述した場合は、全ノード共通のインスタンス監視モニタの設定になります。

InstanceMonitor ステートメントを Node ステートメント内に記述した場合は、該当するノード内のインスタンス監視モニタの設定になります。

InstanceMonitor ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の InstanceMonitor ステートメントを参照してください。

注意: InstanceMonitor ステートメントを省略した場合でも、インスタンス監視は行われます。

◆ ListenerMonitor ステートメント

ListenerMonitor ステートメントを記述することで、対象リスナーの障害を監視するリスナー監視モニタ (lsnrmond) の動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
ListenerMonitor [リスナー名] {  
    [パラメータ名 = パラメータ値]  
}
```

リスナー名 は、監視対象のリスナー名を記述します。

ListenerMonitor ステートメントを全ノードステートメントに記述した場合は、全ノード共通のリスナー監視モニタの設定になります。

ListenerMonitor ステートメントを Node ステートメント内に記述する場合は、*リスナー名* を指定しない場合と指定する場合があります、*リスナー名* を指定しない場合は、該当するノード内すべてのリスナー監視モニタの設定になります。*リスナー名* を指定した場合は、該当するリスナー監視モニタの設定になります。

リスナー名 を指定した ListenerMonitor ステートメントが複数ある場合は、指定した複数のリスナーを監視します。

ListenerMonitor ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の ListenerMonitor ステートメントを参照してください。

注意: 全ノードステートメントに記述した ListenerMonitor ステートメントには、*リスナー名* を指定できません。

注意: 同一の Node ノードステートメント内に重複する *リスナー名* は指定できません。

注意: Node ステートメント内に *リスナー名* を指定した ListenerMonitor ステートメントがない場合は、該当するノード上でリスナー監視モニタの起動(リスナー監視)を行いません。

◆ **TableSpaceMonitor ステートメント**

TableSpaceMonitor ステートメントを記述することで、データベース領域の容量を監視する表領域監視モニタ (tschkmond) の動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
TableSpaceMonitor [表領域監視モニタ ID] {
    [パラメータ名 = パラメータ値]
    TableSpace ステートメント
}
```

表領域監視モニタ ID は、ノード内で表領域監視モニタを識別するための任意の文字列を記述します。

TableSpaceMonitor ステートメントを全ノードステートメントに記述した場合は、全ノード共通の表領域監視モニタの設定になります。

TableSpaceMonitor ステートメントを Node ステートメント内に記述する場合は、表領域監視モニタ ID を指定しない場合と指定する場合があります。表領域監視モニタ ID を指定しない場合は、該当するノード内すべての表領域監視モニタの設定になります。表領域監視モニタ ID を指定した場合は、該当する表領域監視モニタの設定になります。

表領域監視モニタ ID を指定した TableSpaceMonitor ステートメントが複数ある場合は、指定した複数の表領域監視モニタが起動されます。

TableSpaceMonitor ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメータ一覧」の TableSpaceMonitor ステートメントを参照してください。

注意: 全ノードステートメントに記述した TableSpaceMonitor ステートメントには、表領域監視モニタ ID を指定できません。

注意: 同一の Node ステートメント内に重複する 表領域監視モニタ ID は指定できません。

注意: Node ステートメント内に 表領域監視モニタ ID を指定した TableSpaceMonitor ステートメントがない場合は、該当するノード上で表領域監視モニタの起動(データベース領域の容量監視)を行いません。

注意: 表領域監視モニタ ID を指定した TableSpaceMonitor ステートメント内には、必ず TableSpace ステートメントを記述してください。

◆ DiskSpaceMonitor ステートメント

DiskSpaceMonitor ステートメントを記述することで、ローカルディスク領域の容量を監視するディスク領域監視モニタ (fdsmond) の動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
DiskSpaceMonitor {  
    [パラメータ名 = パラメータ値]  
    [Directory ステートメント]  
}
```

DiskSpaceMonitor ステートメントを全ノードステートメントに記述した場合は、全ノード共通のディスク領域監視モニタの設定になります。

DiskSpaceMonitor ステートメントを Node ステートメント内に記述した場合は、該当するノード内のディスク領域監視モニタの設定になります。

DiskSpaceMonitor ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の DiskSpaceMonitor ステートメントを参照してください。

注意: DiskSpaceMonitor ステートメントを省略した場合は、ディスク領域監視モニタの起動(ローカルディスク領域の容量監視)を行いません。

◆ StandbyDatabaseMonitor ステートメント

StandbyDatabaseMonitor ステートメントを記述することで、スタンバイ・インスタンスを監視するモニタ制御デーモン (oramond) およびスタンバイデータベース監視モニタ (stbdbmond) の動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
StandbyDatabaseMonitor {
    [パラメータ名 = パラメータ値]
    Transport ステートメント
    [RedoApply ステートメント]
    [OdgBroker ステートメント]
}
```

StandbyDatabaseMonitor ステートメントを全ノードステートメントに記述した場合は、全ノード共通の設定になります。

StandbyDatabaseMonitor ステートメントを Node ステートメント内に記述した場合は、該当するノード内の設定になります。

StandbyDatabaseMonitor ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の StandbyDatabaseMonitor ステートメントを参照してください。

注意: StandbyDatabaseMonitor ステートメントを省略した場合、スタンバイ・インスタンスの監視を行いません。

また、StandbyDatabaseMonitor ステートメントを記述しても、DataGuardSite ステートメントを省略した場合、スタンバイ・インスタンスの監視を行いません。

注意: StandbyDatabaseMonitor ステートメントを記述した場合は、必ず Transport ステートメントを記述してください。

◆ StatsMonitor ステートメント

StatsMonitor ステートメントを記述することで、Oracle の統計情報を採取する統計情報採取モニタ (statsmond) の動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
StatsMonitor [統計情報採取モニタ ID] {  
    [パラメータ名 = パラメータ値]  
    Stats ステートメント  
}
```

統計情報採取モニタ ID は、ノード内で統計情報採取モニタを識別するための任意の文字列を記述します。

StatsMonitor ステートメントを全ノードステートメントに記述した場合は、全ノード共通の統計情報採取モニタの設定になります。

StatsMonitor ステートメントを Node ステートメント内に記述する場合は、*統計情報採取モニタ ID* を指定しない場合と指定する場合があります。統計情報採取モニタ ID を指定しない場合は、該当するノード内すべての統計情報採取モニタの設定になります。統計情報採取モニタ ID を指定した場合は、該当する統計情報採取モニタの設定になります。

統計情報採取モニタ ID を指定した StatsMonitor ステートメントが複数ある場合は、指定した複数の統計情報採取モニタを起動します。

StatsMonitor ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の StatsMonitor ステートメントを参照してください。

注意: 全ノードステートメントに記述した StatsMonitor ステートメントには、*統計情報採取モニタ ID* を指定できません。

注意: 同一の Node ステートメント内に重複する *統計情報採取モニタ ID* は指定できません。

注意: Node ステートメント内に *統計情報採取モニタ ID* を指定した StatsMonitor ステートメントがない場合は、該当するノード上で統計情報採取モニタの起動(統計情報の採取)を行いません。

注意: *統計情報採取モニタ ID* を指定した StatsMonitor ステートメント内には、必ず Stats ステートメントを記述してください。

◆ **SystemStateDaemon ステートメント**

SystemStateDaemon ステートメントを記述することで、Oracle のシステムステート・ダンプを採取するシステムステートダンプ採取デーモン (ssdump) の動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
SystemStateDaemon {
    [パラメータ名 = パラメータ値]
}
```

SystemStateDaemon ステートメントを全ノードステートメントに記述した場合は、全ノード共通のシステムステートダンプ採取デーモンの設定になります。

SystemStateDaemon ステートメントを Node ステートメント内に記述した場合は、該当するノード内のシステムステートダンプ採取デーモンの設定になります。

SystemStateDaemon ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメータ一覧」の SystemStateDaemon ステートメントを参照してください。

注意: SystemStateDaemon ステートメントを省略した場合でも、障害発生時のシステムステート・ダンプの採取は行われます。明示的に採取しない場合は、GET_DUMP パラメータを NO に設定してください。

◆ Table ステートメント

Table ステートメントを記述することで、インスタンス監視モニタ (instmond) がデフォルトの監視と合わせ、指定された表ごとに監視します。Table ステートメントには、指定した表を監視する際のインスタンス監視モニタの動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
Table 表名 {  
    [パラメータ名 = パラメータ値]  
}
```

表名 は、監視対象の表を記述します。半角で最大 30 文字まで指定できます。スペース、半角小文字を含める場合は、ダブルクォーテーション (") で囲ってください。

Table ステートメントは、Node ステートメント内の InstanceMonitor ステートメント内のみ記述できます。

表名 を指定した Table ステートメントが複数ある場合は、指定した複数の表を監視します。

Table ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の Table ステートメントを参照してください。

注意: 表名 は省略できません。

注意: 「3.1 インスタンス監視表の作成」で作成した監視表は指定しないでください。

◆ **TableSpace ステートメント**

TableSpace ステートメントを記述することで、表領域監視モニタ (tschkmond) が設定された内容で指定された表領域の容量不足を監視します。TableSpace ステートメントには、指定した表領域を監視する際の表領域監視モニタの動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
TableSpace 表領域名 {
    [パラメータ名 = パラメータ値]
}
```

表領域名 は、監視対象の表領域を記述します。半角で最大 30 文字まで指定できます。スペース、小文字を含める場合は、ダブルクォーテーション (") で囲ってください。

TableSpace ステートメントは、Node ステートメント内の TableSpaceMonitor ステートメント内にのみ記述できます。

表領域名 を指定した TableSpace ステートメントが複数ある場合は、指定した複数の表領域を監視します。

TableSpace ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の TableSpace ステートメントを参照してください。

注意: **表領域名** は省略できません。

◆ Directory ステートメント

Directory ステートメントを記述することで、ディスク領域監視モニタ (fdsmond) が指定された監視対象ごとに、設定された内容でローカルディスク領域の容量不足を監視します。Directory ステートメントには、監視対象種別ごとのディスク領域監視モニタの動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
Directory 種別 {
    [パラメータ名 = パラメータ値]
}
```

種別 は、HOME、ARCHIVE、TRACE、ALERT、CORE、AUDIT のいずれかを記述します。それぞれの *種別* の監視対象は、以下のとおりです。

種別	監視対象
HOME	ORACLE_HOME
ARCHIVE	アーカイブログ出力先
TRACE	トレース出力先
ALERT	アラート・ログ出力先
CORE	コアダンプ出力先
AUDIT	監査証跡出力先

Directory ステートメントは、DiskSpaceMonitor ステートメント内にのみ記述できます。

Directory ステートメントを記述した場合は、*種別* ごとに設定された内容で監視が行われます。Directory ステートメントを省略した場合は、すべての領域について、DiskSpaceMonitor ステートメントで設定された内容で監視が行われます。

Directory ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の Directory ステートメントを参照してください。

注意: *種別* は省略できません。

◆ Transport ステートメント

Transport ステートメントを記述することで、スタンバイデータベース監視モニタ (stbldbmond) がプライマリ・サイトからスタンバイ・サイトへの REDO ログの転送状況を監視します。Transport ステートメントには、REDO ログの転送状況を監視する際のスタンバイデータベース監視モニタの動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
Transport {
    [パラメータ名 = パラメータ値]
    Destination ステートメント
}
```

Transport ステートメントは、StandbyDatabaseMonitor ステートメント内にのみ記述できます。

Transport ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の Transport ステートメントを参照してください。

注意: StandbyDatabaseMonitor ステートメントを記述した場合、Transport ステートメントは省略できません。

注意: Transport ステートメントを記述した場合は、必ず Destination ステートメントを記述してください。

◆ Destination ステートメント

Destination ステートメントを記述することで、スタンバイデータベース監視モニタ (stbldbmond) がプライマリ・サイトからスタンバイ・サイトへの REDO ログの転送状況を監視します。Destination ステートメントには、REDO ログの転送状況を監視する際の通信情報をサイトごとに設定します。

記述形式は、以下のとおりです。

記述形式:

```
Destination サイト間通信 ID {  
    [パラメータ名 = パラメータ値]  
}
```

サイト間通信 ID は、DataGuardSite ステートメントに指定した サイト間通信 ID を記述します。

Destination ステートメントが複数ある場合は、指定した複数のサイトで REDO ログの転送状況の監視を行います。

Destination ステートメントは、Transport ステートメント内にのみ記述できます。

Destination ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の Destination ステートメントを参照してください。

注意: Transport ステートメントを記述した場合、Destination ステートメントは省略できません。

注意: サイト間通信 ID は省略できません。

注意: 重複する サイト間通信 ID は指定できません。

注意: Destination ステートメントに記述した サイト間通信 ID と DataGuardSite ステートメントに指定した サイト間通信 ID を一致させてください。

◆ RedoApply ステートメント

RedoApply ステートメントを記述することで、スタンバイデータベース監視モニタ (stbldbmond) がプライマリ・サイトからスタンバイ・サイトへの REDO ログの適用状況を監視します。RedoApply ステートメントには、REDO ログの適用状況を監視する際のスタンバイデータベース監視モニタの動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
RedoApply {
    [パラメータ名 = パラメータ値]
}
```

RedoApply ステートメントは、StandbyDatabaseMonitor ステートメント内にのみ記述できます。

RedoApply ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の RedoApply ステートメントを参照してください。

注意: RedoApply ステートメントを省略した場合でも、DataGuardSite ステートメントおよび StandbyDatabaseMonitor ステートメントを記述すると、プライマリ・サイトからスタンバイ・サイトへの REDO ログの適用状況の監視は行われます。

◆ OdgBroker ステートメント

OdgBroker ステートメントを記述することで、モニタ制御デーモン (oramond) が Oracle Data Guard Broker プロセスの状態を監視します。OdgBroker ステートメントには、Oracle Data Guard Broker プロセスの状態を監視する際のモニタ制御デーモンの動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
OdgBroker {  
    [パラメータ名 = パラメータ値]  
}
```

OdgBroker ステートメントは、StandbyDatabaseMonitor ステートメント内にのみ記述できます。

OdgBroker ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の OdgBroker ステートメントを参照してください。

注意: OdgBroker ステートメントを省略した場合でも、DataGuardSite ステートメントおよび StandbyDatabaseMonitor ステートメントを記述すると、Oracle Data Guard Broker プロセスの状態監視は行われます。

◆ Stats ステートメント

Stats ステートメントを記述することで、統計情報採取モニタ (statsmond) が指定された Oracle の統計情報を採取します。Stats ステートメントには、指定した統計情報種別を採取する際の統計情報採取モニタの動作を設定します。

記述形式は、以下のとおりです。

記述形式:

```
Stats 統計情報種別 {
    [パラメータ名 = パラメータ値]
}
```

統計情報種別 は、SGA_SPACE を記述します。*統計情報種別* の採取対象は、以下のとおりです。

統計情報種別	採取対象
SGA_SPACE	SGA 情報

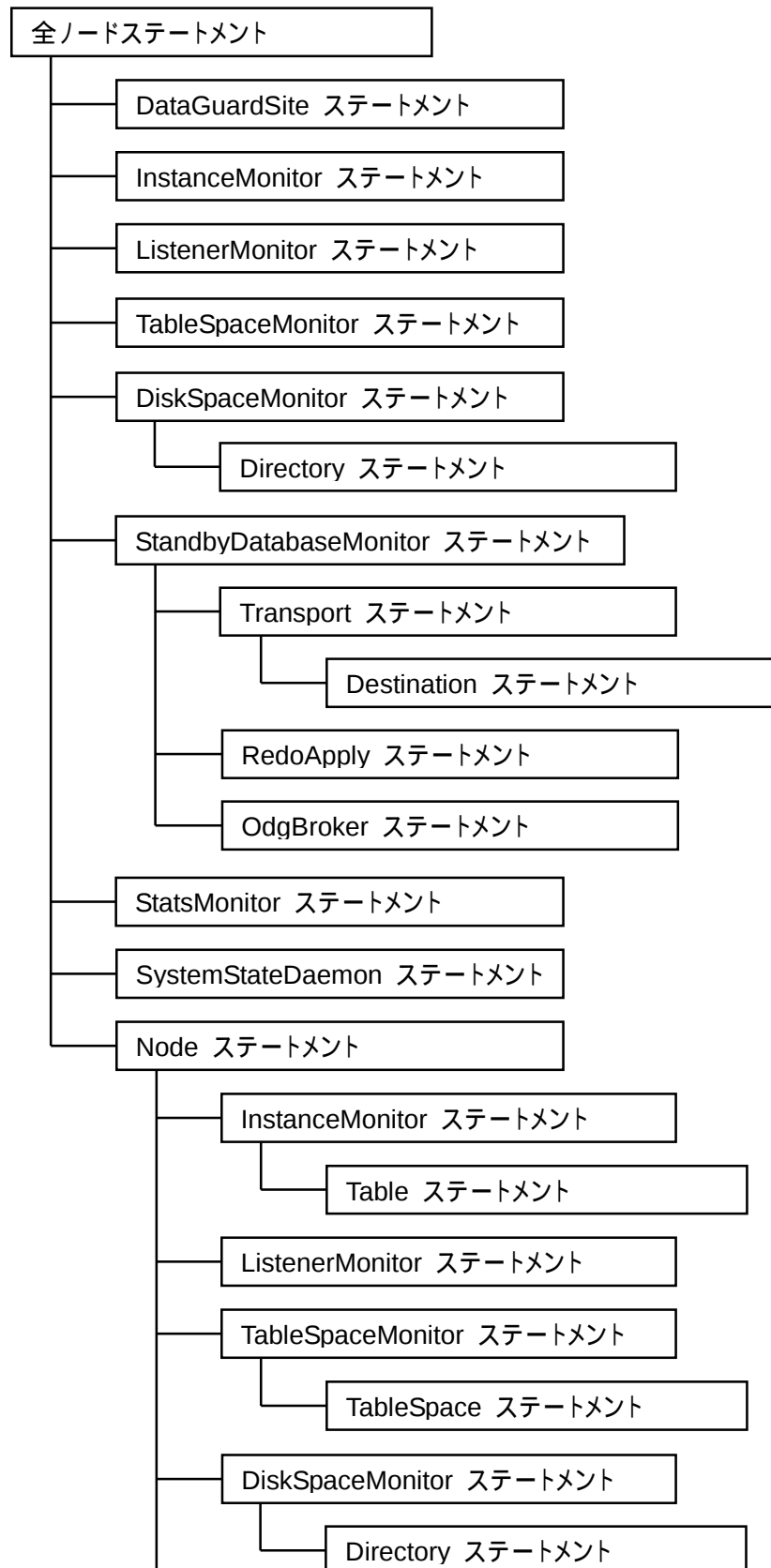
Stats ステートメントは、Node ステートメント内の StatsMonitor ステートメント内にのみ記述できます。

Stats ステートメントに記述できるパラメータの詳細は、「3.2.7 パラメーター一覧」の Stats ステートメントを参照してください。

注意: *統計情報種別* は省略できません。

3.2.5. ステートメントの階層

ステートメントは、以下のように階層で表すことができます。



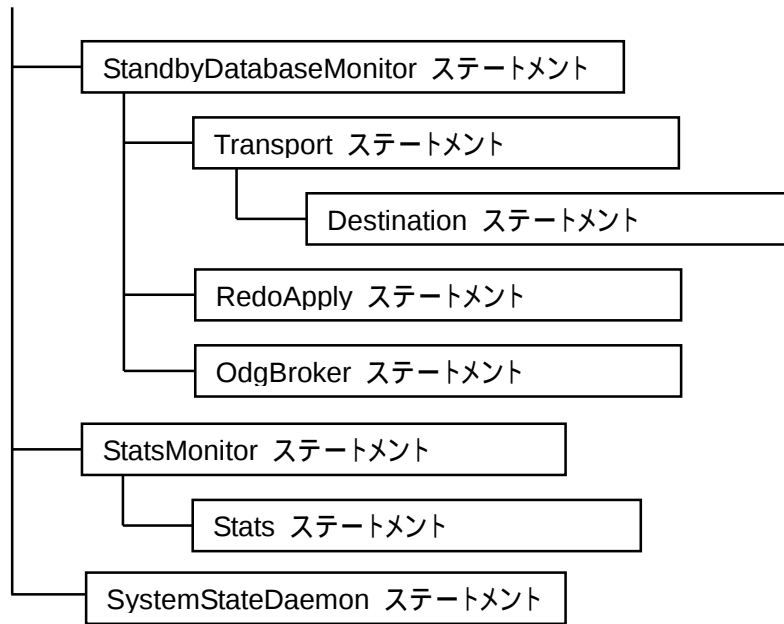


図 3-7 ステートメントの階層

LOG_LEVEL パラメータのように、それぞれのステートメントで記述可能なパラメータは、下位のステートメントで省略した場合、上位のステートメントで設定された値が適用されます。また、下位のステートメントで設定された場合、設定したステートメント以下のステートメントに適用されます。

3.2.6. パラメータとステートメントの記述順

設定ファイル (oramond.conf) に記述するパラメータとステートメントは、以下の順に記述する必要があります。

```

全ノードステートメントのパラメータ
サイト間通信 ID を指定した DataGuardSite ステートメント {
    DataGuardSite ステートメントのパラメータ
}
InstanceMonitor ステートメント {
    InstanceMonitor ステートメントのパラメータ
}
ListenerMonitor ステートメント {
    ListenerMonitor ステートメントのパラメータ
}
TableSpaceMonitor ステートメント {
    TableSpaceMonitor ステートメントのパラメータ
}
DiskSpaceMonitor ステートメント {
    DiskSpaceMonitor ステートメントのパラメータ
    種別を指定した Directory ステートメント {
        Directory ステートメントのパラメータ
    }
}
StandbyDatabaseMonitor ステートメント {
    StandbyDatabaseMonitor ステートメントのパラメータ
    Transport ステートメント {
        Transport ステートメントのパラメータ
        サイト間通信 ID を指定した Destination ステートメント {
            Destination ステートメントのパラメータ
        }
    }
}
RedoApply ステートメント {
    RedoApply ステートメントのパラメータ
}
OdgBroker ステートメント {
    OdgBroker ステートメントのパラメータ
}
StatsMonitor ステートメント {
    StatsMonitor ステートメントのパラメータ
}
SystemStateDaemon ステートメント {
    SystemStateDaemon ステートメントのパラメータ
}
Node ステートメント {
    Node ステートメントのパラメータ
    InstanceMonitor ステートメント {
        InstanceMonitor ステートメントのパラメータ
        表名を指定した Table ステートメント {
            Table ステートメントのパラメータ
        }
    }
}
リスナー名を指定しない ListenerMonitor ステートメント {
    ListenerMonitor ステートメントのパラメータ
}
リスナー名を指定した ListenerMonitor ステートメント {
    ListenerMonitor ステートメントのパラメータ
}

```

```

}
表領域監視モニタ ID を指定しない TableSpaceMonitor ステートメント {
    TableSpaceMonitor ステートメントのパラメータ
}
表領域監視モニタ ID を指定した TableSpaceMonitor ステートメント {
    TableSpaceMonitor ステートメントのパラメータ
    表領域名を指定した TableSpace ステートメント {
        TableSpace ステートメントのパラメータ
    }
}
}
DiskSpaceMonitor ステートメント {
    DiskSpaceMonitor ステートメントのパラメータ
    種別を指定した Directory ステートメント {
        Directory ステートメントのパラメータ
    }
}
}
StandbyDatabaseMonitor ステートメント {
    StandbyDatabaseMonitor ステートメントのパラメータ
    Transport ステートメント {
        Transport ステートメントのパラメータ
        サイト間通信 ID を指定した Destination ステートメント {
            Destination ステートメントのパラメータ
        }
    }
}
RedoApply ステートメント {
    RedoApply ステートメントのパラメータ
}
}
OdgBroker ステートメント {
    OdgBroker ステートメントのパラメータ
}
}
統計情報採取モニタ ID を指定しない StatsMonitor ステートメント {
    StatsMonitor ステートメントのパラメータ
}
統計情報採取モニタ ID を指定した StatsMonitor ステートメント {
    StatsMonitor ステートメントのパラメータ
    統計情報種別を指定した Stats ステートメント {
        Stats ステートメントのパラメータ
    }
}
}
SystemStateDaemon ステートメント {
    SystemStateDaemon ステートメントのパラメータ
}
}
}

```

正しく設定ができたかどうかは、管理コマンド (oraadmin) の check リクエストで確認をしてください。

管理コマンドの詳細は、「6 コマンドリファレンス」を参照してください。

注意: 上記の順で記述されていない場合は、設定が正しく行えないことがあります。

3 製品の設定

3.2.7. パラメーター一覧

全ノードステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	モニタ制御デーモン (oramond)、監視モニタ (instmond, lsnrmond, tschkmond, fdsmond, stbdbmond, altmond) および採取モニタ (statsmond, ssdump) のログの出力レベルを全ノード共通に指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。		○	○
	2	0 ~ 3		
COMPONENT_ID	構成番号を指定します。 同一ノード上で複数のインスタンス監視を行う場合は、設定ファイル (oramond.conf) を複数作成し、設定ファイルごとに異なる構成番号を指定してください。			×
	0	0 ~ 9		
MONTYPE	監視対象の Oracle のバージョンを指定します。 省略できません。	11G 11GR2 12C	×	×
NODE_NAME	監視対象の Oracle が動作するノード名を指定します。 指定できるノード数は、最大 16 ノードです。 詳細は、「3.2.9 ノード名の設定」を参照してください。		×	×
	省略できません。	なし		
SERVICE_PORT	モニタ制御デーモン (oramond) 間で通信に使用するポート番号を指定します。ADMIN_PORT パラメータと違う番号を指定することも可能です。 同一ノード上で複数のインスタンス監視を行う場合は、設定ファイル (oramond.conf) を複数作成し、設定ファイルごとに異なるポート番号を指定してください。 ポート番号が他のアプリケーションと競合する可能性がある環境では、競合しない番号を指定してください。			×
	25312	1024 ~ 65535		
ADMIN_PORT	モニタ制御デーモン (oramond) が管理コマンド (oraadmin) からの要求を受信するポート番号を指定します。 SERVICE_PORT パラメータと違う番号を指定することも可能です。 同一ノード上で複数のインスタンス監視を行う場合は、設定ファイル (oramond.conf) を複数作成し、設定ファイルごとに異なるポート番号を指定してください。 ポート番号が他のアプリケーションと競合する可能性がある環境では、競合しない番号を指定してください。			×
	25312	1024 ~ 65535		
INTERSITE_PORT	各サイトのモニタ制御デーモン (oramond) が通信に使用する受信用のポート番号を指定します。 ADMIN_PORT パラメータと異なるポート番号を指定してください。 同一ノード上で複数のインスタンス監視を行う場合は、設定ファイル (oramond.conf) を複数作成し、設定ファイルごとに異なるポート番号を指定してください。 ポート番号が他のアプリケーションと競合する可能性がある環境では、競合しない番号を指定してください。			○
	25311	1024 ~ 65535		

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
STANDBY_DB_CHECK	モニタ制御デーモン (oramond) およびスタンバイデータベース監視モニタ (stbdbmond) がスタンバイ・インスタンスを監視するかどうかを指定します。 Oracle Data Guard Broker 構成の環境で指定が有効なパラメータです。 詳細は、「3.2.10 スタンバイ・インスタンス監視の設定 (Oracle Data Guard Broker 構成のみ)」を参照してください。	YES NO	○	○
SWITCHOVER_TIMEOUT	モニタ制御デーモン (oramond) が Oracle Data Guard Broker による自動フェイルオーバーの完了を待ち合わせ、監視を再開するまでの時間を秒で全ノード共通に指定します。この待ち時間内にフェイルオーバーの処理が完了しない場合、システム障害として扱われます。	30 秒 1 ~ 3600	○	○
HALT_CONTROL_POLICY	停止する障害ノードの選択条件を指定します。 RAC 構成の環境で指定が有効なパラメータです。 詳細は、「3.2.11 停止ノード選択ポリシーの設定 (RAC 構成のみ)」を参照してください。	FIRST_FAILED FIRST_FAILED CONFIGURED	○	○
HALT_CONTROL_OPTION	障害ノード上の ApplicationMonitor の停止動作を指定します。 詳細は、「3.2.12 障害ノード上の ApplicationMonitor の停止動作の設定」を参照してください。	ALWAYS ALWAYS NEVER OUTLIVE	○	○
HALT_CONTROL_METHOD	障害ノード停止時に Oracle の構成プロセスに対して kill コマンドを実行するかどうかを指定します。 詳細は、「3.2.13 Oracle 構成プロセス停止の設定」を参照してください。	NONE KILL NONE	○	○
ORACLE_BASE	Oracle ディレクトリ構成の基本となるディレクトリを全ノード共通に指定します。db ユーザーの環境変数 ORACLE_BASE に設定されたパスを指定してください。 このパラメータを全ノードステートメントで省略した場合、すべての Node ステートメントで記述する必要があります。	なし なし	○	×
ORACLE_HOME	Oracle ソフトウェアをインストールしたディレクトリを全ノード共通に指定します。db ユーザーの環境変数 ORACLE_HOME に設定されたパスを指定してください。 このパラメータを全ノードステートメントで省略した場合、すべての Node ステートメントで記述する必要があります。	なし なし	○	×
ORA_NLS	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリを全ノード共通に指定します。db ユーザーの環境変数 ORA_NLS10 に設定されたパスを指定してください。 このパラメータを全ノードステートメントで省略した場合、すべての Node ステートメントで記述する必要があります。	なし なし	○	×

3 製品の設定

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
SHLIB_PATH	db ユーザー環境の 64bit 共有ライブラリのパスを全ノード共通に指定します。 このパラメータを全ノードステートメントで省略した場合、すべての Node ステートメントで記述する必要があります。		○	×
	なし	なし		
NLS_LANG	モニタ制御デーモン (oramond)、監視モニタ (instmond, lsnrmond, tschkmond, fdsmond, stbdblmond, altmond) および採取モニタ (statsmond, ssdump) が Oracle にアクセスする際の言語、地域およびキャラクタセットを全ノード共通に指定します。		○	×
	AMERICAN_AMERICA.US7ASCII	なし		
TNS_ADMIN	Oracle Net Services 構成が格納されているディレクトリを全ノード共通に指定します。db ユーザーの環境変数 TNS_ADMIN に設定されたパスを指定してください。		○	×
	指定なし	なし		
GRID_BASE	Oracle Grid Infrastructure のディレクトリ構成の基本となるディレクトリを指定します。grid ユーザーの環境変数 ORACLE_BASE に設定されたパスを指定してください。 Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境で指定が有効なパラメータです。		○	○
	指定なし	なし		
GRID_HOME	Oracle Grid Infrastructure をインストールしたディレクトリを指定します。grid ユーザーの環境変数 ORACLE_HOME に設定されたパスを指定してください。 Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境で指定が有効なパラメータです。		○	○
	指定なし	なし		
GRID_USER	障害検出時に実行する Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの実効ユーザー名を指定します。 Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境で指定が有効なパラメータです。		○	○
	指定なし	なし		
MONITOR_USER	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)、監視モニタ (instmond, lsnrmond, tschkmond, fdsmond, stbdblmond, altmond)、採取モニタ (statsmond, ssdump) および障害検出時に実行する Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの実効ユーザー名を全ノード共通に指定します。		○	×
	oracle	なし		
BOOT_TIMEOUT	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)、監視モニタ (instmond, lsnrmond, tschkmond, fdsmond, stbdblmond, altmond) および採取モニタ (statsmond, ssdump) の起動を確認するまでの待ち時間を秒で全ノード共通に指定します。この待ち時間内に起動できない場合、システム障害として扱われます。		○	○
	10 秒	1 ~ 3600		
ORACLE_USER	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)、監視モニタ (instmond, tschkmond, fdsmond, stbdblmond, altmond) および採取モニタ (statsmond, ssdump) が Oracle に接続するための Oracle ユーザー名を全ノード共通に指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。		○	○
	指定なし	なし		

3.2 設定ファイル (oramond.conf) の作成と適用

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
ORACLE_PASS	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)、監視モニタ (instmond, tschkmond, fdsmond, stbdblmond, altmond) および採取モニタ (statsmond, ssdump) が Oracle に接続するための Oracle ユーザーのパスワードを全ノード共通に指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	なし	○	○
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、SYSDBA 権限で接続するかどうかを全ノード共通に指定します。 モニタ制御デーモン (oramond) の Oracle Data Guard Broker プロセスの状態監視は、SYSDBA 権限で接続する必要があります。そのため、設定内容は Oracle Data Guard Broker プロセスの状態監視の設定には反映されません。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。	YES NO	○	○
PDB_NAME	監視モニタ (instmond, tschkmond, stbdblmond, altmond) および採取モニタ (statsmond, ssdump) が PDB 上の各対象を監視および採取する場合、PDB 名を全ノード共通に指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。	なし	○	○
MONITOR_CONTROL	モニタ制御デーモン (oramond) から監視モニタ (lsnrmond, tschkmond, fdsmond) および統計情報採取モニタ (statsmond) の制御を行うかどうかを全ノード共通に指定します。 詳細は、「3.2.17 監視モニタ制御の設定」を参照してください。	YES NO	○	○
SCRIPT_USER	障害検出時に実行するシェルスクリプトの実効ユーザー名を指定します。	root なし	○	○
SCRIPT_NAME	障害検出時に実行するシェルスクリプトのパス名(絶対パス)を指定します。 必ず実行権が付いていることを確認してください。	なし	○	○
SCRIPT_TIMEOUT	障害検出時に実行するシェルスクリプトの実行時間の上限を秒で指定します。	30 秒 1 ~ 3600	○	○
ORACLE_HANG_CHECK	監視対象 Oracle データベースのバージョンが 11g (11.1.0.7 以降)の際、Oracle ハング検知機能を利用した監視を行うかどうかを指定します。 詳細は、「3.2.18 Oracle ハング検知機能を利用した監視 (RAC 構成のみ)」を参照してください。	NO YES NO	○	○

3 製品の設定

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
CRS_CMD_TIMEOUT	障害検出時に実行する Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの実行時間の上限を秒で指定します。 Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境で指定が有効なパラメータです。		○	○
	90 秒	1 ~ 3600		
GET_STATS	監視モニタ (instmond, lsnrmond, tschmond) および統計情報採取モニタ (statsmond) で統計情報の採取を行うかどうかを全ノード共通に指定します。 詳細は、「3.2.19 統計情報採取の設定」を参照してください。		○	○
	YES	YES NO		
WAIT_ASM_RESTART	モニタ制御デーモン (oramond) が ASM インスタンス停止に伴うデータベース再起動の待ち合わせを行うかどうかを全ノード共通に指定します。 詳細は、「3.2.20 ASM インスタンス停止に伴うデータベース再起動の待ち合わせの設定」を参照してください。		○	○
	YES	YES NO		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
は、複数のインスタンスを監視する場合、設定ファイル (oramond.conf) 間で値が重複することがないように設定してください。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

DataGuardSite ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
DB_UNIQUE_NAME	各サイトのモニタ制御デーモン (oramond) が通信を行うための監視対象サイトのデータベース・インスタンス名を指定します。 Oracle の初期化パラメータ DB_UNIQUE_NAME に設定された値を指定してください。		×	○
	DataGuardSite ステートメントを指定した場合は、省略できません。	なし		
INTERSITE_PORT	各サイトのモニタ制御デーモン (oramond) が通信に使用する送信用のポート番号を指定します。 監視対象サイトの設定ファイル (oramond.conf) の受信信用 INTERSITE_PORT パラメータに指定したポート番号を指定してください。		○	○
	25311	1024 ~ 65535		
SITE_NODE	各サイトのモニタ制御デーモン (oramond) が通信を行うための監視対象サイトのノード名を指定します。NODE_NAME パラメータに指定した監視対象のノード名を指定してください。 指定できるノード数は、最大 16 ノードです。 監視対象サイト内のノード数分指定してください。 Destination ステートメント内の PRIMARY_NET_SERVICE_NAME パラメータと同様の順番になるよう指定してください。		×	○
	DataGuardSite ステートメントを指定した場合は、省略できません。	なし		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

3 製品の設定

Node ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	モニタ制御デーモン (oramond)、監視モニタ (instmond, lsnrmond, tschkmond, fdsmond, stbdblmond, altmond) および採取モニタ (statsmond, ssdump) のログの出力レベルをノードごとに指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。		○	○
	上位のステートメントに指定した値	0 ~ 3		
ORACLE_SID	Oracle のインスタンス識別子を指定します。Oracle の環境変数 ORACLE_SID に設定された値を指定してください。		×	×
	省略できません。	なし		
ORACLE_BASE	Oracle ディレクトリ構成の基本となるディレクトリをノードごとに指定します。db ユーザーの環境変数 ORACLE_BASE に設定されたパスを指定してください。		○	×
	上位のステートメントに指定したパス	なし		
ORACLE_HOME	Oracle ソフトウェアをインストールしたディレクトリノードごとに指定します。db ユーザーの環境変数 ORACLE_HOME に設定したパスを指定してください。		○	×
	上位のステートメントに指定したパス	なし		
ORA_NLS	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリをノードごとに指定します。db ユーザーの環境変数 ORA_NLS10 に設定されたパスを指定してください。		○	×
	上位のステートメントに指定したパス	なし		
SHLIB_PATH	db ユーザー環境の 64bit 共有ライブラリのパスをノードごとに指定します。		○	×
	上位のステートメントに指定したパス	なし		
NLS_LANG	モニタ制御デーモン (oramond)、監視モニタ (instmond, lsnrmond, tschkmond, stbdblmond, fdsmond, altmond) および採取モニタ (statsmond, ssdump) が Oracle にアクセスする際の言語、地域およびキャラクタ・セットをノードごとに指定します。		○	×
	上位のステートメントに指定した値	なし		
TNS_ADMIN	Oracle Net Services 構成が格納されているディレクトリをノードごとに指定します。db ユーザーの環境変数 TNS_ADMIN に設定されているパスを指定してください。		○	×
	上位のステートメントに指定したパス	なし		
MONITOR_USER	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)、監視モニタ (instmond, lsnrmond, tschkmond, fdsmond, stbdblmond, altmond)、採取モニタ (statsmond, ssdump) および障害検出時に実行する Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの実効ユーザー名をノードごとに指定します。		○	×
	上位のステートメントに指定した値	なし		
BOOT_TIMEOUT	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)、監視モニタ (instmond, lsnrmond, tschkmond, fdsmond, stbdblmond, altmond) および採取モニタ (statsmond, ssdump) の起動を確認するまでの待ち時間を秒でノードごとに指定します。この待ち時間内に起動できない場合、システム障害として扱われます。		○	○
	上位のステートメントに指定した値	1 ~ 3600		

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
ORACLE_USER	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)、監視モニタ (instmon, tschmon, fdsmon, stbmon, altmon) および採取モニタ (statsmon, ssdump) が Oracle に接続するための Oracle ユーザー名をノードごとに指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。		○	○
	上位のステートメントに指定した値	なし		
ORACLE_PASS	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)、監視モニタ (instmon, tschmon, fdsmon, stbmon, altmon) および採取モニタ (statsmon, ssdump) が Oracle に接続するための Oracle ユーザーのパスワードをノードごとに指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。		○	○
	上位のステートメントに指定した値	なし		
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、SYSDBA 権限で接続するかどうかをノードごとに指定します。 モニタ制御デーモン (oramond) の Oracle Data Guard Broker プロセスの状態監視は、SYSDBA 権限で接続する必要があります。そのため、設定内容は Oracle Data Guard Broker プロセスの状態監視の設定には反映されません。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。		○	○
	上位のステートメントに指定した値	YES NO		
PDB_NAME	監視モニタ (instmon, tschmon, stbmon, altmon) および採取モニタ (statsmon, ssdump) が PDB 上の各対象を監視および採取する場合、PDB 名をノードごとに指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。		○	○
	上位のステートメントに指定した値	なし		
MONITOR_CONTROL	モニタ制御デーモン (oramond) から監視モニタ (lsnrm, tschmon, fdsmon) および統計情報採取モニタ (statsmon) の制御を行うかどうかをノードごとに指定します。 詳細は、「3.2.17 監視モニタ制御の設定」を参照してください。		○	○
	上位のステートメントに指定した値	YES NO		
SCRIPT_USER	障害検出時に実行するスクリプトの実効ユーザー名を指定します。		○	○
	上位のステートメントに指定した値	なし		
SCRIPT_NAME	障害検出時に実行するシェルスクリプトのパス名 (絶対パス) を指定します。 必ず実行権が付いていることを確認してください。		○	○
	上位のステートメントに指定したパス	なし		
GET_STATS	監視モニタ (instmon, lsnrm, tschmon) および統計情報採取モニタ (statsmon) で統計情報の採取を行うかどうかをノードごとに指定します。 詳細は、「3.2.19 統計情報採取の設定」を参照してください。		○	○
	上位のステートメントに指定した値	YES NO		

3 製品の設定

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
WAIT_ASM_RESTART	モニタ制御デーモン (oramond) が ASM インスタンス停止に伴うデータベース再起動の待ち合わせを行うかどうかをノードごとに指定します。 詳細は、「3.2.20 ASM インスタンス停止に伴うデータベース再起動の待ち合わせの設定」を参照してください。		○	○
	上位のステートメントに指定した値	YES NO		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

