CLUSTERPRO MC ApplicationMonitor 2.9 for Linux ユーザーズガイド

© 2024(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 2.9 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 OS に対応します。

- Red Hat Enterprise Linux 7.0～7.9, 8.1～8.9, 9.1～9.3
- Oracle Linux 7.0～7.9, 8.1～8.9, 9.1～9.3
-

（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-G02	CLUSTERPRO MC ApplicationMonitor 2.9 for Linux (1CPU ライセンス)	2.9
UL4437-G12	CLUSTERPRO MC ApplicationMonitor 2.9 for Linux (1CPU ライセンス)(他社機版)	2.9
UL4437-G22	CLUSTERPRO MC ApplicationMonitor 2.9 for Linux VM (1 ノードライセンス)	2.9
UL4437-G32	CLUSTERPRO MC ApplicationMonitor 2.9 for Linux VM (1 ノードライセンス)(他社機版)	2.9
UL4437-G01	CLUSTERPRO MC ApplicationMonitor CD 2.9	2.9

（3）本書は、次の用語を使用します。

用語	意味
インスタンス	Oracle データベース・インスタンス
ASM インスタンス	Oracle Automatic Storage Management インスタンス
ASM リスナー	Oracle Flex ASM 構成で使用されるリスナー
CDB	マルチテナント・コンテナ・データベース
PDB	プラガブル・データベース
マルチテナント構成	マルチテナント・アーキテクチャーによるマルチテナント・コンテナ・データベースの構成 PDB が一つのシングルテナント構成も含める
非マルチテナント構成	non-CDB 構成
単一リスナー	リスナーとネット・サービスが1対1で構成されたリスナー
冗長化リスナー	リスナーとネット・サービスがn対1で構成されたリスナー

また、以下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 と Java は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。
- ✓ 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 5
1.2.2. マルチテナント構成の PDB 障害の監視と障害時アクション 5
1.2.3. リスナー障害の監視と障害時アクション 5
1.2.4. 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 6
1.2.5. 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 6
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 7
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 7
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 8
1.2.9. Oracle Clusterware／Oracle Restart 障害の監視と障害時アクション 8
1.2.10. ASM インスタンス障害の監視と障害時アクション 9
1.2.11. ASM リスナー障害の監視と障害時アクション 9
1.2.12. SCANリスナー障害の監視と障害時アクション 9
1.2.13. 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 9
1.2.14. Oracle Clusterware／Oracle Restart によるインスタンス／リスナー／ASM インスタンス再起動の待ち合わせ 9
1.2.15. 障害検出時の ApplicationMonitor 停止の設定 10
1.2.16. Oracle Data Guard Broker によるファスト・スタート・フェイルオーバ（自動フェイルオーバ）の待ち合わせ 10
1.2.17. 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 10
1.2.18. 統計情報の採取 10
1.2.19. ApplicationMonitor の一括設定 11
1.2.20. 監視、採取の一時停止と再開 11
1.2.21. 監視、採取の状態および設定の表示と最大監視応答時間の表示 12
1.2.22. 設定ファイルのテンプレート作成機能 12
1.2.23. SNMPトラップを使用した統合運用管理製品との連携 12
1.2.24. 監視タイムアウトの時間帯での自動変更 12
1.2.25. 複数インスタンス監視 13
1.2.26. CLUSTERPRO へ通知する障害種別の指定 13
1.2.27. 単独監視、単独採取 13

1.2.28. Oracle エラーのフィルタリング設定 13
1.2.29. 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 13
1.2.30. 疑似障害モード 14
2. 製品のソフトウェア構成 15
2.1. 監視プロセス 21
2.1.1. モニタ制御デーモン (oramond) 21
2.1.2. インスタンス監視モニタ (instmond) 23
2.1.3. PDB 監視モニタ (pdbmond) 24
2.1.4. リスナー監視モニタ (lsnrmond) 25
2.1.5. 表領域監視モニタ (tschkmond) 26
2.1.6. ディスク領域監視モニタ (fdsmond) 27
2.1.7. スタンバイデータベース監視モニタ (stbdbmond) 28
2.1.8. アラートログ監視モニタ (altmond) 28
2.1.9. CRS 監視モニタ (crsmond) 29
2.1.10. ASM 監視モニタ (asmmond) 30
2.1.11. ASM リスナー監視モニタ (alsnrmond) 31
2.1.12. SCAN リスナー監視モニタ (slsnrmond) 32
2.2. 採取プロセス 33
2.2.1. 統計情報採取モニタ (statsmond) 33
2.2.2. システムステートダンプ採取デーモン (ssdump) 33
2.3. 管理インタフェース 34
2.3.1. 管理コマンド (oraadmin) 34
2.3.2. 構成情報管理サーバ (oraconfd) 35
2.4. 設定ファイル 35
2.4.1. 設定ファイル (oramond.conf) 35
2.4.2. 構成情報ファイル (oramond<構成番号>.bin) 35
2.4.3. CRS 監視設定ファイル (crsmond.conf) 35
2.4.4. ASM 監視設定ファイル (asmmond.conf) 35
2.4.5. ASM リスナー監視設定ファイル (alsnrmond.conf) 36
2.4.6. SCAN リスナー監視設定ファイル (slsnrmond.conf) 36
2.4.7. フィルタリング設定ファイル (inst.conf、lsnr.conf、tschk.conf、fds.conf、stbdb.conf、asm.conf、alsnr.conf、slsnr.conf、stats.conf、ssdump.conf) 36
2.5. 起動用スクリプト 37
2.5.1. ASM 監視モニタ起動スクリプト (start_asmmond.sh) 37
2.5.2. ASM リスナー監視モニタ起動スクリプト (start_alsnrmond.sh) 37
2.5.3. 単独起動モニタ用起動スクリプト (start_alone_monitor.sh) 37
3. 製品の設定 39
3.1. インスタンス監視表の作成 40

3.1.1. インスタンス監視表の作成方法（非マルチテナント構成の場合） 40
3.1.2. インスタンス監視表の作成方法（マルチテナント構成の場合） 45
3.1.3. インスタンス監視表の一括作成（非マルチテナント構成の場合） 55
3.1.4. インスタンス監視表の一括作成（マルチテナント構成の場合） 56
3.2. 設定ファイル (oramond.conf) の作成と適用 58
3.2.1. 設定ファイルの記述例 59
3.2.2. コメントの記述形式 67
3.2.3. パラメータの記述形式 67
3.2.4. ステートメントの記述形式 68
3.2.5. ステートメントの階層 89
3.2.6. パラメータとステートメントの記述順 91
3.2.7. パラメーター覧 93
3.2.8. ログレベルの設定 131
3.2.9. ノード名の設定 132
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） 132
3.2.11. 停止ノード選択ポリシーの設定（RAC 構成のみ） 133
3.2.12. 障害ノード上の ApplicationMonitor の停止動作の設定 133
3.2.13. Oracle 構成プロセス停止の設定 133
3.2.14. Oracle DBA 監査 (audit) 機能対応の設定 134
3.2.15. Oracle パスワードの扱い 134
3.2.16. パスワード・ファイル認証での接続 135
3.2.17. 監視モニタ制御の設定 135
3.2.18. Oracle ハング検知機能を利用した監視（RAC 構成のみ） 136
3.2.19. 待機中の待機イベントの確認（RAC 構成のみ） 136
3.2.20. 統計情報採取の設定 137
3.2.21. 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 139
3.2.22. インスタンス監視方法の設定 140
3.2.23. PDB 監視の設定（マルチテナント構成のみ） 142
3.2.24. 監視タイムアウトの時間帯での自動変更の設定 142
3.2.25. インスタンス監視表のスキーマの設定 143
3.2.26. PDB 障害検出時の動作の設定（マルチテナント構成のみ） 143
3.2.27. PDB クローズ検出時の動作の設定（マルチテナント構成のみ） 144
3.2.28. インスタンス／PDB／リスナー障害時の CLUSTERPRO への障害通知の設定 144
3.2.29. インスタンスの繰り返し障害の監視の設定 145
3.2.30. リスナー再起動方法の設定 146
3.2.31. リスナーの繰り返し障害の監視 146
3.2.32. リスナー監視方法の設定 147

3.2.33. 表領域監視、ローカルディスク監視のしきい値設定 148
3.3. 情報採取スクリプト (oradiag) の設定 149
3.4. PDB監視モニタの単独起動 150
3.5. リスナー監視モニタの単独起動 150
3.6. 表領域監視モニタの単独起動 151
3.7. ディスク領域監視モニタの単独起動 151
3.8. 統計情報採取モニタの単独起動 152
3.9. CRS 監視設定ファイル (crsmond.conf) の作成 153
3.9.1. CRS 監視設定ファイルの記述例 153
3.9.2. コメントの記述形式 154
3.9.3. パラメータの記述形式 154
3.9.4. パラメーター覧 155
3.9.5. Oracle Clusterware／Oracle Restart のデーモンの状態表示用コマンドの設定 157
3.9.6. 監視対象デーモンの設定（Oracle Clusterware 環境のみ） 157
3.9.7. 障害デーモン再起動の待ち合わせ動作の設定 158
3.9.8. Oracle Clusterware／Oracle Restart 障害時の CLUSTERPRO への障害通知の設定 158
3.9.9. 障害検出時のクラスタ全体での統合的判断の設定 159
3.9.10. 障害ノード上の CRS監視モニタの停止動作の設定 159
3.10. ASM 監視設定ファイル (asmmond.conf) の作成 160
3.10.1. ASM 監視設定ファイルの記述例 160
3.10.2. コメントの記述形式 161
3.10.3. パラメータの記述形式 161
3.10.4. パラメーター覧 162
3.10.5. パスワード・ファイル認証での接続 164
3.10.6. ASM インスタンス強制停止の設定 164
3.10.7. ASM インスタンス障害時の CLUSTERPRO への障害通知の設定 164
3.11. ASM 監視モニタ起動スクリプト (start_asmmond.sh) の設定 165
3.11.1. ASM 監視モニタ起動スクリプトの記述例 165
3.11.2. パラメーター覧 166
3.11.3. Oracle 環境変数一覧 166
3.12. ASM リスナー監視設定ファイル (alsnrmond.conf) の作成 167
3.12.1. ASM リスナー監視設定ファイルの記述例 168
3.12.2. コメントの記述形式 168
3.12.3. パラメータの記述形式 168
3.12.4. パラメーター覧 169
3.13. SCANリスナー監視設定ファイル (slsnrmond.conf) の作成 170
3.13.1. SCANリスナー監視設定ファイルの記述例 171