InstanceMonitor ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	インスタンス監視モニタ (instmond) のログの出力レベルを指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。 上位のステートメントに指定した値	0 ~ 3	○	○
MONITOR_USER	インスタンス監視モニタ (instmond) の実効ユーザー名を指定します。 上位のステートメントに指定した値	なし	○	×
BOOT_TIMEOUT	インスタンス監視モニタ (instmond) の起動を確認するまでの待ち時間を秒で指定します。この待ち時間内に起動確認ができない場合、システム障害として扱われます。 上位のステートメントに指定した値	1 ~ 3600	○	○
ORACLE_USER	インスタンス監視モニタ (instmond) が Oracle に接続するための Oracle ユーザー名を指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
ORACLE_PASS	インスタンス監視モニタ (instmond) が Oracle に接続するための Oracle ユーザーのパスワードを指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、インスタンス監視モニタ (instmond) が SYSDBA 権限で接続するかどうかを指定します。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。 上位のステートメントに指定した値	YES NO	○	○
PDB_NAME	インスタンス監視モニタ (instmond) が PDB 上の表を使用して監視する場合、PDB 名を指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。 上位のステートメントに指定した値	なし	○	○
OWNER_NAME	インスタンス監視表のスキーマを指定します。 Oracle 12c の環境で指定が有効なパラメータです。 詳細は、「3.2.21 インスタンス監視表のスキーマの設定 (Oracle 12c のみ)」を参照してください。 監視対象の Oracle のバージョンが 12c の場合、省略できません。	なし	×	○
UNOPEN_PDB_DOWN	インスタンス監視モニタ (instmond) が PDB 上の表を使用して監視し、Table ステートメントを記述して特定の表を監視する場合、PDB の状態が OPEN 以外なら障害として扱うかどうかを指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。 詳細は、「3.2.22 指定表監視時の障害の設定 (Oracle 12c のマルチテナント構成のみ)」を参照してください。 YES	YES NO	○	○

3 製品の設定

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
POLL_INTERVAL	インスタンスを監視する間隔を秒で指定します。 このパラメータは、すべてのノードで同じ値を指定してください。		○	○
	90 秒	5 ~ 2678400		
POLL_TIMEOUT	インスタンスのストールと判断するまでの上限時間を秒で指定します。 このパラメータは、監視開始時、監視対象の Oracle データベースの環境に合わせて、自動的に調整される場合があります。		○	○
	120 秒	1 ~ 86400		
REFORM_TIMEOUT	Oracle の再構成が完了するまでの最大待ち合わせ時間を秒で指定します。Oracle の再構成中は監視を停止しますが、このパラメータに指定された秒数を経過した際、監視を再開します。		○	○
	30 秒	1 ~ 3600		
POLL_DOWN_RETRY	インスタンスのダウン検出時、インスタンスを再確認する回数を指定します。 RAC 構成の環境では、すべてのノードで同じ値を指定してください。		○	○
	0 回	0 ~ 10		
POLL_STALL_RETRY	インスタンスのストール検出時、インスタンスを再確認する回数を指定します。 RAC 構成の環境では、すべてのノードで同じ値を指定してください。		○	○
	0 回	0 ~ 10		
SERVICE_DOWN	インスタンスの障害時、RESTART_COUNT パラメータに設定された回数のインスタンス再起動を行っても再起動できなかった場合に、CLUSTERPRO へ障害を通知するかどうかを指定します。 詳細は、「3.2.23 インスタンス / リスナー 障害時の CLUSTERPRO への障害通知の設定」を参照してください。		○	○
	YES	YES NO NO_DOWN NO_STALL		
RESTART_COUNT	インスタンスの障害時、インスタンスの再起動を待ち合わせる回数を指定します。 Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境で指定が有効なパラメータです。		○	○
	2 回	0 ~ 999		
RESTART_TIMEOUT	障害検出後、次の監視を行うまでの待ち合わせ時間を秒で指定します。 Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境で指定が有効なパラメータです。		○	○
	60 秒	0 ~ 3600		
GET_STATS	インスタンス監視モニタ (instmond) で統計情報の採取を行うかどうかを指定します。 詳細は、「3.2.19 統計情報採取の設定」を参照してください。		○	○
	上位のステートメントに指定した値	YES NO		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

3 製品の設定

ListenerMonitor ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	リスナー監視モジュール (lsnrmond) のログの出力レベルを指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。 上位のステートメントに指定した値	0 ~ 3	○	○
MONITOR_USER	リスナー監視モジュール (lsnrmond) および障害検出時に実行する Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの実効ユーザー名を指定します。 上位のステートメントに指定した値	なし	○	×
BOOT_TIMEOUT	リスナー監視モジュール (lsnrmond) の起動を確認するまでの待ち時間を秒で指定します。この待ち時間内に起動確認ができない場合、システム障害として扱われます。 上位のステートメントに指定した値	1 ~ 3600	○	○
MONITOR_CONTROL	モジュール制御デーモン (oramond) からリスナー監視モジュール (lsnrmond) の制御を行うかどうかを指定します。 詳細は、「3.2.17 監視モジュール制御の設定」を参照してください。 上位のステートメントに指定した値	YES NO	○	○
POLL_INTERVAL	リスナーを監視する間隔を秒で指定します。 60 秒	5 ~ 2678400	○	○
POLL_TIMEOUT	リスナーの障害と判断するまでの上限時間を秒で指定します。 90 秒	1 ~ 86400	○	○
SERVICE_DOWN	リスナーの障害時、RESTART_COUNT パラメータに設定された回数のリスナー再起動を行っても再起動できなかった場合に、CLUSTERPRO へ障害を通知するかどうかを指定します。 詳細は、「3.2.23 インスタンス / リスナー障害時の CLUSTERPRO への障害通知の設定」を参照してください。 NO	YES NO NO_DOWN NO_STALL	○	○
RESTART_COUNT	リスナーの障害時、リスナーの再起動を行う回数を指定します。 5 回	0 ~ 999	○	○
RESTART_INTERVAL	リスナーの再起動失敗から次に再起動を行うまでの間隔を秒で指定します。 3 秒	1 ~ 10	○	○
NET_SERVICE_NAME	リスナーのネット・サービス名を指定します。 ListenerMonitor ステートメントにリスナー名を指定した場合は、省略できません。	なし	×	×
TNSPING	Oracle の tnsping コマンドのコマンド名を絶対パスで指定します。 \${ORACLE_HOME}/bin/tnsping	なし	○	○
LSNRCTL	リスナー再起動シェルスクリプトを絶対パスで指定します。 /etc/opt/HA/AM/bin/lsnr_control.sh	なし	○	○
RESTART_TIMEOUT	障害検出後、次の監視を行うまでの待ち合わせ時間を秒で指定します。 Oracle 11g R2 の Oracle Clusterware / Oracle Restart 環境で指定が有効なパラメータです。 10 秒	0 ~ 3600	○	○

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
TARGET_RESTART	障害検出時、ApplicationMonitor によるリスナーの再起動を行うかどうかを指定します。 詳細は、「3.2.24 リスナー再起動方法の設定」を参照してください。	YES NO	○	○
ORACLE_BASE	Oracle または Oracle Grid Infrastructure のディレクトリ構成の基本となるディレクトリを指定します。db ユーザーまたは grid ユーザーの環境変数 ORACLE_BASE に設定されたパスを指定してください。	上位のステートメントに指定したパス なし	○	×
ORACLE_HOME	Oracle または Oracle Grid Infrastructure をインストールしたディレクトリを指定します。db ユーザーまたは grid ユーザーの環境変数 ORACLE_HOME に設定されたパスを指定してください。	上位のステートメントに指定したパス なし	○	×
ORA_NLS	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリを指定します。db ユーザーまたは grid ユーザーの環境変数 ORA_NLS10 に設定されたパスを指定してください。	上位のステートメントに指定したパス なし	○	×
SHLIB_PATH	db ユーザーまたは grid ユーザー環境の 64bit 共有ライブラリのパスを指定します。	上位のステートメントに指定したパス なし	○	×
NLS_LANG	リスナー監視モジュール (lsnrmond) が Oracle にアクセスする際の言語、地域およびキャラクタ・セットを指定します。	上位のステートメントに指定した値 なし	○	×
TNS_ADMIN	Oracle Net Services 構成が格納されているディレクトリを指定します。db ユーザーまたは grid ユーザーの環境変数 TNS_ADMIN に設定されたパスを指定してください。	上位のステートメントに指定したパス なし	○	×
UP_DOWN_RETRY	リスナー監視時、動作環境が不安定な場合など、リスナーを再起動しても障害を繰り返す際に、繰り返し障害を障害と判断する上限回数を指定します。 詳細は、「3.2.25 リスナーの繰り返し障害の監視」を参照してください。	5 回 0 ~ 10	○	○
UPTIME_COUNT	リスナー監視時、リスナーが安定稼動状態と判断するまでの回数を指定します。 詳細は、「3.2.25 リスナーの繰り返し障害の監視」を参照してください。	1 回 1 ~ 60	○	○
GET_STATS	リスナー監視モジュール (lsnrmond) で統計情報の採取を行うかどうかを指定します。 詳細は、「3.2.19 統計情報採取の設定」を参照してください。	上位のステートメントに指定した値 YES NO	○	○

1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。

2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

3 製品の設定

ListenerMonitor ステートメントに指定するリスナー名は、以下を参照してください。

`${ORACLE_HOME}/network/admin/listener.ora`

```
LISTENER =      ←この部分がリスナー名になります
(DESCRIPTION_LIST =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = sales-server)(PORT = 1521))
    )
  )
)
```

NET_SERVICE_NAME パラメータに指定するリスナーのネット・サービス名は、以下を参照してください。

`${ORACLE_HOME}/network/admin/tnsnames.ora`

```
SALES =          ←この部分がネット・サービス名になります
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = sales-server)(PORT = 1521))
  (CONNECT_DATA =
    (SERVICE_NAME = sales.xxx.yyy.nec.co.jp))
)
```

注意： ListenerMonitor ステートメントのリスナー名およびリスナー監視モニタ数については、動的な設定変更はできません。モニタ制御デーモン (oramond) を停止してから設定の変更をしてください。

注意： リモートからリスナー制御ユーティリティ・コマンド (lsnrctl) を実行する場合、パスワードを設定する必要があります。

ApplicationMonitor では、設定ファイル (oramond.conf) の確認時、lsnrctl コマンドにより、リスナーの存在確認を行っているため、リスナー名にフローティング IP を使用したリスナーを指定した場合、設定ファイルの確認が失敗する可能性があります。フローティング IP を使用したリスナーの監視設定の確認および設定配布(動的変更含む)時にリクエストが失敗する場合は、管理コマンド (oraadmin) の "-F" オプションにより、Oracle 関連の確認を省略してください。

管理コマンドの詳細は、「6 コマンドリファレンス」を参照してください。

もしくは、以下の設定を行い、エラーの発生を回避してください。

1. あらかじめ、lsnrctl コマンドを実行するノード上で、lsnrctl を使用してパスワードを設定してください。

```
# lsnrctl
LSNRCTL> CHANGE_PASSWORD LISTENER ... 監視対象のリスナー名を指定
Old password:
New password: lsnrpasswd ... 設定するパスワードを入力
Reenter new password: lsnrpasswd ... 設定するパスワードを再入力
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=tpc)(HOST=sales-server)(PORT=1521)))
Password changed for LISTENER
The command completed successfully
LSNRCTL> SAVE_CONFIG LISTENER ... 監視対象のリスナー名を指定

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)))
Saved LISTENER configuration parameters.
Listener Parameter File /oracle/network/admin/listener.ora
Old Parameter File /oracle/network/admin/listener.bak
The command completed successfully
LSNRCTL> EXIT
#
```

入力時にパスワードは表示されません。

2. リスナー再起動シェルスクリプト (/etc/opt/HA/AM/lbin/lsnr_control.sh) の LISTENER_PASSWD に 1. で設定したパスワードを指定してください。

```
...
LISTENER_PASSWD=lsnrpasswd ... ここにパスワードを設定
...
```

複数のリスナー監視を行う場合は、リスナー再起動シェルスクリプトをコピーし、監視対象とするリスナーに合わせてパスワードを設定した上で、使用してください。

また、修正したリスナー再起動シェルスクリプトは、それぞれのリスナー監視モニタの LSNRCTL パラメータに設定してください。

設定方法の詳細は、Oracle のマニュアル「Oracle Database Net Services 管理者ガイド」を参照してください。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

3 製品の設定

TableSpaceMonitor ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	表領域監視モニタ (tschkmond) のログの出力レベルを指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。	上位のステートメントに指定した値	0 ~ 3	○ ○
MONITOR_USER	表領域監視モニタ (tschkmond) の実効ユーザー名を指定します。	上位のステートメントに指定した値	なし	○ ×
BOOT_TIMEOUT	表領域監視モニタ (tschkmond) の起動を確認するまでの待ち時間を秒で指定します。 この待ち時間内に起動確認ができない場合、システム障害として扱われます。	上位のステートメントに指定した値	1 ~ 3600	○ ○
ORACLE_USER	表領域監視モニタ (tschkmond) が Oracle に接続するための Oracle ユーザー名を指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	上位のステートメントに指定した値	なし	○ ○
ORACLE_PASS	表領域監視モニタ (tschkmond) が Oracle に接続するための Oracle ユーザーのパスワードを指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	上位のステートメントに指定した値	なし	○ ○
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、表領域監視モニタ (tschkmond) が SYSDBA 権限で接続するかどうかを指定します。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。	上位のステートメントに指定した値	YES NO	○ ○
PDB_NAME	表領域監視モニタ (tschkmond) が PDB 上の表領域を監視する場合、PDB 名を指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。	上位のステートメントに指定した値	なし	○ ○
MONITOR_CONTROL	モニタ制御デーモン (oramond) から表領域監視モニタ (tschkmond) の制御を行うかどうかを指定します。 詳細は、「3.2.17 監視モニタ制御の設定」を参照してください。	上位のステートメントに指定した値	YES NO	○ ○
POLL_INTERVAL	表領域の容量を監視する間隔を秒で指定します。	3600 秒	5 ~ 2678400	○ ○
POLL_TIMEOUT	表領域の容量監視が完了するまで、またはインスタンスのストールと判断するまでの上限時間を秒で指定します。	60 秒	1 ~ 86400	○ ○
COMMAND_NAME	表領域の使用可能エクステント、もしくは空き容量率が指定したしきい値を下回った時に実行するコマンドを絶対パスで指定します。 必ず実行権が付いていることを確認してください。 コマンドがシェルスクリプトの場合は、スクリプトの先頭行に使用するシェルが記述されていることを確認してください。	指定なし	なし	○ ○

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
GET_STATS	表領域監視モニタ (tschkmond) で統計情報の採取を行うかどうかを指定します。 詳細は、「3.2.19 統計情報採取の設定」を参照してください。		○	○
	上位のステートメントに指定した値	YES NO		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

注意: TableSpaceMonitor ステートメントの表領域監視モニタ ID、表領域監視モニタ数については、動的な設定変更はできません。モニタ制御デーモン (oramond) を停止してから設定の変更をしてください。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

3 製品の設定

DiskSpaceMonitor ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	ディスク領域監視モニタ (fdsmond) のログの出力レベルを指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。 上位のステートメントに指定した値	0 ~ 3	○	○
MONITOR_USER	ディスク領域監視モニタ (fdsmond) の実効ユーザー名を指定します。 上位のステートメントに指定した値	なし	○	×
BOOT_TIMEOUT	ディスク領域監視モニタ (fdsmond) の起動を確認するまでの待ち時間を秒で指定します。 この待ち時間内に起動確認ができない場合、システム障害として扱われます。 上位のステートメントに指定した値	1 ~ 3600	○	○
ORACLE_USER	ディスク領域監視モニタ (fdsmond) が Oracle に接続するための Oracle ユーザー名を指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
ORACLE_PASS	ディスク領域監視モニタ (fdsmond) が Oracle に接続するための Oracle ユーザーのパスワードを指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、ディスク領域監視モニタ (fdsmond) が SYSDBA 権限で接続するかどうかを指定します。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。 上位のステートメントに指定した値	YES NO	○	○
MONITOR_CONTROL	モニタ制御デーモン (oramond) からディスク領域監視モニタ (fdsmond) の制御を行うかどうかを指定します。 詳細は、「3.2.17 監視モニタ制御の設定」を参照してください。 上位のステートメントに指定した値	YES NO	○	○
POLL_INTERVAL	ローカルディスク領域の容量を監視する間隔を秒で指定します。 3600 秒	5 ~ 2678400	○	○
POLL_TIMEOUT	ローカルディスク領域の容量監視が完了するまで、またはインスタンスのストールと判断するまでの上限時間を秒で指定します。 60 秒	1 ~ 86400	○	○
COMMAND_NAME	ローカルディスク領域の空き容量率が指定したしきい値を下回った時に実行するコマンドを絶対パスで指定します。 必ず実行権が付いていることを確認してください。 コマンドがシェルスクリプトの場合は、スクリプトの先頭行に使用するシェルが記述されていることを確認してください。 指定なし	なし	○	○
FREE_SPACE	ローカルディスク領域の空き容量率のしきい値を割合で指定します。 10 %	0 ~ 100	○	○

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

3 製品の設定

StandbyDatabaseMonitor ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	スタンバイデータベース監視モニタ (stbdbmond) のログの出力レベルを指定します。 モニタ制御デーモン (oramond) による Oracle Data Guard Broker プロセスの状態監視は、全ノードステートメントまたは Node ステートメントの LOG_LEVEL パラメータの設定で動作します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。 上位のステートメントに指定した値	0～3	○	○
MONITOR_USER	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)およびスタンバイデータベース監視モニタ (stbdbmond) の実効ユーザー名を指定します。 上位のステートメントに指定した値	なし	○	×
BOOT_TIMEOUT	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)およびスタンバイデータベース監視モニタ (stbdbmond) の起動を確認するまでの待ち時間を秒で指定します。 この待ち時間内に起動確認ができない場合、システム障害として扱われます。 上位のステートメントに指定した値	1～3600	○	○
ORACLE_USER	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)およびスタンバイデータベース監視モニタ (stbdbmond) が Oracle に接続するための Oracle ユーザー名を指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
ORACLE_PASS	モニタ制御デーモン (oramond) (スタンバイ・インスタンス監視時)およびスタンバイデータベース監視モニタ (stbdbmond) が Oracle に接続するための Oracle ユーザーのパスワードを指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、スタンバイデータベース監視モニタ (stbdbmond) が SYSDBA 権限で接続するかどうかを指定します。 モニタ制御デーモン (oramond) による Oracle Data Guard Broker プロセスの状態監視は、SYSDBA 権限で接続する必要があります。そのため、設定内容は Oracle Data Guard Broker プロセスの状態監視の設定には反映されません。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。 上位のステートメントに指定した値	YES NO	○	○
PDB_NAME	スタンバイデータベース監視モニタ (stbdbmond) がスタンバイ・インスタンスの PDB 上の表を使用して監視する場合、PDB 名を指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。 上位のステートメントに指定した値	なし	○	○

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
POLL_INTERVAL	スタンバイ・インスタンス (REDO ログ転送状況、REDO ログ適用状況、Oracle Data Guard Broker プロセスの状態) を監視する間隔を秒で指定します。 60 秒	1 ~ 2678400	○	○
POLL_TIMEOUT	モニタ制御デーモン (oramond) が Oracle Data Guard Broker プロセスの状態を障害と判断するまで、またはスタンバイデータベース監視モニタ (sttdbmond) が REDO ログ転送状況、REDO ログ適用状況の監視を完了するまでの上限時間を秒で指定します。 90 秒	1 ~ 86400	○	○
RETRY_COUNT	REDO ログ転送状況、Oracle Data Guard Broker プロセス状態の障害時または REDO ログ適用状況の監視が完了しない場合、スタンバイ・インスタンスの状態を再確認する回数を指定します。 0 回	0 ~ 10	○	○
COMMAND_NAME	スタンバイ・インスタンス (REDO ログ転送状況、REDO ログ適用状況、Oracle Data Guard Broker プロセスの状態) の障害時に実行するコマンドを絶対パスで指定します。 必ず実行権が付いていることを確認してください。 コマンドがシェルスクリプトの場合は、スクリプトの先頭行に使用するシェルが記述されていることを確認してください。 指定なし	なし	○	○

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

3 製品の設定

StatsMonitor ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	統計情報採取モニタ (statsmond) のログの出力レベルを指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。	上位のステートメントに指定した値	○	○
		0 ~ 3		
MONITOR_USER	統計情報採取モニタ (statsmond) の実効ユーザー名を指定します。	上位のステートメントに指定した値	○	×
		なし		
BOOT_TIMEOUT	統計情報採取モニタ (statsmond) の起動を確認するまでの待ち時間を秒で指定します。 この待ち時間内に起動確認ができない場合、システム障害として扱われます。	上位のステートメントに指定した値	○	○
		1 ~ 3600		
ORACLE_USER	統計情報採取モニタ (statsmond) が Oracle に接続するための Oracle ユーザー名を指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	上位のステートメントに指定した値	○	○
		なし		
ORACLE_PASS	統計情報採取モニタ (statsmond) が Oracle に接続するための Oracle ユーザーのパスワードを指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	上位のステートメントに指定した値	○	○
		なし		
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、統計情報採取モニタ (statsmond) が SYSDBA 権限で接続するかどうかを指定します。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。	上位のステートメントに指定した値	○	○
		YES NO		
PDB_NAME	統計情報採取モニタ (statsmond) が PDB 上の統計情報を採取する場合、PDB 名を指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。	上位のステートメントに指定した値	○	○
		なし		
MONITOR_CONTROL	モニタ制御デーモン (oramond) から統計情報採取モニタ (statsmond) の制御を行うかどうかを指定します。 詳細は、「3.2.17 監視モニタ制御の設定」を参照してください。	上位のステートメントに指定した値	○	○
		YES NO		
POLL_INTERVAL	統計情報を採取する間隔を秒で指定します。	3600 秒	○	○
		5 ~ 2678400		
POLL_TIMEOUT	統計情報の採取が完了するまで、またはインスタンスのストールと判断するまでの上限時間を秒で指定します。	60 秒	○	○
		1 ~ 86400		
GET_STATS	統計情報採取モニタ (statsmond) で統計情報の採取を行うかどうかを指定します。 詳細は、「3.2.19 統計情報採取の設定」を参照してください。	上位のステートメントに指定した値	○	○
		YES NO		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

注意: StatsMonitor ステートメントの統計情報採取モニタ ID、統計情報採取モニタ数については、動的な設定変更はできません。モニタ制御デーモン (oramond) を停止してから設定の変更をしてください。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

3 製品の設定

SystemStateDaemon ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
LOG_LEVEL	システムステートダンプ採取デーモン (ssdump) のログの出力レベルを指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。	上位のステートメントに指定した値	○	○
		0 ~ 3		
MONITOR_USER	システムステートダンプ採取デーモン (ssdump) の実効ユーザー名を指定します。	上位のステートメントに指定した値	○	×
		なし		
BOOT_TIMEOUT	システムステートダンプ採取デーモン (ssdump) の起動を確認するまでの待ち時間を秒で指定します。 この待ち時間内に起動確認ができない場合、システム障害として扱われます。	上位のステートメントに指定した値	○	○
		1 ~ 3600		
ORACLE_USER	システムステートダンプ採取デーモン (ssdump) が Oracle に接続するための Oracle ユーザー名を指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	上位のステートメントに指定した値	○	○
		なし		
ORACLE_PASS	システムステートダンプ採取デーモン (ssdump) が Oracle に接続するための Oracle ユーザーのパスワードを指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	上位のステートメントに指定した値	○	○
		なし		
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、システムステートダンプ採取デーモン (ssdump) が SYSDBA 権限で接続するかどうかを指定します。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。	上位のステートメントに指定した値	○	○
		YES NO		
PDB_NAME	システムステートダンプ採取デーモン (ssdump) が PDB 上のシステムステート・ダンプを採取する場合、PDB 名を指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。	上位のステートメントに指定した値	○	○
		なし		
POLL_INTERVAL	インスタンスとの接続が切れていないか確認する間隔を秒で指定します。	60 秒	○	○
		5 ~ 2678400		
POLL_TIMEOUT	インスタンスのストールと判断するまでの上限時間を秒で指定します。	60 秒	○	○
		1 ~ 86400		
GET_DUMP	システムステート・ダンプを採取するかどうかを指定します。	YES	○	○
		YES NO		
DUMP_INTERVAL	システムステート・ダンプを採取する間隔を秒で指定します。	30 秒	○	○
		1 ~ 60		
DUMP_COUNT	システムステート・ダンプを採取する回数を指定します。	3 回	○	○
		1 ~ 10		
DUMP_LEVEL	採取するシステムステート・ダンプのレベルを指定します。	10	○	○
		0 ~ 266		

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
DUMP_TIMEOUT	システムステート・ダンプを採取する上限時間を秒で指定します。 このパラメータは、全ノードステートメント内の SystemStateDaemon ステートメントにのみ指定できます。		○	○
	90 秒	1 ~ 3600		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

個々のパラメータの関係の詳細は、「4 製品の動作」を参照してください。

3 製品の設定

Table ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
TABLE_USER	表にアクセス可能なユーザー名を指定します。 指定できる文字列は、半角で最大 30 文字です。 ユーザー名にスペース、小文字を含める場合、ダブルクォーテーション (") で囲んでください。		×	○
	Table ステートメントを指定した場合は、省略できません。	なし		
TABLE_PASSWORD	表にアクセス可能なユーザー名のパスワードを指定します。 指定できる文字列は、半角で最大 30 文字です。 詳細は、「3.2.15 Oracle パスワードの扱い」を参照してください。		×	○
	Table ステートメントを指定した場合は、省略できません。 Oracle 12c のマルチテナント構成で PDB_NAME パラメータを指定した場合は、省略できます。	なし		
PDB_NAME	PDB 上の表を監視する場合、PDB 名を指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。		○	○
	上位のステートメントに指定した値	なし		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

TableSpace ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
USEABLE_EXTENT_BLOCK	表領域の使用可能エクステントのしきい値をブロック数で指定します。0 を指定した場合、使用可能エクステントサイズの監視を行いません。		○	○
	0 ブロック	0 ~ 1048576		
FREE_SPACE	表領域の空き容量率のしきい値を割合で指定します。		○	○
	20 %	0 ~ 100		
GET_STATS	統計情報の採取を行うかどうかを指定します。 詳細は、「3.2.19 統計情報採取の設定」を参照してください。		○	○
	上位のステートメントに指定した値	YES NO		
PDB_NAME	PDB 上の表領域を監視する場合、PDB 名を指定します。 Oracle 12c のマルチテナント構成の環境で指定が有効なパラメータです。		○	○
	上位のステートメントに指定した値	なし		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

3 製品の設定

Directory ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
COMMAND_NAME	ローカルディスク領域の空き容量率が指定したしきい値を下回った時に実行するコマンドを絶対パスで指定します。 必ず実行権が付いていることを確認してください。 コマンドがシェルスクリプトの場合は、スクリプトの先頭行に使用するシェルが記述されていることを確認してください。		○	○
	指定なし	なし		
FREE_SPACE	ローカルディスク領域の空き容量率の閾値を割合で指定します。		○	○
	上位のステートメントに指定した値	0 ~ 100		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

Transport ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
POLL_INTERVAL	スタンバイ・インスタンス (REDO ログ転送状況) を監視する間隔を秒で指定します。 上位のステートメントに指定した値	1 ~ 2678400	○	○
POLL_TIMEOUT	スタンバイ・インスタンス (REDO ログ転送状況) の監視が完了するまでの上限時間を秒で指定します。 上位のステートメントに指定した値	1 ~ 86400	○	○
RETRY_COUNT	スタンバイ・インスタンス (REDO ログ転送状況) の障害時、スタンバイ・インスタンスの状態を再確認する回数を指定します。 上位のステートメントに指定した値	0 ~ 10	○	○
COMMAND_NAME	スタンバイ・インスタンス (REDO ログ転送状況) の障害時に実行するコマンドを絶対パスで指定します。 必ず実行権が付いていることを確認してください。 コマンドがシェルスクリプトの場合は、スクリプトの先頭行に使用するシェルが記述されていることを確認してください。 上位のステートメントに指定した値	なし	○	○
PRIMARY_ORACLE_USER	スタンバイデータベース監視モータ (stbdbmond) が REDO ログ転送状況を監視する際に、プライマリ・サイトの Oracle に接続するための Oracle ユーザー名を全サイト共通に指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した ORACLE_USER パラメータの値	なし	○	○
PRIMARY_ORACLE_PASS	スタンバイデータベース監視モータ (stbdbmond) が REDO ログ転送状況を監視する際に、プライマリ・サイトの Oracle に接続するための Oracle ユーザーのパスワードを全サイト共通に指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した ORACLE_PASS パラメータの値	なし	○	○
PRIMARY_SYSDBA_CONNECT	PRIMARY_ORACLE_USER パラメータを指定した際、スタンバイデータベース監視モータ (stbdbmond) が SYSDBA 権限で接続するかどうかを全サイト共通に指定します。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。 上位のステートメントに指定した SYSDBA_CONNECT パラメータの値	YES NO	○	○

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

3 製品の設定

Destination ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
PRIMARY_NET_SERVICE_NAME	スタンバイデータベース監視モニタ (stbdbmond) が REDO ログ転送状況を監視する際に、スタンバイ・サイトからプライマリ・サイトヘリスナー経由で接続するためのネット・サービス名をサイトごとに指定します。 指定できるネット・サービス数は、最大 16 個です。 監視対象サイト内のノード数分指定してください。 DataGuardSite ステートメント内の SITE_NODE パラメータと同様の順番になるよう指定してください。	なし	×	○
PRIMARY_ORACLE_USER	スタンバイデータベース監視モニタ (stbdbmond) が REDO ログ転送状況を監視する際に、プライマリ・サイトの Oracle に接続するための Oracle ユーザー名をサイトごとに指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	なし	○	○
PRIMARY_ORACLE_PASS	スタンバイデータベース監視モニタ (stbdbmond) が REDO ログ転送状況を監視する際に、プライマリ・サイトの Oracle に接続するための Oracle ユーザーのパスワードをサイトごとに指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。	なし	○	○
PRIMARY_SYSDBA_CONNECT	PRIMARY_ORACLE_USER パラメータを指定した際、スタンバイデータベース監視モニタ (stbdbmond) が SYSDBA 権限で接続するかどうかをサイトごとに指定します。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。	YES NO	○	○

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

RedoApply ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
POLL_INTERVAL	スタンバイ・インスタンス (REDO ログ適用状況) を監視する間隔を秒で指定します。 上位のステートメントに指定した値	1 ~ 2678400	○	○
POLL_TIMEOUT	スタンバイ・インスタンス (REDO ログ適用状況) の監視が完了するまでの上限時間を秒で指定します。 上位のステートメントに指定した値	1 ~ 86400	○	○
RETRY_COUNT	スタンバイ・インスタンス (REDO ログ適用状況) の監視が完了しない場合、スタンバイ・インスタンスの状態を再確認する回数を指定します。 上位のステートメントに指定した値	0 ~ 10	○	○
COMMAND_NAME	スタンバイ・インスタンス (REDO ログ適用状況) の障害時に実行するコマンドを絶対パスで指定します。 必ず実行権が付いていることを確認してください。 コマンドがシェルスクリプトの場合は、スクリプトの先頭行に使用するシェルが記述されていることを確認してください。 上位のステートメントに指定した値	なし	○	○
ORACLE_USER	スタンバイデータベース監視モジュール (stbdbmon) が REDO ログ適用状況を監視する際に、Oracle に接続するための Oracle ユーザー名を指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
ORACLE_PASS	スタンバイデータベース監視モジュール (stbdbmon) が REDO ログ適用状況を監視する際に、Oracle に接続するための Oracle ユーザーのパスワードを指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、スタンバイデータベース監視モジュール (stbdbmon) が SYSDBA 権限で接続するかどうかを指定します。 詳細は、「3.2.16 パスワード・ファイル認証での接続」を参照してください。 上位のステートメントに指定した値	YES NO	○	○
ALLOWABLE_TIME	プライマリ・サイトからスタンバイ・サイトへの REDO ログ適用の遅延許可時間を秒で指定します。 60 秒	0 ~ 3600	○	○

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

3 製品の設定

OdgBroker ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
POLL_INTERVAL	スタンバイ・インスタンス (Oracle Data Guard Broker プロセスの状態) を監視する間隔を秒で指定します。 上位のステートメントに指定した値	1 ~ 2678400	○	○
POLL_TIMEOUT	スタンバイ・インスタンス (Oracle Data Guard Broker プロセスの状態) の異常と判断するまでの上限時間を秒で指定します。 上位のステートメントに指定した値	1 ~ 86400	○	○
RETRY_COUNT	スタンバイ・インスタンス (Oracle Data Guard Broker プロセスの状態) の障害時、スタンバイ・インスタンスの状態を再確認する回数を指定します。 上位のステートメントに指定した値	0 ~ 10	○	○
COMMAND_NAME	スタンバイ・インスタンス (Oracle Data Guard Broker プロセスの状態) の障害時に実行するコマンドを絶対パスで指定します。 必ず実行権が付いていることを確認してください。 コマンドがシェルスクリプトの場合は、スクリプトの先頭行に使用するシェルが記述されていることを確認してください。 上位のステートメントに指定した値	なし	○	○
BROKER_ORACLE_USER	モニタ制御デーモン (oramond) が Oracle Data Guard Broker プロセスの状態を監視する際に、Oracle に接続するための Broker コマンド用の Oracle ユーザー名を指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○
BROKER_ORACLE_PASS	モニタ制御デーモン (oramond) が Oracle Data Guard Broker プロセスの状態を監視する際に、Oracle に接続するための Broker コマンド用の Oracle ユーザーのパスワードを指定します。 詳細は、「3.2.14 Oracle DBA 監査 (audit) 機能対応の設定」を参照してください。 上位のステートメントに指定した値	なし	○	○

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

Stats ステートメントに記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
POLL_INTERVAL	統計情報を採取する間隔を秒で指定します。		○	○
	上位のステートメントに指定した値	5 ~ 2678400		
POLL_TIMEOUT	統計情報の採取が完了するまで、またはインスタンスのストールと判断するまでの上限時間を秒で指定します。		○	○
	上位のステートメントに指定した値	1 ~ 86400		
GET_STATS	統計情報の採取を行うかどうかを指定します。 詳細は、「3.2.19 統計情報採取の設定」を参照してください。		○	○
	上位のステートメントに指定した値	YES NO		

- 1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。
- 2 モニタ制御デーモン (oramond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

3.2.8. ログレベルの設定

ログの出力レベルを設定するには、LOG_LEVEL パラメータを指定します。ログの出力レベルおよび syslog に出力されるレベルとの関係は、以下のとおりです。

LOG_LEVEL=0	エラーメッセージのみ出力します。 エラーメッセージは、syslog レベル LOG_ERR に対応します。
LOG_LEVEL=1	LOG_LEVEL=0 で出力されるメッセージに加え、警告メッセージを出力します。 警告メッセージは、syslog レベル LOG_WARNING に対応します。
LOG_LEVEL=2	LOG_LEVEL=1 で出力されるメッセージに加え、通知メッセージを出力します。 通知メッセージは、syslog レベル LOG_INFO に対応します。
LOG_LEVEL=3	LOG_LEVEL=2 で出力されるメッセージに加え、デバッグメッセージを出力します。 デバッグメッセージは、syslog レベル LOG_DEBUG に対応します。

出力されるメッセージのファシリティは、LOG_DAEMON です。ファシリティおよびレベルの詳細は、オンラインマニュアルの syslog(3) を参照してください。

出力されるメッセージの詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 1.2 for Linux メッセージ一覧』を参照してください。

注意: StandbyDatabaseMonitor ステートメントの LOG_LEVEL パラメータは、スタンバイデータベース監視モジュール (stbdbmond) のみの設定パラメータです。モニタ制御デーモンによる Oracle Data Guard Broker プロセスの状態を監視する場合は、全ノードステートメントまたは Node ステートメントの LOG_LEVEL パラメータの設定で動作します。

3.2.9. ノード名の設定

ApplicationMonitor では、監視対象の Oracle が動作するノード名の指定が必須です。ノード名は、NODE_NAME パラメータにて指定します。

ノード名には、IP アドレスも指定できます。IP アドレスは、IPv4 形式(例:192.168.0.1)または IPv6 形式(例:2fee:1212::200:2bff:fe2d:0c2c)での指定が可能です。

ノード名が node1 と node2 の場合の記述例は、以下のとおりです。

NODE_NAME=node1 NODE_NAME=node2

ノード名に複数の IP アドレスが付加されている場合、gethostbyname() で取得できる最初の IP アドレスが使用されます。

NODE_NAME パラメータは、複数のネットワーク・インタフェースの指定も可能です。1 つ目のノード名(ネットワーク・インタフェース)に続いてカンマ (,) で区切ることで、別のネットワーク・インタフェースを指定します。1 つ目のネットワーク・インタフェースに障害が発生した場合、2 つ目のネットワーク・インタフェースで通信できます。

注意: NODE_NAME パラメータに仮想 IP (フローティング IP)は設定できません。

設定ファイル (oramond.conf) に記述された NODE_NAME パラメータの順序は、停止ノード選択ポリシーの優先順位となります。

停止ノード選択ポリシーの詳細は、「3.2.11 停止ノード選択ポリシーの設定(RAC 構成のみ)」を参照してください。

3.2.10. スタンバイ・インスタンス監視の設定(Oracle Data Guard Broker 構成のみ)

スタンバイ・インスタンスの監視(REDO ログの転送状況の監視、REDO ログの適用状況の監視および Oracle Data Guard Broker プロセスの状態監視)を設定するには、STANDBY_DB_CHECK パラメータを指定します。STANDBY_DB_CHECK パラメータに指定できる値は、以下のとおりです。

STANDBY_DB_CHECK=YES	スタンバイ・インスタンスの監視を行います。
STANDBY_DB_CHECK=NO	スタンバイ・インスタンスの監視を行いません。

注意: スタンバイ・インスタンスの監視を行うには、STANDBY_DB_CHECK パラメータの設定以外に、DataGuardSite ステートメント、StandbyDatabaseMonitor ステートメントおよび Transport ステートメントの設定を行ってください。

3.2.11. 停止ノード選択ポリシーの設定(RAC 構成のみ)

複数ノードで障害を検出した場合、発生した障害の重要度に応じて、自動的に ApplicationMonitor 停止ノードを選択します。複数ノード障害とは、モニタ制御デーモン (oramond) が 1 つのノードで障害を検出し、障害ノードの停止処理を開始するまでの間に、その他のノードも障害を検出した場合を指します。

複数ノードで発生した障害の重要度が同じ場合に、ApplicationMonitor 停止ノードの選択条件を設定するには、HALT_CONTROL_POLICY パラメータを指定します。指定できる値は、以下のとおりです。

HALT_CONTROL_POLICY=FIRST_FAILED	モニタ制御デーモンが最初に障害を検出したノードが選択されます。
HALT_CONTROL_POLICY=CONFIGURED	設定ファイル (oramond.conf) に記述される NODE_NAME パラメータの優先順位にしたがいます。

CONFIGURED に設定した場合、障害ノードのうち、設定ファイル内で先に記述している NODE_NAME パラメータのノードが ApplicationMonitor 停止ノードとして選択されます。

3.2.12. 障害ノード上の ApplicationMonitor の停止動作の設定

障害が発生したノード上の ApplicationMonitor の停止動作を設定するには、HALT_CONTROL_OPTION パラメータを指定します。指定できる値は、以下のとおりです。

HALT_CONTROL_OPTION=ALWAYS	ApplicationMonitor を停止します。
HALT_CONTROL_OPTION=NEVER	ApplicationMonitor を停止しません。
HALT_CONTROL_OPTION=OUTLIVE	他に ApplicationMonitor が生存しているノードが存在する場合、ALWAYS として扱われます。 他に ApplicationMonitor が生存しているノードが存在しない場合、NEVER として扱われます。

3.2.13. Oracle 構成プロセス停止の設定

障害が発生したノードの停止時、Oracle の構成プロセスに対して kill コマンドを実行するかどうかを設定するには、HALT_CONTROL_METHOD パラメータを指定します。指定できる値は、以下のとおりです。

HALT_CONTROL_METHOD=KILL	kill -SIGKILL を実行します。
HALT_CONTROL_METHOD=NONE	kill -SIGKILL を実行しません。

3.2.14. Oracle DBA 監査 (audit) 機能対応の設定

Oracle の設定により、SYS の監査ログ取得が有効化されている状態で、SYS ユーザーで Oracle に接続した場合、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る可能性があります。

ORACLE_USER パラメータ、ORACLE_PASS パラメータの双方を指定することで、SYS ユーザーではなく、「3.1 インスタンス監視表の作成」で作成した任意の Oracle ユーザーで Oracle に接続するため、大量のログファイルの生成を減らすことができます。

Oracle 12c のマルチテナント構成で、ORACLE_USER パラメータに C## または c## で始まる Oracle ユーザー (共通ユーザー) を指定する場合は、ダブルクォーテーション (") で囲んでください。

ORACLE_PASS パラメータの指定の詳細は、「3.2.15 Oracle パスワードの扱い」を参照してください。

また、PRIMARY_ORACLE_USER パラメータ、PRIMARY_ORACLE_PASS パラメータおよび BROKER_ORACLE_USER パラメータ、BROKER_ORACLE_PASS パラメータも同様です。

注意: ORACLE_USER パラメータ、ORACLE_PASS パラメータをどちらか片方のみ指定することはできません。

注意: 任意の Oracle ユーザーで Oracle に接続している場合でも、監査ログが出力される可能性があります。その場合は、定期的に監査ログを削除してください。

注意: ORACLE_USER パラメータに指定する Oracle 接続ユーザーを変更する場合は、監視表を再作成する必要があります。
監視表の詳細は、「3.1 インスタンス監視表の作成」を参照してください。

3.2.15. Oracle パスワードの扱い

Oracle の設定により、パスワードの大 / 小文字の区別を有効化している状態で、ApplicationMonitor のパラメータ値に小文字を含める場合は、ダブルクォーテーション (") で囲んでください。