3.13.2. コメントの記述形式 171
3.13.3. パラメータの記述形式 171
3.13.4. パラメーター覧 172
3.13.5. 統計情報採取の設定 173
3.14. 単独起動モニタ用起動スクリプト (start_alone_monitor.sh) の設定 174
3.14.1. 単独起動モニタ用起動スクリプトの記述例 174
3.14.2. パラメーター覧 175
3.14.3. Oracle 環境変数一覧 175
3.15. Oracle エラーのフィルタリング設定 176
3.15.1. フィルタリング設定ファイルの記述形式 177
3.15.2. フィルタリングのデフォルト設定 179
3.16. Oracle Clusterware へのリソース登録 181
3.16.1. リソースの登録方法 181
3.16.2. リソースとして登録した監視プロセスの操作方法 188
3.16.3. リソースの削除方法 189
4. 製品の動作 195
4.1. ApplicationMonitor の起動と停止 195
4.2. インスタンス監視時の動作 197
4.3. PDB 監視時の動作 208
4.4. リスナー監視時の動作 214
4.4.1. 単一リスナーの監視 214
4.4.2. 冗長化リスナーの接続監視 226
4.4.3. 冗長化リスナーの状態監視 230
4.5. データベース領域の容量監視時の動作 236
4.6. ローカルディスク領域の容量監視時の動作 237
4.7. スタンバイ・インスタンス監視時の動作 238
4.8. Oracle Clusterware／Oracle Restart 監視時の動作 248
4.9. ASM インスタンス監視時の動作 262
4.10. ASM リスナー監視時の動作 272
4.11. SCAN リスナー監視時の動作 275
4.12. 統計情報採取時の動作 279
4.13. Oracle のシステムステート・ダンプ採取時の動作 280
4.14. シェルスクリプトの実行時の動作 282
4.15. Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトの動作 283
4.16. PDB 再オープンシェルスクリプトの動作 291
4.17. リスナー再起動シェルスクリプトの動作 292
4.18. Oracle Data Guard Broker による自動フェイルオーバ時の監視動作 297

4.19. 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 299
4.20. ライセンスチェックによる機能制限 301
5. 製品の運用 303
5.1. 事前準備 305
5.1.1. Oracle ハング検知機能の設定（RAC 構成のみ） 307
5.2. ApplicationMonitor の設定 308
5.2.1. 設定ファイル (oramond.conf) 308
5.2.2. CRS 監視設定ファイル (crsmond.conf) 314
5.2.3. ASM 監視設定ファイル (asmmond.conf) 315
5.2.4. ASM 監視モニタ起動スクリプト (start_asmmond.sh) 316
5.3. ApplicationMonitor の設定確認 317
5.4. ApplicationMonitor の設定配布 319
5.5. ApplicationMonitor と CLUSTERPRO X SSSの連携 320
5.5.1. EXEC リソースの追加 322
5.5.2. PID モニタリソースの作成 324
5.5.3. 構成情報の反映 325
5.6. ApplicationMonitor の起動確認 326
5.7. ApplicationMonitor 動作中の状態確認 327
5.8. ApplicationMonitor の設定パラメータの調整 329
5.9. ApplicationMonitor 動作中の設定パラメータの変更 331
5.10. ApplicationMonitor の一時停止と再開 334
5.10.1. 監視および採取の一時停止 334
5.10.2. 監視および採取の再開 337
5.10.3. 監視対象 PDB の除外 340
5.11. フェイルオーバグループ停止原因の確認 342
5.12. 疑似障害モードを利用した動作確認 343
5.12.1. 疑似障害モードへの移行 343
5.12.2. インスタンス監視のテスト 343
5.12.3. PDB監視のテスト 344
5.12.4. リスナー監視のテスト 344
5.12.5. 表領域監視のテスト 345
5.12.6. ローカルディスク監視のテスト 345
5.12.7. ASMインスタンス監視のテスト 346
5.12.8. Oracle Clusterware監視のテスト 346
6. コマンドリファレンス 347

1. 製品の概要と特長

1.1. 概要

◆ **現用待機構成**

ApplicationMonitor は、Oracle Database (19c、21c)（以後 Oracle と記載します）のインスタンス障害、マルチテナント構成の PDB 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、Oracle Data Guard Broker 構成のスタンバイ・インスタンス、Oracle Restart 障害、ASM インスタンス障害を監視します。

CLUSTERPRO X または CLUSTERPRO X SingleServerSafe(以降 CLUSTERPRO と記載します) 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。

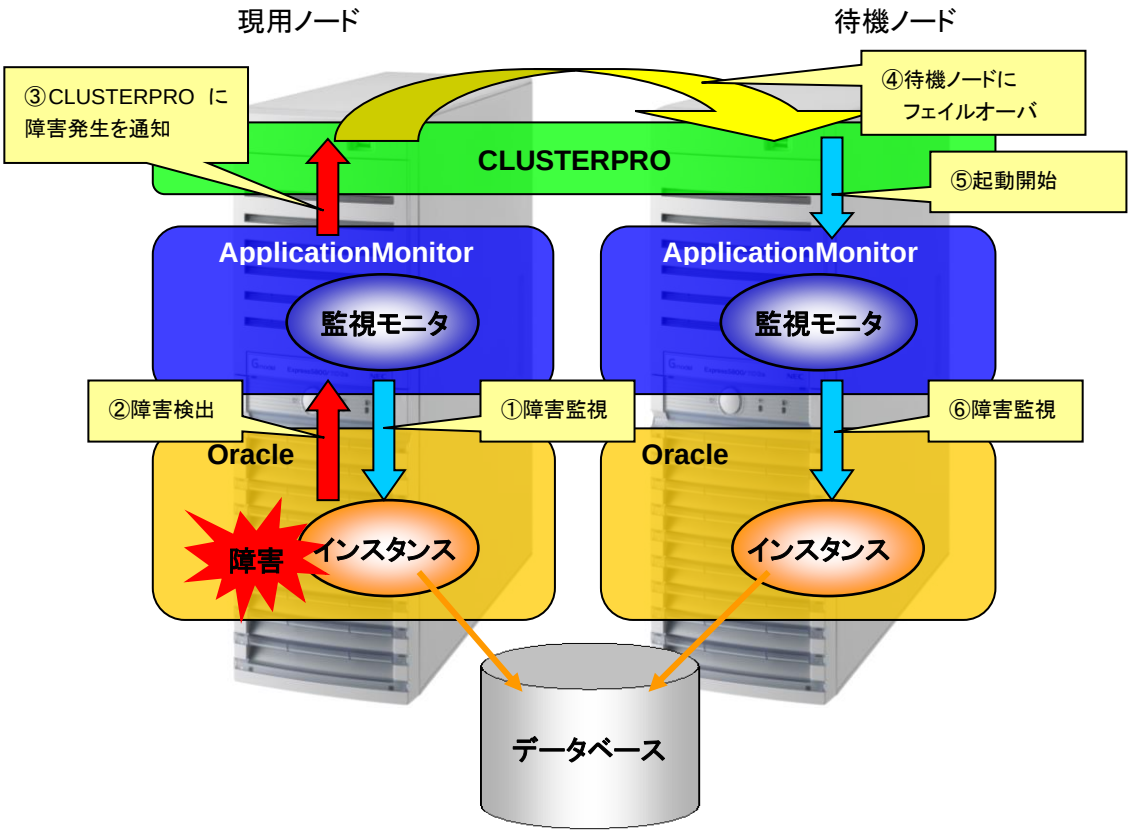


図 1-1 概要（現用待機構成）

◆ **RAC 構成**

ApplicationMonitor は、Oracle Real Application Clusters (19c、21c)（以後 RAC と記載します）のインスタンス障害、マルチテナント構成の PDB 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、Oracle Data Guard Broker 構成のスタンバイ・インスタンス、Oracle Clusterware 障害、ASM インスタンス障害、ASM リスナー障害、SCAN リスナー障害を監視します。