Oracle パスワードの詳細は、Oracle のマニュアル『Oracle Database セキュリティ・ガイド』を参照してください。

3.2.16. パスワード・ファイル認証での接続

パスワード・ファイル認証環境で SYSDBA 権限による監視を設定するには、ORACLE_USER パラメータ、ORACLE_PASS パラメータおよび SYSDBA_CONNECT パラメータを指定します。SYSDBA_CONNECT パラメータに指定できる値は、以下のとおりです。

また、PRIMARY_SYSDBA_CONNECT パラメータも同様です。

SYSDBA_CONNECT=YES	SYSDBA 権限で Oracle に接続します。
SYSDBA_CONNECT=NO	SYSDBA 権限で Oracle に接続しません。

3.2.17. 監視モニタ制御の設定

モニタ制御デーモン (oramond) からリスナー監視モニタ (lsnrmond)、表領域監視モニタ (tschkmond)、ディスク領域監視モニタ (fdsmond) および統計情報採取モニタ (statsmond) を制御(モニタの起動、停止、停止、再開)するかどうかを設定するには、MONITOR_CONTROL パラメータを指定します。指定できる値は、以下のとおりです。

MONITOR_CONTROL=YES	モニタ制御デーモンが各モニタを制御します。
MONITOR_CONTROL=NO	モニタ制御デーモンが各モニタを制御しません。

NO に設定した場合、モニタ制御デーモンによって制御されないため、リスナー監視モニタ、表領域監視モニタ、ディスク領域監視モニタおよび統計情報採取モニタを単独で起動し、監視を行います。

単独起動モニタの詳細は、「3.5 リスナー監視モニタの単独起動」、「3.6 表領域監視モニタの単独起動」、「3.7 ディスク領域監視モニタの単独起動」または「3.8 統計情報採取モニタの単独起動」を参照してください。

3.2.18. Oracle ハング検知機能を利用した監視(RAC 構成のみ)

Oracle 11g (11.1.0.7) より、RAC 固有プロセスのストール監視機能(以後 Oracle ハング検知機能 と記載します)が追加されました。

Oracle ハング検知機能の監視結果を障害判定に利用するかどうかを設定するには、ORACLE_HANG_CHECK パラメータを指定します。指定できる値は、以下のとおりです。

ORACLE_HANG_CHECK=YES	Oracle ハング検知機能を利用します。
ORACLE_HANG_CHECK=NO	Oracle ハング検知機能を利用しません。

3.2.19. 統計情報採取の設定

インスタンス監視モニタ (instmond)、リスナー監視モニタ (lsnrmond)、表領域監視モニタ (tschkmond) および統計情報採取モニタ (statsmond) の情報採取を行うかどうかを設定するには、GET_STATS パラメータを指定します。指定できる値は、以下のとおりです。

GET_STATS=YES	統計情報を採取します。
GET_STATS =NO	統計情報を採取しません。

各モニタが採取する情報は、以下のとおりです。

インスタンス監視モニタ	監視 TAT 情報 ・ 監視対象インスタンス名 ・ 監視 TAT
リスナー監視モニタ	監視 TAT 情報 ・ 監視対象リスナー名 ・ 監視 TAT
表領域監視モニタ	表領域情報 ・ 表領域名 ・ 空き領域のサイズ(ブロック数) ・ 総サイズ(ブロック数) ・ 空き容量率(%)
統計情報採取モニタ	SGA 情報 ・ 空き容量サイズ(ブロック数) ・ 総サイズ(ブロック数) ・ 空き容量率(%)

各モニタは、採取した情報をファイルに出力します。ファイルサイズが 3M バイトを超えた場合、ファイルを新規に作成し、以下のディレクトリに二世代分まで情報を残します。

現在の情報	/var/opt/HA/AM/<構成番号>/data/<モニタ名>_<ID>_1.dat
一世代前の情報	/var/opt/HA/AM/<構成番号>/data/<モニタ名>_<ID>_2.dat
二世代前の情報	/var/opt/HA/AM/<構成番号>/data/<モニタ名>_<ID>_3.dat

<ID> インスタンス監視モニタ :なし
 リスナー監視モニタ :リスナー名
 表領域監視モニタ :表領域監視モニタ ID
 統計情報採取モニタ :統計情報採取モニタ ID

採取した情報は、先頭に情報採取日時を付与し、CSV 形式でファイルに出力されます。各モニタからの出力フォーマットと出力イメージは、以下のようになります。

◆ インスタンス監視モニタ

出力フォーマット: 情報採取日時, SID 名, 監視 TAT

出力イメージ

```
2014/3/13 08:23:40,rac1,1
2014/3/13 08:25:10,rac1,1
2014/3/13 08:26:40,rac1,1
```

◆ リスナー監視モニタ

出力フォーマット: 情報採取日時, リスナー名, 監視 TAT

出力イメージ

```
2014/3/13 08:23:40,LISTENER,1
2014/3/13 08:24:40,LISTENER,1
2014/3/13 08:25:40,LISTENER,1
```

◆ 表領域監視モニタ

出力フォーマット: 情報採取日時, 表領域名, 空き容量, 総サイズ, 空き容量率

出力イメージ

```
2014/3/13 08:23:40,USERS,152,1600,9
2014/3/13 09:23:40,USERS,152,1600,9
2014/3/13 10:23:40,USERS,152,1600,9
```

◆ 統計情報採取モニタ

出力フォーマット: 情報採取日時, 種別, 空き容量, 総サイズ, 空き容量率

出力イメージ

```
2014/3/13 08:23:40, SGA_SPACE,345529792.00,603979776.00,57.00
2014/3/13 09:23:40, SGA_SPACE,345529792.00,603979776.00,57.00
2014/3/13 10:23:40, SGA_SPACE,345529792.00,603979776.00,57.00
```

3.2.20. ASM インスタンス停止に伴うデータベース再起動の待ち合わせの設定

ASM インスタンス停止に伴うデータベース再起動の待ち合わせを設定するには、WAIT_ASM_RESTART パラメータを指定します。WAIT_ASM_RESTART パラメータに指定できる値は、以下のとおりです。

WAIT_ASM_RESTART=YES	ASM インスタンス停止に伴うデータベース再起動を待ち合わせます。
WAIT_ASM_RESTART=NO	ASM インスタンス停止に伴うデータベース再起動を待ち合わせません。

3.2.21. インスタンス監視表のスキーマの設定 (Oracle 12c のみ)

ApplicationMonitor では、監視対象の Oracle のバージョンが Oracle 12c の場合、「3.1 インスタンス監視表の作成」でインスタンス監視表を作成した際のスキーマの指定が必須です。

スキーマは、InstanceMonitor ステートメント内の OWNER_NAME パラメータにて指定します。OWNER_NAME パラメータに指定する値は、以下のとおりです。

監視を行うユーザー	OWNER_NAME パラメータに指定するスキーマ
SYS ユーザー	SYS またはローカル・ユーザー
共通ユーザー	SYS またはローカル・ユーザー
ローカル・ユーザー	ローカル・ユーザー

3.2.22. 指定表監視時の障害の設定 (Oracle 12c のマルチテナント構成のみ)

Table ステートメントを記述して特定の表を監視する際、PDB の状態が OPEN 以外だった場合に障害として判断するかどうかを設定するには、UNOPEN_PDB_DOWN パラメータを指定します。

UNOPEN_PDB_DOWN パラメータに指定できる値は、以下のとおりです。

UNOPEN_PDB_DOWN=YES	PDB の状態が OPEN 以外だった場合、障害として扱います。
UNOPEN_PDB_DOWN=NO	PDB の状態が OPEN 以外だった場合、障害として扱いません。

3.2.23. インスタンス / リスナー障害時の CLUSTERPRO への障害通知の設定

インスタンス / リスナーの障害時、再確認および再起動を行っても改善できなかった場合に、CLUSTERPRO への障害通知を行うかどうかを設定するには、SERVICE_DOWN パラメータを指定します。指定できる値は、以下のとおりです。

SERVICE_DOWN=YES	CLUSTERPRO へ障害を通知します。
SERVICE_DOWN=NO	CLUSTERPRO へ障害を通知しません。
SERVICE_DOWN=NO_DOWN	ダウン障害の場合、CLUSTERPRO へ障害を通知しません。
SERVICE_DOWN=NO_STALL	ストール障害の場合、CLUSTERPRO へ障害を通知しません。

この機能は、リスナー監視モニタ (lsnrmond) がモニタ制御デーモン (oramond) によって制御されているか、単独起動しているかにかかわらず有効です。

また、Oracle Clusterware / Oracle Restart 監視の障害検出時の動作についても、SERVICE_DOWN パラメータを指定することにより、同様の設定が可能です。

Oracle Clusterware / Oracle Restart 監視の設定の詳細は、「3.9 CRS 監視設定ファイル (crsmond.conf) の作成」を参照してください。

3.2.24. リスナー再起動方法の設定

リスナーの障害時、リスナーの再起動方法を設定するには、TARGET_RESTART パラメータを指定します。TARGET_RESTART パラメータに指定できる値は、以下のとおりです。

TARGET_RESTART=YES	ApplicationMonitor によってリスナーを再起動します。
TARGET_RESTART=NO	ApplicationMonitor によってリスナーを再起動しません。

NO に設定した場合、Oracle Clusterware / Oracle Restart によるリスナーの再起動を待ち合わせます。

注意: NO に設定した場合、Oracle 11g R1 の Oracle Clusterware 環境では、Oracle Clusterware によるリスナーの再起動を待ち合わせますが、リスナーの再起動は行われません。

3.2.25. リスナーの繰り返し障害の監視

動作環境が不安定な場合など、障害リスナーを再起動して復旧させても、障害を繰り返す可能性があります。

リスナーが障害を繰り返す状態を障害と確定させるには、UP_DOWN_RETRY パラメータ、UPTIME_COUNT パラメータを指定します。

リスナーの繰り返し障害の累積回数は、障害を検出してリスナーを再起動する初回時 (UP の状態から障害を検出した際) に加算されます。繰り返し障害の累積回数が UP_DOWN_RETRY パラメータに設定した回数を超えた場合、障害と判断します。ただし、リスナー監視の結果、UP の状態が UPTIME_COUNT パラメータに設定した回数続いた場合、安定稼働状態と判断します。その際、繰り返し障害の累積回数をクリアします。

3.3. Oracle エラーのフィルタリング設定

ApplicationMonitor では、各モニタの監視時または採取時に検出する Oracle エラーをエラーとして扱うか、またはエラーとして扱わない(許可エラー)かを設定できます。

Oracle エラーのフィルタリングは、以下のフィルタリング設定ファイルに記述することで設定します。

また、以下のフィルタリング設定ファイルには、インストール時にデフォルト設定されています。デフォルト設定の詳細は、「3.3.2 フィルタリングのデフォルト設定」を参照してください。

インスタンス監視モニタ (instmond) の フィルタリング設定ファイル	/etc/opt/HA/AM/conf/inst.conf
リスナー監視モニタ (lsnrmond) の フィルタリング設定ファイル	/etc/opt/HA/AM/conf/lsnr.conf
表領域監視モニタ (tschkmond) の フィルタリング設定ファイル	/etc/opt/HA/AM/conf/tschk.conf
ディスク領域監視モニタ (fdsmond) の フィルタリング設定ファイル	/etc/opt/HA/AM/conf/fds.conf
スタンバイデータベース監視モニタ (stbdbmond) のフィルタリング設定ファイル	/etc/opt/HA/AM/conf/stbdb.conf
統計情報採取モニタ (statsmond) の フィルタリング設定ファイル	/etc/opt/HA/AM/conf/stats.conf
システムステートダンプ採取デモン (ssdump) のフィルタリング設定ファイル	/etc/opt/HA/AM/conf/ssdump.conf

フィルタリング設定ファイルの記述例は、以下のようになります。

#instmond error check configuration file#	
ERROR_CODE_TYPE = ALLOWABLE_CODE	
-18	#maximum number of session exceeded.
-19	#maximum number of session licenses exceeded.
-20	#maximum number of processes exceeded.
-2020	#too many database links in use.
-2391	#exceeded simultaneous SESSION_PER_USER limit.
-3113	#end-of-file on communication channel.
-8000	#maximum number of session sequence lists exceeded.
-9330	#Session terminated internally by Oracle or by an Oracle DBA.
-12540	#TNS:internal limit restriction exceeded.
-1033	#ORACLE initialization or shutdown in progress.
-1109	#database not open.
-16000	#database open for read-only access.
-1219	#database not open: queries allowed on fixed tables/views only

3.3.1. フィルタリング設定ファイルの記述形式

フィルタリング設定ファイルの記述形式は、各モニタ共通です。

フィルタリング設定ファイルは、コメント、ERROR_CODE_TYPE パラメータ、Oracle エラー番号から構成されます。

◆ コメント

コメントは、シャープ記号 (#) から始まり、行の最後までとなります。

◆ ERROR_CODE_TYPE パラメータ

ERROR_CODE_TYPE パラメータは、このパラメータ以降に記述する Oracle エラー番号に対し、エラーとして扱う Oracle エラー番号なのか、許可エラーとして扱う Oracle エラー番号なのかを記述します。

記述形式は、以下のとおりです。

記述形式：

ERROR_CODE_TYPE = [ERROR_CODE|ALLOWABLE_CODE]

ERROR_CODE_TYPE パラメータに指定できる値は、以下のとおりです。

ERROR_CODE_TYPE = ERROR_CODE	エラーとして扱います。
ERROR_CODE_TYPE = ALLOWABLE_CODE	許可エラーとして扱います。

注意： フィルタリング設定ファイルが正しく記述されている場合、デフォルト設定をすべて無効とし、設定された内容が使用されます。

注意： フィルタリング設定ファイルが正しく記述されていない場合、設定された内容をすべて無効とし、デフォルト設定が使用されます。

注意： ERROR_CODE_TYPE パラメータは、必ず Oracle エラー番号を記述する前の行に記述してください。

3 製品の設定

◆ Oracle エラー番号

フィルタリングする Oracle エラー番号を記述します。1 つの Oracle エラー番号につき、1 行に記述します。

記述形式は、以下のとおりです。

記述形式：

-Oracle エラー番号

Oracle エラー番号の先頭は、必ずマイナス (-) になります。「ORA-00018:最大セッション数を超えました」の場合、-18 と記述します。

3.3.2. フィルタリングのデフォルト設定

インスタンス監視モニタ (instmond) のデフォルト設定は、以下のとおりです。

インスタンス監視モニタの場合、以下の Oracle エラー番号を許可エラーとして扱うように設定しています。

Oracle エラー番号	メッセージ
ORA-00018	最大セッション数を超過しました。
ORA-00019	最大セッション・ライセンスを超過しました。
ORA-00020	最大プロセス数(string)を超過しました。
ORA-02020	使用中のデータベース・リンクが多すぎます。
ORA-02391	同時実行の SESSION_PER_USER 制限を超過しました。
ORA-03113	通信チャンネルでファイルの終わりが検出されました。
ORA-08000	セッション順序番号リスト数が最大値を超過しました。
ORA-09330	Oracle または Oracle DBA により、内部的にセッションが終了しました。
ORA-12540	TNS:内部制限を超過しました。
ORA-1033	Oracle の初期化またはシャットダウン中です。
ORA-1109	データベースがオープンされていません。
ORA-16000	データベースは読取り専用アクセスでオープンされています。
ORA-1219	データベースがオープンしていません: 固定の表/ ビューに対する問合せのみ可能です。

リスナー監視モニタ (lsnrmond) のデフォルト設定は、以下のとおりです。

リスナー監視モニタの場合、以下の Oracle エラー番号をエラーとして扱うように設定しています。

Oracle エラー番号	メッセージ
TNS-12541	TNS:リスナーがありません。

表領域監視モニタ (tschkmond)、ディスク領域監視モニタ (fdsmond)、統計情報採取モニタ (statsmond) およびシステムステート採取デーモン (ssdump) のデフォルト設定は、以下のとおりです。

表領域監視モニタ、ディスク領域監視モニタ、統計情報採取モニタおよびシステムステート採取デーモンの場合、以下の Oracle エラー番号を許可エラーとして扱うように設定しています。

Oracle エラー番号	メッセージ
ORA-1033	Oracle の初期化またはシャットダウン中です。

3 製品の設定

スタンバイデータベース監視モニタ (stbldbmond) のデフォルト設定は、以下のとおりです。

スタンバイデータベース監視モニタの場合、以下の Oracle エラー番号を許可エラーとして扱うように設定しています。

Oracle エラー番号	メッセージ
ORA-1033	Oracle の初期化またはシャットダウン中です。
ORA-1109	データベースがオープンされていません。
ORA-16000	データベースは読取り専用アクセスでオープンされています。
ORA-1219	データベースがオープンしていません: 固定の表/ ビューに対する問合せのみ可能です。

3.4. 情報採取スクリプト (oradiag) の設定

ApplicationMonitor では、情報採取スクリプトを使用することにより、障害検出時、`/var/opt/HA/AM/<構成番号>/oradiag` ディレクトリに障害解析情報を採取できます。

情報採取スクリプトは、`/opt/HA/AM/bin` 配下にインストールされます。障害検出時にモニタ制御デーモン (oramond) が自動で情報採取スクリプトを実行する場合は、設定ファイル (oramond.conf) の `SCRIPT_NAME` パラメータに情報採取スクリプトの絶対パス名を設定してください。

<code>SCRIPT_NAME = /opt/HA/AM/bin/oradiag</code>

情報採取スクリプトは、障害を検出したノード上のモニタ制御デーモンの構成番号 (設定ファイルの `COMPONENT_ID` パラメータ) とインスタンスの監視結果を引数として実行されます。

監視結果は、以下のとおりです。

監視結果	意味
UP	全ノードで障害情報を採取し、自ノード上のインスタンスが問題なく動作していた場合に指定されます。
DOWN	自ノード上のインスタンス監視でインスタンス停止などのエラー障害を検出した場合に指定されます。
STALL	自ノード上のインスタンス監視でストール障害、無応答障害を検出した場合に指定されます。
ERROR	上記以外の異常を検出し、監視の続行が不可能な場合に指定されます。

情報採取スクリプトの `DOWN_sequence` および `STALL_sequence` の各シェル関数内に任意の追加処理を記述することで、監視結果がダウンの場合には `DOWN_sequence` の処理、監視結果がストールの場合には `STALL_sequence` の処理を実行できます。

情報採取スクリプトは、情報採取時に前回採取した情報が残っている場合、以下のディレクトリに二世代分まで情報を残します。

現在の情報	<code>/var/opt/HA/AM/<構成番号>/oradiag</code>
一世代前の情報	<code>/var/opt/HA/AM/<構成番号>/oradiag.old</code>
二世代前の情報	<code>/var/opt/HA/AM/<構成番号>/oradiag.old2</code>

注意: 設定ファイルの `SCRIPT_NAME` パラメータに情報採取スクリプトを設定した場合は、必ず `SCRIPT_USER` パラメータに `root` ユーザーを設定してください。

なお、`SCRIPT_NAME` パラメータには、任意のシェルスクリプトを設定することができます。

3.5. リスナー監視モニタの単独起動

リスナー監視モニタ (lsnrmond) は、設定ファイル (oramond.conf) の MONITOR_CONTROL パラメータを NO に指定することにより、モニタ制御デーモン (oramond) からの制御を行わず、単独で起動することができます。

リスナー監視モニタを単独で起動する場合は、必ず root または oracle コマンドの実行権限があるユーザーで実行してください。

リスナー監視モニタの起動方法の詳細は、「6 コマンドリファレンス」を参照してください。

3.6. 表領域監視モニタの単独起動

表領域監視モニタ (tschkmond) は、設定ファイル (oramond.conf) の MONITOR_CONTROL パラメータを NO に設定することにより、モニタ制御デーモン (oramond) からの制御を行わず、単独で起動することができます。

表領域監視モニタを単独で起動する場合は、以下の環境変数を必ず指定してください。

ORACLE_SID	Oracle のインスタンス識別子
ORACLE_BASE	Oracle ディレクトリ構成の基本となるディレクトリ
ORACLE_HOME	Oracle ソフトウェアをインストールしたディレクトリ
ORA_NLS10	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリ
NLS_LANG	表領域監視モニタが Oracle にアクセスする際の言語、地域およびキャラクタ・セット
LD_LIBRARY_PATH	db ユーザー環境の 64bit 共有ライブラリのパス

表領域監視モニタの起動方法の詳細は、「6 コマンドリファレンス」を参照してください。

3.7. ディスク領域監視モニタの単独起動

ディスク領域監視モニタ (fdsmond) は、設定ファイル (oramond.conf) の MONITOR_CONTROL パラメータを NO に設定することにより、モニタ制御デーモン (oramond) からの制御を行わず、単独で起動することができます。

ディスク領域監視モニタを単独で起動する場合は、以下の環境変数を必ず指定してください。

ORACLE_SID	Oracle のインスタンス識別子
ORACLE_BASE	Oracle ディレクトリ構成の基本となるディレクトリ
ORACLE_HOME	Oracle ソフトウェアをインストールしたディレクトリ
ORA_NLS10	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリ
NLS_LANG	ディスク領域監視モニタが Oracle にアクセスする際の言語、地域およびキャラクタ・セット
LD_LIBRARY_PATH	db ユーザー環境の 64bit 共有ライブラリのパス

ディスク領域監視モニタの起動方法の詳細は、「6 コマンドリファレンス」を参照してください。

3.8. 統計情報採取モニタの単独起動

統計情報採取モニタ (statsmond) は、設定ファイル (oramond.conf) の MONITOR_CONTROL パラメータを NO に設定することにより、モニタ制御デーモン (oramond) からの制御を行わず、単独で起動することができます。

統計情報採取モニタを単独で起動する場合は、以下の環境変数を必ず指定してください。

ORACLE_SID	Oracle のインスタンス識別子
ORACLE_BASE	Oracle ディレクトリ構成の基本となるベースディレクトリ
ORACLE_HOME	Oracle ソフトウェアをインストールしたホームディレクトリ
ORA_NLS10	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリ
NLS_LANG	統計情報採取モニタが Oracle にアクセスする際の言語、地域およびキャラクタ・セット
LD_LIBRARY_PATH	db ユーザー環境の 64bit 共有ライブラリのパス

統計情報採取モニタの起動方法の詳細は、「6 コマンドリファレンス」を参照してください。

3.9. CRS 監視設定ファイル (crsmond.conf) の作成

ApplicationMonitor で Oracle の Oracle Clusterware / Oracle Restart の無応答障害監視を行う場合は、CRS 監視設定ファイルを作成する必要があります。CRS 監視設定ファイルには、監視対象とする crsd/ohasd の状態を表示するコマンド、CRS 監視モータ (crsmond) の動作設定などを定義します。

CRS 監視設定ファイルのサンプルが /etc/opt/HA/AM/conf/sample ディレクトリにインストールされます。サンプルを参照して、システムの環境に応じた指定を行ってください。

CRS 監視モータの詳細は、「6 コマンドリファレンス」を参照してください。

注意: 作成した CRS 監視設定ファイルは、/etc/opt/HA/AM/conf/crsmond.conf として置いてください。

注意: CRS 監視設定ファイルは、管理コマンド (oraadmin) による全ノードへの配布は行えません。各ノードで作成する必要があります。

3.9.1. CRS 監視設定ファイルの記述例

CRS 監視設定ファイル (crsmond.conf) は、コメント、パラメータによって構成されたテキストファイルです。

設定は、以下のように記述します。

```
# crsmond.conf

ORACLE_VERSION = 12C           # 11G, 11GR2, 12C
CRS_STAT        = "/u01/app/12.1.0/grid/bin/crsctl check crs"
#CRS_STAT       = "/u01/app/11.2.0/grid/bin/crsctl check crs"
#CRS_STAT       = /u01/app/oracle/product/11.1.0/crs/bin/crs_stat
LOG_LEVEL       = 2
POLL_INTERVAL   = 90
POLL_TIMEOUT    = 180
RETRY_COUNT     = 5
WAIT_INTERVAL   = 180
SERVICE_DOWN   = NO
#CUSTOM_USER    = root
#CUSTOM_CMD     = /var/tmp/test.sh
```

3.9.2. コメントの記述形式

コメントは、ダブルクォーテーション (") で囲まれていないシャープ記号 (#) で始まり、行の最後までとなります。

3.9.3. パラメータの記述形式

パラメータは、CRS 監視モニタ (crsmond) の動作に必要な値を数値または文字列で指定します。

パラメータ値にスペースを含める場合は、文字列全体をダブルクォーテーション (") で囲ってください。

1 つのパラメータを 1 行(1021 文字以内)に記述します。パラメータの記述形式は、以下のとおりです。

各種パラメータの詳細は、「3.9.4 パラメーター一覧」を参照してください。

記述形式:

パラメータ名 = パラメータ値

注意: パラメータ値に全角文字、タブを使用することはできません。

また、文字列内にダブルクォーテーション (") および改行コードを使用することはできません。

3.9.4. パラメーター一覧

CRS 監視設定ファイル (crsmond.conf) に記述可能なパラメータは、以下のとおりです。

パラメータ名	意味		1	2
	省略時のデフォルト値	値の範囲		
ORACLE_VERSION	監視対象の Oracle のバージョンを指定します。		×	○
	省略できません。	11G 11GR2 12C		
CRS_STAT	Oracle が提供する crsd/ohasd の状態を表示するコマンドの絶対パスおよび引数を指定します。 詳細は、「3.9.5 crsd/ohasd の状態表示用コマンドの設定」を参照してください。		×	○
	省略できません。	なし		
LOG_LEVEL	CRS 監視モニタ (crsmond) のログの出力レベルを指定します。 詳細は、「3.2.8 ログレベルの設定」を参照してください。		○	○
	2	0 ~ 3		
POLL_INTERVAL	crsd/ohasd を監視する間隔を秒で指定します。		○	○
	90 秒	10 ~ 86400		
POLL_TIMEOUT	crsd/ohasd の無応答と判断するまでの上限時間を秒で指定します。		○	○
	180 秒	10 ~ 86400		
RETRY_COUNT	障害検出時、crsd/ohasd を再確認する回数を指定します。		○	○
	5 回	0 ~ 24		
WAIT_INTERVAL	障害検出時、init(8) による crsd/ohasd 再起動を待ち合わせる時間を秒で指定します。		○	○
	180 秒	60 ~ 86400		
SERVICE_DOWN	障害検出時、init(8) による crsd/ohasd 再起動を待ち合わせ、RETRY_COUNT パラメータに設定された回数の再確認を行っても再起動できなかった場合に、CLUSTERPRO へ障害を通知するかどうかを指定します。 詳細は、「3.2.23 インスタンス / リスナー 障害時の CLUSTERPRO への障害通知の設定」を参照してください。		○	○
	NO	YES NO NO_DOWN NO_STALL		
CUSTOM_USER	障害検出時に実行する CUSTOM_CMD パラメータに指定されたコマンドの実効ユーザー名を指定します。		○	○
	root	なし		
CUSTOM_CMD	障害検出時に実行するコマンドの絶対パスおよび引数を指定します。 必ず実行権が付いていることを確認してください。 コマンドがシェルスクリプトの場合は、スクリプトの先頭行に使用するシェルが記述されていることを確認してください。		○	○
	指定なし	なし		

1 パラメータが省略可能かどうかを示します。可能な場合は○、不可能な場合は×です。

2 CRS 監視モニタ (crsmond) が動作している時、動的な設定変更 (apply リクエスト) が可能かどうかを示します。可能な場合は○、不可能な場合は×です。

3.9.5. crsd/ohasd の状態表示用コマンドの設定

Oracle Clusterware / Oracle Restart の監視を行うには、crsd/ohasd の状態表示用コマンドの指定が必須です。crsd/ohasd の状態表示用コマンドは、CRS_STAT パラメータにて指定します。

CRS_STAT パラメータは、監視対象の Oracle のバージョンおよび環境により、設定するコマンドが異なります。設定するコマンドは、以下のとおりです。

Oracle のバージョン	環境	設定するコマンド
11g	全環境	crs_stat
11g R2 12c	Oracle Restart	crsctl check has
	Oracle Clusterware	crsctl check crs

4. 製品の動作

4.1. ApplicationMonitor の起動と停止

モニタ制御デーモン (oramond) は、各ノード上で CLUSTERPRO から Oracle グループのリソースとして起動されます。モニタ制御デーモンは、起動すると構成情報ファイル (oramond<構成番号>.bin) を読み込み、設定にしたがってインスタンス監視モニタ (instmond)、リスナー監視モニタ (lsnrmond)、表領域監視モニタ (tschkmond)、ディスク領域監視モニタ (fdsmond)、スタンバイデータベース監視モニタ (stbdbmond)、アラートログ監視モニタ (altmond)、統計情報採取モニタ (statsmond)、システムステートダンプ採取デーモン (ssdump) を起動(1、 2)します。

- (1) 監視対象の RAC がプライマリ/セカンダリ構成の場合、セカンダリノードでは、リスナー監視モニタ、表領域監視モニタ、ディスク領域監視モニタ、統計情報採取モニタは、監視および採取を行いません。
- (2) 監視対象がスタンバイ・データベースの場合、各監視モニタは、Oracle の起動状態に合わせた監視を行います。

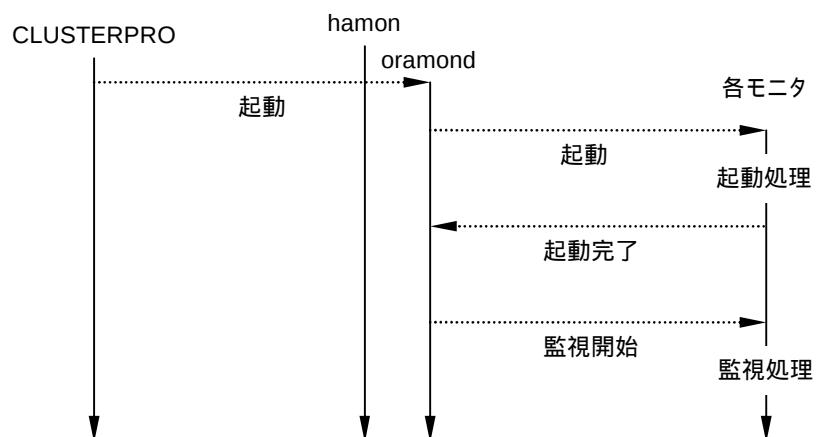


図 4-1 各モニタ起動時の動作

モニタ制御デーモンは、各モニタから起動完了の通知を受け取ると、監視開始および採取開始の通知を各モニタに送信します。これにより、各監視モニタおよび各採取モニタは、動作を開始します。

また、スタンバイ・インスタンスを監視する場合、モニタ制御デーモンは、Oracle Data Guard Broker プロセスの状態を監視します。Oracle Data Guard Broker プロセスの状態監視時の動作の詳細は、「4.6 スタンバイ・インスタンス監視時の動作」を参照してください。

モニタ制御デーモンは、SIGTERM を受信するとすべての監視モニタおよび採取モニタを終了させた後、正常に終了します。

4 製品の動作

各モニタの起動が `BOOT_TIMEOUT` パラメータに設定した時間までに完了しなかった場合、最大 5 回の再起動を試みます。インスタンス監視モニタとアラートログ監視モニタおよび `SERVICE_DOWN` パラメータが `YES` または `NO_STALL` に設定されているリスナー監視モニタについては、再起動しても起動完了の通知を受信できない場合、モニタ制御デモンは、監視モニタの障害として扱います。

障害検出時、モニタ制御デモン停止のタイミングで、`CLUSTERPRO` (`PID` 監視) に障害を通知します。単独起動時のリスナー監視モニタ、`CRS` 監視モニタの `CLUSTERPRO` 連携時の動作についても同様です。

4.2. インスタンス監視時の動作

インスタンス監視モニタ (instmond) は、起動後に POLL_INTERVAL パラメータに設定した間隔でインスタンスを監視します。インスタンスの監視方法は、V\$ 表の参照 (1)、SYSTEM 表領域上のオブジェクトの更新 / 参照、ユーザー指定表の参照です。

なお、データベースが読取り専用アクセスでオープンされている場合、およびデータベースがマウント状態の場合は、インスタンスに対する起動確認のみ行います。

(1) V\$ 表の参照は、SYSTEM 表領域上に作成したビューを通して行います。

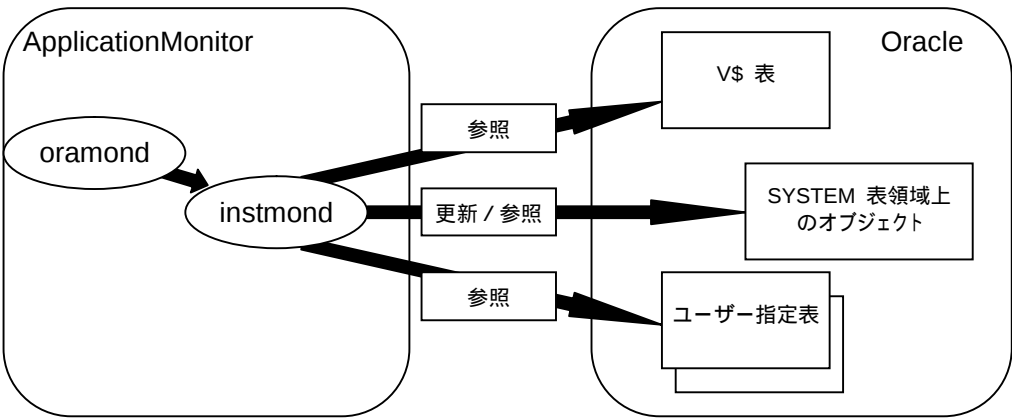


図 4-2 インスタンス監視モニタの概要

SYSTEM 表領域に作成するオブジェクトは、以下のとおりです。

名称	種類	説明
HAAM_TEST_SEQ[0 ~ n]	シーケンス	インスタンス監視時、連番を取得するためのオブジェクト(インスタンスごとに異なるオブジェクトを作成します)。
HAAM_TIME_CHK[0 ~ n]	テーブル	インスタンス監視モニタのアクセス時間を書き込むオブジェクト(インスタンスごとに異なるオブジェクトを作成します)。
HAAM_X...	ビュー	インスタンス監視時、各種 V\$ 表を参照するためのオブジェクト。

4 製品の動作

インスタンスの監視は、以下のような動作になります。

(例)

```
InstanceMonitor {  
  POLL_INTERVAL = 120  
  GET_STATS     = YES  
}
```

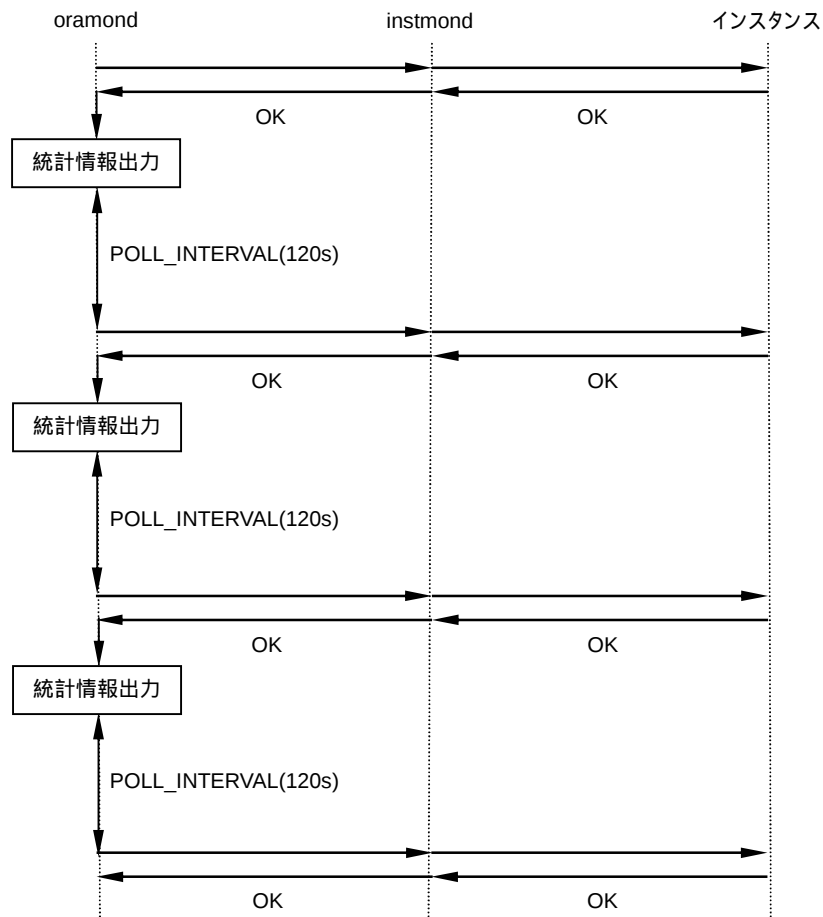


図 4-3 インスタンス監視時の動作

インスタンス監視モニタは、POLL_INTERVAL パラメータに設定した間隔でインスタンスにアクセスします。

GET_STATS パラメータを YES に設定した場合、インスタンスへのアクセスが正常の際に統計情報(監視 TAT)をファイルへ出力します。

インスタンスの監視でエラーを検出し、Oracle Clusterware / Oracle Restart によってインスタンスが再起動された場合、以下のような動作になります。

(例)

```
InstanceMonitor {
  POLL_INTERVAL      = 120
  POLL_DOWN_RETRY    = 2
  SERVICE_DOWN       = YES
  RESTART_COUNT       = 1
  RESTART_TIMEOUT     = 60
}
```

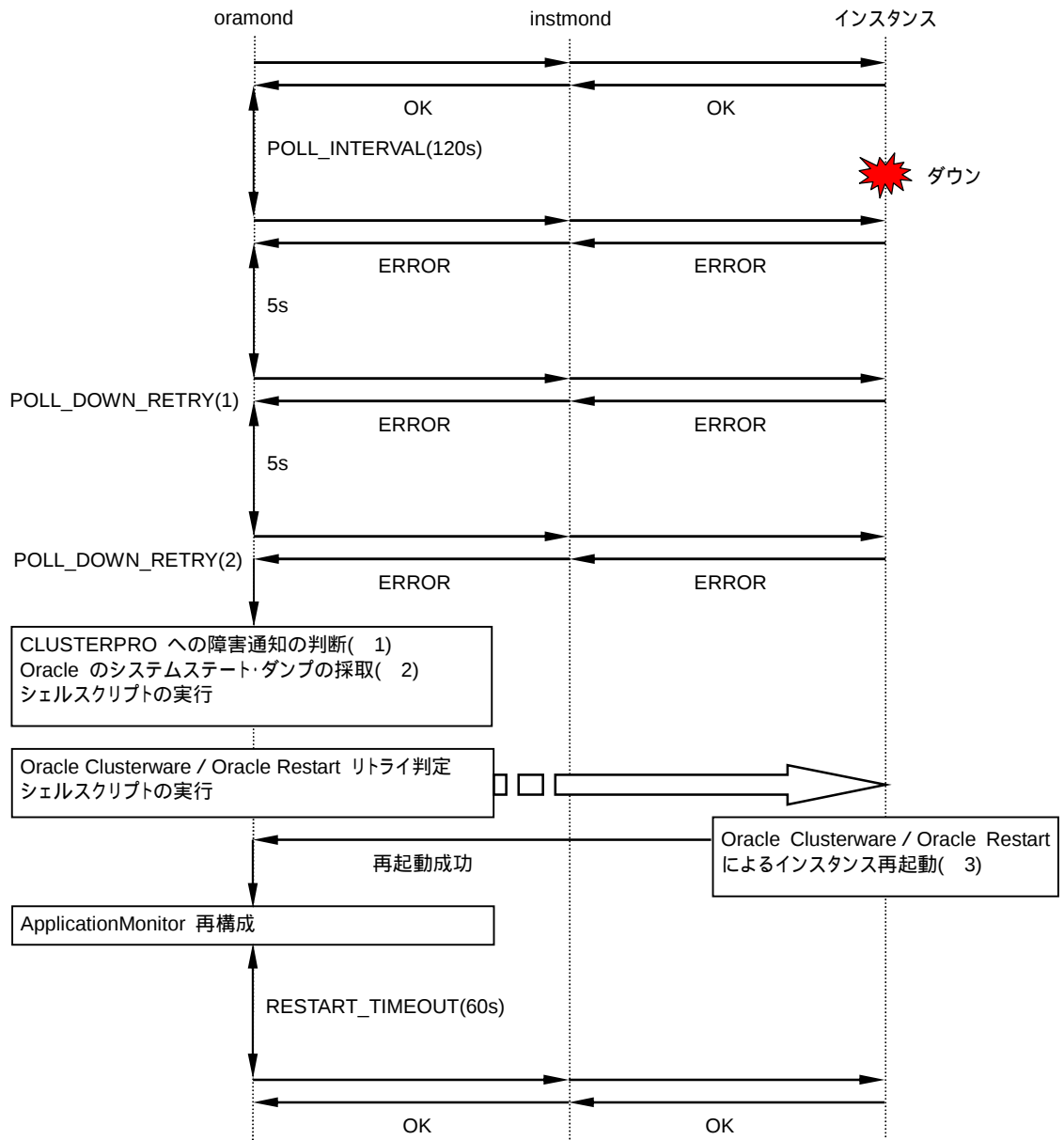


図 4-4 インスタンス監視エラー時の動作 (Oracle Clusterware / Oracle Restart によってインスタンスが再起動された場合)

インスタンスの監視でエラーを検出した場合、`POLL_DOWN_RETRY` パラメータに 1 以上の値が設定されていると、`POLL_DOWN_RETRY` パラメータに設定した回数のインスタンスの再確認を行います。再確認中にインスタンスの監視で正常を検出した場合、障害と判断せず、監視を続けます。

`POLL_DOWN_RETRY` パラメータに設定した回数のインスタンスの再確認を行っても、インスタンス監視結果がエラーになる場合は、`SERVICE_DOWN` パラメータの設定により、`CLUSTERPRO` ノード内の障害を通知するか判断します。インスタンス監視エラーとなる際、`SERVICE_DOWN` パラメータに `YES` または `NO_STALL` が設定されていると、`CLUSTERPRO` ノードへ障害を通知すると判断します (1)。

`CLUSTERPRO` ノードへ障害を通知すると判断した場合、モニタ制御デーモンは、Oracle のシステムステート・ダンプの採取 (2) およびシェルスクリプトを実行し、情報採取を行います。

なお、インスタンス監視エラーとなる際、`SERVICE_DOWN` パラメータに `NO` または `NO_DOWN` が設定されていると、モニタ制御デーモンは、`CLUSTERPRO` ノードへ障害を通知しないと判断するため、監視を続けます。

`POLL_DOWN_RETRY` パラメータに 0 が設定されている場合は、再確認せずに `CLUSTERPRO` ノードへの障害通知の判断、Oracle のシステムステート・ダンプの採取およびシェルスクリプトの実行を行います。

Oracle のシステムステート・ダンプの採取動作の詳細は、「4.8 システムステート・ダンプ採取時の動作」、シェルスクリプトの実行動作の詳細は、「4.9 シェルスクリプトの実行時の動作」を参照してください。

情報採取完了後、`RESTART_COUNT` パラメータに 1 以上の値が設定されていると、Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトを実行し、Oracle Clusterware / Oracle Restart によるインスタンスの再起動 (3) を促して待ち合わせます。

Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの詳細は、「4.10 Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの動作」を参照してください。

インスタンスが再起動された場合は、`ApplicationMonitor` を再構成し、監視を続けます。

- (1) `CLUSTERPRO` ノードへ通知すると判断した障害は、`ApplicationMonitor` を停止することによって通知します。
- (2) インスタンス監視エラー時の Oracle のシステムステート・ダンプ採取は、RAC 構成時のみ非障害ノードにおいて行われます。
- (3) Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境でのみ行われます。

インスタンスの監視でエラーを検出し、Oracle Clusterware / Oracle Restart によってインスタンスが再起動されなかった場合、以下のような動作になります。

(例)

HALT_CONTROL_OPTION = ALWAYS

```
InstanceMonitor {
  POLL_INTERVAL      = 120
  POLL_DOWN_RETRY    = 2
  SERVICE_DOWN       = YES
  RESTART_COUNT       = 1
}
```

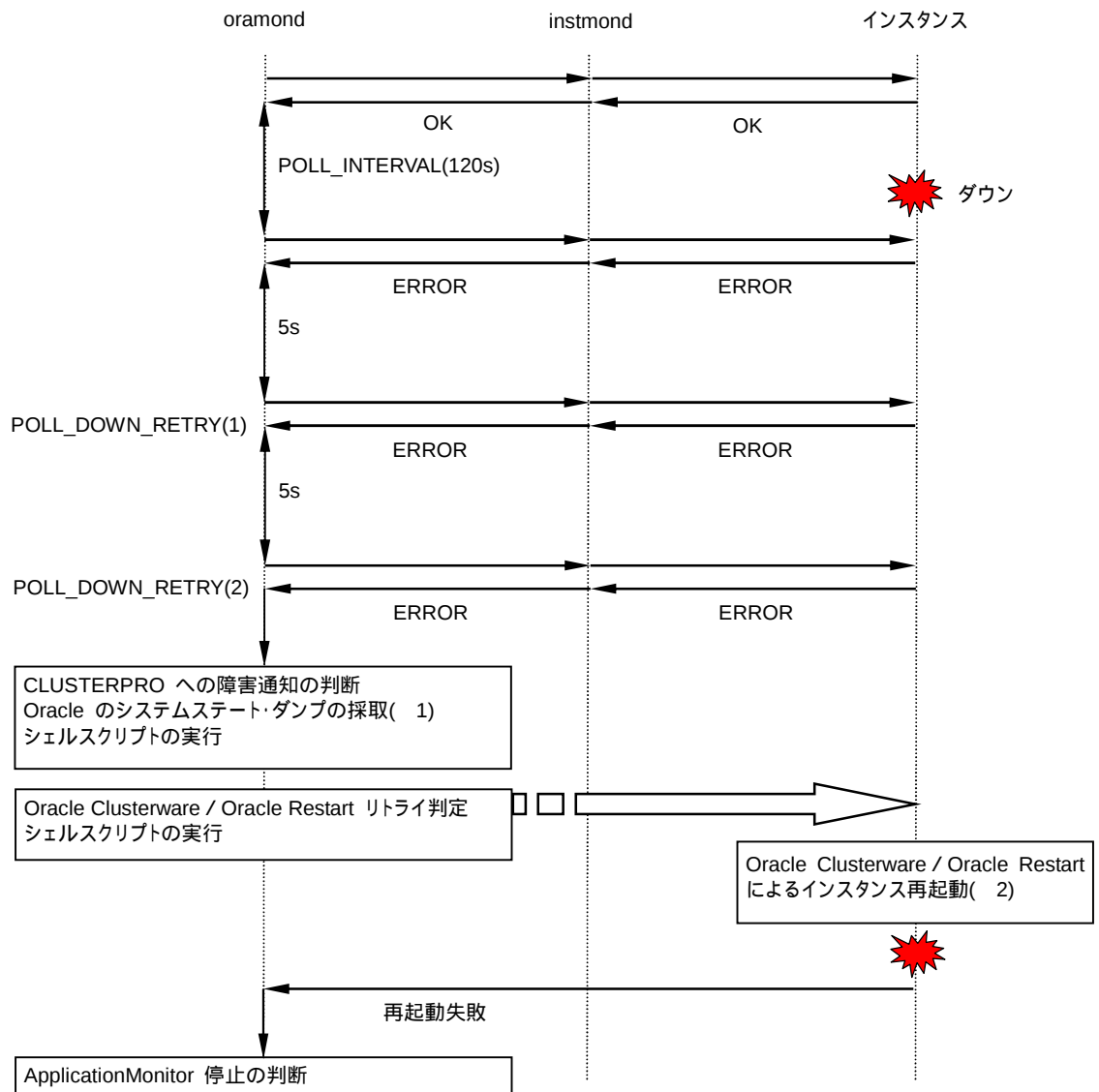


図 4-5 インスタンス監視エラー時の動作 (Oracle Clusterware / Oracle Restart によってインスタンスが再起動されなかった場合)

4 製品の動作

Oracle Clusterware / Oracle Restart によるインスタンスの再起動(2)を待ち合わせても、インスタンスが再起動されなかった場合は、HALT_CONTROL_OPTION パラメータの設定により、障害ノード上の ApplicationMonitor 停止の判断を行います。HALT_CONTROL_OPTION パラメータに ALWAYS が設定されている場合、または OUTLIVE が設定され、他に生存しているノードが存在している場合は、ApplicationMonitor を停止し、CLUSTERPRO へ障害を通知します。

なお、HALT_CONTROL_OPTION パラメータに NEVER が設定されている場合、または OUTLIVE が設定され、他に生存しているノードが存在していない場合は、ApplicationMonitor を再構成し、監視を継続します。CLUSTERPRO への障害通知は行いません。

- (1) インスタンス監視エラー時の Oracle のシステムステート・ダンプ採取は、RAC 構成時のみ非障害ノードにおいて行われます。
- (2) Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境でのみ行われます。

インスタンスの監視で Oracle バックグラウンドプロセスのストールを検出した場合、以下のような動作になります。

(例)

```
InstanceMonitor {  
  POLL_INTERVAL      = 120  
  POLL_TIMEOUT       = 90  
  POLL_STALL_RETRY   = 1  
  SERVICE_DOWN       = YES  
  RESTART_COUNT      = 1  
  RESTART_TIMEOUT    = 60  
}
```

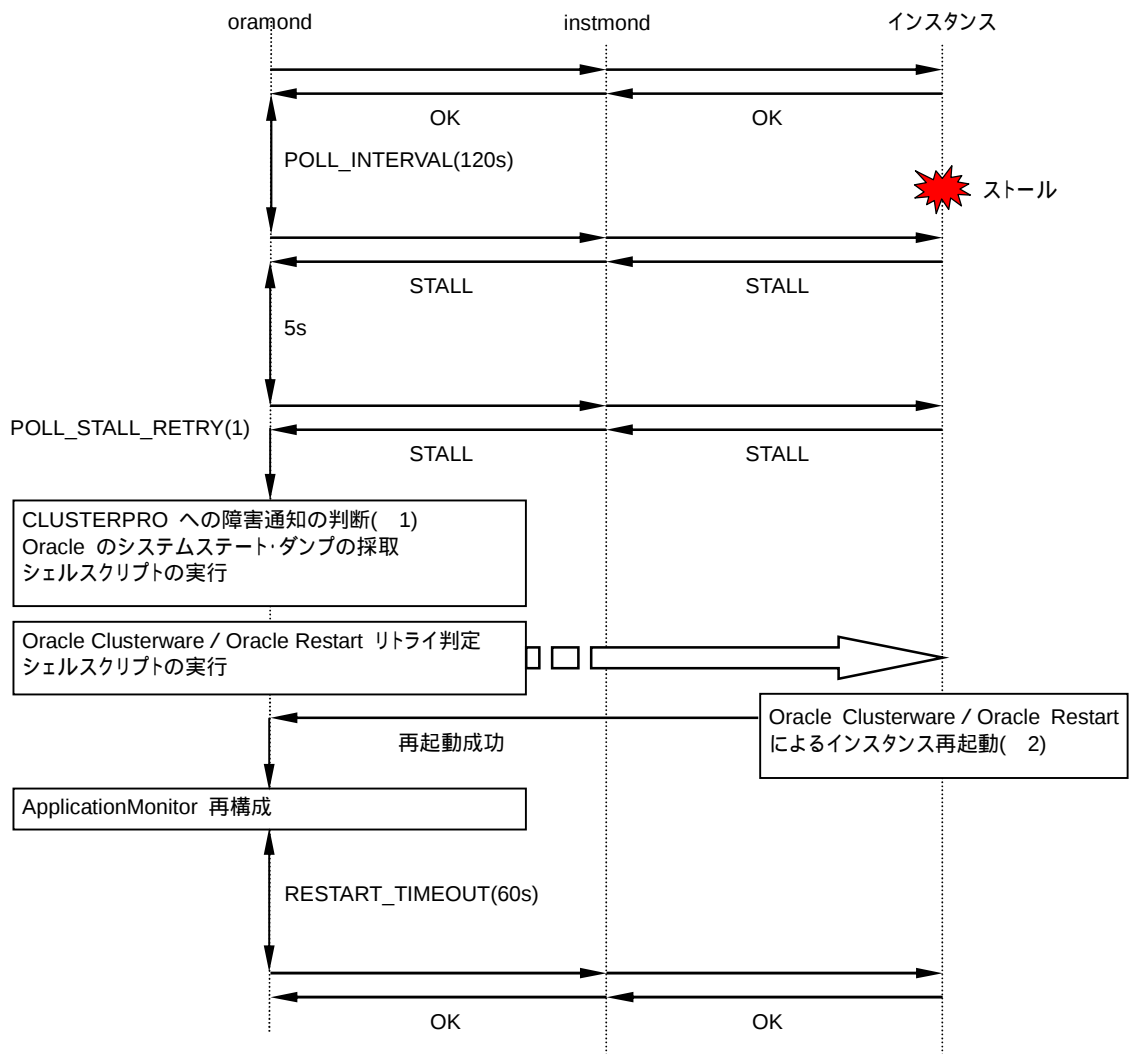


図 4-6 Oracle バックグラウンドプロセスのストール時の動作

4 製品の動作

インスタンスの監視が POLL_TIMEOUT パラメータに設定した時間内に完了しない場合、以下のような動作になります。

(例)

```
InstanceMonitor {  
  POLL_INTERVAL    = 120  
  POLL_TIMEOUT     = 90  
  POLL_STALL_RETRY = 1  
  SERVICE_DOWN     = YES  
  RESTART_COUNT    = 1  
  RESTART_TIMEOUT  = 60  
}
```

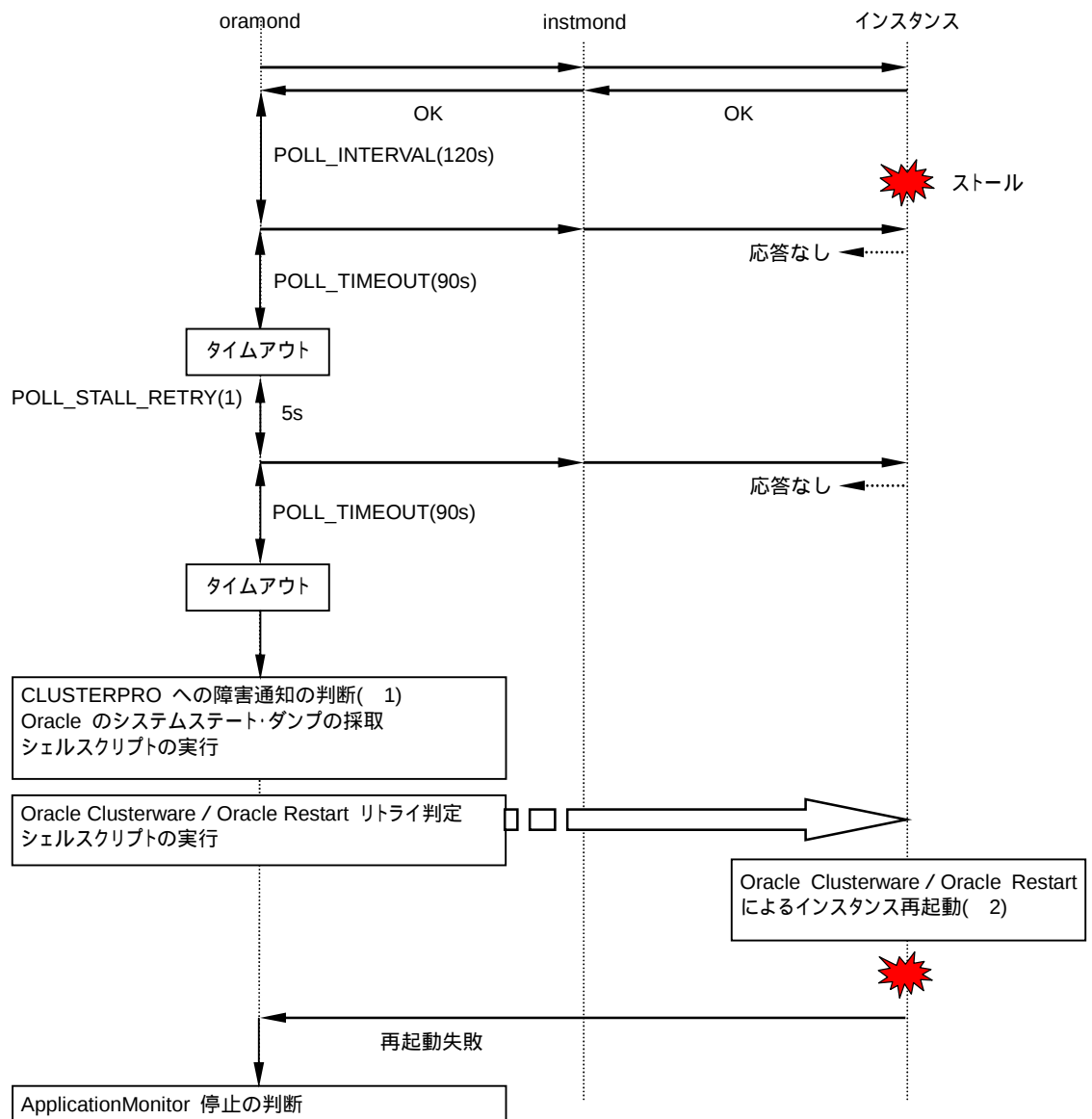


図 4-7 インスタンス監視タイムアウト時の動作

Oracle バックグラウンドプロセスのストールを検出した場合、またはインスタンスへのアクセスが POLL_TIMEOUT パラメータに設定した時間内に完了しない場合、POLL_STALL_RETRY パラメータに 1 以上の値が設定されていると、POLL_STALL_RETRY パラメータに設定した回数の再確認を行います。再確認中にインスタンスの監視で正常を検出した場合、障害と判断せず、監視を続けます。

POLL_STALL_RETRY パラメータに設定した回数のインスタンスの再確認を行っても、インスタンス監視結果がストールになる (Oracle バックグラウンドプロセスのストールまたはインスタンスへのアクセスがタイムアウトする) 場合は、SERVICE_DOWN パラメータの設定により、CLUSTERPRO へノード内の障害を通知するか判断します。インスタンス監視結果がストールとなる際、SERVICE_DOWN パラメータに YES または NO_DOWN が設定されていると、CLUSTERPRO へ障害を通知すると判断します (1)。

CLUSTERPRO へ障害を通知すると判断した場合、モニタ制御デーモンは、Oracle のシステムステート・ダンプの採取およびシェルスクリプトを実行し、情報採取を行います。

なお、インスタンス監視結果がストールとなる際、SERVICE_DOWN パラメータに NO または NO_STALL が設定されていると、モニタ制御デーモンは、CLUSTERPRO へ障害を通知しないと判断するため、監視を続けます。

POLL_STALL_RETRY パラメータに 0 が設定されている場合は、再確認せずに CLUSTERPRO への障害通知の判断、Oracle のシステムステート・ダンプの採取、シェルスクリプトの実行を行います。

Oracle のシステムステート・ダンプの採取動作の詳細は、「4.8 システムステート・ダンプ採取時の動作」、シェルスクリプトの実行動作の詳細は、「4.9 シェルスクリプトの実行時の動作」を参照してください。

情報採取完了後、RESTART_COUNT パラメータに 1 以上の値が設定されていると、Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトを実行し、Oracle Clusterware / Oracle Restart によるインスタンスの再起動 (2) を促して待ち合わせます。

Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの詳細は、「4.10 Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの動作」を参照してください。

インスタンスが再起動された場合は、ApplicationMonitor を再構成し、監視を続けます。

Oracle Clusterware / Oracle Restart によるインスタンスの再起動を待ち合わせても、インスタンスが再起動されなかった場合は、HALT_CONTROL_OPTION パラメータの設定により、障害ノード上の ApplicationMonitor 停止の判断を行います。HALT_CONTROL_OPTION パラメータに ALWAYS が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存

4 製品の動作

在している場合は、ApplicationMonitor を停止し、CLUSTERPRO へ障害を通知します。

なお、HALT_CONTROL_OPTION パラメータに NEVER が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在していない場合は、ApplicationMonitor を再構成し、監視を継続します。CLUSTERPRO への障害通知は行いません。

- (1) CLUSTERPRO へ通知すると判断した障害は、ApplicationMonitor を停止することによって通知します。
- (2) Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境でのみ行われます。

4.3. リスナー監視時の動作

リスナー監視モニタ (lsnrmond) は、起動後に POLL_INTERVAL パラメータに設定した間隔で TNSPING パラメータに設定した Oracle の tnsping コマンドを用いて、リスナーの状態を監視します。

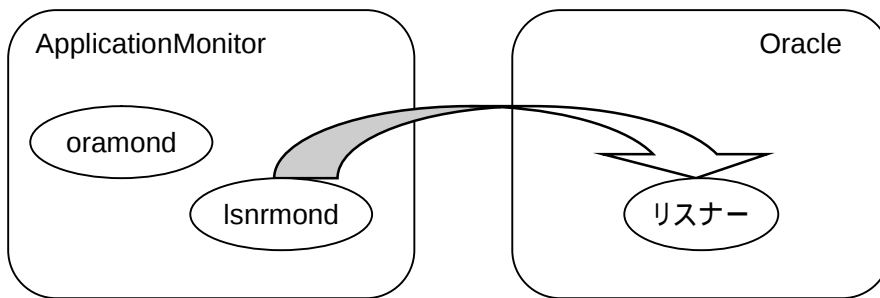


図 4-8 リスナー監視モニタの概要

リスナーの監視は、以下のような動作になります。

(例)

```
ListenerMonitor LISTENER {
  POLL_INTERVAL      = 60
  NET_SERVICE_NAME = LSNR
  GET_STATS          = YES
}
```

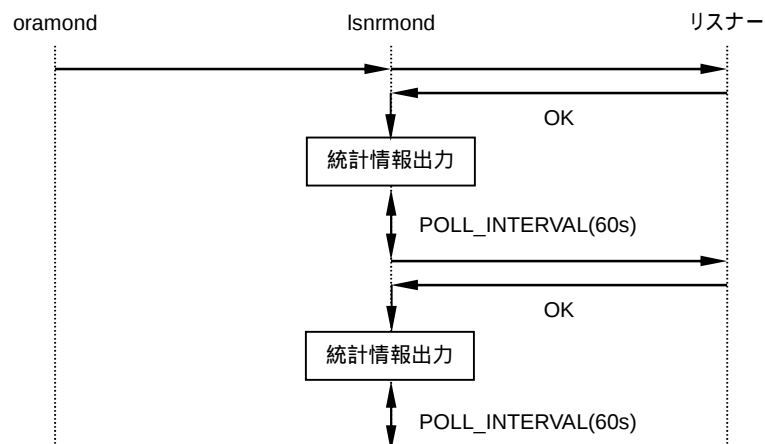


図 4-9 リスナー監視時の動作

リスナー監視モニタは、POLL_INTERVAL パラメータに設定した間隔でリスナーを監視します。

GET_STATS パラメータに YES が設定されている場合、リスナーの状態が正常の際に統計情報(監視 TAT)をファイルへ出力します。

4 製品の動作

リスナーの監視でエラーを検出し、ApplicationMonitor によってリスナーが再起動した場合、以下のような動作になります。

(例)

```
ListenerMonitor LISTENER {  
  POLL_INTERVAL    = 60  
  RESTART_COUNT    = 2  
  RESTART_INTERVAL = 3  
  NET_SERVICE_NAME = LSNR  
  LSNRCTL          = /etc/opt/HA/AM/lbin/lsnr_control.sh  
  TARGET_RESTART   = YES  
}
```

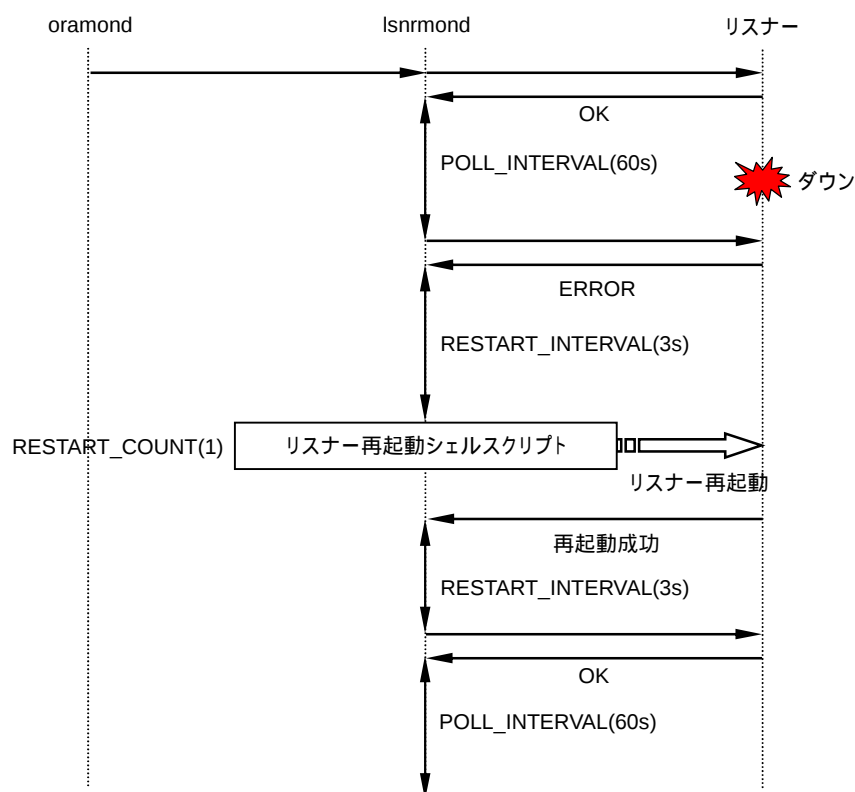


図 4-10 リスナー監視エラー時の動作 (ApplicationMonitor によってリスナーが再起動した場合)

リスナーの監視でエラーを検出した場合、RESTART_COUNT パラメータに 1 以上の値が設定されていると、RESTART_COUNT パラメータに設定した回数のリスナーの再起動 (LSNRCTL パラメータに設定したシェルスクリプトの実行) を行います。リスナーが再起動した場合は、RESTART_INTERVAL パラメータに設定した時間の待ち合わせを行い、監視を再開します。

リスナーの監視でエラーを検出し、ApplicationMonitor によるリスナー再起動が失敗した場合、以下のような動作になります。

(例)

HALT_CONTROL_OPTION = ALWAYS

```
ListenerMonitor LISTENER {
  POLL_INTERVAL    = 60
  SERVICE_DOWN     = YES
  RESTART_COUNT    = 2
  RESTART_INTERVAL = 3
  NET_SERVICE_NAME = LSNR
  LSNRCTL          = /etc/opt/HA/AM/lbin/lsnr_control.sh
  TARGET_RESTART   = YES
}
```

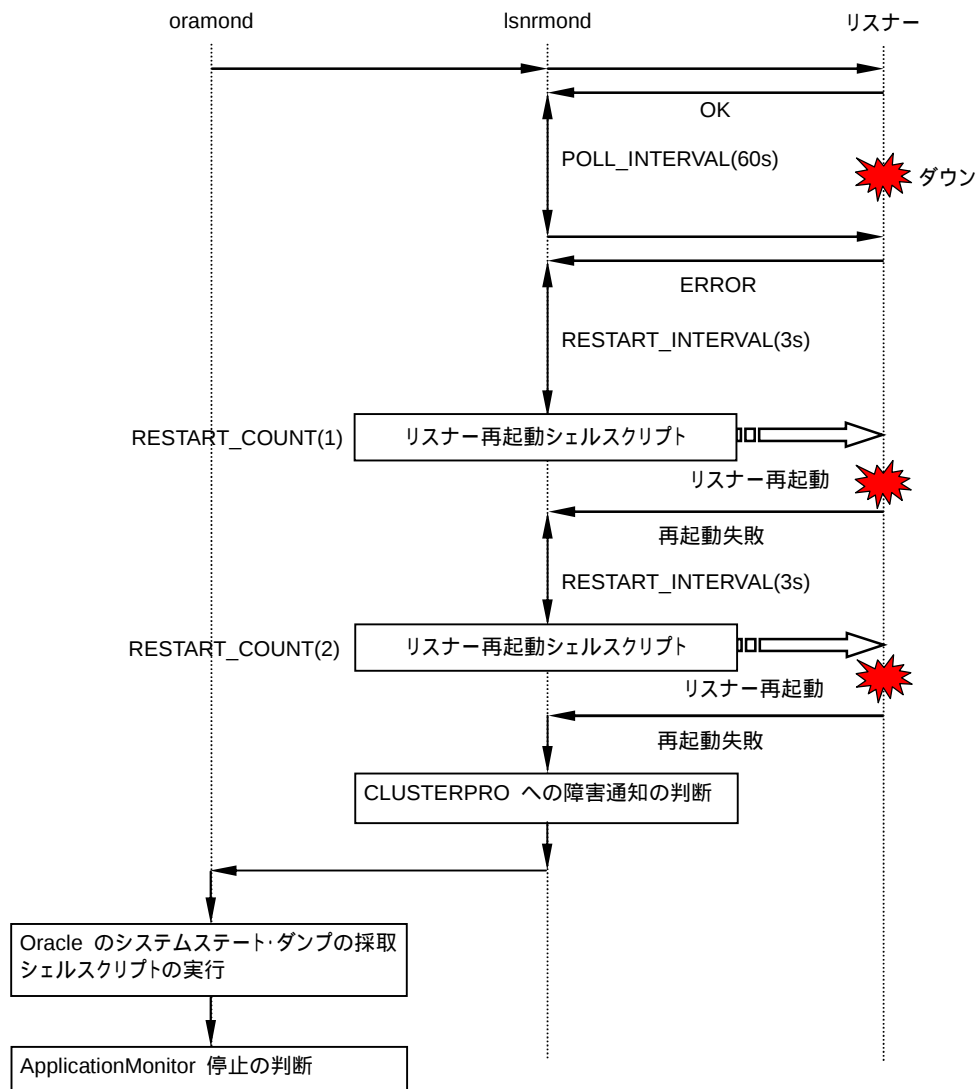


図 4-11 リスナー監視エラー時の動作 (ApplicationMonitor によるリスナー再起動が失敗した場合)

RESTART_COUNT パラメータに設定した回数のリスナーの再起動を行っても、リスナーが起動できない場合は、SERVICE_DOWN パラメータの設定により、CLUSTERPRO ヘノード内の障害を通知するか判断します。リスナー監視結果がエラーとなる際、SERVICE_DOWN パラメータに YES または NO_STALL が設定されていると、CLUSTERPRO へ障害を通知すると判断します。

CLUSTERPRO へ障害を通知すると判断した場合、モニタ制御デーモンは、Oracle のシステムステート・ダンプの採取およびシェルスクリプトを実行し、情報採取を行います。

なお、リスナー監視結果がエラーとなる際、SERVICE_DOWN パラメータに NO または NO_DOWN が設定されていると、モニタ制御デーモンは、CLUSTERPRO へ障害を通知しないと判断するため、監視を継続します。

RESTART_COUNT パラメータに 0 が設定されている場合は、リスナーの再起動を行わずに CLUSTERPRO への障害通知の判断、Oracle のシステムステート・ダンプの採取、シェルスクリプトの実行を行います。

Oracle のシステムステート・ダンプの採取動作の詳細は、「4.8 システムステート・ダンプ採取時の動作」、シェルスクリプトの実行動作の詳細は、「4.9 シェルスクリプトの実行時の動作」を参照してください。

情報採取完了後、HALT_CONTROL_OPTION パラメータの設定により、障害ノード上の ApplicationMonitor 停止の判断を行います。HALT_CONTROL_OPTION パラメータに ALWAYS が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在している場合は、ApplicationMonitor を停止し、CLUSTERPRO へ障害を通知します。

なお、HALT_CONTROL_OPTION パラメータに NEVER が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在していない場合は、ApplicationMonitor を再構成し、監視を継続します。CLUSTERPRO への障害通知は行いません。

リスナーの監視でエラーを検出し、Oracle Clusterware / Oracle Restart によってリスナーが再起動された場合、以下のような動作になります。

(例)

```
ListenerMonitor LISTENER {
  POLL_INTERVAL      = 60
  RESTART_COUNT      = 1
  RESTART_INTERVAL   = 3
  NET_SERVICE_NAME   = LSNR
  LSNRCTL            = /etc/opt/HA/AM/lbin/lsnr_control.sh
  RESTART_TIMEOUT    = 10
  TARGET_RESTART     = NO
}
```

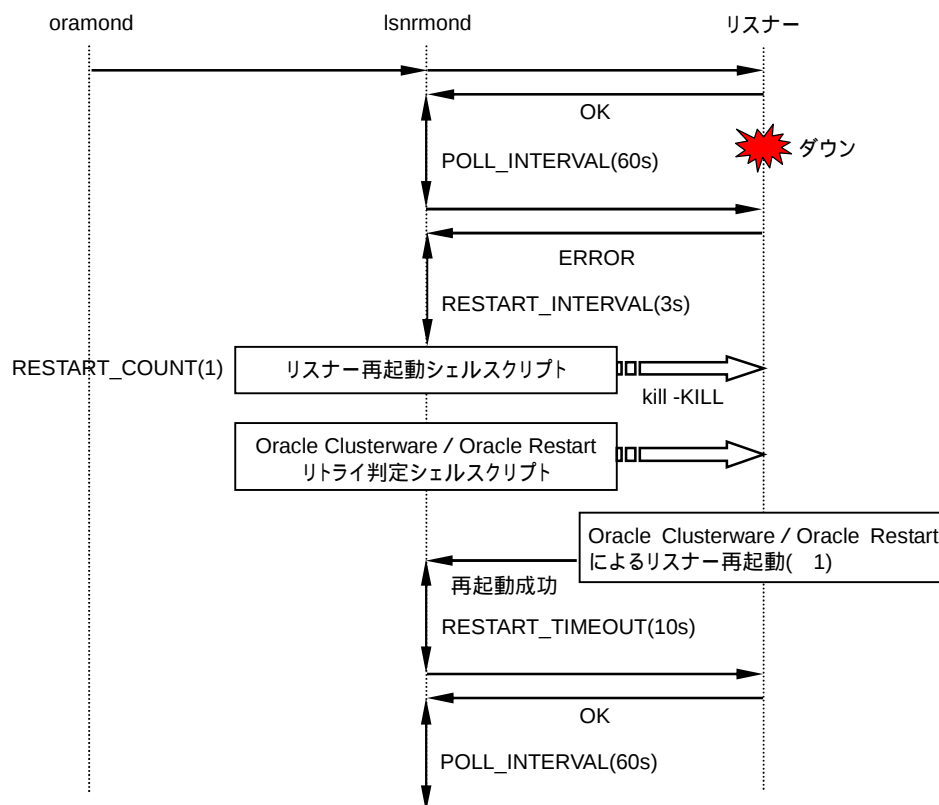


図 4-12 リスナー監視エラー時の動作 (Oracle Clusterware / Oracle Restart によってリスナーが再起動された場合)

リスナーの監視でエラーを検出した場合、RESTART_COUNT パラメータに 1 以上、TARGET_RESTART パラメータの値が NO に設定されていると、LSNRCTL パラメータに設定したシェルスクリプトを実行します。

リスナーに対して KILL コマンドを実行した後、Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトを実行し、Oracle Clusterware / Oracle Restart によるリスナーの再起動 (1) を促して待ち合わせます。

Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの詳細は、「4.10 Oracle

4 製品の動作

Clusterware / Oracle Restart リトライ判定シェルスクリプトの動作」を参照してください。

リスナーが再起動された場合は、RESTART_TIMEOUT パラメータに設定した時間の待ち合わせを行い、監視を継続します。

- (1) Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境でのみ行われます。

リスナーの監視でエラーを検出し、Oracle Clusterware / Oracle Restart によってリスナーが再起動されなかった場合、以下のような動作になります。

(例)

HALT_CONTROL_OPTION = ALWAYS

```
ListenerMonitor LISTENER {
  POLL_INTERVAL      = 60
  SERVICE_DOWN       = YES
  RESTART_COUNT       = 1
  RESTART_INTERVAL    = 3
  NET_SERVICE_NAME    = LSNR
  LSNRCTL             = /etc/opt/HA/AM/lbin/lsnr_control.sh
  TARGET_RESTART      = NO
}
```

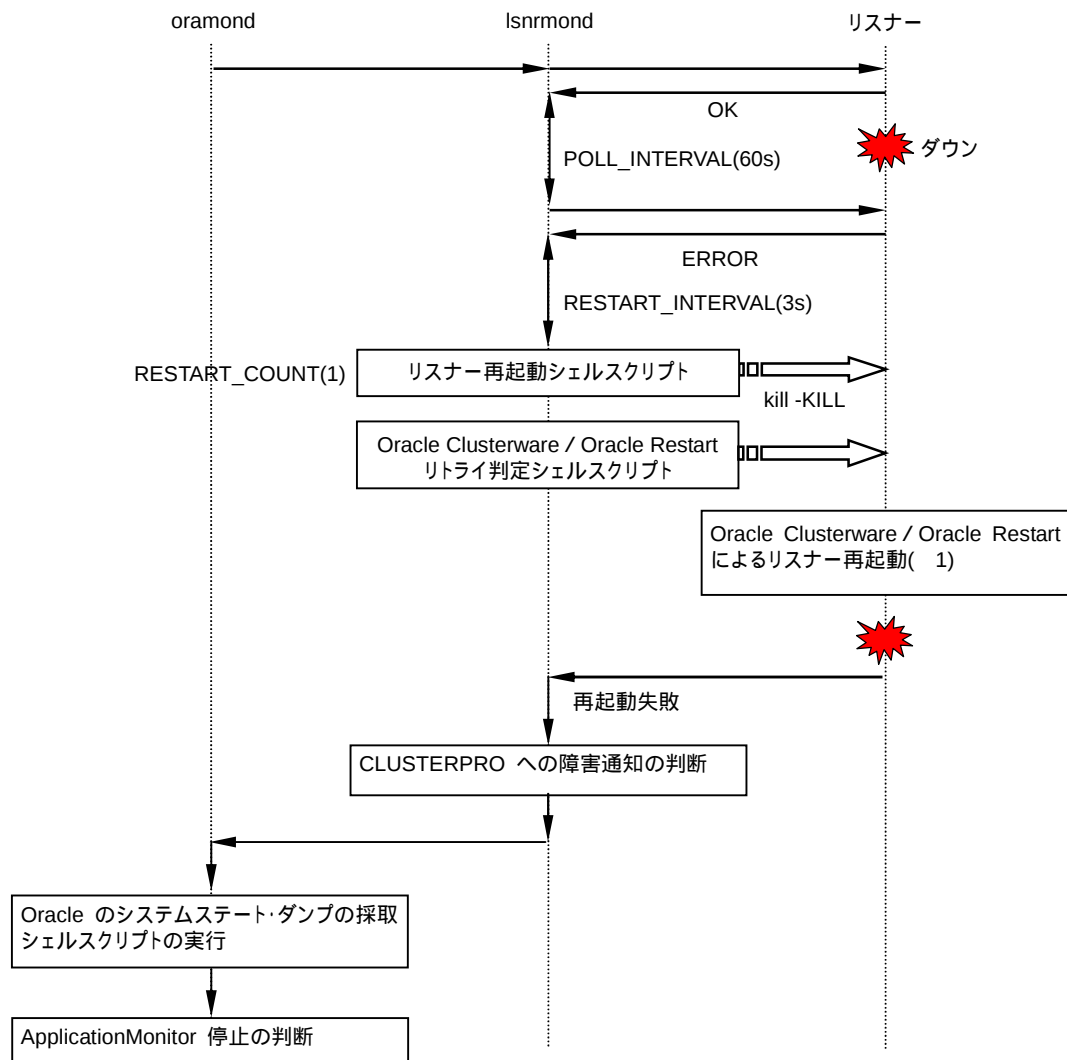


図 4-13 リスナー監視エラー時の動作 (Oracle Clusterware / Oracle Restart によってリスナーが再起動されなかった場合)

Oracle Clusterware / Oracle Restart によるリスナーの再起動(1)を待ち合わせても、リスナーが再起動されなかった場合は、SERVICE_DOWN パラメータの設定により、CLUSTERPRO へノード内の障害を通知するか判断します。リスナー監視結果がエラーとなる際、SERVICE_DOWN パラメータに YES または NO_STALL が設定されていると、CLUSTERPRO へ障害を通知すると判断します。

CLUSTERPRO へ障害を通知すると判断した場合、モニタ制御デーモンは、Oracle のシステムステート・ダンプの採取およびシェルスクリプトを実行し、情報採取を行います。

なお、リスナー監視結果がエラーとなる際、SERVICE_DOWN パラメータに NO または NO_DOWN が設定されていると、モニタ制御デーモンは、CLUSTERPRO へ障害を通知しないと判断するため、監視を継続します。

RESTART_COUNT パラメータに 0 が設定されている場合は、リスナー再起動の待ち合わせを行わずに CLUSTERPRO への障害通知の判断、Oracle のシステムステート・ダンプの採取、シェルスクリプトの実行を行います。

Oracle のシステムステート・ダンプの採取動作の詳細は、「4.8 システムステート・ダンプ採取時の動作」、シェルスクリプトの実行動作の詳細は、「4.9 シェルスクリプトの実行時の動作」を参照してください。

情報採取完了後、HALT_CONTROL_OPTION パラメータの設定により、障害ノード上の ApplicationMonitor 停止の判断を行います。HALT_CONTROL_OPTION パラメータに ALWAYS が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在している場合は、ApplicationMonitor を停止し、CLUSTERPRO へ障害を通知します。

なお、HALT_CONTROL_OPTION パラメータに NEVER が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在していない場合は、ApplicationMonitor を再構成し、監視を継続します。CLUSTERPRO への障害通知は行いません。

(1) Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境でのみ行われます。

リスナーの監視で POLL_TIMEOUT パラメータに設定した時間内に完了せず、ApplicationMonitor によるリスナー再起動が失敗した場合、以下のような動作になります。

(例)

HALT_CONTROL_OPTION = ALWAYS

```
ListenerMonitor LISTENER {
  POLL_INTERVAL    = 60
  POLL_TIMEOUT     = 90
  SERVICE_DOWN     = YES
  RESTART_COUNT    = 2
  RESTART_INTERVAL = 3
  NET_SERVICE_NAME = LSNR
  LSNRCTL          = /etc/opt/HA/AM/lbin/lsnr_control.sh
  TARGET_RESTART   = YES
}
```