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。

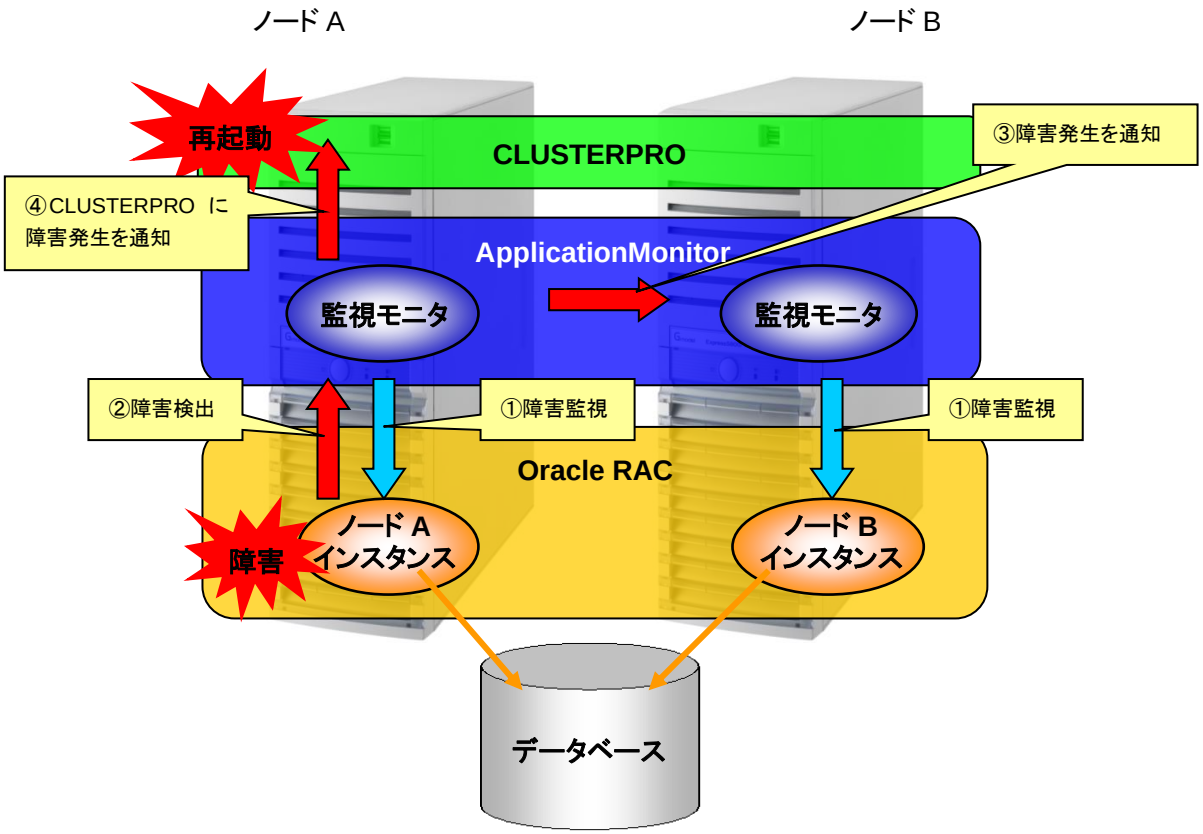


図 1-2 概要（RAC 構成）

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
- マルチテナント構成の PDB 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 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
- Oracle Clusterware／Oracle Restart 障害の監視
- ASM インスタンス障害の監視
- ASM リスナー障害の監視
- SCAN リスナー障害の監視

◆ 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
- PDB 障害検出時にコマンドを実行および PDB を再オープンまたはクローズ
- 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／ASM インスタンスの再起動を待ち合わせることが可能
- Oracle Data Guard Broker によるファスト・スタート・フェイルオーバ（自動フェイルオーバ）を待ち合わせることが可能
- 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
- 監視、採取の一時停止と再開
- 監視、採取の状態および設定の表示
- 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
- SNMP トラップを使用した統合運用管理製品との連携
- 監視タイムアウトの時間帯での自動変更
- 疑似障害モード

◆ 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
- CLUSTERPRO へ通知する障害種別を指定することが可能
- PDB 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（以後 ユーザー指定表 と記載します）を指定すると、ユーザー指定表で発生するインスタンス障害も監視します。

障害検出時、インスタンスの再確認を設定した回数行います。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、syslog にメッセージを出力します。その後、CLUSTERPRO へ障害を通知します。

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）では、RAC のノード間通信に関連するバックグラウンドプロセスを LMHB プロセスが監視し、ハングアップ状態を検知します。

RAC 構成時、ApplicationMonitor は、Oracle ハング検知機能の監視結果を障害判定に利用することができます。これにより、複数ノードで同時にストール障害を検出した際、障害の原因となっているノードを特定することが可能です。

また、Oracle ハング検知機能で検知されたハングアップ状態のプロセスが待機中の待機イベントを確認することで、より詳細なストールの判断を行うことができます。

注意: Grid Infrastructure Management Repository (GIMR) データベース（以後 MGMTDB と記載します）の監視はできません。

1.2.2. マルチテナント構成の PDB 障害の監視と障害時アクション

ApplicationMonitor は、一定時間ごとに PDB の参照／更新などを行い、PDB 障害を監視します。また、ユーザー指定表を指定すると、ユーザー指定表で発生する PDB 障害も監視します。

障害検出時、PDB の再確認を設定した回数行います。設定回数内に PDB の参照／更新が成功しない場合には、syslog にメッセージを出力します。その際に、任意のコマンドの実行および PDB を再オープンまたはクローズすることも可能です。

PDB の再オープンまたはクローズは行わずに、インスタンス障害時のアクションを実行することも可能です。

1.2.3. リスナー障害の監視と障害時アクション

ApplicationMonitor は、単一リスナーの監視では、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。

冗長化リスナーの接続監視では、一定時間ごとに冗長化されたリスナーに対するネット・サービスの有無を監視します。

冗長化リスナーの状態監視では、一定時間ごとに冗長化された各リスナーの状態を監視します。

障害検出時、単一リスナーの監視および冗長化リスナーの状態監視では、リスナーの再起動を設定した回数行います。冗長化リスナーの接続監視では、リスナーの再起動は行わずに指定した回数のリトライを試みます。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、syslog にメッセージを出力します。その後、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。

注意: Grid Infrastructure Management Repository (GIMR) リスナー（以後 MGMTLSNR と記載します）の監視はできません。

注意: 単一リスナー監視の監視方式は、旧リリースのリスナー監視と同等です。

1.2.4. 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、syslog にメッセージを出力します。その際に、任意のコマンドを実行することも可能です。

さらに、しきい値を 2 段階に指定し、コマンド実行前に警告を通知することも可能です。

1.2.5. 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や指定した ASM ディスクグループ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、syslog にメッセージを出力します。その際に、任意のコマンドを実行することも可能です。

さらに、しきい値を 2 段階に指定し、コマンド実行前に警告を通知することも可能です。

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、REDO ログの転送状況の再確認を設定した回数行います。設定回数内に状態が回復しない場合には、syslog にメッセージを出力します。その際に、任意のコマンドを実行することも可能です。

注意: 以下の構成では監視できません。

- **カスケード・スタンバイ構成**
- **Active Data Guard 遠隔同期構成**
- **非 Oracle Data Guard Broker 構成**

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、syslog にメッセージを出力します。その際に、任意のコマンドを実行することも可能です。

注意: 以下の構成では監視できません。

- **カスケード・スタンバイ構成**
- **Active Data Guard 遠隔同期構成**
- **非 Oracle Data Guard Broker 構成**

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、Oracle Data Guard Broker プロセスの状態の再確認を設定した回数行います。設定回数内に状態が回復しない場合には、syslog にメッセージを出力します。その際に、任意のコマンドを実行することも可能です。

注意: 以下の構成では監視できません。

- **カスケード・スタンバイ構成**
- **Active Data Guard 遠隔同期構成**
- **非 Oracle Data Guard Broker 構成**

1.2.9. Oracle Clusterware／Oracle Restart 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、Oracle Clusterware／Oracle Restart の状態の再確認を設定した回数行います。設定回数内に状態が回復しない場合には、syslog にメッセージを出力します。その際に、任意のコマンドを実行することも可能です。コマンド実行後は、CLUSTERPRO へ障害を通知します。

また、クラスタ内の Oracle Clusterware の状態を統合的に判断し、複数ノードで障害検出したときでも 全ノードダウンを回避できます。

1.2.10. ASM 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 ASM インスタンスの参照を行い、ASM インスタンス障害を監視します。

障害検出時、ASM インスタンスの状態の再確認を設定した回数行います。設定回数内に状態が回復しない場合には、syslog にメッセージを出力します。その際に、任意のコマンドを実行することも可能です。コマンド実行後は、CLUSTERPRO へ障害を通知します。

1.2.11. ASM 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 ASM リスナーの起動状態を確認することで、ASM リスナー障害を監視します。

障害検出時、syslog にメッセージを出力します。その後、Oracle Clusterware による ASM リスナーの再起動を待ち合わせます。

1.2.12. SCAN 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 指定したネットワーク番号を使用する SCAN リスナーの起動状態を確認することで、SCAN リスナー障害を監視します。

障害検出時、syslog にメッセージを出力します。

1.2.13. 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していますが、任意のシェルスクリプトを実行することも可能です。

1.2.14. Oracle Clusterware／Oracle Restart によるインスタンス／リスナー／ASM インスタンス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／ASM インスタンス再起動を "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. 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。

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

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

1.2.17. 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、ASM 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。

1.2.18. 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。

1.2.19. ApplicationMonitor の一括設定

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。

注意: ASM インスタンスを監視する場合は、監視するそれぞれのノードで ASM 監視設定ファイルを設定してください。

注意: ASM リスナーを監視する場合は、監視するそれぞれのノードで ASM リスナー監視設定ファイルを設定してください。

注意: SCAN リスナーを監視する場合は、監視するそれぞれのノードで SCAN リスナー監視設定ファイルを設定してください。

1.2.20. 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 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、ApplicationMonitor のプロセス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またはスイッチオーバを実施する場合は、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。
フェイルオーバまたはスイッチオーバが完了後、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。

1.2.21. 監視、採取の状態および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、監視および採取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、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させるまでの時間短縮につながります。

1.2.22. 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 2.9 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。

1.2.23. SNMP トラップを使用した統合運用管理製品との連携

ApplicationMonitor は、SNMP マネージャーを使用した様々な統合運用管理製品に障害を通知することができます。

SNMP トラップを使用した統合運用管理製品との連携機能の詳細は、別冊の『CLUSTERPRO MC LogMonitor 2.9 for Linux ユーザーズガイド (SNMP トラップ送信機能)』を参照してください。

『CLUSTERPRO MC LogMonitor 2.9 for Linux ユーザーズガイド (SNMP トラップ送信機能)』は、CD-R 媒体に /Util/mclogmon/Linux/manual/Linux_MCLOGMON_readme.pdf の名前で含まれています。