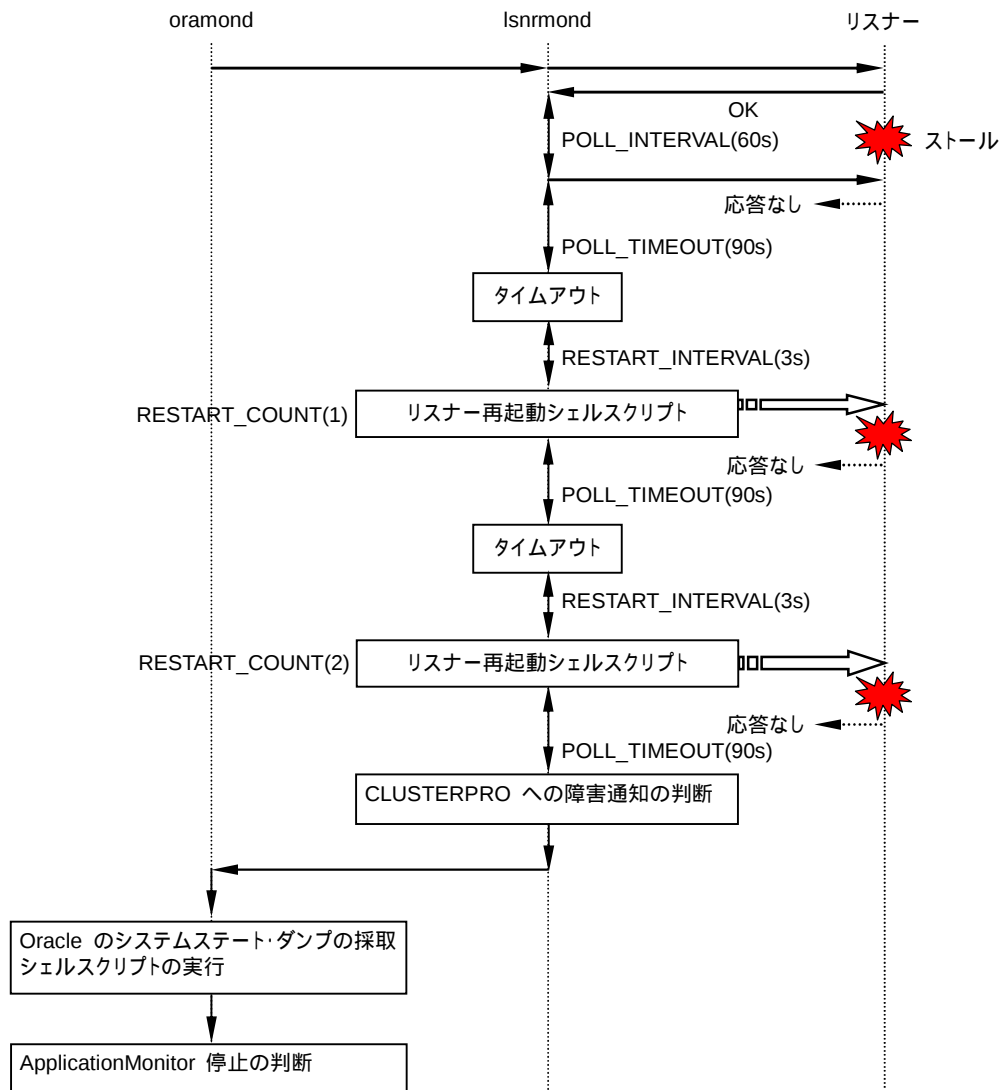


図 4-14 リスナー監視タイムアウト時の動作 (ApplicationMonitor によるリスナー再起動が失敗した場合)

リスナーの監視が POLL_TIMEOUT パラメータに設定した時間内に完了しない場合、RESTART_COUNT パラメータに 1 以上の値が設定されていると、RESTART_COUNT パラメータに設定した回数のリスナーの再起動(LSNRCTL パラメータに設定したシェルスクリプトの実行)を行います。

RESTART_COUNT パラメータに設定した回数のリスナーの再起動を行っても、リスナー監視結果がストールになる(タイムアウトする)場合は、SERVICE_DOWN パラメータの設定により、CLUSTERPRO へノード内の障害を通知するか判断します。リスナー監視結果がストールとなる際、SERVICE_DOWN パラメータに YES または NO_DOWN が設定されていると、CLUSTERPRO へ障害を通知すると判断します。

CLUSTERPRO へ障害を通知すると判断した場合、モニタ制御デーモンは、Oracle のシステムステート・ダンプの採取およびシェルスクリプトを実行し、情報採取を行います。

なお、リスナー監視結果がストールとなる際、SERVICE_DOWN パラメータに NO または NO_STALL が設定されていると、モニタ制御デーモンは、CLUSTERPRO へ障害を通知しないと判断するため、監視を継続します。

RESTART_COUNT パラメータに 0 が設定されている場合は、リスナーの再起動を行わずに CLUSTERPRO への障害通知の判断、Oracle のシステムステート・ダンプの採取、シェルスクリプトの実行を行います。

Oracle のシステムステート・ダンプの採取動作の詳細は、「4.8 システムステート・ダンプ採取時の動作」、シェルスクリプトの実行動作の詳細は、「4.9 シェルスクリプトの実行時の動作」を参照してください。

情報採取完了後、HALT_CONTROL_OPTION パラメータの設定により、障害ノード上の ApplicationMonitor 停止の判断を行います。HALT_CONTROL_OPTION パラメータに ALWAYS が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在している場合は、ApplicationMonitor を停止し、CLUSTERPRO へ障害を通知します。

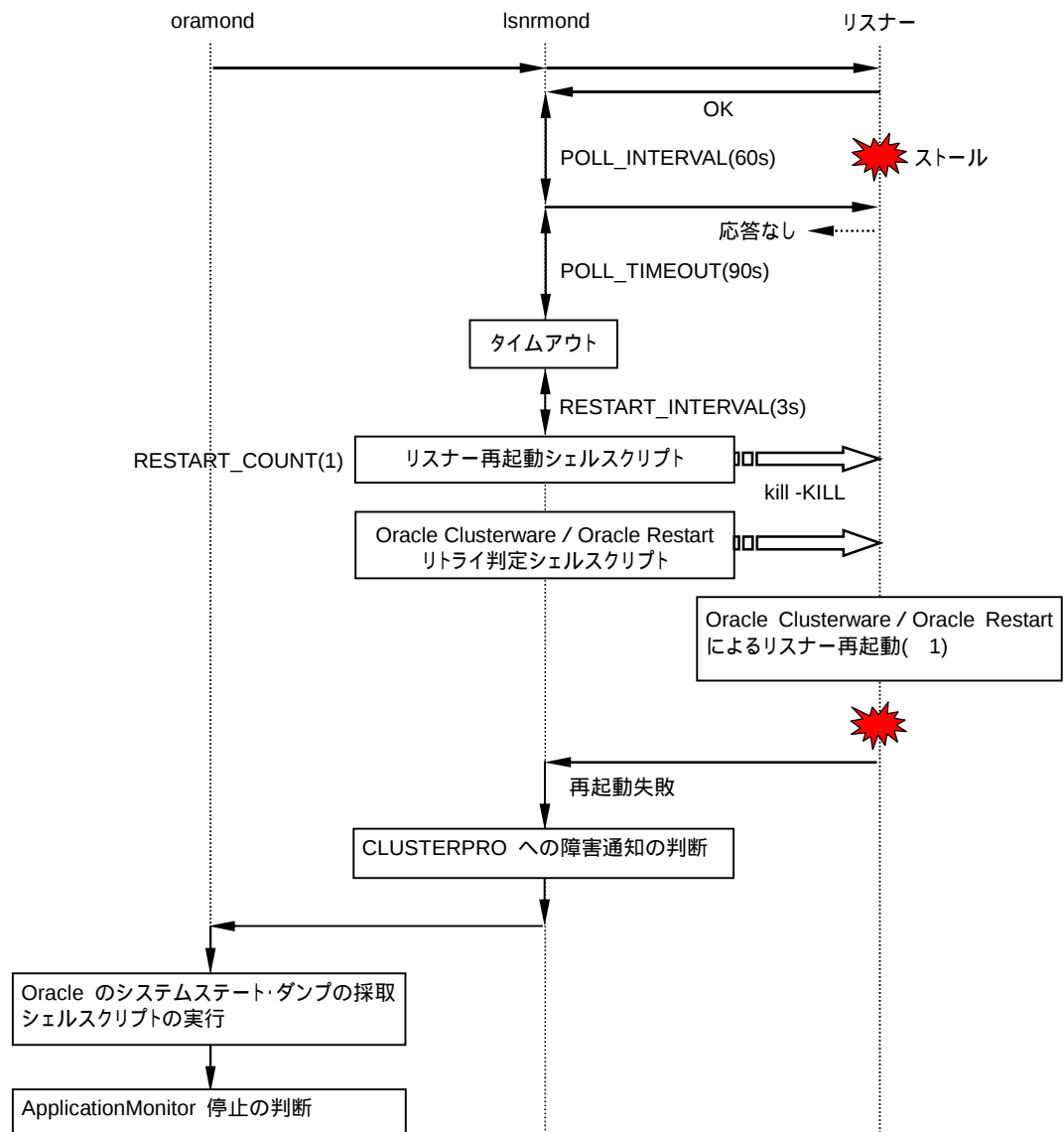
なお、HALT_CONTROL_OPTION パラメータに NEVER が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在していない場合は、ApplicationMonitor を再構成し、監視を継続します。CLUSTERPRO への障害通知は行いません。

リスナーの監視で POLL_TIMEOUT パラメータに設定した時間内に完了せず、Oracle Clusterware / Oracle Restart によってリスナーが再起動されなかった場合、以下のような動作になります。

(例)

HALT_CONTROL_OPTION = ALWAYS

```
ListenerMonitor LISTENER {
  POLL_INTERVAL      = 60
  POLL_TIMEOUT       = 90
  SERVICE_DOWN       = YES
  RESTART_COUNT      = 1
  RESTART_INTERVAL   = 3
  NET_SERVICE_NAME   = LSNR
  LSNRCTL             = /etc/opt/HA/AM/lbin/lsnr_control.sh
  TARGET_RESTART     = NO
}
```



**図 4-15 リスナー監視タイムアウト時の動作(Oracle Clusterware / Oracle Restart
によってリスナーが再起動されなかった場合)**

Oracle Clusterware / Oracle Restart によるリスナーの再起動(1)を待ち合わせても、リスナーが再起動されなかった場合は、SERVICE_DOWN パラメータの設定により、CLUSTERPRO へノード内の障害を通知するか判断します。リスナー監視結果がストールになる(タイムアウトする)際、SERVICE_DOWN パラメータに YES または NO_DOWN が設定されていると、CLUSTERPRO へ障害を通知すると判断します。

CLUSTERPRO へ障害を通知すると判断した場合、モニタ制御デーモンは、Oracle のシステムステート・ダンプの採取およびシェルスクリプトを実行し、情報採取を行います。

なお、リスナー監視結果がストールとなる際、SERVICE_DOWN パラメータに NO または NO_STALL が設定されていると、モニタ制御デーモンは、CLUSTERPRO へ障害を通知しないと判断するため、監視を継続します。

情報採取完了後、HALT_CONTROL_OPTION パラメータの設定により、障害ノード上の ApplicationMonitor 停止の判断を行います。HALT_CONTROL_OPTION パラメータに ALWAYS が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在している場合は、ApplicationMonitor を停止し、CLUSTERPRO へ障害を通知します。

なお、HALT_CONTROL_OPTION パラメータに NEVER が設定されている場合または OUTLIVE が設定され、他に生存しているノードが存在していない場合は、ApplicationMonitor を再構成し、監視を継続します。CLUSTERPRO への障害通知は行いません。

(1) Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境でのみ行われます。

4.4. データベース領域の容量監視時の動作

表領域監視モニタ (tschkmond) は、起動後に POLL_INTERVAL パラメータに設定した間隔でデータベース領域の容量不足を監視します。監視方法は、データディクショナリ表の参照です。

データベース領域の容量監視は、以下のような動作になります。

(例)

```
TableSpaceMonitor TSCHKMOND {
  POLL_INTERVAL = 3600
  TableSpace TABLESPACE {
    USEABLE_EXTENT_BLOCK = 10000
    FREE_SPACE           = 20
    GET_STATS            = YES
  }
}
```

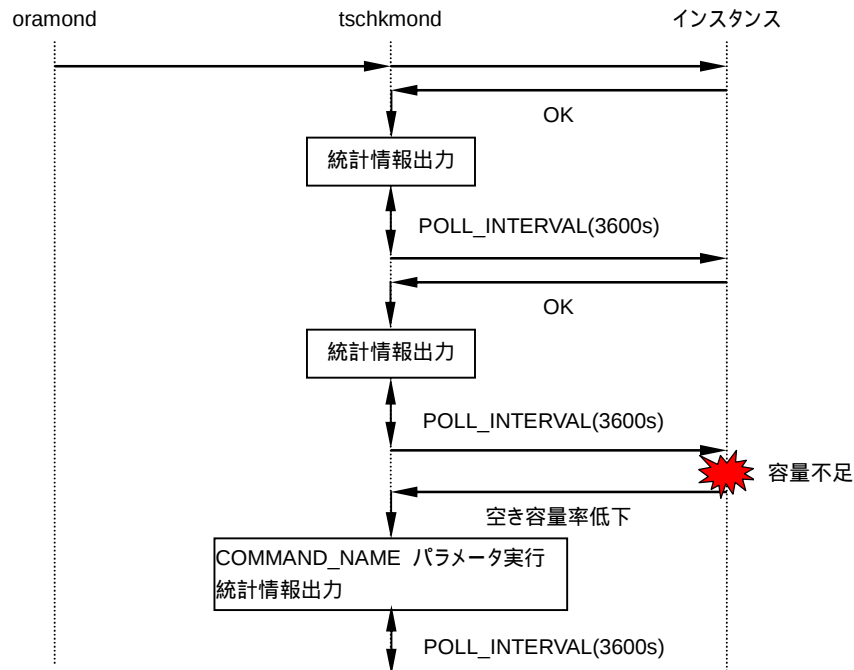


図 4-16 データベース領域の容量監視時の動作

表領域監視モニタは、POLL_INTERVAL パラメータに設定した間隔でインスタンスへ接続し、表領域の最大使用可能エクステントサイズおよび空き容量率を監視します。

GET_STATS パラメータに YES が設定されている場合、統計情報 (表領域情報) をファイルへ出力します。

監視対象の表領域の最大使用可能エクステントサイズまたは空き容量率が設定したしきい値を下回った場合、COMMAND_NAME パラメータに設定したコマンドを実行し、監視を継続します。

なお、データベースがマウント状態の場合は、インスタンスに対する起動確認のみ行います。

4.5. ローカルディスク領域の容量監視時の動作

ディスク領域監視モニタ (fdsmond) は、起動後に POLL_INTERVAL パラメータに設定した間隔でローカルディスク領域の容量不足を監視します。

ローカルディスク領域の容量監視は、以下のような動作になります。

(例)

```
DiskSpaceMonitor {
  POLL_INTERVAL = 3600
  FREE_SPACE    = 10
}
```

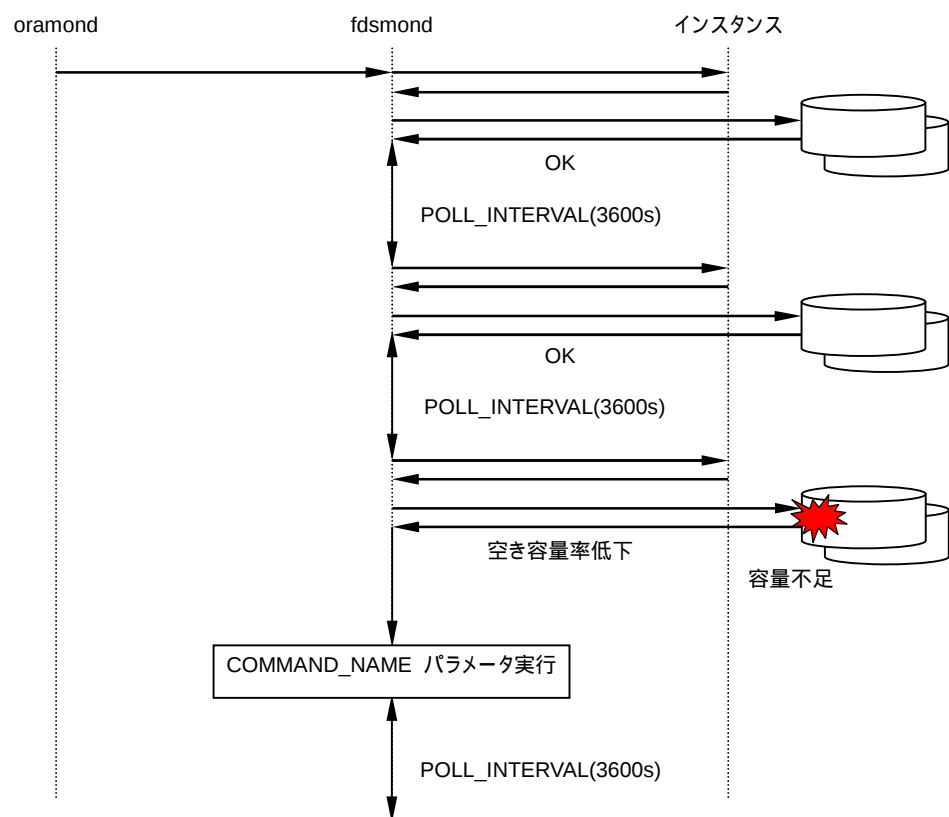


図 4-17 ローカルディスク領域の容量監視時の動作

ディスク領域監視モニタは、POLL_INTERVAL パラメータに設定した間隔でインスタンスへ接続し、アーカイブログ出力先、ユーザー・トレース出力先、バックグラウンド・プロセス・トレース出力先、コアファイル出力先および監査証跡ファイル出力先の空き容量率を監視します。

監視対象のディスク領域の空き容量率が設定したしきい値を下回った場合、COMMAND_NAME パラメータに設定したコマンドを実行し、監視を続けます。

なお、データベースがマウント状態の際、任意の Oracle ユーザーによる監視を行う (ORACLE_USER / ORACLE_PASS パラメータが指定されている) 場合は、インスタンスに対する起動確認のみ行います。

4.6. スタンバイ・インスタンス監視時の動作

スタンバイデータベース監視モニタ (stbdbmond) は、起動後に SYSTEM 表領域上の表の更新または POLL_INTERVAL パラメータに設定した間隔でスタンバイ・インスタンスを監視します。

スタンバイ・インスタンスの REDO ログ転送状況の監視は、以下のような動作になります。

(例)

```
#
# プライマリ・サイトの設定
#

INTERSITE_PORT      = 25311
STANDBY_DB_CHECK = YES

DataGuardSite SiteStb {
    DB_UNIQUE_NAME = "db_stb"
    INTERSITE_PORT  = 25312
    SITE_NODE       = node2
}

StandbyDatabaseMonitor {
    Transport {
        POLL_INTERVAL = 60
        Destination SiteStb {
            PRIMARY_NET_SERVICE_NAME = "dg_stb1"
        }
    }
}

#
# スタンバイ・サイトの設定
#

INTERSITE_PORT      = 25312
STANDBY_DB_CHECK = YES

DataGuardSite SitePri {
    DB_UNIQUE_NAME = "db_pri"
    INTERSITE_PORT  = 25311
    SITE_NODE       = node1
}

StandbyDatabaseMonitor {
    Transport {
        POLL_INTERVAL = 60
        Destination SitePri {
            PRIMARY_NET_SERVICE_NAME = "dg_pri1"
        }
    }
}
```

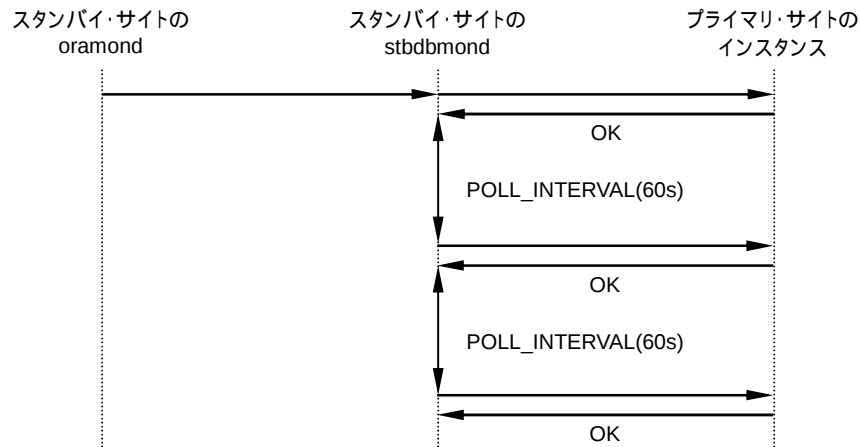


図 4-18 スタンバイ・インスタンスの REDO ログ転送状況監視時の動作
(一定時間経過による監視)

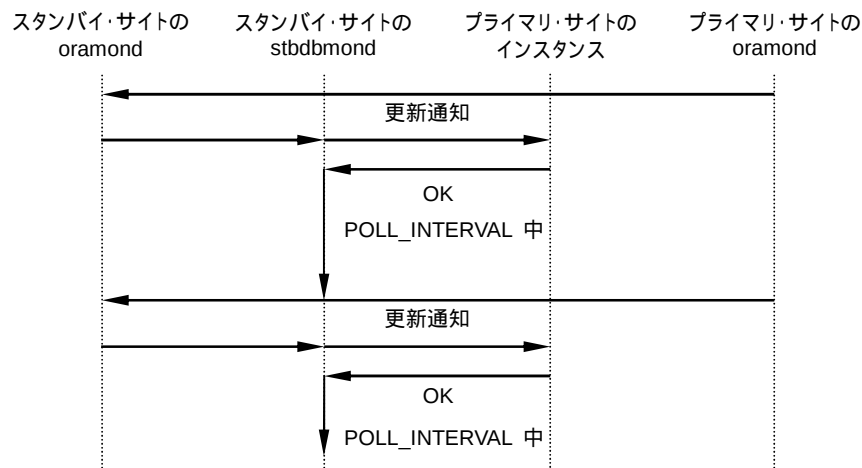


図 4-19 スタンバイ・インスタンスの REDO ログ転送状況監視時の動作
(更新通知による監視)

スタンバイデータベース監視モニタは、POLL_INTERVAL パラメータに設定した間隔でプライマリ・サイトのインスタンスにリスナー経由でアクセスし、REDO ログの転送状況を監視します。

また、プライマリ・サイトとスタンバイ・サイトの間でモニタ制御デーモン (oramond) 同士が通信を行っています。プライマリ・サイトで SYSTEM 表領域上の表を更新(1)すると、スタンバイ・サイトのモニタ制御デーモンがプライマリ・サイトのモニタ制御デーモンから更新通知を受け取り、POLL_INTERVAL パラメータの待ち合わせ中であっても REDO ログの転送状況を監視します。

なお、スタンバイ・サイトがフィジカル・スタンバイ・データベースまたはデータ保護モードが最大パフォーマンスモードの場合は、プライマリ・サイトのインスタンスに対する起動確認のみ行います。

- (1) SYSTEM 表領域上の表の更新動作の詳細は、「4.2 インスタンス監視時の動作」を参照してください。

4 製品の動作

スタンバイ・インスタンスの REDO ログ転送状況の監視でエラーを検出した場合は、以下のような動作になります。

(例)

```
StandbyDatabaseMonitor {  
  Transport {  
    POLL_INTERVAL = 60  
    RETRY_COUNT = 1  
  }  
}
```

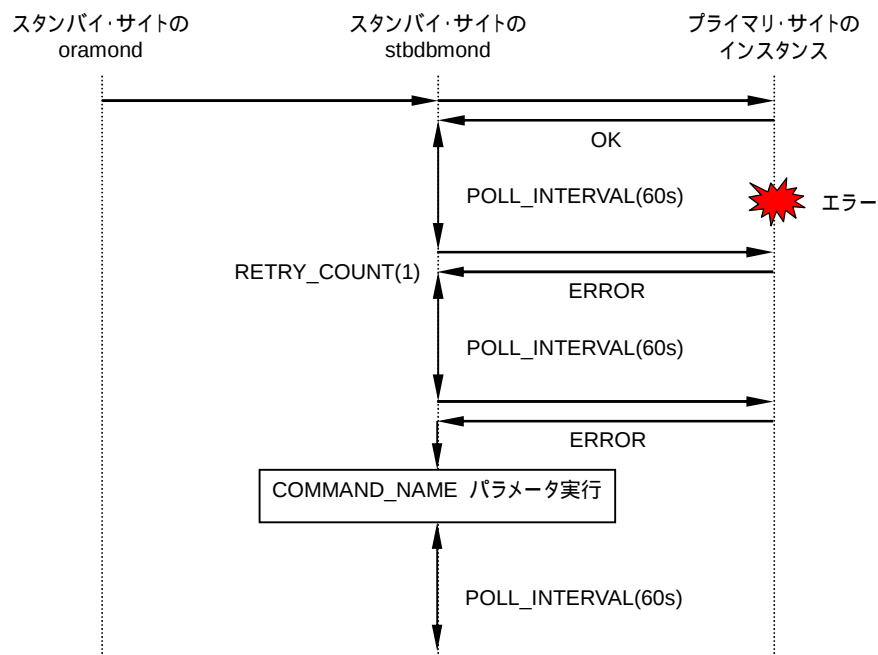


図 4-20 スタンバイ・インスタンスの REDO ログ転送状況監視エラー時の動作

スタンバイ・インスタンスの REDO ログ転送状況監視でエラーを検出した場合、RETRY_COUNT パラメータに 1 以上の値が設定されていると、RETRY_COUNT パラメータに設定した回数の REDO ログ転送状況の再確認を行います。再確認中に正常を検出した場合、障害と判断せず、監視を継続します。

RETRY_COUNT パラメータに設定した回数の再確認を行っても、監視結果がエラーになる場合は、COMMAND_NAME パラメータに設定したコマンドを実行し、監視を継続します。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに COMMAND_NAME パラメータに設定したコマンドの実行を行います。

スタンバイ・インスタンスの REDO ログ適用状況の監視は、以下のような動作になります。

(例)

```
StandbyDatabaseMonitor {
  RedoApply {
    POLL_INTERVAL = 60
  }
}
```

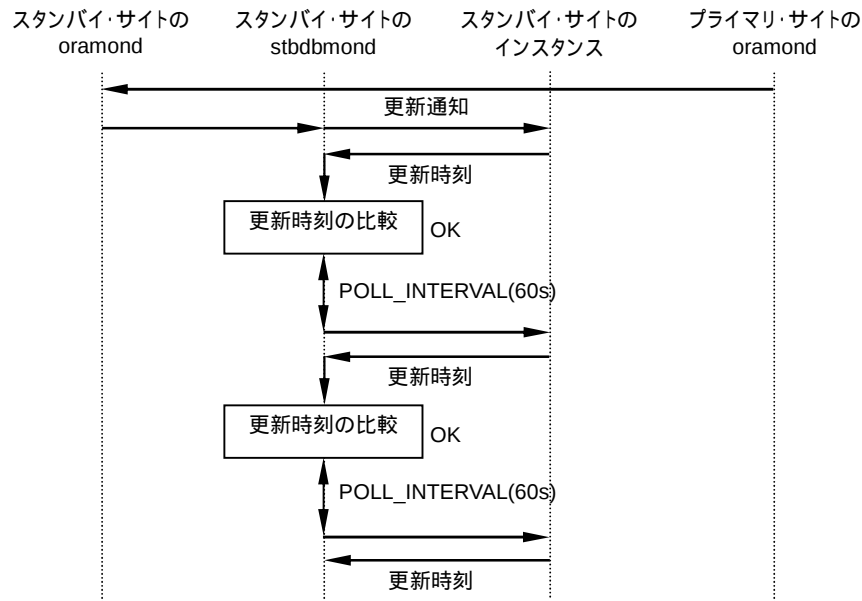


図 4-21 スタンバイ・インスタンスの REDO ログ適用状況監視時の動作
(一定時間経過による監視)

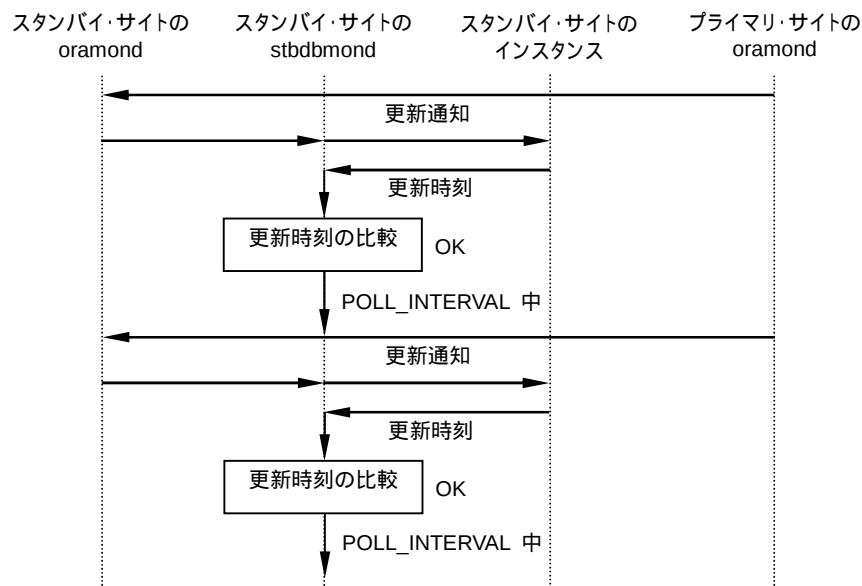


図 4-22 スタンバイ・インスタンスの REDO ログ適用状況監視時の動作
(更新通知による監視)

4 製品の動作

スタンバイデータベース監視モニタは、`POLL_INTERVAL` パラメータに設定した間隔でスタンバイ・サイトのインスタンスにアクセスし、`REDO` ログの適用状況を監視します。

また、プライマリ・サイトとスタンバイ・サイトの間でモニタ制御デーモン同士が通信を行っています。プライマリ・サイトで `SYSTEM` 表領域上の表を更新 (1) すると、スタンバイ・サイトのモニタ制御デーモンがプライマリ・サイトのモニタ制御デーモンから更新通知を受け取り、`POLL_INTERVAL` パラメータの待ち合わせ中であっても `REDO` ログの適用状況を監視します。

なお、スタンバイ・サイトがフィジカル・スタンバイ・データベースまたはスナップショット・スタンバイ・データベースの場合は、スタンバイ・サイトのインスタンスに対する起動確認のみ行います。

- (1) `SYSTEM` 表領域上の表の更新動作の詳細は、「4.2 インスタンス監視時の動作」を参照してください。

スタンバイ・インスタンスの REDO ログ適用状況の監視でエラーを検出した場合は、以下のような動作になります。

(例)

```
StandbyDatabaseMonitor {
  RedoApply {
    POLL_INTERVAL = 60
    ALLOWABLE_TIME = 60
  }
}
```

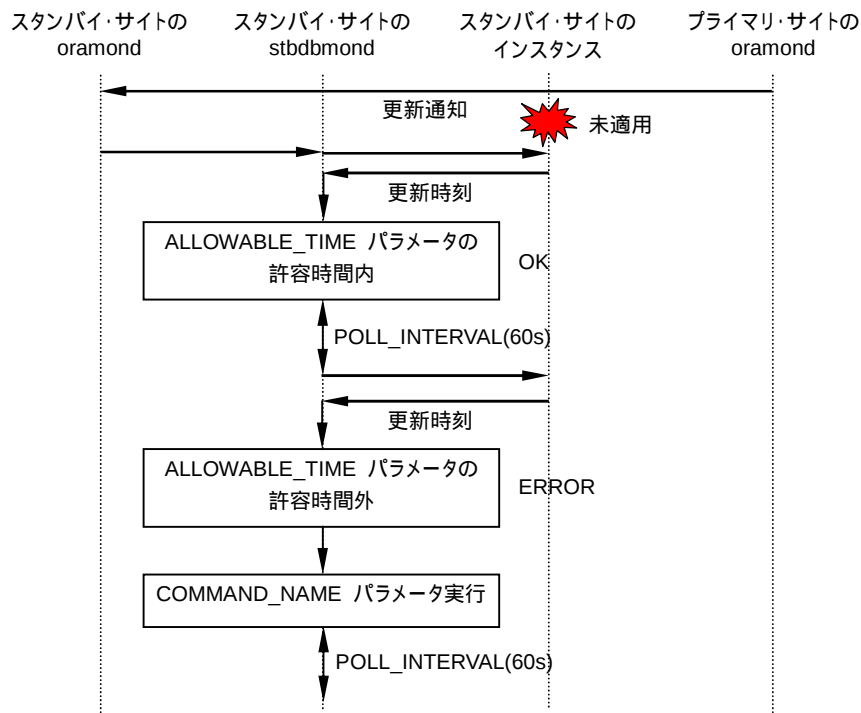


図 4-23 スタンバイ・インスタンスの REDO ログ適用状況監視エラー時の動作

スタンバイ・インスタンスの REDO ログ適用状況監視でエラー (REDO ログの未適用) を検出した場合、ALLOWABLE_TIME パラメータに設定された時間内の遅延は、正常として判断します。

POLL_INTERVAL パラメータに設定した時間の経過または更新通知の受け取り時に REDO ログの適用状況を再確認します。

ALLOWABLE_TIME パラメータに設定した時間の遅延を検出した場合は、エラーとして判断します。エラーと判断した際は、COMMAND_NAME パラメータに設定したコマンドを実行し、監視を続けます。

4 製品の動作

モニタ制御デーモンは、Oracle の Data Guard コマンドを用いて、すべてのサイトの中からプライマリ・サイトを特定し、Oracle Data Guard Broker プロセスの状態を監視します。

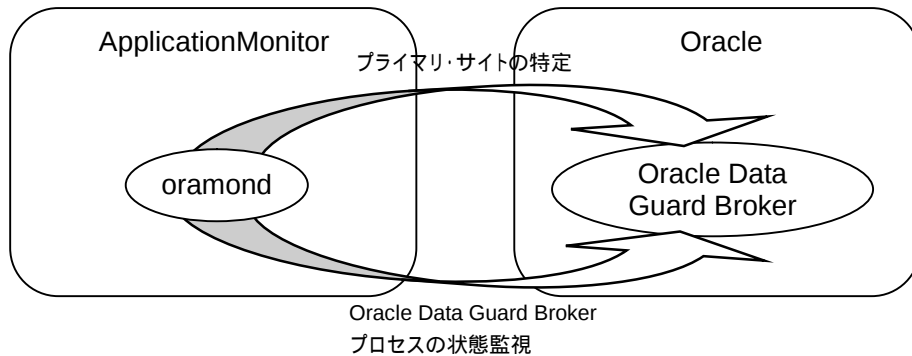


図 4-24 スタンバイ・インスタスの Oracle Data Guard Broker プロセスの状態監視時の概要

スタンバイ・インスタスの Oracle Data Guard Broker プロセスの状態監視は、以下のような動作になります。

(例)

```
StandbyDatabaseMonitor {  
  OdgBroker {  
    POLL_INTERVAL = 60  
  }  
}
```

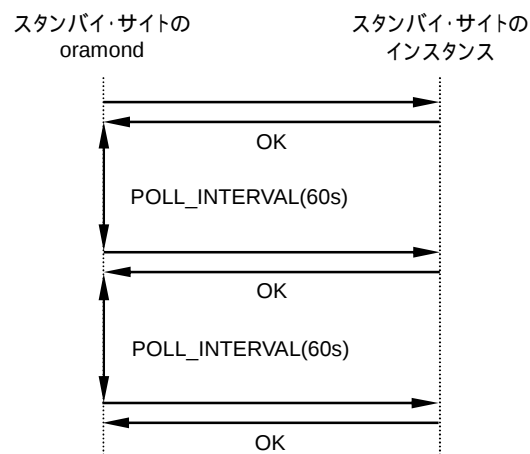


図 4-25 スタンバイ・インスタスの Oracle Data Guard Broker プロセスの状態監視時の動作(一定時間経過による監視)

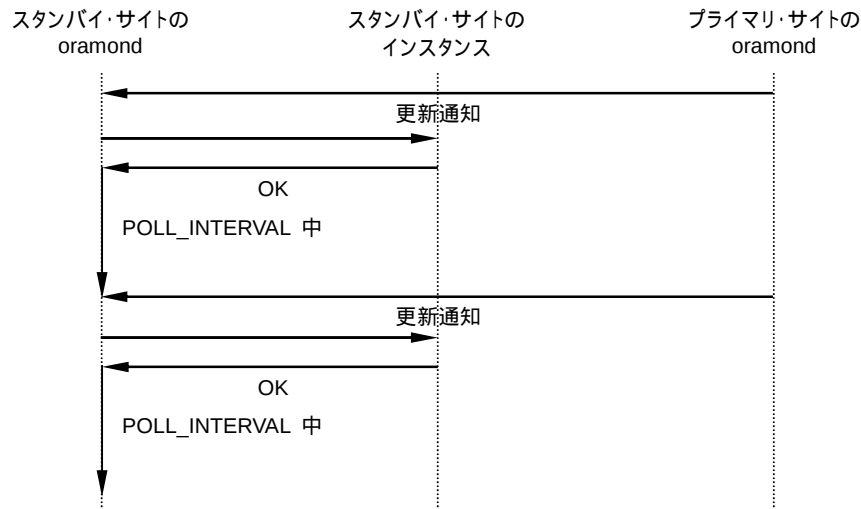


図 4-26 スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視時の動作(更新通知による監視)

スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視は、POLL_INTERVAL パラメータに設定した間隔で監視します。

また、プライマリ・サイトとスタンバイ・サイトの間でモニタ制御デーモン同士が通信を行っています。プライマリ・サイトの停止などのプライマリ・サイトの特定が必要な場合、スタンバイ・サイトのモニタ制御デーモンがプライマリ・サイトのモニタ制御デーモンから更新通知を受け取り、POLL_INTERVAL パラメータの待ち合わせ中であっても Oracle Data Guard Broker プロセスの状態を監視します。

なお、スタンバイ・サイトがフィジカル・スタンバイ・データベースの場合は、スタンバイ・サイトのインスタンスに対する起動確認のみ行います。

4 製品の動作

スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視でエラーを検出した場合は、以下のような動作になります。

(例)

```
StandbyDatabaseMonitor {  
  OdgBroker {  
    POLL_INTERVAL = 60  
    RETRY_COUNT = 1  
  }  
}
```

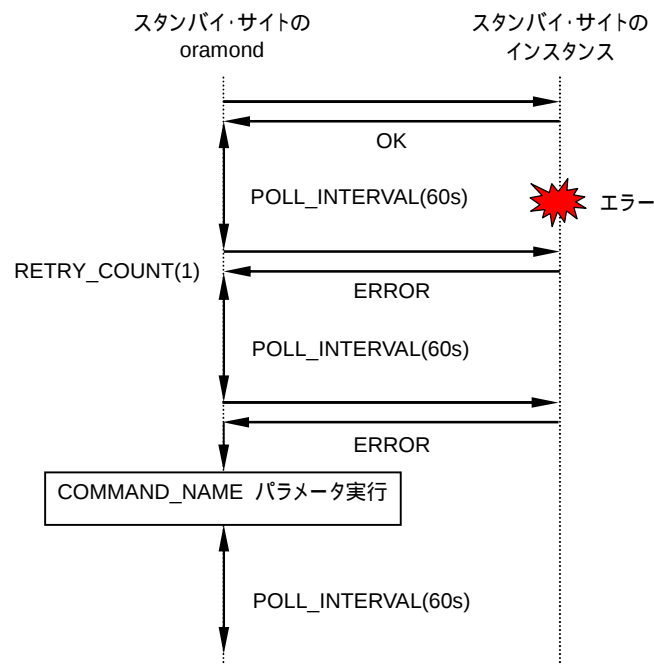


図 4-27 スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視エラー時の動作

スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視でエラーを検出した場合、RETRY_COUNT パラメータに 1 以上の値が設定されていると、RETRY_COUNT パラメータに設定した回数の Oracle Data Guard Broker プロセス状態を再確認します。再確認中に正常を検出した場合、障害と判断せず、監視を継続します。

RETRY_COUNT パラメータに設定した回数の再確認を行っても、監視結果がエラーになる場合は、COMMAND_NAME パラメータに設定したコマンドを実行し、監視を継続します。

4.7. 統計情報採取時の動作

統計情報採取モニタ (statsmond) は、起動後に POLL_INTERVAL パラメータに設定した間隔で統計情報を採取します。

統計情報の採取は、以下のような動作になります。

(例)

```
StatsMonitor STATSMOND {
  POLL_INTERVAL = 3600
  Stats SGA_SPACE {
    GET_STATS = YES
  }
}
```

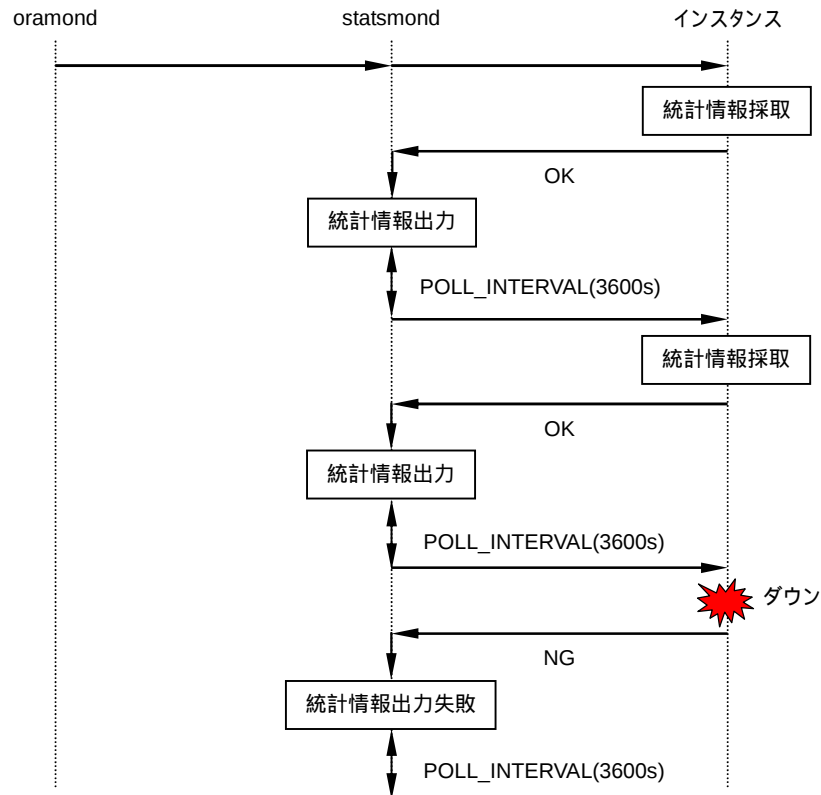


図 4-28 統計情報採取時の動作

統計情報採取モニタは、POLL_INTERVAL_パラメータに設定した間隔でインスタンスへ接続し、統計情報の採取および採取した統計情報をファイルへ出力します。

統計情報の採取または出力に失敗した場合、出力せずに統計情報の採取を継続します。

なお、データベースがマウント状態の際、任意の Oracle ユーザーによる採取を行う (ORACLE_USER / ORACLE_PASS パラメータが指定されている) 場合は、インスタンスに対する起動確認のみ行います。

4.8. システムステート・ダンプ採取時の動作

モニタ制御デーモン (oramond) は、Oracle の障害を検出した時、GET_DUMP パラメータを YES に設定したすべてのノードのシステムステートダンプ採取デーモン (ssdump) に採取要求を送ります。この要求によってシステムステートダンプ採取デーモンは、システムステート・ダンプを採取します。

シェルスクリプトの実行が指定されている場合は、システムステート・ダンプの採取と同時にを行います。

システムステート・ダンプの採取は、以下のような動作になります。

(例)

```
SystemStateDaemon {
  GET_DUMP      =YES
  DUMP_INTERVAL =30
  DUMP_COUNT    =3
  DUMP_TIMEOUT  =90
}
```

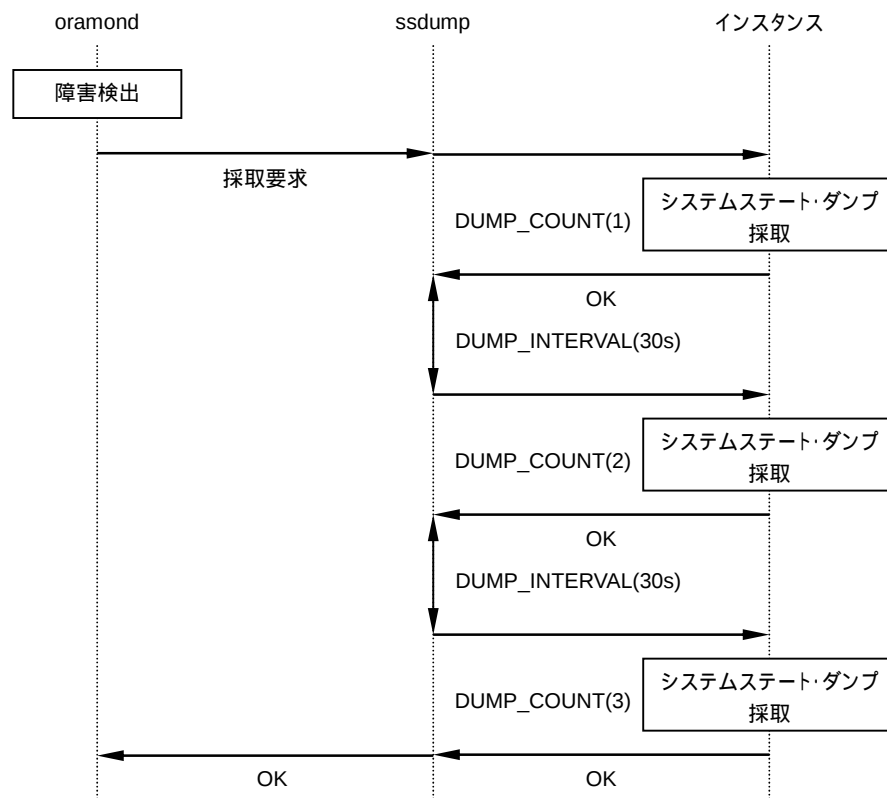


図 4-29 システムステート・ダンプ採取時の動作

システムステートダンプ採取デーモンは、DUMP_INTERVAL パラメータに設定した間隔でシステムステート・ダンプを採取します。システムステート・ダンプの採取は、DUMP_COUNT パラメータに

設定した回数行います。

モニタ制御デーモンは、DUMP_TIMEOUT パラメータに設定した時間までにシステムステート・ダンプの採取が完了しなかった場合、システムステート・ダンプの採取を停止します。

また、Oracle 12c のマルチテナント構成で、PDB 上のシステムステート・ダンプを採取する設定をした際、PDB への接続ができない中で障害を検出した場合は、CDB 上でシステムステート・ダンプを採取します。

4.9. シェルスクリプトの実行時の動作

モニタ制御デーモン (oramond) は、Oracle の障害を検出した時、SCRIPT_NAME パラメータに設定したシェルスクリプトをすべてのノードで実行します。

システムステート・ダンプの採取が指定されている場合は、シェルスクリプトの実行と同時にを行います。

モニタ制御デーモンは、SCRIPT_TIMEOUT パラメータに設定した時間までにシェルスクリプトの処理が完了しなかった場合、シェルスクリプトを停止します。

4.10. Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの動作

Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプト (crs_control.sh) は、インスタンス監視で障害を検出した場合、またはリスナー監視で障害を検出した場合に、Oracle Clusterware / Oracle Restart によるインスタンス / リスナーの再起動が可能かどうかを判断し、再起動を促すシェルスクリプトです。

インスタンス監視時の Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトは、以下のような動作になります。

(例)

```
GRID_USER          = grid
CRS_CMD_TIMEOUT    = 90

InstanceMonitor {
  RESTART_COUNT    = 1
}
```

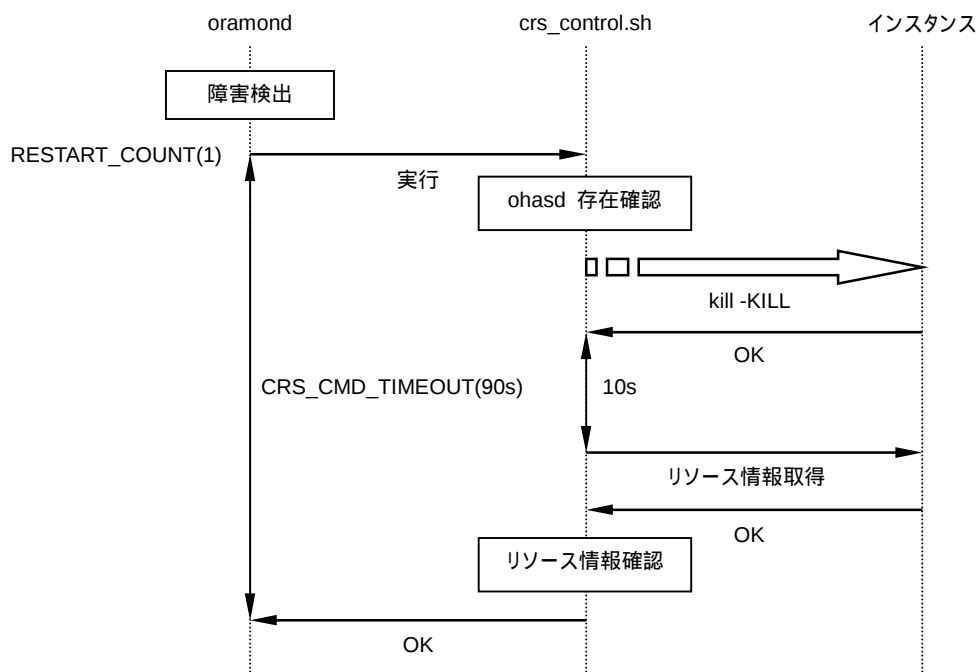


図 4-30 インスタンス監視時の Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトの動作

インスタンス監視で障害を検出した場合、RESTART_COUNT パラメータに 1 以上の値が設定されていると、モニタ制御デーモン (oramond) が Oracle Clusterware / Oracle Restart リトライ

判定シェルスクリプトを実行します。

Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトは、ps コマンドを実行し、ohasd の存在確認を行います。

ohasd の存在確認後、SMON プロセスに対して KILL コマンドを実行し、Oracle Clusterware / Oracle Restart によるインスタンスの再起動(1)を促します。

10 秒待機後、GRID_USER パラメータまたは MONITOR_USER パラメータ(2)に設定したユーザーで crsctl コマンドを実行し、リソース情報を取得します。

リソースの TARGET、RESTART_ATTEMPTS、RESTART_COUNT、STATE および INTERNAL_STATE リソース属性のそれぞれの値を確認します。各リソース属性の値が期待値だった場合、Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトは、正常終了します。

なお、ohasd が存在しない、SMON プロセスの KILL に失敗、リソース情報が取得できない、または各リソース属性の値が期待値ではない場合、Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトは、異常終了します。

また、モニタ制御デモンは、CRS_CMD_TIMEOUT パラメータに設定した時間までに Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトが終了しなかった場合、異常終了と判断します。

- (1) Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境でのみ行われます。
- (2) 双方のパラメータが設定されている場合は、GRID_USER パラメータに設定されているユーザーで実行します。MONITOR_USER パラメータのみ設定されている場合は、全ノードステートメントまたは Node ステートメントに設定されているユーザーで実行します。

リスナー監視時の Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトは、以下のような動作になります。

(例)

```
CRS_CMD_TIMEOUT = 90
```

```
ListenerMonitor LISTENER {
  MONITOR_USER      = oracle
  RESTART_COUNT     = 1
  NET_SERVICE_NAME = LSNR
  TARGET_RESTART    = NO
}
```

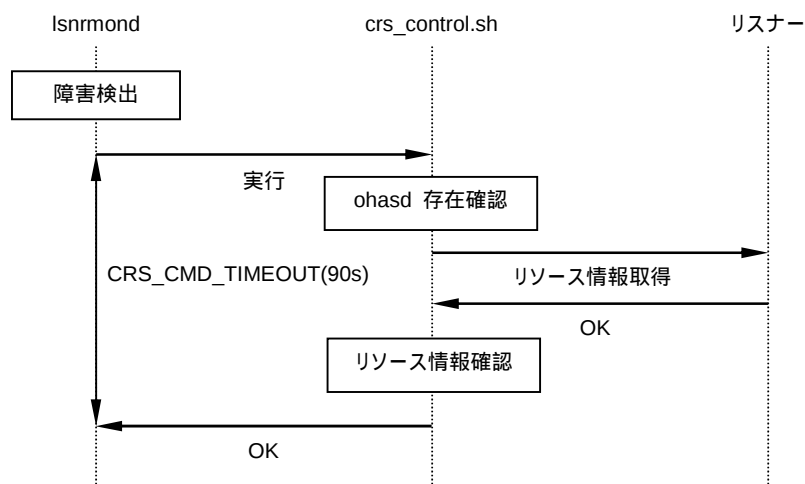


図 4-31 リスナー監視時の Oracle Clusterware / Oracle Restart
リトライ判定シェルスクリプトの動作

リスナー監視で障害を検出した場合、RESTART_COUNT パラメータに 1 以上、TARGET_RESTART パラメータの値が NO に設定されていると、リスナー監視モニタ (lsnrmond) が Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトを実行します。

Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトは、ps コマンドを実行し、ohasd の存在確認を行います。

ohasd の存在確認後、MONITOR_USER パラメータに設定したユーザーで crsctl コマンドを実行し、リソース情報を取得します。

リソースの TARGET、RESTART_ATTEMPTS、RESTART_COUNT、STATE および INTERNAL_STATE リソース属性のそれぞれの値を確認します。各リソース属性の値が期待値だった場合、Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトは、正常終了します。

なお、ohasd が存在しない、リソース情報が取得できない、または各リソース属性の値が期待値

4 製品の動作

ではない場合、Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトは、異常終了します。

また、リスナー監視モニタは、CRS_CMD_TIMEOUT パラメータに設定した時間までに Oracle Clusterware / Oracle Restart リトライ判定シェルスクリプトが終了しなかった場合、異常終了と判断します。

- (1) Oracle 11g R2 以降の Oracle Clusterware / Oracle Restart 環境でのみ行われます。

4.11. Oracle Data Guard Broker による自動フェイルオーバー時の監視動作

ApplicationMonitor では、データベース・インスタンスのフェイルオーバー実施状況を監視することにより、Oracle Data Guard Broker による自動フェイルオーバーを検出し、自動フェイルオーバー処理を待ち合わせることができます (1)。

- (1) Oracle Data Guard で手動フェイルオーバーおよびスイッチオーバーを実行した場合は、フェイルオーバーを検出できません。

Oracle Data Guard Broker による自動フェイルオーバーが行われた場合、以下のような動作になります。

(例)

SWITCHOVER_TIMEOUT = 30

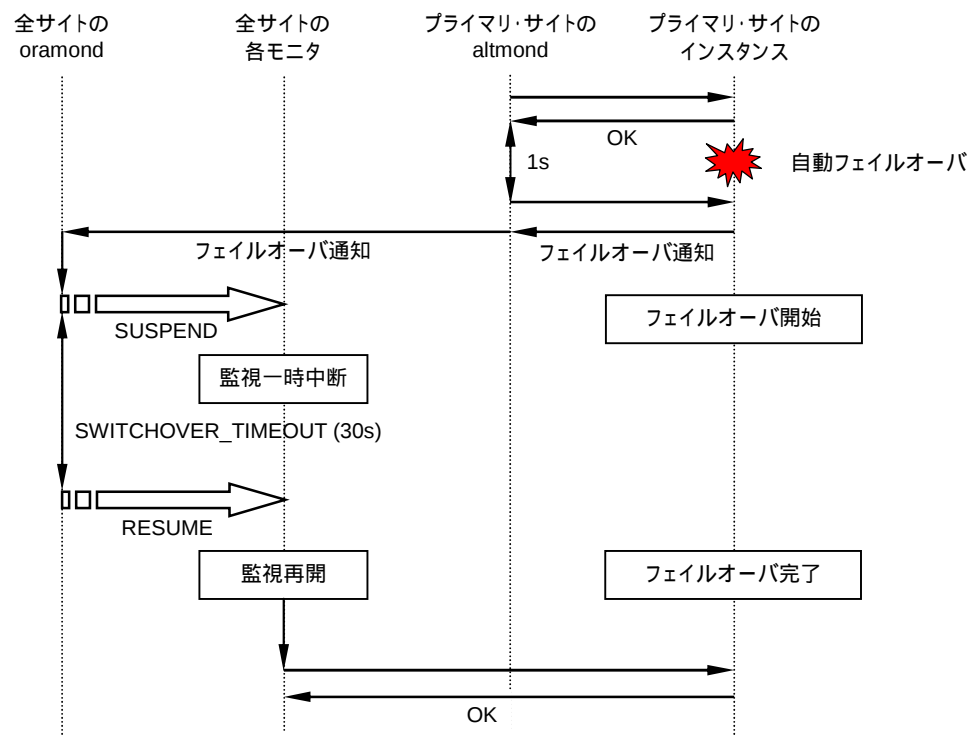


図 4-32 Oracle Data Guard Broker による自動フェイルオーバー時の監視動作

Oracle Data Guard Broker によって自動フェイルオーバーが実行された場合、アラートログ監視モニタ (altmond) が自動フェイルオーバーを検出し、モニタ制御デーモン (oramond) に通知します。

通知を受け取ったモニタ制御デーモンは、各監視モニタに対して SUSPEND 要求 (一時停止要求) を送ります。この要求により、各監視モニタは監視を一時停止します。

4 製品の動作

各モニタの一時停止完了後、自動フェイルオーバーの完了を待ち合わせます。
SWITCHOVER_TIMEOUT パラメータに設定した時間までに自動フェイルオーバーが完了しなかった場合、モニタ制御デモンは、各監視モニタに対して RESUME 要求(監視再開要求)を送ります。この要求により、各監視モニタは監視を再開します。

4.12. ASM インスタンス停止に伴うデータベース再起動時の監視動作

ASM インスタンスの停止および再起動が行われると、データベース・インスタンスも再起動されます。ApplicationMonitor では、ASM インスタンスとデータベース・インスタンスの通信状態を監視することにより、ASM インスタンス停止に影響されることなく、監視をすることができます。

ASM インスタンス停止に伴うデータベースの再起動が行われた場合、以下のような動作になります。

(例)

WAIT_ASM_RESTART = YES

```
InstanceMonitor {
  RESTART_TIMEOUT = 60
}
```

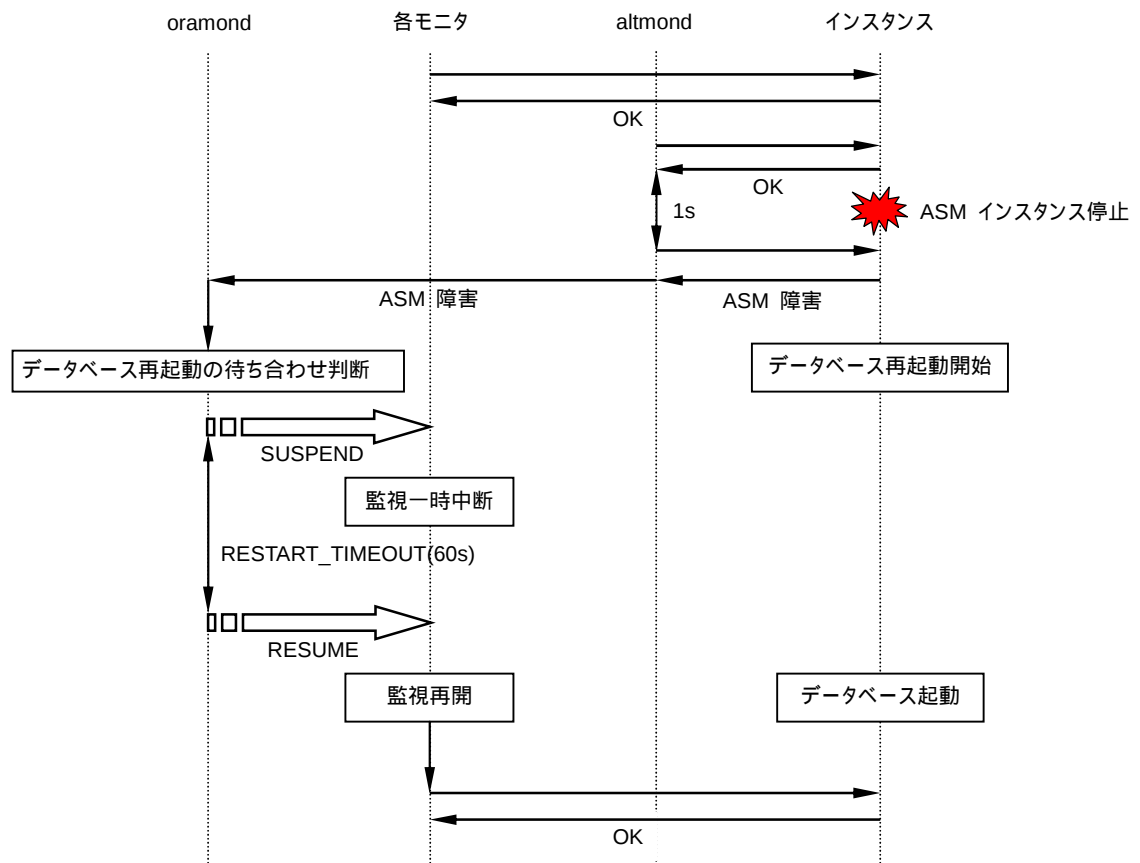


図 4-33 ASM インスタンス停止に伴うデータベースの再起動時の監視動作

ASM インスタンスが停止した場合、アラートログ監視モニタ (altmond) が ASM 障害を検出し、モニタ制御デーモン (oramond) に通知します。

通知を受け取ったモニタ制御デーモンは、WAIT_ASM_RESTART パラメータの設定により、

データベースの再起動を待ち合わせるか判断します。WAIT_ASM_RESTART パラメータに YES が設定されていると、データベースの再起動を待ち合わせます。

データベースの再起動を待ち合わせると判断した場合、モニタ制御デーモンは、各監視モニタに対して SUSPEND 要求(一時停止要求)を送ります。この要求により、各監視モニタは監視を一時停止します。

各モニタの一時停止完了後、データベース再起動の完了を待ち合わせます。InstanceMonitor ステートメントの RESTART_TIMEOUT パラメータに設定した時間までにデータベースの再起動が完了しなかった場合、モニタ制御デーモンは、各監視モニタに対して RESUME 要求(監視再開要求)を送ります。この要求により、各監視モニタは監視を再開します。

なお、WAIT_ASM_RESTART パラメータに NO が設定されていると、データベースの再起動を待ち合わせないと判断するため、監視を継続します。

4.13. Oracle Clusterware / Oracle Restart 監視時の動作

CRS 監視モニタ (crsmond) は、常に単独起動モニタとして動作します。

CRS 監視モニタは、起動後に CRS 監視設定ファイル (crsmond.conf) の POLL_INTERVAL パラメータに設定した間隔で CRS_STAT パラメータに設定したコマンドを用いて、crsd/ohasd の状態を監視します。

Oracle Clusterware / Oracle Restart の監視は、以下のような動作になります。

(例)

```
ORACLE_VERSION = 12C
CRS_STAT       = "/u01/app/12.1.0/grid/bin/crsctl check crs"
POLL_INTERVAL  = 90
```

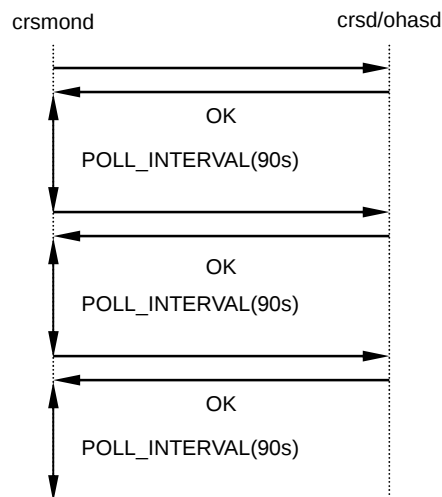


図 4-34 Oracle Clusterware / Oracle Restart 監視時の動作

CRS 監視モニタが起動時にエラーを検出した場合、以下のような動作になります。

(例)

```
ORACLE_VERSION = 12C
CRS_STAT        = "/u01/app/12.1.0/grid/bin/crsctl check crs"
POLL_INTERVAL   = 90
RETRY_COUNT      = 2
WAIT_INTERVAL   = 180
SERVICE_DOWN   = NO
```

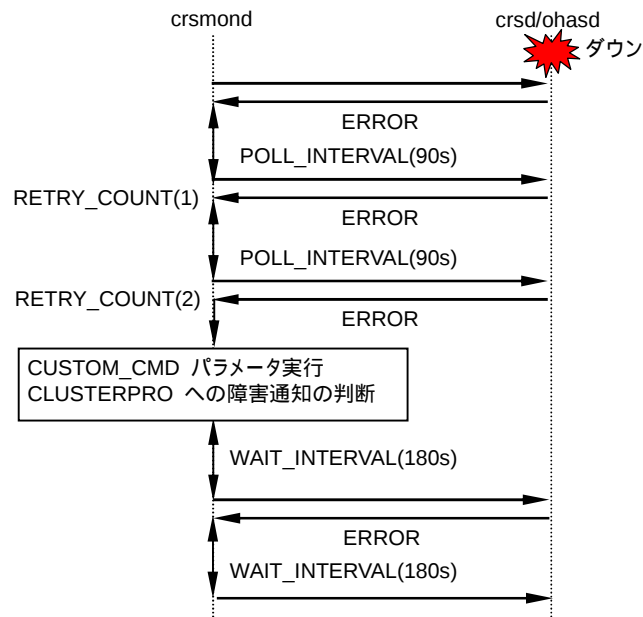


図 4-35 CRS 監視モニタ起動エラー時の動作

CRS 監視モニタが起動時にエラーを検出した場合、RETRY_COUNT パラメータに 1 以上の値が設定されていると、RETRY_COUNT パラメータに設定されている回数の crsd/ohasd の状態を再確認します。再確認中に起動を確認した場合、障害と判断せず、監視を継続します。

RETRY_COUNT パラメータに設定した回数の crsd/ohasd の再確認を行っても起動を確認できなかった場合は、CUSTOM_CMD パラメータの設定により、コマンドの実行および SERVICE_DOWN パラメータの設定により、CLUSTERPRO へノード内の障害を通知するか判断します。起動がエラーとなる際、SERVICE_DOWN パラメータに NO または NO_DOWN が設定されていると、CRS 監視モニタは、CLUSTERPRO へ障害を通知しないと判断するため、監視を継続します。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに CUSTOM_CMD パラメータの設定したコマンドの実行、CLUSTERPRO への障害通知の判断を行います。

また、起動がエラーとなる際、SERVICE_DOWN パラメータに YES または NO_STALL が設定されていると、CLUSTERPRO へ障害を通知すると判断し、ApplicationMonitor を停止します。

CLUSTERPRO へ障害を通知しないと判断した場合、CRS 監視モニタは、WAIT_INTERVAL パラメータに設定した間隔で crsd/ohasd の状態確認を継続します。この動作は、crsd/ohasd 起動の確認または明示的に CRS 監視モニタを停止するまで繰り返します。

4 製品の動作

crsd/ohasd の監視でエラーを検出し、init(8) によって crsd/ohasd が再起動された場合、以下のような動作になります。

(例)

```
ORACLE_VERSION = 12C
CRS_STAT        = "/u01/app/12.1.0/grid/bin/crsctl check crs"
POLL_INTERVAL   = 90
RETRY_COUNT      = 2
WAIT_INTERVAL    = 180
```

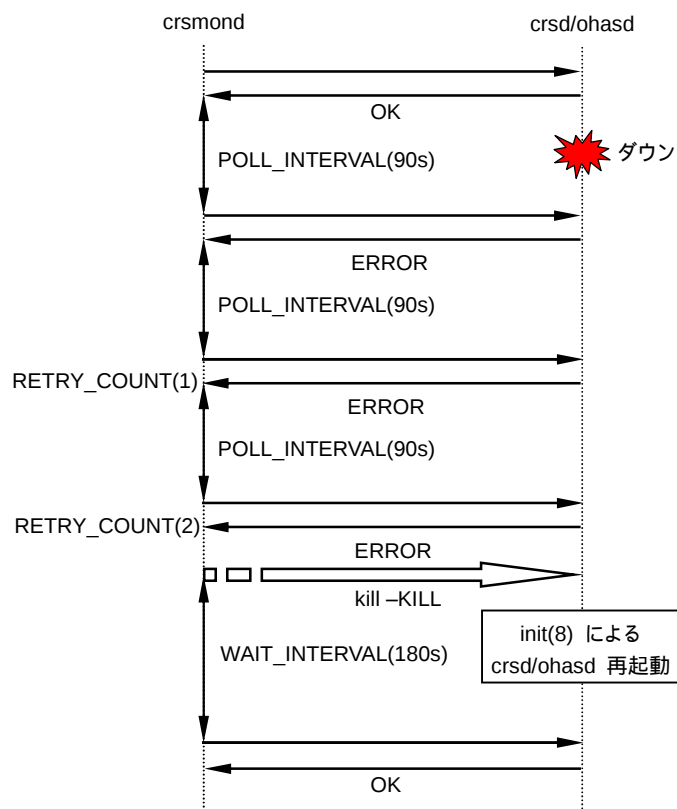


図 4-36 crsd/ohasd 監視エラー時の動作 (init(8) によって crsd/ohasd が再起動された場合)

crsd/ohasd の監視でエラーを検出した場合、RETRY_COUNT パラメータに 1 以上の値が設定されていると、RETRY_COUNT パラメータに設定した回数の crsd/ohasd の状態を再確認します。再確認中に crsd/ohasd の監視で正常を検出した場合、障害と判断せず、監視を継続します。

RETRY_COUNT パラメータに設定した回数の crsd/ohasd の再確認を行っても、crsd/ohasd の監視結果がエラーになる場合は、crsd/ohasd プロセスに対して KILL コマンドを実行し、init(8) による crsd/ohasd 再起動を促します。

KILL コマンド実行後、WAIT_INTERVAL パラメータに設定した時間の待ち合わせを行い、監視を再開します。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに crsd/ohasd プロセスに対して KILL コマンドを実行します。

4 製品の動作

crsd/ohasd の監視でエラーを検出し、init(8) によって crsd/ohasd が再起動されなかった場合、以下のような動作になります。

(例)

```
ORACLE_VERSION = 12C
CRS_STAT        = "/u01/app/12.1.0/grid/bin/crsctl check crs"
POLL_INTERVAL   = 90
RETRY_COUNT      = 2
WAIT_INTERVAL    = 180
SERVICE_DOWN    = YES
```

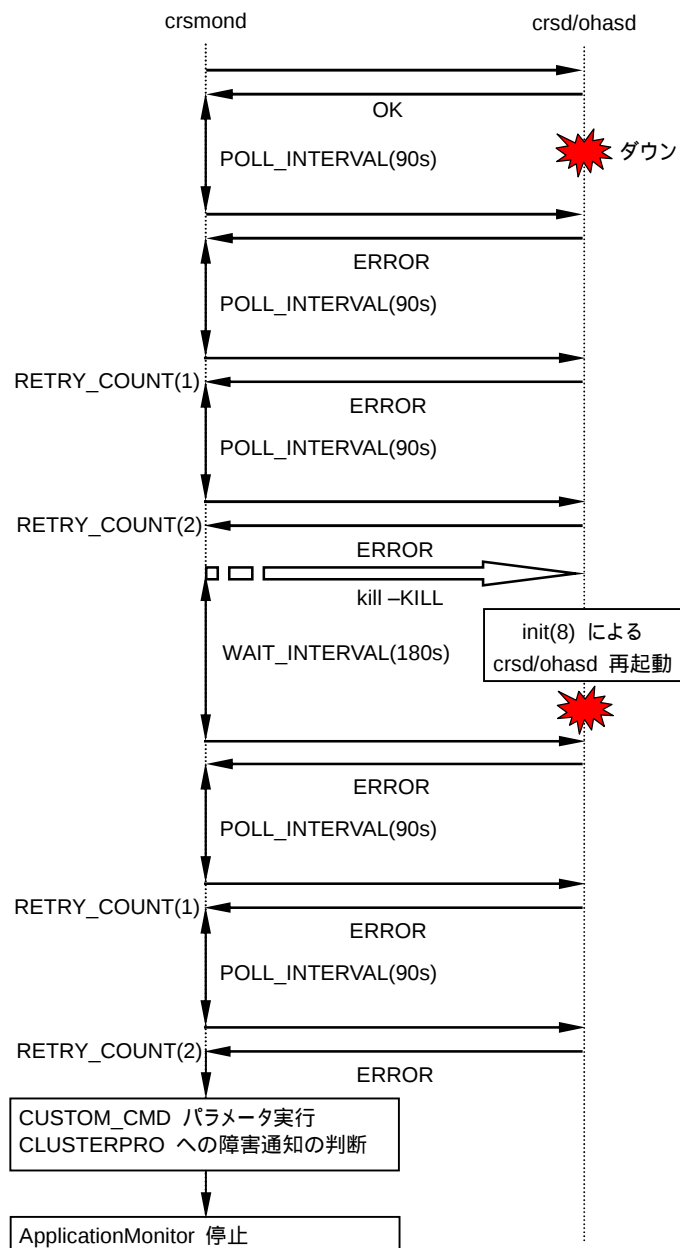


図 4-37 crsd/ohasd 監視エラー時の動作(init(8) によって crsd/ohasd が再起動されなかった場合)

KILL コマンド実行後、WAIT_INTERVAL パラメータに設定した時間の待ち合わせを行っても、crsd/ohasd の監視結果がエラーとなる場合は、RETRY_COUNT パラメータに設定した回数の crsd/ohasd の確認を再度行います。

RETRY_COUNT パラメータに設定した回数の crsd/ohasd の再確認を行っても crsd/ohasd の監視結果がエラーになる場合は、CUSTOM_CMD パラメータの設定により、コマンドの実行および SERVICE_DOWN パラメータの設定により、CLUSTERPRO ノード内の障害を通知するか判断します。crsd/ohasd の監視結果がエラーとなる際、SERVICE_DOWN パラメータに YES または NO_STALL が設定されていると、CLUSTERPRO ノードへ障害を通知すると判断し、ApplicationMonitor を停止します。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに CUSTOM_CMD パラメータの設定したコマンドの実行、CLUSTERPRO ノードへの障害通知の判断を行います。

また、crsd/ohasd の監視結果がエラーとなる際、SERVICE_DOWN パラメータに NO または NO_DOWN が設定されていると、CRS 監視モニタは、CLUSTERPRO ノードへ障害を通知しないと判断するため、監視を継続します。

4 製品の動作

crsd/ohasd の監視で POLL_TIMEOUT パラメータに設定した時間内に完了しない場合、以下のような動作になります。

(例)

```
ORACLE_VERSION = 12C
CRS_STAT        = "/u01/app/12.1.0/grid/bin/crsctl check crs"
POLL_INTERVAL   = 90
POLL_TIMEOUT    = 180
RETRY_COUNT     = 1
WAIT_INTERVAL   = 180
SERVICE_DOWN   = YES
```

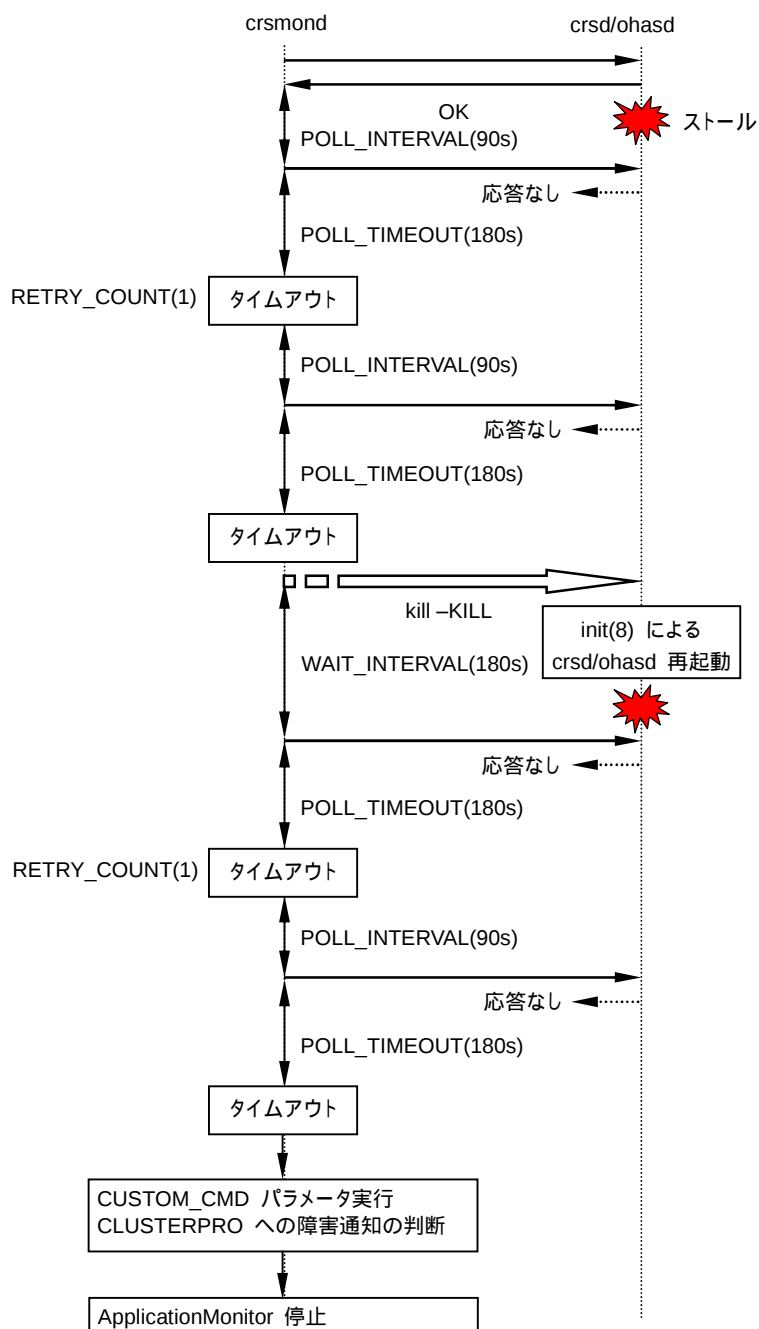


図 4-38 crsd/ohasd 監視タイムアウト時の動作

crsd/ohasd の監視が POLL_TIMEOUT パラメータに設定した時間内に完了しなかった場合、RETRY_COUNT パラメータに 1 以上の値が設定されていると、RETRY_COUNT パラメータに設定した回数の crsd/ohasd の状態を再確認します。再確認中に crsd/ohasd の監視で正常を検出した場合、障害と判断せず、監視を継続します。

RETRY_COUNT パラメータに設定した回数の crsd/ohasd の再確認を行っても、crsd/ohasd の監視結果がストールとなる(タイムアウトする)場合は、crsd/ohasd プロセスに対して KILL コマンドを実行し、init(8) による crsd/ohasd 再起動を促します。

KILL コマンド実行後、WAIT_INTERVAL パラメータに設定した時間の待ち合わせを行っても、crsd/ohasd の監視結果がストールとなる場合は、RETRY_COUNT パラメータに設定した回数の crsd/ohasd の確認を再度行います。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに crsd/ohasd プロセスに対して KILL コマンドを実行します。

RETRY_COUNT パラメータに設定した回数の crsd/ohasd の再確認を行っても crsd/ohasd の監視結果がストールになる場合は、CUSTOM_CMD パラメータの設定により、コマンドの実行および SERVICE_DOWN パラメータの設定により、CLUSTERPRO ノード内の障害を通知するか判断します。crsd/ohasd の監視結果がストールとなる際、SERVICE_DOWN パラメータに YES または NO_DOWN が設定されていると、CLUSTERPRO へ障害を通知すると判断し、ApplicationMonitor を停止します。

なお、crsd/ohasd の監視結果がストールとなる際、SERVICE_DOWN パラメータに NO または NO_STALL が設定されていると、CRS 監視モニタは、CLUSTERPRO へ障害を通知しないと判断するため、監視を継続します。

5. 製品の運用

ApplicationMonitor を運用するための手順を説明します。

本書では、以下の構成を説明します。

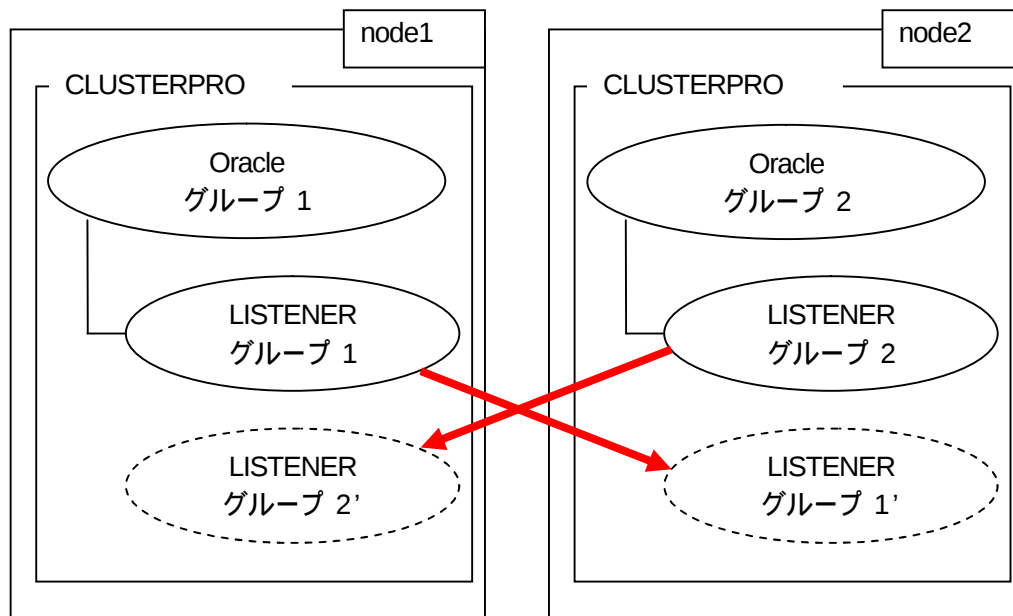


図 5-1 構成例

CLUSTERPRO 上に構成されている 2 ノード RAC 構成の環境です。Oracle Clusterware / Oracle Restart 起動の延長で Oracle インスタンスを起動させるフェイルオーバーグループ(以後 Oracle グループと記載します)に対して、ApplicationMonitor の監視機能を追加します。

また、Oracle グループとは別にリロケータブル IP を使用するリスナーを起動させるフェイルオーバーグループ(以後 LISTENER グループと記載します)を作成しています。

5.1. 事前準備

以下の事前準備が必要です。

- ◆ CLUSTERPRO でクラスタ化が正しく行われている。
- ◆ CLUSTERPRO で Oracle グループが作成されている (1、 2)。
- ◆ CLUSTERPRO で LISTENER グループが作成されている。
- ◆ CLUSTERPRO で上記のフェイルオーバーグループ内のすべてのリソースが起動する。
- ◆ RENS 連携をする場合、RENS モニタプロセス (hamon) の起動設定が行われている (3)。

- (1) デフォルト設定では、Oracle Clusterware / Oracle Restart が OS 起動の延長で自動起動されます。自動起動されないように設定する場合は、root ユーザーで以下を行います。

Oracle 11g R1 の RAC 構成の環境

```
# <$ORA_CRS_HOME> /bin/crsctl disable crs
```

Oracle 11g R2 以降の環境

```
# <grid ホーム> /bin/crsctl disable has
```

- (2) Oracle Clusterware / Oracle Restart 起動の延長で Oracle インスタンスが自動起動されないように設定する場合は、Oracle ユーザーで Oracle Clusterware / Oracle Restart が管理するリソースの AUTO_START パラメータを "never" に変更してください。

```
% srvctl modify database -d <database_name> -y manual
```