1.2.24. 監視タイムアウトの時間帯での自動変更

ApplicationMonitor では、時間帯を指定して監視タイムアウトの設定を自動で変更することができます。

データベースが高負荷状態となる夜間バッチ処理の時間帯の設定を自動で変更するなど、システムの運用に合わせた監視を行うことが可能です。

なお、以下の監視は対象外です。

- Oracle Clusterware ／ Oracle Restart の監視
- リスナー監視
- ASM リスナー監視
- SCAN リスナー監視

1.2.25. 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 の監視対象リソースとして複数登録することで実現します。

1.2.26. CLUSTERPRO へ通知する障害種別の指定

インスタンス監視、PDB 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 CLUSTERPRO 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、CLUSTERPRO 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 の監視および ASM インスタンスの監視も同様の選択が可能です。

1.2.27. 単独監視、単独採取

PDB 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および統計情報の採取は、それぞれ単独で行えます。CLUSTERPRO の監視対象リソースをインスタンス監視とは別に登録することで実現します。これにより、柔軟に構成することが可能です。

ASM リスナー障害の監視および SCAN リスナー障害の監視は、Oracle Clusterware のリソースとして登録し、常に単独で監視を行います。

注意： PDB 障害の単独での監視は、Oracle 21c の環境でのみ有効です。

1.2.28. 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 へ通知する障害を運用形態に合わせて指定できます。

1.2.29. 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。

1.2.30. 疑似障害モード

ApplicationMonitor は、oraadminコマンドの指定でDB障害検出時の動作を疑似的に実現できます。実際にDB障害を発生させずにクラスタ連携の動作確認などが安全に行えます。

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、起動用スクリプトから構成されます。

◆ 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
- PDB 監視モニタ (pdbmond) ※1、2
- 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、3
- アラートログ監視モニタ (altmond) ※1
- CRS 監視モニタ (crsmond) ※4
- ASM 監視モニタ (asmmond) ※1、4
- ASM リスナー監視モニタ (alsnrmond) ※6
- SCAN リスナー監視モニタ (slsnrmond) ※4

◆ 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) ※5
- CRS 監視設定ファイル (crsmond.conf) ※4
- ASM 監視設定ファイル (asmmond.conf) ※4
- ASM リスナー監視設定ファイル (alsnrmond.conf) ※6
- SCAN リスナー監視設定ファイル (slsnrmond.conf) ※4
- フィルタリング設定ファイル (inst.conf、lsnr.conf、tschk.conf、fds.conf、stbdb.conf、asm.conf、alsnr.conf、slsnr.conf、stats.conf、ssdump.conf)

◆ 起動用スクリプト

- ASM 監視モニタ起動スクリプト (start_asmmond.sh) ※4
- ASM リスナー監視モニタ起動スクリプト (start_alsnrmond.sh) ※6
- 単独起動モニタ用起動スクリプト (start_alone_monitor.sh)

※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、"_19C" または "_21C" が末尾に付きますが、本文中では省略しています。

※2 マルチテナント構成が対象です。

※3 Oracle Data Guard Broker 構成が対象です。

※4 Oracle Clusterware／Oracle Restart 環境が対象です。

※5 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与します。COMPONENT_ID パラメータを省略した場合は、0 となります。

※6 Oracle Flex ASM 環境が対象です。

ApplicationMonitor のソフトウェア構成を CLUSTERPRO 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。