なお、上記 manual を automatic に変更するとデフォルト値に戻ります。

注意: 自動起動されないように設定した場合は、Oracle グループとは別に Oracle インスタンスを起動するフェイルオーバーグループを作成する必要があります。

- (3) RENS 連携できるモニタは、モニタ制御デーモン (oramond)、単独起動のリスナー監視モニタ (lsnrmond) (4)および CRS 監視モニタ (crsmond) です。

- (4) 単独起動のリスナー監視モニタと RENS が連携する場合、プロセスを実行するユーザーは、root または rens グループに含まれるユーザーである必要があります。

リスナー監視モニタを root 以外のユーザーで実行する場合または設定ファイル (oramond.conf) の MONITOR_USER パラメータに root 以外のユーザーを指定する場合は、該当ユーザーを rens グループにも所属させてください。

```
# usermod -a -G rens <ユーザー名>
```

CLUSTERPRO および RENS の設定などの詳細は、CLUSTERPRO のマニュアルおよびサーバ管理基盤のマニュアルを参照してください。

5.1.1. Oracle ハング検知機能の停止(RAC 構成のみ)

Oracle では、Oracle ハング検知機能がデフォルトで有効になっています。

Oracle 11g (11.1.0.7) の Oracle ハング検知機能には、不具合が報告されているため、機能を停止します。また、ApplicationMonitor を RAC 構成の環境で監視する場合は、Oracle のバージョンにかかわらず、Oracle ハング検知機能を停止することを推奨します。

Oracle ハング検知機能を停止するには、すべてのノードで Oracle 初期化パラメータファイルに以下のパラメータを設定し、インスタンスを再起動してください。

SPFILE を使用している場合は、以下を実行してください。

```
$ sqlplus "/ as sysdba"
SQL> alter system set "_lm_rcvr_hang_kill"=false scope=spfile;
```

PFILE を使用している場合は、PFILE に以下を追加してください。

```
_lm_rcvr_hang_kill=false
```

以上で、Oracle ハング検知機能の停止設定は終了です。

5.1.2. Oracle Clusterware / Oracle Restart の監視対象設定

Oracle Clusterware / Oracle Restart は、デフォルト設定の場合、Oracle データベース、インスタンス、ノードアプリケーションなどを監視し、監視対象のプロセス停止障害発生時に自動で再起動を行います。

Oracle Clusterware / Oracle Restart を監視する場合は、以下の構成を推奨します。

◆ Oracle 11g R1 の RAC 構成の環境

- ApplicationMonitor によるリスナー監視 (Oracle vip を使用している vip 用のリスナーは除く)
- ApplicationMonitor によるインスタンス監視
- Oracle Clusterware によるリソース監視 (ApplicationMonitor で監視しているリスナーは除く)

この構成は、インスタンス監視のみ Oracle Clusterware と ApplicationMonitor 両方からの監視となります。

Oracle Clusterware / Oracle Restart がデフォルト設定の場合、障害検出時にインスタンスの再起動を実施しますが、その際に ApplicationMonitor が障害と判断する可能性があります。そのため、Oracle Clusterware によるインスタンスの再起動を抑止する設定を行う、もしくは ApplicationMonitor のパラメータを調整し、ApplicationMonitor が障害と判断しないように設定してください。

設定例は、以下のとおりです。

- **Oracle Clusterware による監視対象の再起動を行わない場合 (推奨)**

Oracle Clusterware の RESTART_ATTEMPTS パラメータを以下の手順で変更します。

- 1 **oracle** ユーザーで、すべてのインスタンスとノードアプリケーションを停止します。

```
% srvctl stop instance -d <dbname> -i <instname>
% srvctl stop nodeapps -n <hostname>
```

- 2 **root** ユーザーで、再起動を行わない設定にする対象リソース (Oracle Clusterware が高可用性のために管理するエンティティ) に関する情報を Oracle Cluster Registry からテキストに出力します。

```
# cd $ORA_CRS_HOME/bin  
# ./crs_stat -p <対象リソース名> > $ORA_CRS_HOME/crs/profile/<対象リソース名>.cap
```

- 3 上記で作成された .cap ファイルの RESTART_ATTEMPTS パラメータの値を 0 に修正します。

```
# ./crs_profile -update <対象リソース名> -o ra=0
```

- 4 Oracle Cluster Registry に変更内容を登録します。

```
# ./crs_register <対象リソース名> -u
```

- 5 RESTART_ATTEMPTS パラメータの値が 0 に変更されたことを確認します。

```
# ./crs_stat -p <対象リソース名> | grep RESTART_ATTEMPTS
```

- 6 **oracle** ユーザーに戻り、インスタンス、ノードアプリケーションおよびその他の必要なリソースを起動します。

```
% srvctl start nodeapps -n <hostname>  
% srvctl start instance -d <dbname> -i <instname>
```

・ ApplicationMonitor のパラメータを変更する場合

設定ファイル (oramond.conf) の InstanceMonitor ステートメント内の POLL_DOWN_RETRY パラメータおよび POLL_STALL_RETRY パラメータの設定値を、インスタンスの起動からデータベースがオープンになるまでの間、障害と判断しないように調整してください。

パラメータの詳細は、「3.2.7 パラメーター一覧」を参照してください。

◆ Oracle 11g R2 以降の環境

- ApplicationMonitor によるリスナー監視 (Oracle scan-vip を使用している scan-vip 用リスナーは除く)
- ApplicationMonitor によるインスタンス監視
- Oracle Clusterware / Oracle Restart によるリソース監視

この構成は、Oracle Clusterware / Oracle Restart の状況に合わせて動作するため、設定は必要ありません。

Oracle Clusterware / Oracle Restart の詳細は、Oracle のマニュアル『Oracle Clusterware 管理およびデプロイメント・ガイド』を参照してください。

以上で、Oracle Clusterware / Oracle Restart の監視対象設定は終了です。

5.2. ApplicationMonitor の設定

Oracle の設定を参照し、サンプルファイルをもとに ApplicationMonitor の設定ファイル (oramond.conf) を作成します。

```
# cd /etc/opt/HA/AM/conf
# cp sample/oramond.conf.sample ./oramond.conf
# vi oramond.conf
```

設定ファイルテンプレートから設定ファイルを作成する場合は、"sample/oramond.conf.sample" の部分を "oramond.tmp" に読み替えてください。

ApplicationMonitor で Oracle の監視を行うためには、以下のステートメントおよびパラメータの設定が最低限必要です。

◆ 全ノードステートメント

- ・ **NODE_NAME** パラメータ
Oracle の設定を参照し、監視対象のノード名をそれぞれ指定してください。
- ・ **ORACLE_BASE, ORACLE_HOME, ORA_NLS, SHLIB_PATH** パラメータ
Oracle の設定を参照し、各パラメータを指定してください。
誤ったパスを指定した場合、監視が正常に行われません。
- ・ **MONTYPE** パラメータ
監視対象の Oracle のバージョンに合わせて指定してください。
誤った値を指定した場合、監視が正常に行われません。
- ・ **MONITOR_USER** パラメータ
Oracle の OSDBA(dba) グループに所属するユーザー名を指定してください。

◆ DataGuardSite ステートメント

スタンバイ・インスタンスの監視機能を使用する場合、監視対象のサイト数分指定してください。

指定した DataGuardSite ステートメント指示行に任意のサイト間通信 ID を指定してください。

- ・ DB_UNIQUE_NAME パラメータ

スタンバイ・インスタンスの監視機能を使用する場合、Oracle の設定を参照し、監視対象サイトのデータベース・インスタンス名 (初期化パラメータ DB_UNIQUE_NAME の値) を指定してください。

誤った値を指定した場合、監視が正常に行われません。

- ・ SITE_NODE パラメータ

スタンバイ・インスタンスの監視機能を使用する場合、Oracle の設定を参照し、監視対象サイト内のノード名をノード数分指定してください。

誤った値を指定した場合、監視が正常に行われません。

- ◆ Node ステートメント

監視対象のノード数分指定してください。

Node ステートメント指示行に全ノードステートメントの NODE_NAME パラメータに設定したノード名と同じ値を指定してください。

- ・ ORACLE_SID パラメータ

各 Node ステートメント内に Oracle インスタンスの識別子を指定してください。

誤った値を指定した場合、監視が正常に行われません。

- ◆ InstanceMonitor ステートメント

- ・ OWNER_NAME パラメータ

Oracle 12c を監視対象とする場合、インスタンス監視表のスキーマを指定してください。

誤った値を指定した場合、監視が正常に行われません。

- ◆ ListenerMonitor ステートメント

リスナー障害の監視機能を使用する場合、Node ステートメントごとに存在する ListenerMonitor ステートメント指示行に Oracle の設定を参照して、監視対象のリスナー名を指定してください。

- ・ TNSPING パラメータ

リスナー障害の監視機能を使用する場合、監視対象の Oracle がインストールされているディレクトリを参照し、Oracle の tnsping コマンドのパスを指定してください。

誤ったパスを指定した場合、監視が正常に行われません。

- ・ NET_SERVICE_NAME パラメータ

リスナー障害の監視機能を使用する場合、Node ステートメントごとに存在する監視対象のリスナーを設定した ListenerMonitor ステートメント内に監視対象リスナーのネット・サービス名を指定してください。

誤った値を指定した場合、監視が正常に行われません。

- ◆ TableSpaceMonitor ステートメント

データベース容量不足の監視機能を使用する場合、Node ステートメントごとに TableSpaceMonitor ステートメントを指定してください。

指定した TableSpaceMonitor ステートメント指示行に 任意の表領域監視モニタ ID を指定してください。

- ◆ TableSpace ステートメント

データベース容量不足の監視機能を使用する場合、表領域監視モニタ ID を設定した TableSpaceMonitor ステートメント内に TableSpace ステートメントを指定してください。

指定した TableSpace ステートメント指示行に Oracle の設定を参照して、監視対象の表領域名を指定してください。

- ◆ DiskSpaceMonitor ステートメント

ローカルディスク容量不足の監視機能を使用する場合、Node ステートメントごとに DiskSpaceMonitor ステートメントを指定してください。

- ◆ StandbyDatabaseMonitor ステートメント

スタンバイ・インスタンスの監視機能を使用する場合、Node ステートメントごとに StandbyDatabaseMonitor ステートメントを指定してください。

- ◆ Transport ステートメント

スタンバイ・インスタンスの監視機能を使用する場合、StandbyDatabaseMonitor ステートメント内に Transport ステートメントを指定してください。

- ◆ Destination ステートメント

スタンバイ・インスタンスの監視機能を使用する場合、Transport ステートメント内に Destination ステートメントを指定してください。

指定した Destination ステートメント指示行に DataGuardSite ステートメントで指定したサイト間通信 ID を指定してください。

誤った値を指定した場合、監視が正常に行われません。

- ・ PRIMARY_NET_SERVICE_NAME パラメータ

スタンバイ・インスタンスの監視機能を使用する場合、監視対象サイトのノードごとに存在する、スタンバイ・サイトからプライマリ・サイトヘリスナー経由で接続するためのネット・サービス名を指定してください。

DataGuardSite ステートメントの SITE_NODE パラメータに設定したノード名と同じ順番で指定してください。

誤った値または誤った順番を指定した場合、監視が正常に行われません。

- ◆ RedoApply ステートメント

スタンバイ・インスタンスの監視機能を使用する場合、StandbyDatabaseMonitor ステートメント内に RedoApply ステートメントを指定してください。

- ◆ OdgBroker ステートメント

スタンバイ・インスタンスの監視機能を使用する場合、StandbyDatabaseMonitor ステートメント内に OdgBroker ステートメントを指定してください。

- ◆ StatsMonitor ステートメント

統計情報の採取機能を使用する場合、Node ステートメントごとに StatsMonitor ステートメントを指定してください。

指定した StatsMonitor ステートメント指示行に任意の統計情報採取モニタ ID を指定してください。

- ◆ Stats ステートメント

統計情報の採取機能を使用する場合、統計情報採取モニタ ID を設定した StatsMonitor ステートメント内に Stats ステートメントを指定してください。

指定した Stats ステートメント指示行に監視対象の統計情報種別を指定してください。

「3.2 設定ファイル (oramond.conf) の作成と適用」を参照し、必要に応じて各ステートメントおよび各パラメータを設定してください。

Oracle Clusterware / Oracle Restart の監視をする場合は、Oracle の設定を参照し、サンプルファイルをもとに CRS 監視設定ファイル (crsmond.conf) を作成します。

```
# cd /etc/opt/HA/AM/conf
# cp sample/crsmond.conf.sample ./crsmond.conf
# vi crsmond.conf
```

ApplicationMonitor で Oracle Clusterware / Oracle Restart の監視を行うためには、以下のパラメータの設定が最低限必要です。

- ・ ORACLE_VERSION パラメータ
監視対象の Oracle のバージョンに合わせて指定してください。
誤った値を指定した場合、監視が正常に行われません。
- ・ CRS_STAT パラメータ
構成および監視対象の Oracle のバージョンに合わせて、crsd/ohasd の状態を表示するコマンドの絶対パスを指定してください。
誤った値を指定した場合、監視が正常に行われません。

「3.9 CRS 監視設定ファイル (crsmond.conf) の作成」を参照し、必要に応じて各パラメータを設定してください。

以上で、ApplicationMonitor の基本的な設定は終了です。

5.3. ApplicationMonitor の設定確認

作成した設定ファイル (oramond.conf) の内容を確認します。

Oracle グループおよび LISTENER グループが動作している状態で、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c check -f /etc/opt/HA/AM/conf/oramond.conf
Finished to check configuration-file.
#
```

設定確認が正常に完了すると、上記の実行結果になります。エラーメッセージが出力されていないことを確認してください。

```
# /opt/HA/AM/bin/oraadmin -c check -f /etc/opt/HA/AM/conf/oramond.conf
Oracle or listener is DOWN, so can't check parameter. Node=node1
Oracle or listener is DOWN, so can't check parameter. Node=node2
Finished to check configuration-file.
#
```

リロケータブル IP を使用するリスナーを監視する RAC 構成の環境で、設定確認が正常に完了すると、上記の実行結果になります。上記は、LISTENER グループ 1 のリスナーが node2 で、LISTENER グループ 2 のリスナーが node1 で動作していない場合の実行結果です。上記以外のメッセージが出力されていないことを確認してください。

```
# /opt/HA/AM/bin/oraadmin -c check -f /etc/opt/HA/AM/conf/oramond.conf
Oracle or listener is DOWN, so can't check parameter. Node=node2
Finished to check configuration-file.
#
```

現用待機構成の環境で、設定確認が正常に完了すると、上記の実行結果になります。上記は、Oracle が node1 で動作しており、node2 では動作していない場合の実行結果です。Oracle が動作しているノード上のエラーメッセージが出力されていないことを確認してください。

メッセージの詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 1.2 for Linux メッセージ一覧』を参照してください。また、エラーメッセージが出力される場合は、「3.2.7 パラメーター一覧」を参照し、設定パラメータに誤りがないか確認してください。

注意： 現用待機構成の環境では、Oracle が動作していないノード上での設定は確認しません。可能であれば、Oracle 起動リソースを待機系ノードに移動させた状態で、再度管理コマンドを実行し、待機系ノードの設定も確認されることを推奨します。

注意: CRS 監視設定ファイル (crsmond.conf) の設定確認は、管理コマンド (oraadmin) では行えません。CRS 監視設定ファイルの設定確認は、CRS 監視モニタ (crsmond) 起動時に行います。

以上で、ApplicationMonitor の設定確認は終了です。

5.4. ApplicationMonitor の設定配布

作成した設定ファイル (oramond.conf) から構成情報ファイル (oramond<構成番号>.bin) を生成し、設定した各ノードに配布します。

Oracle グループおよび LISTENER グループが動作している状態で、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c apply -f /etc/opt/HA/AM/conf/oramond.conf
Finished to check configuration-file.
Succeeded in apply configuration file.
#
```

設定配布が正常に完了すると、上記の実行結果になります。**"Succeeded in apply configuration file."** が出力されたことを確認してください。

```
# /opt/HA/AM/bin/oraadmin -c apply -f /etc/opt/HA/AM/conf/oramond.conf
Oracle or listener is DOWN, so can't check parameter. Node=node1
Oracle or listener is DOWN, so can't check parameter. Node=node2
Finished to check configuration-file.
Succeeded in apply configuration file.
#
```

リロケータブル IP を使用するリスナーを監視する RAC 構成の環境で、設定配布が正常に完了すると、上記の実行結果になります。上記は、LISTENER グループ 1 のリスナーが node2 で、LISTENER グループ 2 のリスナーが node1 で動作していない場合の実行結果です。

"Succeeded in apply configuration file." が出力されたことを確認してください。

```
# /opt/HA/AM/bin/oraadmin -c apply -f /etc/opt/HA/AM/conf/oramond.conf
Oracle or listener is DOWN, so can't check parameter. Node=node2
Finished to check configuration-file.
Succeeded in apply configuration file.
#
```

ApplicationMonitor を現用待機構成の環境で、設定配布が正常に完了すると、上記の実行結果になります。上記は、Oracle が node1 で動作しており、node2 では動作していない場合の実行結果です。**"Succeeded in apply configuration file."** が出力されたことを確認してください。

メッセージの詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 1.2 for Linux メッセージ一覧』を参照してください。

注意: CRS 監視設定ファイル (crsmond.conf) の設定配布は、管理コマンド (oraadmin) では行えません。各ノードで CRS 監視設定ファイルを作成してください。

以上で、ApplicationMonitor の設定配布は終了です。

5.5. ApplicationMonitor の CLUSTERPRO 連携設定

設定を終えた ApplicationMonitor と CLUSTERPRO を連携します。以下は、CLUSTERPRO WebManager にて行います。

注意: 以下は、変更が必要な設定のみ記載しています。その他の設定については、環境に合わせて設定してください。

CLUSTERPRO の設定および表示などに関する詳細は、CLUSTERPRO のマニュアルを参照してください。RENS 連携をする場合、設定などに関する詳細は、サーバ管理基盤のマニュアルを参照してください。

5.5.1. EXEC リソースの追加

Oracle グループに、ApplicationMonitor を起動 / 停止するリソース (EXEC リソース) を追加します。

以下は、PID モニタリソースによる監視を行う場合の手順です。

- 1 Oracle グループ (マニュアル上では、failover_oracle) を右クリックし、「リソースの追加」を選択します。
- 2 「グループのリソース定義」ダイアログボックスが開きます。
「タイプ」で「execute resource」を選択し、「名前」に任意の EXEC リソース名 (マニュアル上では、「exec_am」) を入力し、「次へ」をクリックします。
- 3 依存関係を設定します。
「既定の依存関係に従う」のチェックボックスのチェックを外します。
「利用可能なリソース」から Oracle 起動用の EXEC リソースを選択し、「追加」をクリックします。
「依存するリソース」に Oracle 起動用の EXEC リソースが追加されたことを確認し、「次へ」をクリックします。
- 4 復旧動作の設定画面が表示されますが、「次へ」をクリックします。
- 5 開始スクリプトおよび終了スクリプトを設定します。
「この製品で作成したスクリプト」にチェックし、「スクリプト一覧」を選択し、「編集」をクリックするとエディタが起動します。

- 5.1 「スクリプト一覧」から「start.sh」を選択し、「編集」をクリックします。

CLUSTERPRO が提供している開始スクリプト (start.sh) に ApplicationMonitor を起動する処理を追加します。単独起動する場合も同様です。

記述例は、以下のとおりです。

```
/etc/opt/HA/AM/lbin/oramond [-C 構成番号]
```

ApplicationMonitor の起動の詳細は、「6 コマンドリファレンス」を参照し、環境に合わせて開始スクリプトを設定してください。

- 5.2 CLUSTERPRO が提供している終了スクリプト (stop.sh) に ApplicationMonitor を停止する処理を追加します。

「スクリプト一覧」から「stop.sh」を選択し、「編集」をクリックします。

注意: ApplicationMonitor の停止処理は、以下のようにモニタ制御デーモンに対して SIGTERM を送るように終了スクリプトを設定してください。単独起動する場合も同様です。

記述例は、以下のとおりです。

```
kill <モニタ制御デーモン PID>
```

- 5.3 開始スクリプトおよび終了スクリプトのパラメータを設定します。

「調整」をクリックすると、「exec リソース調整プロパティ」ダイアログボックスが開きます。

「パラメータ」タブで「開始スクリプト」は、「非同期」を選択します。「終了スクリプト」は、「同期」を選択し、「OK」をクリックします。

- 6 「グループのリソース定義」を終了します。

「完了」をクリックします。

注意: 同一ノード上で複数のインスタンスを監視する場合、インスタンス数分の EXEC リソースを追加してください。また、開始スクリプト設定時に "-C" オプションを付与して構成番号を指定してください。

各モニタの単独起動を合わせて行う場合も、起動するモニタ数分の EXEC リソースを追加してください。

ApplicationMonitor の起動の詳細は、「6 コマンドリファレンス」を参照してください。

以下は、PID モニタリソースによる監視を行わずに RENS と連携する場合の手順です。

- 1 Oracle グループ (マニュアル上では、failover_oracle) を右クリックし、「リソースの追加」を選択します。
- 2 「グループのリソース定義」ダイアログボックスが開きます。
「タイプ」で「execute resource」を選択し、「名前」に任意の EXEC リソース名 (マニュアル上では、「exec_am」) を入力し、「次へ」をクリックします。
- 3 依存関係を設定します。
「既定の依存関係に従う」のチェックボックスのチェックを外します。
「利用可能なリソース」から Oracle 起動用の EXEC リソースを選択し、「追加」をクリックします。
「依存するリソース」に Oracle 起動用の EXEC リソースが追加されたことを確認し、「次へ」をクリックします。
- 4 復旧動作の設定画面が表示されますが、「次へ」をクリックします。
- 5 開始スクリプトおよび終了スクリプトを設定します。
「この製品で作成したスクリプト」にチェックし、「スクリプト一覧」を選択し、「編集」をクリックするとエディタが起動します。

- 5.1 「スクリプト一覧」から「start.sh」を選択し、「編集」をクリックします。

CLUSTERPRO が提供している開始スクリプト (start.sh) に ApplicationMonitor を起動する処理を追加します。単独起動する場合も同様です。

記述例は、以下のとおりです。

```
/etc/opt/HA/AM/lbin/oramond [-C 構成番号] &
```

注意: バックグラウンド指定するなど、開始スクリプトが終了するように記述してください。

ApplicationMonitor の起動の詳細は、「6 コマンドリファレンス」を参照し、環境に合わせて開始スクリプトを設定してください。

- 5.2 CLUSTERPRO が提供している終了スクリプト (stop.sh) に ApplicationMonitor を停止する処理を追加します。
「スクリプト一覧」から「stop.sh」を選択し、「編集」をクリックします。

注意: ApplicationMonitor の停止処理は、以下のようにモニタ制御デーモンに対して SIGTERM を送るよう終了スクリプトを設定してください。単独起動する場合も同様です。

記述例は、以下のとおりです。

kill <モニタ制御デーモン PID>

5.3 開始スクリプトおよび終了スクリプトのパラメータを設定します。

「調整」をクリックすると、「exec リソース調整プロパティ」ダイアログボックスが開きます。

「パラメータ」タブで「開始スクリプト」は、「同期」を選択します。「終了スクリプト」は、「同期」を選択し、「OK」をクリックします。

6 「グループのリソース定義」を終了します。

「完了」をクリックします。

注意: 同一ノード上で複数のインスタンスを監視する場合、インスタンス数分の EXEC リソースを追加してください。また、開始スクリプト設定時に "-C" オプションを付与して構成番号を指定してください。

各モニタの単独起動を合わせて行う場合も、起動するモニタ数分の EXEC リソースを追加してください。

ApplicationMonitor の起動の詳細は、「6 コマンドリファレンス」を参照してください。

注意: RENS と連携する場合、開始スクリプトが終了しても「5.5.3 外部連携モニタリソースの作成」で作成する RENS に対する外部連携モニタリソースの登録が完了している (ApplicationMonitor が RENS のリソースとして認識されている) とは限りません。リソースの登録完了前に外部連携モニタが活性化すると障害と判断する可能性があります。

そのため、開始スクリプトに sleep コマンドの処理を追加し、外部連携モニタリソースの登録が完了後に開始スクリプトを終了させてください。

以上で、EXEC リソースの追加は終了です。

5.5.2. PID モニタリソースの作成

ApplicationMonitor を監視する PID モニタリソースを作成します。

PID モニタリソースによる監視を行わない場合は、作成する必要はありません。

- 1 「Monitor」を右クリックし、「モニタリソースの追加」を選択します。
- 2 「モニタリソースの定義」ダイアログボックスが開きます。
「タイプ」で「pid monitor」を選択し、「名前」に任意のモニタリソース名(マニュアル上では、pidw_am)を入力し、「次へ」をクリックします。
- 3 監視条件を設定します。
「監視タイミング」の「活性時」を選択し、「参照」をクリックします。
ApplicationMonitor 起動用の EXEC リソース (exec_am) を選択し、「次へ」をクリックします。
- 4 回復動作を設定します。
「回復対象」の「参照」をクリックし、Oracle グループ (failover_oracle) を選択します。
「最大再活性回数」を 0 回に設定し、「完了」をクリックします。

注意: 同一ノード上で複数のインスタンスを監視する場合、インスタンス数分の PID モニタリソースを追加してください。

各モニタの単独起動を合わせて行う場合も、起動するモニタ数分の PID モニタリソースを追加してください。

以上で、PID モニタリソースの作成は終了です。

5.5.3. 外部連携モニタリソースの作成

RENS と連携する場合、CLUSTERPRO の外部連携モニタリソースを作成する必要があります。

RENS と連携しない場合は、作成する必要はありません。

- 1 「Monitor」を右クリックし、「モニタリソースの追加」を選択します。
- 2 「モニタリソースの定義」ダイアログボックスが開きます。
「タイプ」で「message receive monitor」を選択し、「名前」に任意のモニタリソース名を入力し、「次へ」をクリックします。
- 3 監視条件を設定します。
「監視タイミング」の「活性時」を選択し、「参照」をクリックします。
ApplicationMonitor 起動用の EXEC リソース (exec_am) を選択し、「次へ」をクリックします。
- 4 「共通」タブで「カテゴリ」に「HAAM」を入力し、「次へ」をクリックします。
- 5 回復動作を設定します。
「回復対象」の「参照」をクリックし、Oracle グループ (failover_oracle) を選択します。
「完了」をクリックします。

以上で、外部連携モニタリソースの作成は終了です。

5.5.4. クラスタ構成情報の反映

クラスタ構成情報の内容を CLUSTERPRO に反映します。

- 1 「ファイル」メニューから「設定の反映」を選択します。
- 2 確認ダイアログが表示されますので「OK」をクリックします。
- 3 反映に成功すると確認ダイアログが表示されます。「了解」をクリックし、ダイアログを閉じます。

注意： 設定した情報によって表示されるメッセージが異なります。表示されたメッセージにしたがって操作を行ってください。

詳細は、CLUSTERPRO のマニュアルを参照してください。

クラスタが停止状態の場合は、クラスタを開始します。

WebManager の「表示」メニューより、「操作モード」を選択し、「サービス」メニューから「クラスタ開始」を選択してください。

次に、設定が反映されていることを確認します。

- 1 WebManager の「表示」メニューより、「操作モード」を選択します。
- 2 フェイルオーバーグループの Oracle 起動用 EXEC リソース「exec_am」が「起動済」であることを確認してください。
- 3 ApplicationMonitor 監視用のモニタリソース「pidw_am」が「正常」であることを確認してください。

以上で、ApplicationMonitor の CLUSTERPRO 連携設定は終了です。

5.6. ApplicationMonitor の起動

クラスタ構成情報の内容を CLUSTERPRO に反映すると、ApplicationMonitor は起動します。

Oracle が起動したノード上の syslog メッセージを参照し、ApplicationMonitor が起動したことを確認してください。

```
...
oramond[6705]: O:ApplicationMonitor 1.2
oramond[6705]: O>Loading configuration file /etc/opt/HA/AM/conf/oramond0. bin.
oramond[6705]: O:Logging level is 2.
oramond[6705]: O:altmond[6708] started.
...
oramond[6705]: O:Monitoring activity on local node is started.
oramond[6705]: O:instmond[6729] sid1 started.
oramond[6705]: O:altmond[6708] status is UP.
oramond[6705]: O:ssdump[6731] started.
oramond[6705]: O:lsnrmond[6732] LISTENER1 started.
oramond[6705]: O:tschkmond[6733] TSCHKMOND1 started.
oramond[6705]: O:fdsmmond[6734] started.
oramond[6705]: O:statsmond[6735] STATSMOND1 started.
oramond[6705]: O:stbdbmond[6736] started.
oramond[6705]: O:stbdbmond[6736] status is UP.
oramond[6705]: O:lsnrmond[6732] LISTENER1 status is UP.
oramond[6705]: O:ssdump[6731] status is UP.
oramond[6705]: O:fdsmmond[6734] status is UP.
oramond[6705]: O:statsmond[6735] STATSMOND1 status is UP.
oramond[6705]: O:tschkmond[6733] TSCHKMOND1 status is UP.
oramond[6705]: O:instmond[6729] sid1 status is UP.
oramond[6705]: O:odgbroker status is UP.
...
```

ApplicationMonitor が起動し、監視対象に指定したインスタンス名、リスナー名などに "**status is UP**" と表示されていれば、正常に監視が行われています。

Oracle が起動しない場合は、表示される syslog メッセージを参照し、設定を再度見直してください。

ApplicationMonitor によって出力されるメッセージの詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 1.2 for Linux メッセージ一覧』を参照してください。また、エラーメッセージが出力される場合は、「3.2.7 パラメーター一覧」を参照し、設定パラメータに誤りがないか確認してください。

以上で、ApplicationMonitor の起動は終了です。

5.7. ApplicationMonitor 動作中の状態確認

ApplicationMonitor の状態を確認することができます。

以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: node1 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	0/180	NO
INSTANCE	n/a	ACTIVE	UP	90	1/120	YES
LISTENER	LISTENER1	ACTIVE	UP	60	0/90	YES
LISTENER	LISTENER2	ACTIVE	UP	60	1/90	NO
TABLESPACE	TSCHKMOND1	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND1	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
NODE: node2 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	0/180	NO
INSTANCE	n/a	ACTIVE	UP	90	1/120	YES
LISTENER	LISTENER3	ACTIVE	UP	60	3/90	NO
TABLESPACE	TSCHKMOND2	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	ACTIVE	UP	60	1/90	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STATS	STATSMOND2	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
#
```

監視対象のインスタンス、リスナーの状態 (t-status) が **"UP"** であることが確認できます。

また、単独起動しているモニタの状態も確認できます。モニタ制御デーモン (oramond) による制御 (control) が "NO" で表示されているモニタは、単独起動モニタになります。

5 製品の運用

ApplicationMonitor を現用待機構成の環境でご使用の際、以下のような表示となります。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: node1 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	UNKNOWN	UNKNOWN	n/a	n/a	NO
INSTANCE	n/a	ACTIVE	UP	90	4/120	YES
LISTENER	LISTENER	ACTIVE	UP	60	5/90	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
NODE: node2 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	UNKNOWN	UNKNOWN	n/a	n/a	NO
INSTANCE	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
LISTENER	LISTENER	UNKNOWN	UNKNOWN	n/a	n/a	YES
SSDUMP	n/a	UNKNOWN	n/a	n/a	n/a	YES

```
#
```

Oracle が動作していないノードは、"**UNKNOWN**" と表示されます。

RENS と連携する場合は、表示項目が増えます。表示項目は、以下のとおりです。

resource-name: RENS リソース名

RENS リソース名は、以下の規則で自動的に作成されます。

モニタ	RENS リソース名
CRS 監視モニタ	CRS
インスタンス監視モニタ	Instance_ < SID 名 >
リスナー監視モニタ	Listener_ < リスナー名 >

r-status: RENS に通知済みのリソース値

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: node1 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control	resource-name	r-status
CRS	n/a	ACTIVE	UP	90	3/180	NO	CRS	UP
INSTANCE	n/a	ACTIVE	UP	90	4/120	YES	Instance_sid1	UP
LISTENER	LISTENER	ACTIVE	UP	60	5/90	YES	Listener_LISTENER	UP
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES		

```
#
```

注意: RENS がインストールされていない環境では、RENS の項目 (resource-name、r-status) は表示されません。

以上で、ApplicationMonitor 動作中の状態確認は終了です。

5.8. ApplicationMonitor の設定パラメータの調整

ApplicationMonitor は、監視対象の Oracle に対するアクセスが一定時間経過しても終了しない場合をストールと判断します。そのため、ご使用の環境に合わせて POLL_TIMEOUT パラメータの値を調整する必要があります。

ApplicationMonitor を動作させた状態で、Oracle に対して通常の業務を想定した負荷を与えた上で、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: node1 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	113 /180	NO
INSTANCE	n/a	ACTIVE	UP	90	67 /120	YES
LISTENER	LISTENER1	ACTIVE	UP	60	35 /90	YES
LISTENER	LISTENER2	ACTIVE	UP	60	30 /90	NO
TABLESPACE	TSCHKMOND1	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND1	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
NODE: node2 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	109 /180	NO
INSTANCE	n/a	ACTIVE	UP	90	65 /120	YES
LISTENER	LISTENER3	ACTIVE	UP	60	37 /90	NO
TABLESPACE	TSCHKMOND2	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	ACTIVE	UP	60	41 /90	YES
STATS	STATSMOND2	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
#
```

現在までに Oracle のアクセスが完了するまでにかかった最大監視応答時間 (tat) が表示されます。表示される値を参考にし、ピーク負荷時にストールと判断しない範囲内で POLL_TIMEOUT パラメータの値を調整してください。

POLL_TIMEOUT パラメータを小さくすると早期にストールを検出できますが、ストールと判断する可能性が高くなります。そのため、システムで許容されるダウンタイム内で大きい値を設定することを推奨します。

また、現在の設定で既にストールと判断してしまうようであれば、POLL_TIMEOUT パラメータに

は、さらに大きい値を設定してください。

以上で、ApplicationMonitor の設定パラメータの調整は終了です。

5.9. ApplicationMonitor 動作中の設定パラメータの変更

POLL_TIMEOUT パラメータを変更する場合など、ApplicationMonitor を動作させた状態で、新しい設定を適用させることができます。

設定ファイル (oramond.conf) を編集後、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c apply -f /etc/opt/HA/AM/conf/oramond.conf
Finished to check configuration-file.
Succeeded in apply configuration file.
#
```

ApplicationMonitor の動作中に管理コマンド (oraadmin) の apply リクエストを実行した場合、新しい構成情報ファイル (oramond<構成番号>.bin) を全ノードに配布し、動作中の ApplicationMonitor に対して新しい設定内容を適用します。

```
# /opt/HA/AM/bin/oraadmin -c apply -f /etc/opt/HA/AM/conf/oramond.conf
Oracle or listener is DOWN, so can't check parameter. Node=node1
Oracle or listener is DOWN, so can't check parameter. Node=node2
Finished to check configuration-file.
Succeeded in apply configuration file.
#
```

リロケータブル IP を使用するリスナーを監視する RAC 構成の環境で、設定内容の適用が正常に完了すると、上記の実行結果になります。上記は、LISTENER グループ 1 のリスナーが node2 で、LISTENER グループ 2 のリスナーが node1 で動作していない場合の実行結果です。

単独起動しているモニタがある場合は、上記を実行したあと、各単独起動モニタに対して個別に以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c apply -m lsnr -s LISTENER2 -n node1
Success: lsnrmond (node = node1, listener_name = LISTENER2) is accepted (request = apply).
#
# /opt/HA/AM/bin/oraadmin -c apply -m lsnr -s LISTENER3 -n node2
Success: lsnrmond (node = node2, listener_name = LISTENER3) is accepted (request = apply).
#
```

動作中に設定変更が可能なパラメータは、「3.2.7 パラメーター一覧」を参照してください。

注意: リスナー監視モニタ数、表領域監視モニタ数および統計情報採取モニタ数を動的に変更することはできません。

変更する場合は、以下の手順にしたがってください。

1. CLUSTERPRO の Oracle グループを停止する。
2. 設定ファイルを編集する。
3. 管理コマンドから check リクエストを実行し、設定を確認する。
4. 管理コマンドから apply リクエストを実行し、設定内容の適用および構成情報ファイルを配布する。
5. CLUSTERPRO の Oracle グループを起動する。
6. 管理コマンドから show リクエストを実行し、変更内容が適用されているか確認する。

注意: 監視対象がフローティング IP によって構成されており、かつそのアドレスが無効化されている場合、check / apply リクエストが失敗します。
該当アドレスを有効な状態にして、check / apply リクエストを行うようにしてください。
CLUSTERPRO 連携を行っている場合、以下手順にしたがって設定を変更してください。

1. CLUSTERPRO の Oracle グループを停止する。
2. Oracle グループから、ApplicationMonitor を起動するモニタリソースを外す。
3. フローティング IP を有効化するため、Oracle グループを起動する。
4. 管理コマンドから check リクエストを実行し、設定を確認する。
5. 管理コマンドから apply リクエストを実行し、設定内容の適用および構成情報ファイルを配布する。
6. Oracle グループに ApplicationMonitor を起動するモニタリソースを再登録する。
7. Oracle および ApplicationMonitor の起動を確認する。
8. 管理コマンドから show リクエストを実行し、設定内容が適用されているか確認する。

注意: Oracle がマウント状態時、ApplicationMonitor の動作中に check / apply リクエストを実行した場合、Oracle 関連パラメータの確認に失敗することがあります。
ApplicationMonitor を停止した状態で check / apply リクエストを実行するか、管理コマンドの "-F" オプションを付与して check / apply リクエストを実行してください。
"-F" オプションを付与した場合、Oracle 関連パラメータの確認を省略しますので、設定値に問題がないことを十分ご確認ください。
管理コマンドの詳細は、「6 コマンドリファレンス」を参照してください。

以上で、ApplicationMonitor 動作中の設定パラメータの変更は終了です。

5.10. ApplicationMonitor の一時停止と再開

Oracle を動作させたままの状態、データベースのバックアップや表領域のメンテナンスなどの作業を実施する場合、ApplicationMonitor の管理コマンド (oraadmin) を用いて、監視および採取を一時的に停止させることができます。

これによって、メンテナンスに伴う過大な負荷や長時間の表ロックなど、障害と判断して CLUSTERPRO のフェイルオーバーグループが停止することを避けることができます。

5.10.1. 監視および採取の一時停止

ApplicationMonitor の監視および採取を一時停止するには、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c suspend
Connect to monitor-control(master).
Succeeded in suspend.
#
```

ApplicationMonitor の状態を確認するため、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: node1 SUSPEND
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	0/180	NO
INSTANCE	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
LISTENER	LISTENER1	UNKNOWN	UNKNOWN	n/a	n/a	YES
LISTENER	LISTENER2	ACTIVE	UP	60	1/90	NO
TABLESPACE	TSCHKMOND1	UNKNOWN	n/a	n/a	n/a	YES
DISKSPACE	n/a	UNKNOWN	n/a	n/a	n/a	YES
STANDBYDB	n/a	UNKNOWN	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND1	UNKNOWN	n/a	n/a	n/a	YES
SSDUMP	n/a	UNKNOWN	n/a	n/a	n/a	YES

```
NODE: node2 SUSPEND
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	0/180	NO
INSTANCE	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
LISTENER	LISTENER3	ACTIVE	UP	60	3/90	NO
TABLESPACE	TSCHKMOND2	UNKNOWN	n/a	n/a	n/a	YES
DISKSPACE	n/a	UNKNOWN	n/a	n/a	n/a	YES
STANDBYDB	n/a	UNKNOWN	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND2	UNKNOWN	n/a	n/a	n/a	YES
SSDUMP	n/a	UNKNOWN	n/a	n/a	n/a	YES

```
#
```

単独起動しているモニタがある場合は、各単独起動モニタに対して個別に以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c suspend -m lsnr -s LISTENER2 -n node1
Success: lsnrmond (node = node1, listener_name = LISTENER2) is accepted (request = suspend).
#
# /opt/HA/AM/bin/oraadmin -c suspend -m lsnr -s LISTENER3 -n node2
Success: lsnrmond (node = node2, listener_name = LISTENER3) is accepted (request = suspend).
#
# /opt/HA/AM/bin/oraadmin -c suspend -m crs -n node1
Success: crsmond (node = node1) is accepted (request = suspend).
#
# /opt/HA/AM/bin/oraadmin -c suspend -m crs -n node2
Success: crsmond (node = node2) is accepted (request = suspend).
#
```

ApplicationMonitor の状態を確認するため、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: node1 SUSPEND
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	SUSPEND	UP	90	0/180	NO
INSTANCE	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
LISTENER	LISTENER1	UNKNOWN	UNKNOWN	n/a	n/a	YES
LISTENER	LISTENER2	SUSPEND	UP	60	1/90	NO
TABLESPACE	TSCHKMOND1	UNKNOWN	n/a	n/a	n/a	YES
DISKSPACE	n/a	UNKNOWN	n/a	n/a	n/a	YES
STANDBYDB	n/a	UNKNOWN	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND1	UNKNOWN	n/a	n/a	n/a	YES
SSDUMP	n/a	UNKNOWN	n/a	n/a	n/a	YES