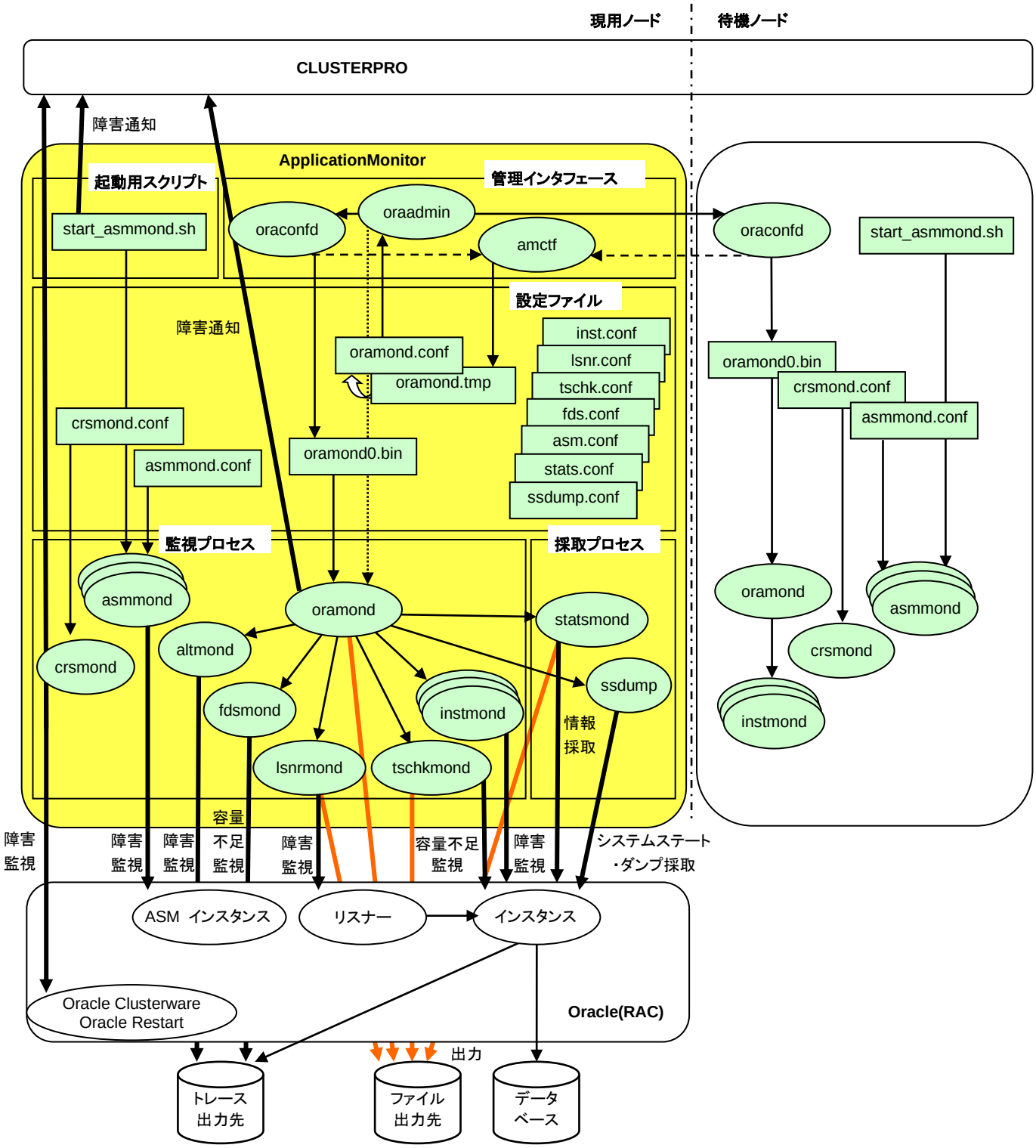


図 2-1 ソフトウェア構成

ASM リスナー監視および SCAN リスナー監視を行う場合の ApplicationMonitor のソフトウェア構成を示すと図 2-3 のようになります。

なお、ASM リスナー監視および SCAN リスナー監視に関連する監視プロセス、管理インタフェース、設定ファイルのみ記載しています。

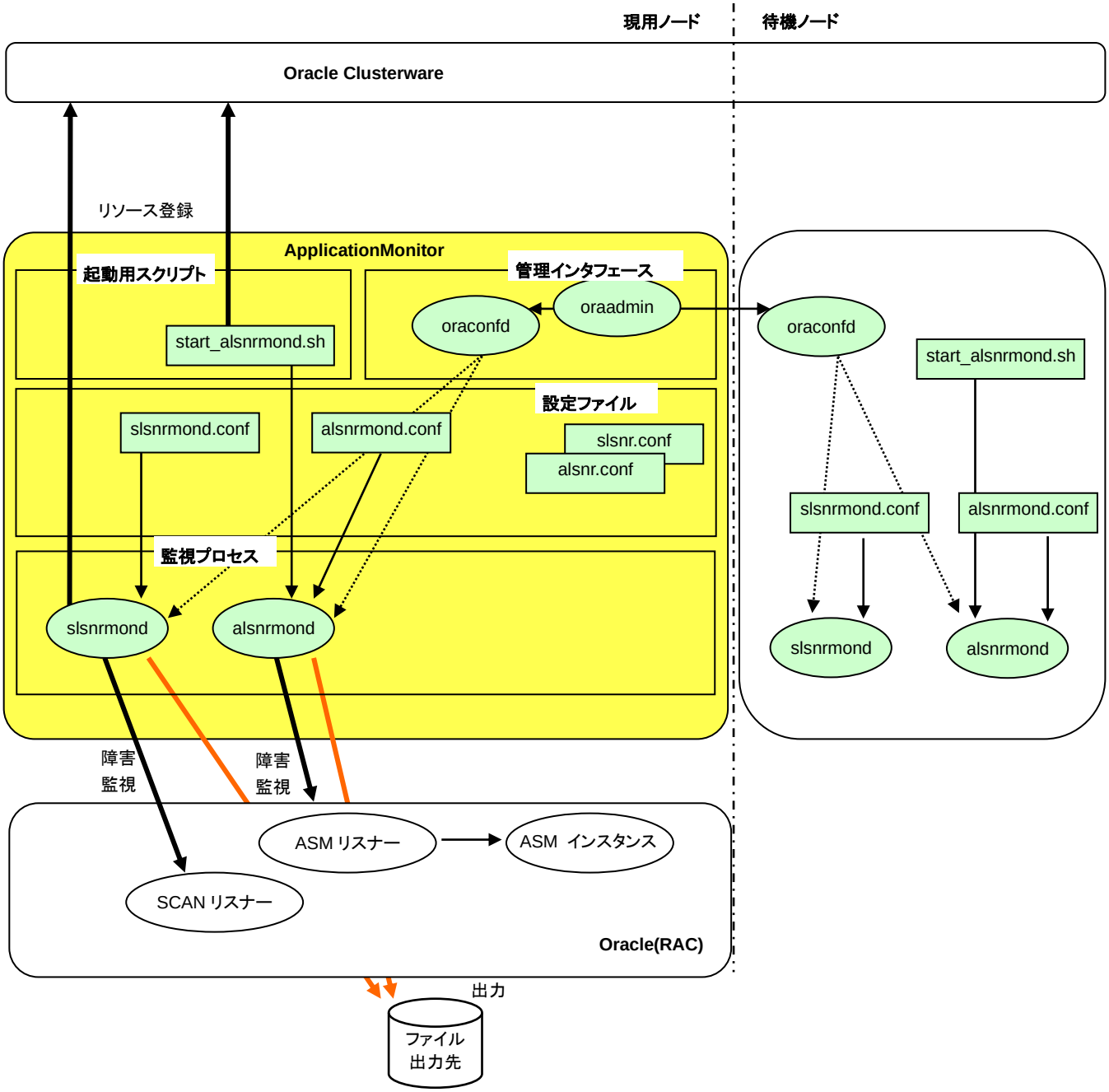


図 2-2 ソフトウェア構成（Oracle Clusterware と連携する監視）

マルチテナント構成で PDB 監視を行う場合の ApplicationMonitor のソフトウェア構成を示すと図 2-3 のようになります。

なお、PDB 監視に関連する監視プロセス、管理インタフェース、設定ファイルのみ記載しています。

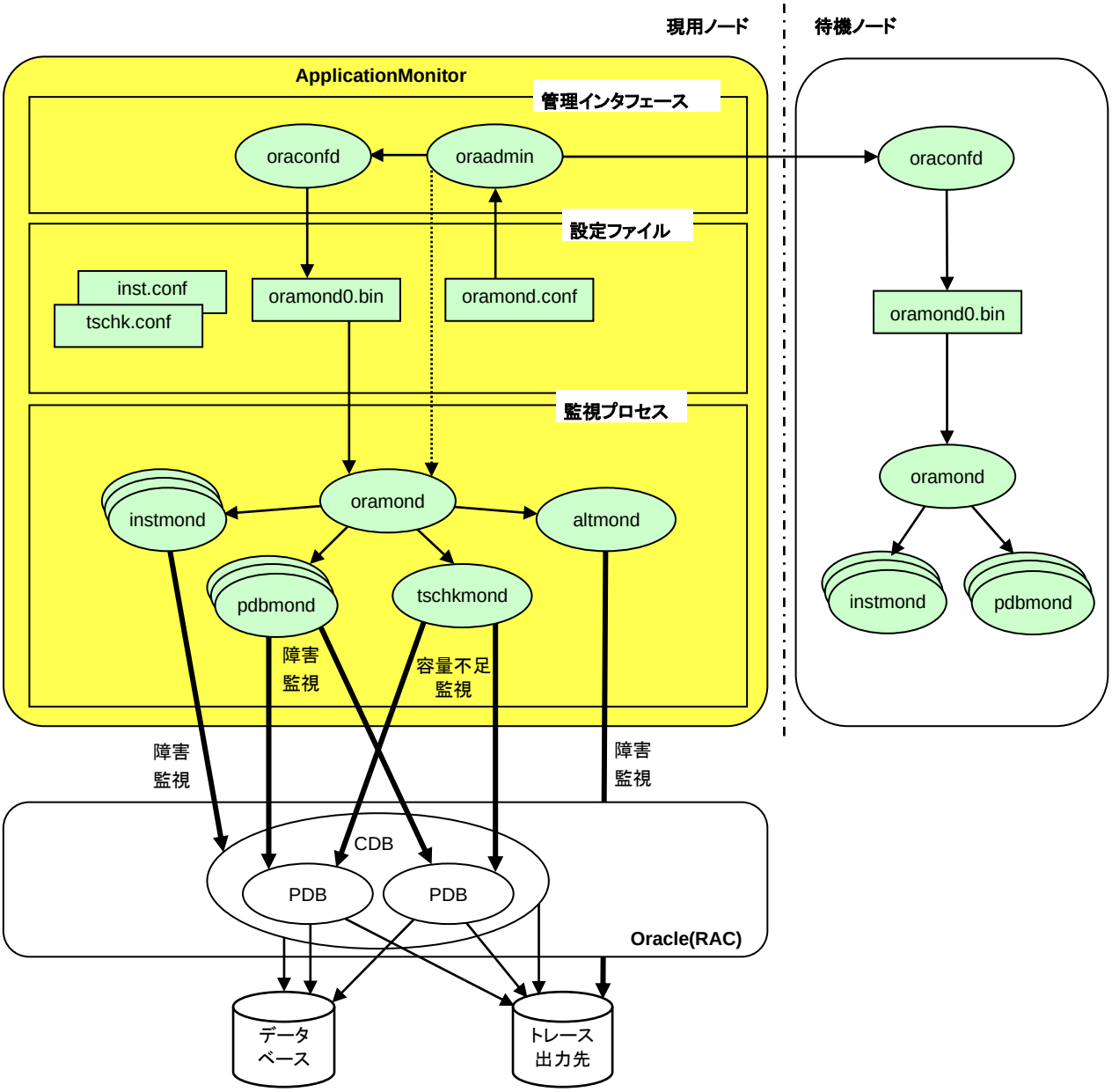


図 2-3 ソフトウェア構成
（マルチテナント構成の PDB 監視）

Oracle Data Guard Broker 構成でスタンバイ・インスタンス監視を行う場合の ApplicationMonitor のソフトウェア構成を示すと図 2-4 のようになります。

なお、スタンバイ・インスタンス監視に関連する監視プロセス、設定ファイルのみ記載しています。

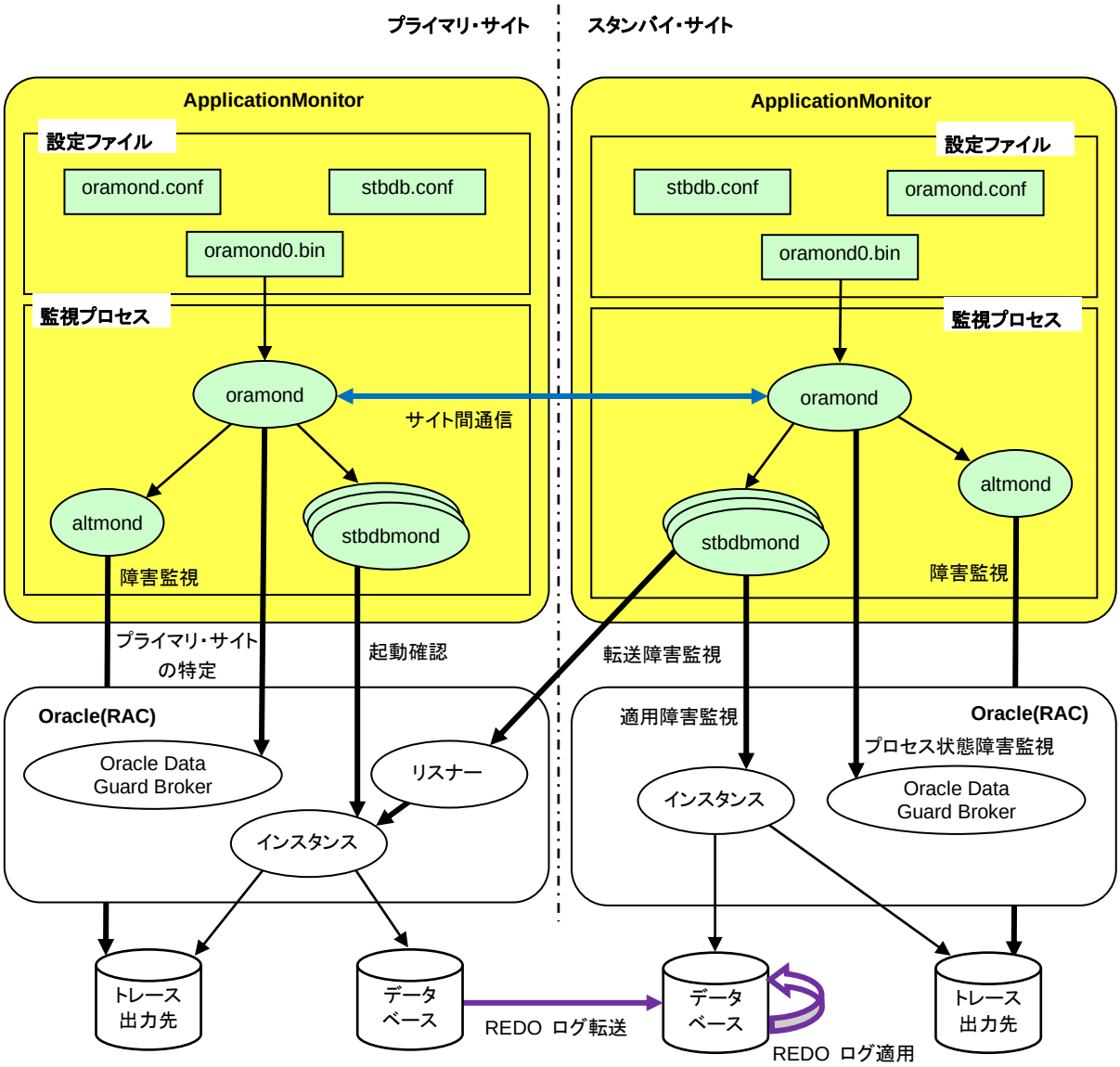


図 2-4 ソフトウェア構成（Oracle Data Guard Broker 構成の スタンバイ・インスタンス監視）

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)、PDB 監視モニタ (pdbmond)、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) 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、PDB 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。