```
NODE: node2 SUSPEND
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	SUSPEND	UP	90	0/180	NO
INSTANCE	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
LISTENER	LISTENER3	SUSPEND	UP	60	3/90	NO
TABLESPACE	TSCHKMOND2	UNKNOWN	n/a	n/a	n/a	YES
DISKSPACE	n/a	UNKNOWN	n/a	n/a	n/a	YES
STANDBYDB	n/a	UNKNOWN	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND2	UNKNOWN	n/a	n/a	n/a	YES
SSDUMP	n/a	UNKNOWN	n/a	n/a	n/a	YES

```
#
```

単独起動モニタも含めて、ApplicationMonitor の状態が "**SUSPEND**" に変わったことを確認し、必要な作業を実施してください。

なお、一時停止した ApplicationMonitor は、Oracle に対するすべてのアクセスが抑止されます。

注意: 時刻補正を伴う作業を行う場合は、"-M" オプションを付与して suspend リクエストを実行し、モニタ制御デーモン (oramond) 内部で動作する無応答監視を一時停止してください。

監視および採取の一時停止の説明は、以上です。

5.10.2. 監視および採取の再開

作業終了後、一時停止中の監視および採取を再開させるには、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c resume
Connect to monitor-control(master).
Succeeded in resume.
```

ApplicationMonitor の状態を確認するため、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: node1 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	SUSPEND	UP	90	0/180	NO
INSTANCE	n/a	ACTIVE	UP	90	1/120	YES
LISTENER	LISTENER1	ACTIVE	UP	60	0/90	YES
LISTENER	LISTENER2	SUSPEND	UP	60	1/90	NO
TABLESPACE	TSCHKMOND1	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND1	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
NODE: node2 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	SUSPEND	UP	90	0/180	NO
INSTANCE	n/a	ACTIVE	UP	90	1/120	YES
LISTENER	LISTENER3	SUSPEND	UP	60	3/90	NO
TABLESPACE	TSCHKMOND2	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	ACTIVE	UP	60	1/90	YES
STATS	STATSMOND2	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
#
```

単独起動しているモニタがある場合は、各単独起動モニタに対して個別に以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c resume -m lsnr -s LISTENER2 -n node1
Success: lsnrmond (node = node1, listener_name = LISTENER2) is accepted (request = resume).
#
# /opt/HA/AM/bin/oraadmin -c resume -m lsnr -s LISTENER3 -n node2
Success: lsnrmond (node = node2, listener_name = LISTENER3) is accepted (request = resume).
#
# /opt/HA/AM/bin/oraadmin -c resume -m crs -n node1
Success: crsmond (node = node1) is accepted (request = resume).
#
# /opt/HA/AM/bin/oraadmin -c resume -m crs -n node2
Success: crsmond (node = node2) is accepted (request = resume).
#
```

ApplicationMonitor の状態を確認するため、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: node1 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	0/180	NO
INSTANCE	n/a	ACTIVE	UP	90	1/120	YES
LISTENER	LISTENER1	ACTIVE	UP	60	0/90	YES
LISTENER	LISTENER2	ACTIVE	UP	60	1/90	NO
TABLESPACE	TSCHKMOND1	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND1	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
NODE: node2 RESUME
```

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	0/180	NO
INSTANCE	n/a	ACTIVE	UP	90	1/120	YES
LISTENER	LISTENER3	ACTIVE	UP	60	3/90	NO
TABLESPACE	TSCHKMOND2	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	ACTIVE	UP	60	1/90	YES
STATS	STATSMOND2	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

```
#
```

単独起動モニタも含めて、ApplicationMonitor の状態が "ACTIVE" に変わったことを確認してください。

注意: ApplicationMonitor のモニタは、起動時に Oracle のライブラリをリンクしています。
suspend リクエストにて一時停止を行った場合、Oracle に対するアクセスは停止しますが、モニタプロセス自体は動作しているため、Oracle パッチ適用の際は、Oracle グループを停止してください。

以上で、Oracle のメンテナンスは終了です。

5.11. フェイルオーバーグループ停止原因の確認

Oracle グループが異常終了した場合、フェイルオーバーグループ停止の原因が ApplicationMonitor による障害の検出かどうかを確認する必要があります。

停止ノードの syslog を参照し、ApplicationMonitor のメッセージを確認してください。

```
...
instmond[6729]: [0] sql_error:Oracle error occurred.
instmond[6729]: [0] sql_error: ORA-01034: ORACLE not available
instmond[6729]: [0] sql_error: ORA-27101: shared memory realm does not exist
instmond[6729]: [0] sql_error: Linux-x86_64 Error: 2: No such file or directory
...
oramond[6705]: 0:instmond[6729] sid1 down retry over.
oramond[6705]: 0:instmond[6729] sid1 status is DOWN.
oramond[6705]: 0:Local node has been shifted to disaster mode.
...
oramond[6705]: 0:Issuing AM cluster reconstruction.
oramond[6705]: 0:oramond on this node is terminated.
...
clusterpro: <type: rm><event: 9> Detected an error in monitoring pidw_am. (1 :
Process does not exist. (pid=13145))
clusterpro: <type: apisv><event: 38> There was a request to failover
group(failover_oracle) from the clprm process(IP=<xxx.xxx.xxx.xxx>).
...
```

failover_oracle:	フェイルオーバーグループ名
pidw_am:	ApplicationMonitor モニタリソース名
13145:	ApplicationMonitor 起動スクリプトの PID
xxx.xxx.xxx.xxx:	環境の IP アドレス

"oramond on this node is terminated." と表示されていた場合、ApplicationMonitor が Oracle の異常を検出し、フェイルオーバーグループが停止しています。

上記の出力は、該当ノード上の Oracle インスタンス "sid1" が停止したことにより、ApplicationMonitor がインスタンス状態をダウンと判断し、自身を停止しています。その後、CLUSTERPRO によるフェイルオーバーグループの切り替えが発生しています。

障害の検出時は、出力される Oracle のエラーコード、別冊の『CLUSTERPRO MC ApplicationMonitor 1.2 for Linux メッセージ一覧』を参考にし、原因を判断してください。

なお、設定ファイル (oramond.conf) の SCRIPT_NAME パラメータに情報採取スクリプト (oradiag) を設定した場合は、/var/opt/HA/AM/<構成番号>/oradiag/ ディレクトリに障害解析に有益な情報が採取されます。必要に応じて、NEC カスタマーサポートセンターへ提示してください。

以上で、フェイルオーバーグループ停止原因の確認は終了です。

6. コマンドリファレンス

oraadmin

名 前 oraadmin - 管理コマンド

形 式 /opt/HA/AM/bin/oraadmin

機能説明 管理コマンドは、ApplicationMonitor の設定、状態表示などの運用および管理を行うためのコマンドです。

/opt/HA/AM/bin/oraadmin -c show

以下の show リクエストは、動作しているすべてのモニタに対する要求です。

```
-c show [-C <component_id>] [-p <port_number>]
-c show -t [-C <component_id>] [-p <port_number>]
-c show -d [-C <component_id>] [-p <port_number>]
-c show -b [-C <component_id>] [-p <port_number>]
-c show -S [-C <component_id>] [-p <port_number>]
-c show -a [-C <component_id>] [-p <port_number>]
```

<component_id> は、構成番号 (COMPONENT_ID パラメータ) を指定します。

<port_number> は、構成情報管理サーバ (oraconfd) のポート番号を指定します。

show	単独起動しているリスナー監視モニタ (lsnrmond)、表領域監視モニタ (tschkmond)、ディスク領域監視モニタ (fdsmond) および統計情報採取モニタ (statsmond) を含むモニタ情報を表示します。 また、-C の指定にかかわらず、各ノードで動作している CRS 監視モニタ (crsmond) のモニタ情報も表示します。
show -t	監視対象の表領域の情報を表示します。
show -d	監視対象のディスク領域の情報を種別単位に表示します。
show -b	監視対象のスタンバイ・インスタンスの情報を REDO ログ転送状況、REDO ログ適用状況ごとに表示します。
show -S	採取対象の統計情報を表示します。
show -a	上記オプション指定を含む show リクエストで得られるすべてのモニタ情報の詳細を表示します。

/opt/HA/AM/bin/oraadmin -c suspend

以下の suspend リクエストは、モニタ制御デーモン (oramond) によって管理されているモニタのみへの要求です。

そのため、単独起動しているリスナー監視モニタ、表領域監視モニタ、ディスク領域監視モニタ、統計情報採取モニタおよび CRS 監視モニタは含まれません。

```
-c suspend [-C <component_id>] [-n <node_name>] [-n <node_name>] . . .
-c suspend -M [-C <component_id>] [-n <node_name>] [-n <node_name>] . . .
```

<component_id> は、構成番号を指定します。

<node_name> は、監視対象のノード名 (NODE_NAME パラメータ) を指定します。

suspend	全ノードまたは指定したノードのモニタを一時停止します。
suspend -M	全ノードまたは指定したノードのモニタを一時停止すると同時に、モニタ制御デーモン自身の監視スレッドを一時停止します。

以下の suspend リクエストは、単独起動しているモニタのみへの要求です。

モニタ制御デーモンで管理されているモニタへの要求は行えません。

単独起動しているモニタへの要求では、構成番号およびノード名の指定が必要です。

```
-c suspend -m lsnr -s <リスナー名> -C <component_id> -n <node_name> [-p <port_number>]
-c suspend -m tschk -i <表領域監視モニタ ID> -C <component_id> -n <node_name> [-p <port_number>]
```

```
-c suspend -m fds -C <component_id> -n <node_name> [-p <port_number>]
-c suspend -m stats -i <統計情報採取モニタ ID> -C <component_id> -n <node_name> [-p <port_number>]
-c suspend -m crs -n <node_name> [-p <port_number>]
```

<リスナー名> は、監視対象のリスナー名を指定します。
 <component_id> は、構成番号を指定します。
 <node_name> は、監視対象のノード名を指定します。
 <port_number> は、構成情報管理サーバのポート番号を指定します。
 <表領域監視モニタ ID> は、表領域監視モニタの識別子を指定します。
 <統計情報採取モニタ ID> は、統計情報採取モニタの識別子を指定します。

```
suspend -m lsnr      単独起動しているリスナー監視モニタを一時停止します。
suspend -m tschk     単独起動している表領域監視モニタを一時停止します。
suspend -m fds       単独起動しているディスク領域監視モニタを一時停止します。
suspend -m stats     単独起動している統計情報採取モニタを一時停止します。
suspend -m crs       CRS 監視モニタを一時停止します。
```

/opt/HA/AM/bin/oraadmin -c resume

以下の resume リクエストは、モニタ制御デーモンによって管理されているモニタのみへの要求です。そのため、単独起動しているリスナー監視モニタ、表領域監視モニタ、ディスク領域監視モニタ、統計情報採取モニタおよび CRS 監視モニタは含まれません。

```
-c resume [-C <component_id>] [-n <node_name>] [-p <port_number>] . . .
```

<component_id> は、構成番号を指定します。
 <node_name> は、監視対象のノード名を指定します。

```
resume              全ノードまたは指定したノードのモニタを再開します。
                   また、監視スレッドが停止していた場合は、監視スレッドも監視を再開します。
```

以下の resume リクエストは、単独起動しているモニタのみへの要求です。
 モニタ制御デーモンで管理されているモニタへの要求は行えません。

単独起動しているモニタへの要求では、構成番号およびノード名の指定が必要です。

```
-c resume -m lsnr -s <リスナー名> -C <component_id> -n <node_name> [-p <port_number>]
-c resume -m tschk -i <表領域監視モニタ ID> -C <component_id> -n <node_name> [-p <port_number>]
-c resume -m fds -C <component_id> -n <node_name> [-p <port_number>]
-c resume -m stats -i <統計情報採取モニタ ID> -C <component_id> -n <node_name> [-p <port_number>]
-c resume -m crs -n <node_name> [-p <port_number>]
```

<リスナー名> は、監視対象のリスナー名を指定します。
 <component_id> は、構成番号を指定します。
 <node_name> は、監視対象のノード名を指定します。
 <port_number> は、構成情報管理サーバのポート番号を指定します。
 <表領域監視モニタ ID> は、表領域監視モニタの識別子を指定します。
 <統計情報採取モニタ ID> は、統計情報採取モニタの識別子を指定します。

```
resume -m lsnr      単独起動しているリスナー監視モニタを再開します。
resume -m tschk     単独起動している表領域監視モニタを再開します。
resume -m fds       単独起動しているディスク領域監視モニタを再開します。
resume -m stats     単独起動している統計情報採取モニタを再開します。
resume -m crs       CRS 監視モニタを再開します。
```

/opt/HA/AM/bin/oraadmin -c apply

ApplicationMonitor 動作中の場合、以下の apply リクエストは、モニタ制御デーモンによって管理されているモニタのみへの要求です。そのため、単独起動しているリスナー監視モニタ、表領域監視モニタ、ディスク領域監視モニタ、統計情報採取モニタおよび CRS 監視モニタは含まれません。

ApplicationMonitor 停止中の場合、以下の apply リクエストは、単独起動しているリスナー監視モニタ、表領域監視モニタ、ディスク領域監視モニタおよび統計情報採取モニタも含まれます。

```
-c apply -f <configure_file_name> [-p <port_number>] [-F] [-E]
```

<configure_file_name> は、任意の設定ファイル名を指定します。

<port_number> は、構成情報管理サーバのポート番号を指定します。

apply 指定された設定ファイルの構成情報を配布し、設定内容を全ノードに適用します。

-F オプションが指定された場合、ノードごとの Oracle 関連パラメータの確認を省略し、設定内容を適用します。主にフローティング IP 無効時に使用します。

-E オプションは、システムの運用上、Oracle グループを停止できない場合など、リスナー監視モニタ、表領域監視モニタおよび統計情報採取モニタの追加や削除を行った設定内容の動的変更を可能とします。しかし、単独起動しているモニタが動作中の場合、単独起動モニタは停止します。

単独起動モニタ動作中の -E オプション指定は、動作保証対象外となります。

以下の apply リクエストは、単独起動しているモニタのみへの要求です。

モニタ制御デモンで管理されているモニタへの要求は行えません。

単独起動しているモニタへの要求では、構成番号およびノード名の指定が必要です。

```
-c apply -m lsnr -s <リスナー名> -C <component_id> -n <node_name> [-p <port_number>]
-c apply -m tschk -i <表領域監視モニタ ID> -C <component_id> -n <node_name> [-p <port_number>]
-c apply -m fds -C <component_id> -n <node_name> [-p <port_number>]
-c apply -m stats -i <統計情報採取モニタ ID> -C <component_id> -n <node_name> [-p <port_number>]
-c apply -m crs -n <node_name> [-p <port_number>]
```

<リスナー名> は、監視対象のリスナー名を指定します。

<component_id> は、構成番号を指定します。

<node_name> は、監視対象のノード名を指定します。

<port_number> は、構成情報管理サーバのポート番号を指定します。

<表領域監視モニタ ID> は、表領域監視モニタの識別子を指定します。

<統計情報採取モニタ ID> は、統計情報採取モニタの識別子を指定します。

apply -m lsnr	単独起動しているリスナー監視モニタに新しい設定内容を適用します。
apply -m tschk	単独起動している表領域監視モニタに新しい設定内容を適用します。
apply -m fds	単独起動しているディスク領域監視モニタに新しい設定内容を適用します。
apply -m stats	単独起動している統計情報採取モニタに新しい設定内容を適用します。
apply -m crs	CRS 監視モニタに新しい設定内容を適用します。

/opt/HA/AM/bin/oraadmin -c check

以下の check リクエストは、構成情報管理サーバに対する要求です。

各監視モニタに対して直接的な要求は行いません。

```
-c check -f <configure_file_name> [-p <port_number>] [-F] [-E]
```

<configure_file_name> は、任意の設定ファイル名を指定します。

<port_number> は、構成情報管理サーバのポート番号を指定します。

check 指定された設定ファイルのチェックを行います。

-F オプションが指定された場合、ノードごとの Oracle 関連パラメータの確認を省略します。主にフローティング IP 無効時に使用します。

-E オプションが指定されている場合、リスナー監視モニタ、表領域監視モニタおよび統計情報採取モニタの追加や削除を行った設定内容の動的確認を可能とします。

/opt/HA/AM/bin/oraadmin -c delete

以下の delete リクエストは、構成情報管理サーバに対する要求です。

各監視モニタに対して直接的な要求は行いません。

```
-c delete [-p <port_number>] [-C <component_id>]
```

<port_number> は、構成情報管理サーバのポート番号を指定します。

<component_id> は、構成番号を指定します。

delete 構成情報ファイル (oramond<構成番号>.bin) を削除します。

`/opt/HA/AM/bin/oraadmin -c view`

以下の view リクエストは、構成情報管理サーバに対する要求です。

各監視モニタに対して直接的な要求は行いません。

`-c view [-p <port_number>] [-C <component_id >]`

<port_number> は、構成情報管理サーバのポート番号を指定します。

<component_id> は、構成番号を指定します。

view 現在の構成情報を表示します。

使 用 例 コマンド

`oraadmin -c show`

を実行した場合、次のような出力になります。

Connect to monitor-control(master).

NODE: node1 RESUME

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	0/180	NO
INSTANCE	n/a	ACTIVE	UP	90	1/120	YES
LISTENER	LISTENER1	ACTIVE	UP	60	0/90	YES
LISTENER	LISTENER2	ACTIVE	UP	60	1/90	NO
TABLESPACE	TSCHKMOND1	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	UNKNOWN	UNKNOWN	n/a	n/a	YES
STATS	STATSMOND1	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

NODE: node2 RESUME

monitors	name	m-status	t-status	interval	tat/limit	control
CRS	n/a	ACTIVE	UP	90	0/180	NO
INSTANCE	n/a	ACTIVE	UP	90	1/120	YES
LISTENER	LISTENER3	ACTIVE	UP	60	3/90	NO
TABLESPACE	TSCHKMOND2	ACTIVE	n/a	3600	n/a	YES
DISKSPACE	n/a	ACTIVE	n/a	3600	n/a	YES
STANDBYDB	n/a	ACTIVE	n/a	n/a	n/a	YES
ODGBROKER	n/a	ACTIVE	UP	60	1/90	YES
STATS	STATSMOND2	ACTIVE	n/a	n/a	n/a	YES
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES

ノード単位に状態が出力されます。

上記の例では、node1 および node2 がノード名になります。

ノード名の後ろに、各ノード上のモニタ制御デーモンの状態が出力されます。

UNKNOWN	状態不明または停止中
SUSPEND	一時停止中
SUSPEND (Meta-Control Stopped)	一時停止中 (モニタ制御デーモン自己監視スレッド停止中)
RESUME	動作中

モニタ制御デーモンの状態に続き、各ノード上のモニタの状態が出力されます。

列名とその意味を以下に示します。

monitors	監視モニタおよび採取モニタ	
CRS	CRS	CRS 監視モニタ
INSTANCE	INSTANCE	インスタンス監視モニタ
LISTENER	LISTENER	リスナー監視モニタ
TABLESPACE	TABLESPACE	表領域監視モニタ
DISKSPACE	DISKSPACE	ディスク領域監視モニタ
STANDBYDB	STANDBYDB	スタンバイデータベース監視モニタ

	ODGBROKER	モニタ制御デーモン (Oracle Data Guard Broker プロセス状態)
	STATS	統計情報採取モニタ
	SSDUMP	システムステートダンプ採取デーモン
name	リスナー監視モニタ、表領域監視モニタおよび統計情報採取モニタの場合、監視対象のリスナー名、表領域名および採取対象の統計情報名が出力されます。 上記以外は、"n/a" が出力されます。	
m-status	監視モニタおよび採取モニタの状態	
	UNKNOWN	状態不明または停止中
	ACTIVE	動作中
	SUSPEND	単独起動モニタの一時停止
t-status	監視対象および採取対象の状態	
	UNKNOWN	状態不明または停止中
	UP	正常
	DOWN	停止
	STALL	無応答
	SECONDARY	監視対象はセカンダリ・インスタンス
	ERROR	エラー
	n/a	モニタが表領域監視モニタ、ディスク領域監視モニタ、スタンバイデータベース監視モニタ、統計情報採取モニタおよびシステムステートダンプ採取デーモンの場合に出力
interval	各モニタの監視間隔が出力されます。 モニタ制御デーモン (スタンバイ・インスタンス監視時のプライマリ・サイト)、スタンバイデータベース監視モニタ、採取モニタおよび停止中の監視モニタは、"n/a" が出力されます。	
tat	Oracle へのアクセスが完了するまでにかった時間の最大値が出力されます。 モニタがモニタ制御デーモン (スタンバイ・インスタンス監視時のプライマリ・サイト)、表領域監視モニタ、ディスク領域監視モニタ、スタンバイデータベース監視モニタ、統計情報採取モニタおよびシステムステートダンプ採取デーモンの場合またはモニタが停止中の場合は、"n/a" が出力されます。	
limit	POLL_TIMEOUT パラメータに設定した値が出力されます。 モニタがモニタ制御デーモン (スタンバイ・インスタンス監視時のプライマリ・サイト)、表領域監視モニタ、ディスク領域監視モニタ、スタンバイデータベース監視モニタ、統計情報採取モニタおよびシステムステートダンプ採取デーモンの場合またはモニタが停止中の場合は、"n/a" が出力されます。	
control	モニタ制御デーモンによる制御	
	YES	モニタ制御デーモンの制御下
	NO	単独起動モニタ

/opt/HA/AM/bin/oraadmin -c show

を実行の際に RENS と連携している場合、以下の 2 項目が追加されます。

Connect to monitor-control(master).

NODE: node1 RESUME

monitors	name	m-status	t-status	interval	tat/limit	control	resource-name	r-status
CRS	n/a	ACTIVE	UP	90	3/180	NO	CRS	UP
INSTANCE	n/a	ACTIVE	UP	90	4/120	YES	Instance_sid1	UP
LISTENER	LISTENER	ACTIVE	UP	60	5/90	YES	Listener_LISTENER	UP
SSDUMP	n/a	ACTIVE	n/a	n/a	n/a	YES		
resource-name	RENS リソース名							
	Instance_ < SID 名 >						インスタンス監視モニタの監視情報	
	Listener_ < リスナー名 >						リスナー監視モニタの監視情報	
	CRS						CRS 監視モニタの監視情報	
r-status	RENS に通知したリソース値							
	UNKNOWN						状態不明	
	UP						正常	
	DOWN						停止	

6 コマンドリファレンス

STALL	無応答
SUSPEND	監視一時停止中
MONITOR_HALT	監視停止

ファイル /opt/HA/AM/bin/oraadmin

- 注 意 管理コマンドは、同一ノード、別ノードを問わず、同時に複数実行することはできません。
モニタ制御デーモン起動中、Oracle 障害発生中および apply リクエスト直後は、管理コマンドを実行できません。
管理コマンドは、root ユーザーのみ実行できます。
ApplicationMonitor 動作中の delete リクエストは、実行できません。
Oracle Data Guard で手動フェイルオーバーまたはスイッチオーバーを実施する場合は、実施前に suspend リクエストを実行して ApplicationMonitor を一時停止してください。フェイルオーバーまたはスイッチオーバーが完了後、apply リクエストを実行して ApplicationMonitor を再構成してください。ApplicationMonitor 再構成後、resume リクエストを実行して ApplicationMonitor の動作を再開してください。

oraconfd

名 前 oraconfd - 構成情報管理サーバ

形 式 **/etc/opt/HA/AM/lbin/oraconfd [-p *ポート番号*]**

機能説明 構成情報管理サーバは、各ノード上の構成情報ファイル (oramond<構成番号>.bin) を管理するためのデーモンです。
通常、/etc/rc.d/init.d/haam_oraconfd により、システム立ち上げ時に自動起動されます。

-p ポート番号を指定します。省略した場合は、25310 です。

使用例 コマンド
 oraconfd
 oraconfd -p 25312

ファイル /etc/opt/HA/AM/lbin/oraconfd

注 意 構成情報管理サーバは、各ノード上で 1 つ起動している必要があります。
構成情報管理サーバの複数起動はできません。
構成情報管理サーバは、root ユーザーでのみ起動できます。

oramond

名 前 oramond - モニタ制御デーモン

形 式 **/etc/opt/HA/AM/lbin/oramond [-C 構成番号]**

機能説明 モニタ制御デーモンは、ApplicationMonitor の中心となるデーモンです。
 インスタンス監視モニタ (instmond)、リスナー監視モニタ (lsnrmond)、表領域監視モニタ (tschkmond)、ディスク領域監視モニタ (fdsmond)、スタンバイデータベース監視モニタ (stbdbmond)、アラートログ監視モニタ (altmond)、統計情報採取モニタ (statsmond) およびシステムステートダンプ採取デーモン (ssdump) の起動と管理を制御するためのデーモンです。
 通常は、CLUSTERPRO にフェイルオーバーグループのグループリソースとして登録し、クラスタ起動時に Oracle グループのモニタリソースとして起動します。

また、スタンバイ・インスタンス監視に Oracle Data Guard Broker プロセスの状態を監視します。
 障害検出時は、syslog に警告メッセージを表示します。

-C 構成番号を指定します。省略した場合は、0 です。

使 用 例 コマンド
 oramond
 oramond -C 1

フ ァ イ ル /etc/opt/HA/AM/lbin/oramond

注 意 モニタ制御デーモンは、root ユーザーでのみ起動できます。

lsnrmond

名 前 lsnrmond - リスナー監視モニタ

形 式 **/etc/opt/HA/AM/lbin/lsnrmond -s リスナー名 [-C 構成番号]**

機能説明 リスナー監視モニタは、リスナー障害を監視するためのデーモンです。
障害検出時に停止することで、モニタ制御デーモン (oramond) または CLUSTERPRO へ障害を通知します。

-s 設定ファイル (oramond.conf) に記述した ListenerMonitor ステートメントのリスナー名を指定します。
-C 構成番号を指定します。省略した場合は、0 です。

使用例 コマンド
lsnrmond -s LISTENER
lsnrmond -s LISTENER -C 1

ファイル /etc/opt/HA/AM/lbin/lsnrmond

注 意 リスナー監視モニタは、root または OSDBA(dba) グループに所属するユーザーで起動する必要があります。
リスナー監視モニタを単独起動する場合は、設定ファイルの ListenerMonitor ステートメント内の MONITOR_CONTROL パラメータを NO に設定する必要があります。
リスナー監視モニタの単独起動と RENS を連携する場合は、root または rens グループに所属するユーザーで起動する必要があります。

tschkmond

名 前 tschkmond_11G, tschkmond_11GR2, tschkmond_12C - 表領域監視モニタ

形 式 **/etc/opt/HA/AM/lbin/tschkmond_11G -i 表領域監視モニタ ID [-C 構成番号]**
/etc/opt/HA/AM/lbin/tschkmond_11GR2 -i 表領域監視モニタ ID [-C 構成番号]
/etc/opt/HA/AM/lbin/tschkmond_12C -i 表領域監視モニタ ID [-C 構成番号]

機能説明 表領域監視モニタは、データベース領域の容量不足を監視するためのデーモンです。
 容量不足時は、syslog にエラーメッセージを表示します。

-i 設定ファイル (oramond.conf) に記述した TableSpaceMonitor ステートメントの
 表領域監視モニタ ID を指定します。
 -C 構成番号を指定します。省略した場合は、0 です。

使 用 例 コマンド
 tschkmond_11G -i TSCHKMOND
 tschkmond_11G -i TSCHKMOND -C 1
 tschkmond_11GR2 -i TSCHKMOND
 tschkmond_11GR2 -i TSCHKMOND -C 1
 tschkmond_12C -i TSCHKMOND
 tschkmond_12C -i TSCHKMOND -C 1

フ ァ イ ル /etc/opt/HA/AM/lbin/tschkmond_11G
 /etc/opt/HA/AM/lbin/tschkmond_11GR2
 /etc/opt/HA/AM/lbin/tschkmond_12C

注 意 表領域監視モニタは、OSDBA(dba) グループに所属するユーザーで起動する必要があります。
 表領域監視モニタを単独起動する場合は、次の環境変数が設定されている必要があります。

ORACLE_SID	Oracle のインスタンス識別子
ORACLE_BASE	Oracle ディレクトリ構成の基本となるディレクトリ
ORACLE_HOME	Oracle ソフトウェアをインストールしたディレクトリ
ORA_NLS10	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されている ディレクトリ
NLS_LANG	表領域監視モニタが Oracle にアクセスする際の言語、地域およびキャラ クタ・セット
LD_LIBRARY_PATH	db ユーザー環境の 64bit 共有ライブラリのパス

表領域監視モニタを単独起動する場合は、設定ファイルの TableSpaceMonitor ステートメント内の
 MONITOR_CONTROL パラメータを NO に設定する必要があります。

fdsmond

名 前 fdsmond_11G, fdsmond_11GR2, fdsmond_12C - ディスク領域監視モニタ

形 式 **/etc/opt/HA/AM/lbin/fdsmond_11G [-C 構成番号]**
/etc/opt/HA/AM/lbin/fdsmond_11GR2 [-C 構成番号]
/etc/opt/HA/AM/lbin/fdsmond_12C [-C 構成番号]

機能説明 ディスク領域監視モニタは、ローカルディスク領域の容量不足を監視するためのデーモンです。
容量不足時は、syslog にエラーメッセージを表示します。

-C 構成番号を指定します。省略した場合は、0 です。

使 用 例 コマンド
fdsmond_11G
fdsmond_11G -C 1
fdsmond_11GR2
fdsmond_11GR2 -C 1
fdsmond_12C
fdsmond_12C -C 1

フ ァ イ ル /etc/opt/HA/AM/lbin/fdsmond_11G
/etc/opt/HA/AM/lbin/fdsmond_11GR2
/etc/opt/HA/AM/lbin/fdsmond_12C

注 意 ディスク領域監視モニタは、OSDBA(dba) グループに所属するユーザーで起動する必要があります。
ディスク領域監視モニタを単独起動する場合は、次の環境変数が設定されている必要があります。

ORACLE_SID	Oracle のインスタンス識別子
ORACLE_BASE	Oracle ディレクトリ構成の基本となるディレクトリ
ORACLE_HOME	Oracle ソフトウェアをインストールしたディレクトリ
ORA_NLS10	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリ
NLS_LANG	ディスク領域監視モニタが Oracle にアクセスする際の言語、地域およびキャラクタ・セット
LD_LIBRARY_PATH	db ユーザー環境の 64bit 共有ライブラリパス

ディスク領域監視モニタを単独起動する場合は、設定ファイル (oramond.conf) の DiskSpaceMonitor ステートメント内の MONITOR_CONTROL パラメータを NO に設定する必要があります。

statsmond

名 前 statsmond_11G, statsmond_11GR2, statsmond_12C - 統計情報採取モニタ

形 式 /etc/opt/HA/AM/lbin/statsmond_11G -i 統計情報採取モニタ ID [-C 構成番号]
 /etc/opt/HA/AM/lbin/statsmond_11GR2 -i 統計情報採取モニタ ID [-C 構成番号]
 /etc/opt/HA/AM/lbin/statsmond_12C -i 統計情報採取モニタ ID [-C 構成番号]

機能説明 統計情報採取モニタは、統計情報を採取するためのデーモンです。
 定期的に統計情報を採取し、ファイルに出力します。採取失敗時は、syslog にエラーメッセージを表示します。

-i 設定ファイル (oramond.conf) に記述した StatsMonitor ステートメントの統計情報採取モニタ ID を指定します。
 -C 構成番号を指定します。省略した場合は、0 です。

使 用 例 コマンド
 statsmond_11G -i STATSMOND
 statsmond_11G -i STATSMOND -C 1
 statsmond_11GR2 -i STATSMOND
 statsmond_11GR2 -i STATSMOND -C 1
 statsmond_12C -i STATSMOND
 statsmond_12C -i STATSMOND -C 1

フ ァ イ ル /etc/opt/HA/AM/lbin/statsmond_11G
 /etc/opt/HA/AM/lbin/statsmond_11GR2
 /etc/opt/HA/AM/lbin/statsmond_12C

注 意 統計情報採取モニタは、OSDBA(dba) グループに所属するユーザーで起動する必要があります。
 統計情報採取モニタを単独起動する場合は、次の環境変数が設定されている必要があります。

ORACLE_SID	Oracle のインスタンス識別子
ORACLE_BASE	Oracle ディレクトリ構成の基本となるディレクトリ
ORACLE_HOME	Oracle ソフトウェアをインストールしたディレクトリ
ORA_NLS10	言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリ
NLS_LANG	統計情報採取モニタが Oracle にアクセスする際の言語、地域およびキャラクタ・セット
LD_LIBRARY_PATH	db ユーザー環境の 64bit 共有ライブラリパス

統計情報採取モニタを単独起動する場合は、設定ファイルの StatsMonitor ステートメント内の MONITOR_CONTROL パラメータを NO に設定する必要があります。

crsmond

名 前 crsmond - CRS 監視モニタ

形 式 **/etc/opt/HA/AM/ibin/crsmond**

機 能 説 明 CRS 監視モニタは、crsd/ohasd の無応答障害を監視するためのプロセスです。
障害検出時に停止することで、CLUSTERPRO へ障害を通知します。

使 用 例 コマンド
crsmond

フ ァ イ ル /etc/opt/HA/AM/ibin/crsmond

注 意 CRS 監視モニタは、モニタ制御デーモン (oramond) により制御されません。そのため、設定ファイル (oramond.conf) ではなく、CRS 監視設定ファイル (crsmond.conf) を設定する必要があります。
CRS 監視モニタは、CRS 監視設定ファイル名が crsmond.conf 以外の名称では動作しません。
CRS 監視設定ファイルには、ORACLE_VERSION パラメータ、CRS_STAT パラメータを設定する必要があります。
CRS 監視モニタは、root ユーザーのみ実行できます。
同一ノード上で複数の CRS 監視モニタを起動することはできません。

索引

A

altmond	18
ApplicationMonitor 停止	6
ASM	7
ASM インスタンス停止に伴うデータベース再起動 時の監視動作	171
ASM インスタンス停止に伴うデータベース再起動 の待ち合わせ	7, 109

C

CDB	15
CLUSTERPRO 連携設定	199
CRS 監視設定ファイル	21, 120
CRS 監視モニタ	18, 233
crsd	18
crsd/ohasd の状態表示用コマンド	123
crsmond	18, 233
crsmond.conf	21, 120

D

DataGuardSite ステートメント	47
DataGuardSite ステートメントのパラメータ	73
Destination ステートメント	60
Destination ステートメントのパラメータ	98
Directory ステートメント	58
Directory ステートメントのパラメータ	96
DiskSpaceMonitor ステートメント	52
DiskSpaceMonitor ステートメントのパラメータ ..	86

E

EXEC リソースの追加	199
--------------------	-----

F

fds.conf	21
fdsmond	17

fdsmond_11G	231
fdsmond_11GR2	231
fdsmond_12C	231

H

haam.dict	21
haam_stat.dict	21
haam_stat_map.dict	21
hasd	18

I

inst.conf	21
InstanceMonitor ステートメント	49
InstanceMonitor ステートメントのパラメータ	77
instmond	15

L

ListenerMonitor ステートメント	50
ListenerMonitor ステートメントのパラメータ	80
lsnr.conf	21
lsnrmond	16, 229

N

Node ステートメント	48
Node ステートメントのパラメータ	74

O

OdgBroker ステートメント	62
OdgBroker ステートメントのパラメータ	100
oraadmin	20, 221
Oracle Clusterware の監視対象設定	187
Oracle Clusterware 監視	173
Oracle Clusterware によるインスタンス / リスナー 再起動の待ち合わせ	6
Oracle Clusterware 無応答障害の監視	5
Oracle Clusterware / Oracle Restart リトライ判定	

シェルスクリプトの動作	165
Oracle Data Guard Broker による自動フェイル オーバーの待ち合わせ	7
Oracle Data Guard Broker による自動フェイル オーバー時の監視動作	169
Oracle Data Guard Broker プロセスの状態による スタンバイ・インスタンスの監視	6
Oracle DBA 監査 (audit) 機能	105
Oracle Restart 監視	173
Oracle Restart によるインスタンス / リスナー再起 動の待ち合わせ	6
Oracle Restart の監視対象設定	187
Oracle Restart 無応答障害の監視	5
Oracle 構成プロセス停止	104
Oracle パスワード	105
Oracle ハング検知機能	106, 186
oraconfd	20, 227
oradiag	117
oramond	14, 228
oramond.conf	21, 36
oramond<構成番号>.bin	21
P	
PDB	15
PID モニタリソースの作成	203
R	
RAC 構成	2
REDO ログ適用状況によるスタンバイ・インスタン スの監視	5
REDO ログ転送状況によるスタンバイ・インスタン スの監視	5
RedoApply ステートメント	61
RedoApply ステートメントのパラメータ	99
RENS	1
RENS 連携用辞書ファイル	21
S	
SGA 情報	7

ssdump	19
ssdump.conf	21
StandbyDatabaseMonitor ステートメント	53
StandbyDatabaseMonitor ステートメントのパラ メータ	88
Stats ステートメント	63
Stats ステートメントのパラメータ	101
stats.conf	21
statsmond	19
statsmond_11G	232
statsmond_11GR2	232
statsmond_12C	232
StatsMonitor ステートメント	54
StatsMonitor ステートメントのパラメータ	90
stbdb.conf	21
stbdbmond	17
SystemStateDaemon ステートメント	55
SystemStateDaemon ステートメントのパラメータ	92
T	
Table ステートメント	56
Table ステートメントのパラメータ	94
TableSpace ステートメント	57
TableSpace ステートメントのパラメータ	95
TableSpaceMonitor ステートメント	51
TableSpaceMonitor ステートメントのパラメータ	84
Transport ステートメント	59
Transport ステートメントのパラメータ	97
tschk.conf	21
tschkmond	16
tschkmond_11G	230
tschkmond_11GR2	230
tschkmond_12C	230
あ	
アラートログ監視モニタ	18

い

一時停止	8, 213
一括設定	7
一括変更	7
インスタンス監視時の動作.....	127
インスタンス監視表	24
インスタンス監視表のスキーマ	109
インスタンス監視モニタ.....	15
インスタンス障害時の CLUSTERPRO への障害 通知.....	110
インスタンス障害の監視	4

う

運用.....	183
---------	-----

か

外部連携モニタリソースの作成.....	204
概要.....	1
各モニタ状態.....	8
各モニタ設定.....	8
監視 TAT 情報	7
監視応答時間	7
監視プロセス.....	14
監視モニタ制御	106
管理インタフェース	20
管理コマンド.....	20, 221

き

起動.....	206
共通ユーザー.....	28

く

クラスタウェアへの障害通知.....	6
クラスタ構成情報の反映.....	205

け

現用待機構成	1
--------------	---

こ

構成情報管理サーバ.....	20, 227
構成情報ファイル.....	21
コメント.....	44, 121

さ

再開	8, 213, 216
採取プロセス	19
最大監視応答時間.....	8
サイト間通信 ID	60
サイト間通信 ID	47

し

シェルスクリプト.....	6
シェルスクリプトの実行時の動作	164
システム・グローバル領域.....	7
システムステート・ダンプ	6
システムステート・ダンプ採取時の動作.....	162
システムステートダンプ採取デモン.....	19
事前準備	184
指定表監視時の障害	109
自動フェイルオーバー.....	7
種別	58
障害ノード上の ApplicationMonitor の停止動作	104
情報採取スクリプト.....	117

す

スキーマ	33
スタンバイ・インスタンス監視時の動作.....	151
スタンバイ・インスタンス監視の設定	103
スタンバイデータベース監視モニタ	17
ステートメント.....	45
ステートメントの階層	64

せ

設定	23, 190
設定確認	195
設定配布	197

設定ファイル	21, 36
設定ファイルのテンプレート作成	8
全ノードステートメント	46
全ノードステートメントのパラメータ	68

そ

ソフトウェア構成	11
----------------	----

た

単独起動	9
------------	---

て

ディザスタ・リカバリ構成	9
停止ノード選択ポリシー	104
ディスク領域監視モニタ	17, 231
ディスク領域監視モニタの単独起動	119
データベース領域の容量監視時の動作	149
データベース領域の容量不足の監視	4

と

統計情報採取時の動作	161
統計情報採取モニタ	19, 232
統計情報採取モニタ ID	54
統計情報採取モニタの単独起動	119
統計情報種別	63
統計情報の採取	7, 107
動作	125
動作中の状態確認	207
動作中の設定パラメータの調整	209
動作中の設定パラメータの変更	211
特長	3

ね

ネット・サービス名	82
-----------------	----

の

ノード名	103
------------	-----

は

パスワード・ファイル認証	106
パラメータ	44, 68, 121, 122

ひ

表名	56
表領域監視モニタ	16, 230
表領域監視モニタ ID	51
表領域監視モニタの単独起動	118
表領域名	57

ふ

ファスト・スタート・フェイルオーバー	7
フィルタリング設定	9, 112
フィルタリング設定ファイル	21, 113
フィルタリングのデフォルト設定	115
フェイルオーバーグループ停止原因の確認	219
複数インスタンス監視	9

も

モニタ制御デーモン	14, 228
-----------------	---------

り

リスナー監視時の動作	137
リスナー監視モニタ	16, 229
リスナー監視モニタの単独起動	118
リスナー再起動方法	110
リスナー障害時の CLUSTERPRO への障害通知	110
リスナー障害の監視	4
リスナーの繰り返し障害	111
リスナー名	50, 82

ろ

ローカル・ユーザー	28
ローカルディスク領域の容量監視時の動作	150
ローカルディスク領域の容量不足の監視	5
ログレベル	102

CLUSTERPRO
MC ApplicationMonitor 1.2 for Linux
ユーザーズガイド

2014 年 4 月 第 1 版
日本電気株式会社
東京都港区芝五丁目7番地1号
TEL (03) 3454-1111 (代表)



© NEC Corporation 2012-2014

日本電気株式会社の許可なく複製、改変などを行うことはできません。
本書の内容に関しては将来予告なしに変更することがあります。

保護用紙