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。

なお、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、サイト間通信を行うことで各サイト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、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、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、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、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。

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、正常な監視および採取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、監視および採取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、以下の構成では監視できません。**

- **カスケード・スタンバイ構成**
- **Active Data Guard 遠隔同期構成**
- **非 Oracle Data Guard Broker 構成**

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 V$ 表の参照、SYSTEM 表領域上の表の更新などを行い、Oracle バックグラウンドプロセス単位で障害を監視します。また、ユーザー指定表を監視対象とした場合は、表の参照を行います。

マルチテナント構成の場合は、CDB 上の V$ 表の参照、SYSTEM 表領域上の表の更新などを行い、障害を監視します。

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。

注意: MGMTDB の監視はできません。

2.1.3. PDB 監視モニタ (pdbmond)

PDB 監視モニタは、PDB 障害を監視するデーモンです。

CDB を経由して PDB へ接続し、一定時間ごとに V$ 表の参照、SYSTEM 表領域上の表の更新などを行います。また、ユーザー指定表を監視対象とした場合は、表の参照を行います。

PDB の障害検出時は、syslog にメッセージを出力し、設定ファイル (oramond.conf) に設定されたコマンドを実行します。また、障害を検出した PDB を再オープンまたはクローズすることが可能です。

PDB の再オープンまたはクローズは行わずに、モニタ制御デーモン (oramond) に障害を通知することも可能です。

注意: マルチテナント構成以外では監視できません。

注意: PDB を監視する場合には、インスタンス監視モニタ (instmond) によって CDB の監視も行うことを推奨します。

注意: 単独で PDB 監視モニタを動作させる場合、Oracle 21c の環境でのみ可能です。

注意: 単独 PDB 監視モニタ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。

2.1.4. 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。

単一リスナー監視では、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。冗長化リスナーの接続監視では、一定時間ごとに Oracle の tnsping コマンドを用いて冗長化リスナーに対するネット・サービスの有無を確認します。冗長化リスナーの状態監視では、一定時間ごとに Oracle の lsnrctl コマンドを用いて冗長化された各リスナーの状態を確認します。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、単一リスナー監視および冗長化リスナーの状態監視では、リスナーの再起動を行います。冗長化リスナーの接続監視では、リスナーの再起動は行わずにリトライを試みます。リスナーの状態が改善しない場合は、モニタ制御デーモン (oramond) に障害を通知します。

単独起動時は、リスナー監視モニタ自身を停止することで CLUSTERPRO へ障害を通知します。

なお、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: 単独リスナー監視モニタ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: MGMTLSNR の監視はできません。

2.1.5. 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。

マルチテナント構成の場合は、CDB 上および PDB 上の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。しきい値は 2 段階に設定可能です。

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。

注意: 単独表領域監視モニタ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。

2.1.6. 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、また指定した ASM ディスクグループ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。しきい値は 2 段階に設定可能です。

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 パラメータおよび 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。

注意： **単独ディスク領域監視モニタ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。

2.1.7. 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。

マルチテナント構成の場合、スタンバイ・サイトの CDB に適用された REDO ログの転送状況および適用状況を監視します。

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、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。

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、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。

注意: 以下の構成では監視できません。

- **カスケード・スタンバイ構成**
- **Active Data Guard 遠隔同期構成**
- **非 Oracle Data Guard Broker 構成**

2.1.8. 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。

2.1.9. CRS 監視モニタ (crsmond)

CRS 監視モニタは、Oracle Clusterware／Oracle Restart のデーモン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、Oracle Clusterware／Oracle Restart の正常動作を確認します。

障害検出時は、syslog にメッセージを出力します。Oracle Clusterware／Oracle Restart のデーモンプロセスが存在した場合、障害デーモンプロセスを強制終了することにより、init(8)／Oracle Clusterware による障害デーモンの再起動を促すことが可能です。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。

CRS 監視モニタ自身を停止することで CLUSTERPRO へ障害を通知します。

RAC 構成時の CRS 監視モニタは各ノード上で動作し、それぞれのノードで Oracle Clusterware 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 CRS 監視モニタ同士が情報を交換します。

複数ノードで障害が発生した場合は 障害検出したノードのうち、CRS 監視モニタ自身を停止するノードを1つ選択します。

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。**

注意: **CRS 監視モニタ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。

2.1.10. ASM 監視モニタ (asmmond)

ASM 監視モニタは、ASM インスタンス障害を監視するデーモンです。
ASM インスタンスへ接続し、一定時間ごとに V$ 表の参照を行います。

ASM インスタンスの障害検出時は、syslog にメッセージを出力します。ASM インスタンスを強制終了することにより、Oracle Clusterware／Oracle Restart による ASM インスタンスの再起動を促します。また、ASM 監視設定ファイル (asmmond.conf) に指定されたコマンドを実行します。

注意: ASM 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。ASM インスタンスを監視する場合、設定ファイル (oramond.conf) ではなく、ASM 監視設定ファイルの設定を行い、別途 ASM 監視モニタを起動してください。

注意: ASM 監視モニタ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。

2.1.11. ASM リスナー監視モニタ (alsnrmond)

ASM リスナー監視モニタは、ASM 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 lsnrctl コマンドを用いて ASM リスナーの起動状態を確認します。

ASM リスナーの障害検出時は、syslog にメッセージを出力します。ASM リスナーを強制終了することにより、Oracle Clusterware による ASM リスナーの再起動を促します。

注意: ASM リスナー監視モニタは、モニタ制御デーモン (oramond) により制御されません。ASM リスナーを監視する場合、設定ファイル (oramond.conf) ではなく ASM リスナー監視設定ファイルへの設定、Oracle Clusterware へのリソース登録を行ってください。

注意: ASM リスナー監視モニタ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。

2.1.12. SCAN リスナー監視モニタ (slsnrmond)

SCAN リスナー監視モニタは、SCAN 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 lsnrctl コマンドを用いて SCAN リスナーの起動状態を確認します。

SCAN リスナーの障害検出時は、syslog にメッセージを出力します。SCAN リスナーを強制終了することにより、Oracle Clusterware による SCAN リスナーの再起動を促します。

注意: **SCAN リスナー監視モニタは、モニタ制御デーモン (oramond) により制御されません。SCAN リスナーを監視する場合、設定ファイル (oramond.conf) ではなく SCAN リスナー監視設定ファイルへの設定、Oracle Clusterware へのリソース登録を行ってください。**

注意: **SCAN リスナー監視モニタ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。

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。

マルチテナント構成の場合は、CDB 上の 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 パラメータおよび 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。

注意: 単独統計情報採取モニタ動作中にシステム時間を変更した場合、正常な採取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、採取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。

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、新規接続ができない障害の場合にも、Oracle のシステムステート・ダンプを採取できます。

マルチテナント構成の場合は、CDB 上の 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 パラメータおよび 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)をサポートしています。

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。または、監視中の PDB を監視対象から一時的に外します。
resume	監視モニタおよび採取モニタを再開します。または、監視対象から一時的に外した PDB を監視対象に戻します。
stop	監視モニタおよび採取モニタを停止します。
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。 または、動作中の監視モニタに対する機能制限の解除を行います。
delete	構成情報ファイルを削除します。
view	構成情報ファイルの内容を表示します。
test	疑似障害モードへの移行/疑似障害モードからの復帰/疑似障害動作の指示を行います。

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、ApplicationMonitor が設定ファイル (oramond.conf) を読み込み用に変換したバイナリファイルです。管理コマンド (oraadmin) から apply リクエストにより生成し、ApplicationMonitor が動作する各ノード上に配布します。構成番号は、設定ファイルの COMPONENT_ID パラメータの番号を付与します。

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もとに動作します。

注意: CRS 監視、ASM 監視、SCAN リスナー監視では、構成情報ファイルを参照しません。

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 (crsmond) 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. ASM 監視設定ファイル (asmmond.conf)

ASM 監視設定ファイルは、ASM 監視モニタ (asmmond) 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。

ASM 監視設定ファイルの詳細は、「3.10 ASM 監視設定ファイル (asmmond.conf) の作成」を参照してください。

2.4.5. ASM リスナー監視設定ファイル (alsnrmond.conf)

ASM リスナー監視設定ファイルは、ASM リスナー監視モニタ (alsnrmond) 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。

ASM リスナー監視設定ファイルの詳細は、「3.12 ASM リスナー監視設定ファイル (alsnrmond.conf) の作成」を参照してください。

2.4.6. SCAN リスナー監視設定ファイル (slsnrmond.conf)

SCAN リスナー監視設定ファイルは、SCAN リスナー監視モニタ (slsnrmond) 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。

SCAN リスナー監視設定ファイルの詳細は、「3.13 SCANリスナー監視設定ファイル (slsnrmond.conf) の作成」を参照してください。

2.4.7. フィルタリング設定ファイル (inst.conf、lsnr.conf、tschk.conf、fds.conf、stbdb.conf、asm.conf、alsnr.conf、slsnr.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、「3.15 Oracle エラーのフィルタリング設定」を参照してください。

2.5. 起動用スクリプト

2.5.1. ASM 監視モニタ起動スクリプト (start_asmmond.sh)

ASM 監視モニタ起動スクリプトは、ASM 監視モニタ (asmmond) を起動するスクリプトです。

ASM インスタンスの障害検出によって ASM 監視モニタが停止した場合、ASM 監視モニタ起動スクリプト自身を停止することで CLUSTERPRO へ障害を通知します。

ASM 監視モニタ起動スクリプトの詳細は、「3.11 ASM 監視モニタ起動スクリプト (start_asmmond.sh) の設定」を参照してください。

2.5.2. ASM リスナー監視モニタ起動スクリプト (start_alsnrmond.sh)

ASM リスナー監視モニタ起動スクリプトは、ASM リスナー監視モニタ (alsnrmond) を起動するスクリプトです。

ASM リスナーの障害検出によって ASM リスナー監視モニタが停止した場合、Oracle Clusterware による ASM リスナーの再起動を待ち合わせます。

通常、設定を行う必要はありません。

注意: Oracle が提供する asmcmd コマンドを root ユーザーで実行できない場合は、GRID_USER パラメータに asmcmd コマンドを実行するユーザーを設定してください。

2.5.3. 単独起動モニタ用起動スクリプト (start_alone_monitor.sh)

単独起動モニタ用起動スクリプトは、PDB 監視モニタ(pdbmond)、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)を単独起動するスクリプトです。

単独起動モニタ用起動スクリプトの詳細は、「3.14 単独起動モニタ用起動スクリプト (start_alone_monitor.sh) の設定」を参照してください。

注意: 単独で PDB 監視モニタを起動する場合、Oracle 21c の環境でのみ可能です。

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
- ◆ 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
- ◆ ASM 監視設定ファイル (asmmond.conf) の作成
- ◆ ASM 監視モニタ起動スクリプト (start_asmmond.sh) の設定
- ◆ ASM リスナー監視設定ファイル (alsnrmond.conf) の作成
- ◆ SCAN リスナー監視設定ファイル（slsnrmond.conf）の作成
- ◆ Oracle エラーのフィルタリング設定
- ◆ Oracle Clusterware へのリソース登録　※1

これらの設定方法について説明します。

※1　ASM リスナー監視を行う場合、SCAN リスナー監視を行う場合、または CLUSTERPRO が使用できない環境で Oracle Clusterware と HA クラススタソフトとして連携する場合のみ必要です。

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。

「3.1.1」、「3.1.2」では スクリプト（/opt/HA/AM/bin/init/haam_init.sh）による設定方法を説明します。

「3.1.3」、「3.1.4」では スクリプト（/opt/HA/AM/bin/init/haam_init_pack.sh）による設定方法を説明します。

haam_init_pack.sh では、haam_init.sh を複数回実行することで行う監視表作成、プロファイル作成、Oracle 接続ユーザーへの権限設定などを一括で行います。

haam_init.sh/haam_init_pack.sh どちらを使用して設定することも可能です。

監視表やプロファイルを削除する場合は haam_init.sh を使う方法のみが可能です。

ApplicationMonitor に添付し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。

監視対象の Oracle の構成	インスタンス監視表の作成方法
非マルチテナント構成	3.1.1、3.1.3 章参照
マルチテナント構成	3.1.2、3.1.4 章参照

3.1.1. インスタンス監視表の作成方法（非マルチテナント構成の場合）

非マルチテナント構成で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。

シェルスクリプトは、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、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 (HAAM_PROFILE) 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。

シェルスクリプトは、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、OSDBA(dba) グループに所属する Oracle ユーザーで Oracle に接続し、監視用の Oracle 接続ユーザーを作成してください。

【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。

シェルスクリプトは、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、確認処理を自動的に実施します。

個別に確認を行う場合は、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。

シェルスクリプトは、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

シェルスクリプトの実行結果より、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

シェルスクリプトの実行結果より、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
```

6 監視表の再作成

監視表が正しく作成された場合は、実行する必要はありません。

監視表が正しく作成されていない場合は、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。

シェルスクリプトは、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。

7　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。

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。

シェルスクリプトは、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（マルチテナント構成の場合）

マルチテナント構成時、CDB 上にインスタンス監視表を作成する必要があります。
CDB の監視パターンは、以下の図のように 2 通りあります。

	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
監視パターン 1	SYS ユーザー	SYS ユーザー
監視パターン 2	共通ユーザー	SYS ユーザー

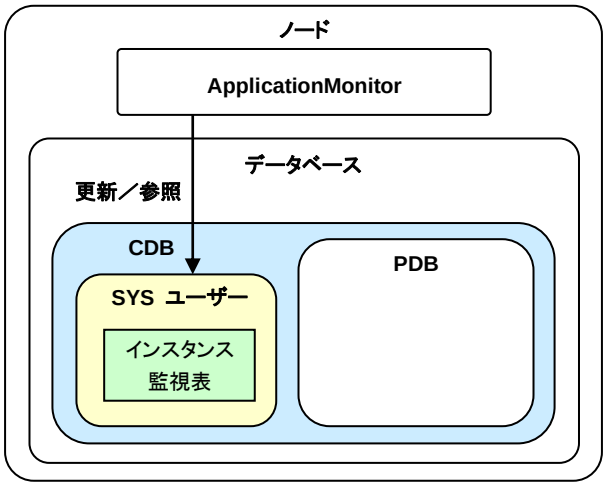


図 3-1 CDB の監視パターン 1

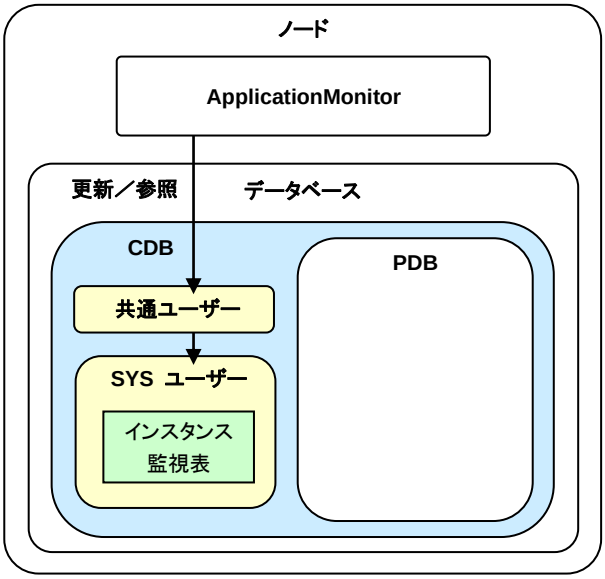


図 3-2 CDB の監視パターン 2

PDB の監視を行う場合は、PDB 上にもインスタンス監視表を作成する必要があります。
PDB の監視パターンは、以下の図のように 4 通りあります。

	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
監視パターン 1	SYS ユーザー	SYS ユーザー
監視パターン 2	SYS ユーザー	ローカル・ユーザー
監視パターン 3	共通ユーザー	SYS ユーザー
監視パターン 4	共通ユーザー	ローカル・ユーザー

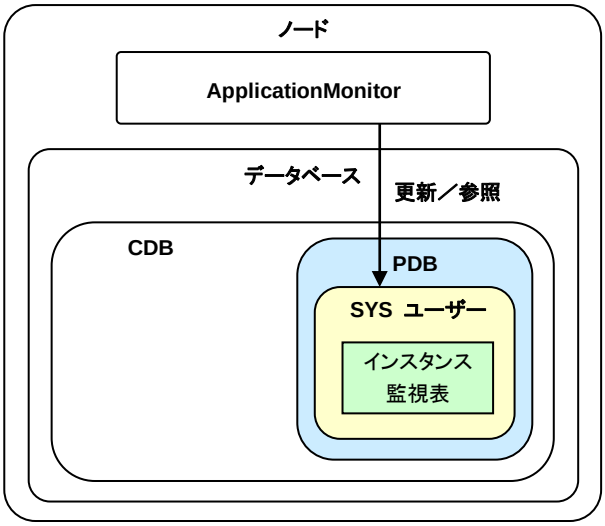


図 3-3 PDB の監視パターン 1

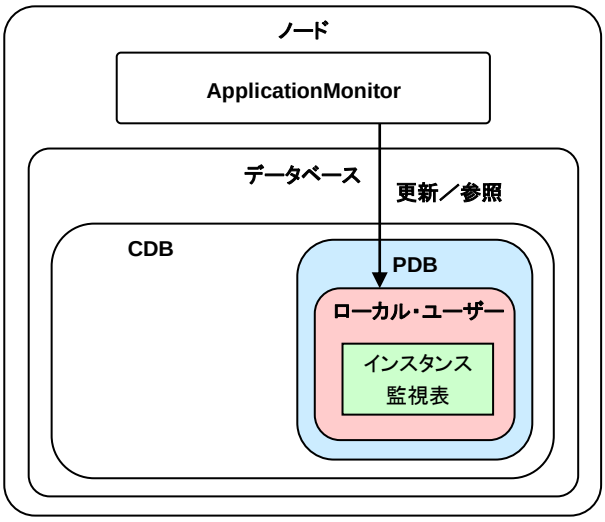


図 3-4 PDB の監視パターン 2

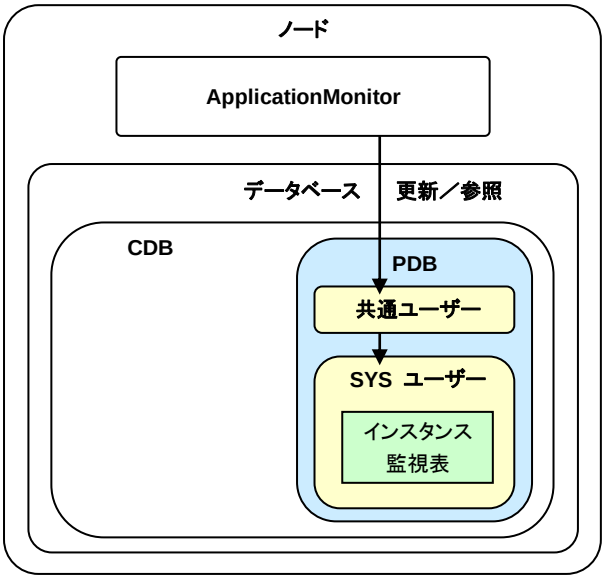


図 3-5 PDB の監視パターン 3

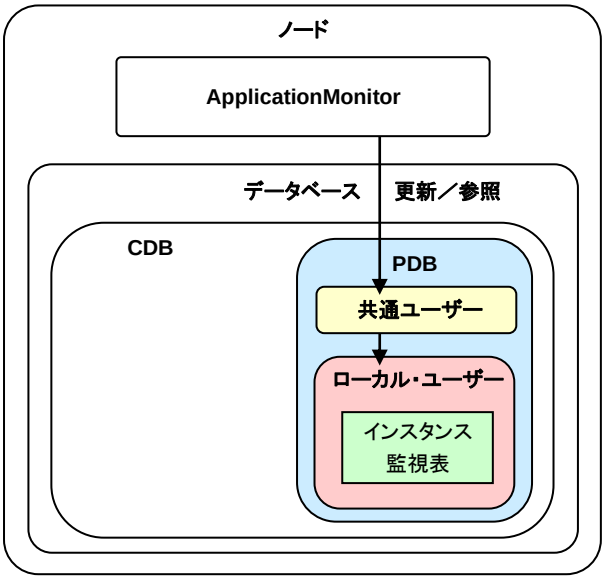


図 3-6 PDB の監視パターン 4

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

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。

Oracle の共通ユーザー名、プロファイル名の "C##" の部分は、データベースの設定により変わります。

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。

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。

シェルスクリプトは、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、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 (C##HAAM_PROFILE) 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。

シェルスクリプトは、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、OSDBA(dba) グループに所属する Oracle ユーザーで Oracle に接続し、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】 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 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。

シェルスクリプトは、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 table <XXX>
```

XXX:　監視対象ノード数を指定します。省略はできません。

PDB の監視を行う場合は、上記に引き続き、監視表を PDB 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。

シェルスクリプトは、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。省略はできません。

SYS またはローカル・ユーザーを指定してください。

ZZZ:　PDB 名を指定します。省略はできません。

【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】 上記のシェルスクリプトは、複数の PDB 名を指定できません。複数の PDB を監視対象とする場合は、PDB 名を 1 つずつ指定し、すべての PDB に対してシェルスクリプトを実行してください。

【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、確認処理を自動的に実施します。

個別に CDB 上の監視表を確認する場合は、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。

シェルスクリプトは、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 verify <XXX>
```

XXX:　監視対象ノード数を指定します。省略はできません。

個別に PDB 上の監視表を確認する場合は、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。

シェルスクリプトは、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。省略はできません。
　　SYS またはローカル・ユーザーを指定してください。

ZZZ:　PDB 名を指定します。省略はできません。

【注意】 上記のシェルスクリプトは、複数の PDB 名を指定できません。複数の PDB を監視対象とする場合は、PDB 名を 1 つずつ指定し、すべての PDB に対してシェルスクリプトを実行してください。

5.1　TABLE オブジェクトの確認

シェルスクリプトの実行結果より、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 CDB 上の SEQUENCE オブジェクトの確認

シェルスクリプトの実行結果より、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
```

6 監視表の再作成

監視表が正しくされた場合は、実行する必要はありません。

CDB 上の監視表が正しく作成されていない場合は、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 CDB 上の監視表を削除したのち、4 を実行して CDB 上の監視表を再作成してください。

シェルスクリプトは、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 drop table
```

PDB 上の監視表が正しく作成されていない場合は、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 PDB 上の監視表を削除したのち、4 を実行して PDB 上の監視表を再作成してください。

シェルスクリプトは、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。省略はできません。
SYS またはローカル・ユーザーを指定してください。

ZZZ: PDB 名を指定します。省略はできません。

【注意】 上記のシェルスクリプトは、複数の PDB 名を指定できません。複数の PDB を監視対象とする場合は、PDB 名を 1 つずつ指定し、すべての PDB に対してシェルスクリプトを実行してください。

7 権限の付与

7.1 CDB を監視する Oracle 接続ユーザー（共通ユーザー）の権限の付与

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

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。

シェルスクリプトは、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。

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

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。

【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。

7.2 PDB を監視する Oracle 接続ユーザー（共通ユーザー）の権限の付与

CDB のみ監視を行う、または SYS ユーザーで PDB の監視を行う場合は、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 PDB の監視を行う場合、3 で作成した Oracle 接続ユーザー（共通ユーザー）に、PDB の監視に必要な権限を付与します。

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。

シェルスクリプトは、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。省略はできません。
SYS またはローカル・ユーザーを指定してください。

ZZZ: PDB 名を指定します。省略はできません。

【注意】 上記のシェルスクリプトは、複数の PDB 名を指定できません。複数の PDB を監視対象とする場合は、PDB 名を 1 つずつ指定し、すべての PDB に対してシェルスクリプトを実行してください。

【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。

3.1.3. インスタンス監視表の一括作成（非マルチテナント構成の場合）

「3.1.1」で行う監視表、プロファイの作成/Oracle 接続ユーザーへの権限設定を haam_init_pack.sh を使用して一括して行います。

非マルチテナント構成で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 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。

シェルスクリプトは、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。

【注意】 Oracle 接続ユーザーを新規作成する場合、大文字で作成されます。

コマンドラインで一括作成する場合の実行例です。

```
$ /opt/HA/AM/bin/init/haam_init_pack.sh non-cdb <XXX> [<YYY[:ZZZ]>]
```

コマンド実行後対話形式で作成する場合の実行例です。

```
$ /opt/HA/AM/bin/init/haam_init_pack.sh
Please input database mode.[non-cdb|cdb|pdb]
haam_init_pack> non-cdb
Please input number of node.
haam_init_pack> XXX
Please input local user name. default[SYS]
haam_init_pack> YYY（または YYY: ZZZ）
Please input password.
haam_init_pack> 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: Oracle ローカル・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。

ZZZ: Oracle ローカル・ユーザーのパスワードを指定します。既存のユーザーを指定する場合はパスワードを省略できます。

実行結果は「3.1.1」の「5 作成した監視表の確認」に従って確認してください。

3.1.4. インスタンス監視表の一括作成（マルチテナント構成の場合）

「3.1.2」で行う監視表、プロファイの作成/Oracle 接続ユーザーへの権限設定を haam_init_pack.sh を使用して一括して行います。

マルチテナント構成で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 監視表の作成

監視表を 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。

シェルスクリプトは、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。

コマンドラインで一括作成する場合の実行例です。

```
$ /opt/HA/AM/bin/init/haam_init_pack.sh cdb <XXX> [<YYY[:ZZZ]>]
```

コマンド実行後対話形式で作成する場合の実行例です。

```
$ /opt/HA/AM/bin/init/haam_init_pack.sh
Please input database mode.[non-cdb|cdb|pdb]
haam_init_pack> cdb
Please input number of node.
haam_init_pack> XXX
Please input common user name. default[SYS]
haam_init_pack> YYY（または YYY: ZZZ）
Please input password.
haam_init_pack> 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: Oracle 接続ユーザー（共通ユーザー）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。

ZZZ: Oracle 共通ユーザーのパスワードを指定します。既存のユーザーを指定する場合はパスワードを省略できます。

PDB の監視を行う場合は、上記に引き続き、監視表を PDB 上に作成します。

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

コマンドラインで一括作成する場合の実行例です。

```
$ /opt/HA/AM/bin/init/haam_init_pack.sh pdb <XXX> <YYY> [<ZZZ>] [<VVV>]
```

コマンド実行後対話形式で作成する場合の実行例です。

```
$ /opt/HA/AM/bin/init/haam_init_pack.sh
Please input database mode.[non-cdb|cdb|pdb]
haam_init_pack> pdb
Please input number of node.
haam_init_pack> XXX
Please input common user name. default[SYS]
haam_init_pack> VVV
Please input pdb name.
haam_init_pack> YYY
Please input owner name. default[SYS]
haam_init_pack> 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: PDB 名を指定します。省略はできません。

ZZZ: 監視表を所有するユーザー名を指定します。
SYS またはローカル・ユーザーを指定してください。
ローカル・ユーザーは作成済みである必要があります。
一括作成時、第 3 引数(ZZZ)、第 4 引数(VVV)ともに SYS ユーザーを指定する場合は省略できます。
対話形式で作成する場合は SYS ユーザーを指定する場合は省略できます。

VVV: 作成する Oracle 接続ユーザー(共通ユーザー)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。

実行結果は「3.1.2」の「5 作成した監視表の確認」に従って確認してください。

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、Oracle に関する情報および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は、/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、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、必ず環境を確認した上で指定してください。

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) に変換し、各ノード上の構成情報管理サーバ (oraconfd) に送信します。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。

注意: 複数のインスタンス監視を行う場合は、監視するインスタンスごとに設定ファイルを作成する必要があります。

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。

Oracle のインスタンス障害の監視、リスナー障害の監視を現用ノードと待機ノードで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 21C ( 19C or 21C )
MONTYPE    = 21C

# Oracle 環境変数
ORACLE_BASE      = /u01/app/oracle
ORACLE_HOME      = /u01/app/oracle/product/21.3.0/dbhome_1
ORA_NLS          = /u01/app/oracle/product/21.3.0/dbhome_1/nls/data
SHLIB_PATH       = /u01/app/oracle/product/21.3.0/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
        }
}
```

2 ノードで構成されたクラスタ環境で、RAC のインスタンス障害の監視、リスナー障害の監視を行う場合は、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     = 1

# モニタタイプ 21C (19C or 21C )
MONTYPE     = 21C

# Oracle 環境変数
ORACLE_BASE      = /u01/app/oracle
ORACLE_HOME      = /u01/app/oracle/product/21.3.0/dbhome_1
ORA_NLS          = /u01/app/oracle/product/21.3.0/dbhome_1/nls/data
SHLIB_PATH       = /u01/app/oracle/product/21.3.0/dbhome_1/lib
NLS_LANG         = japanese_japan.ja16euc
GRID_BASE        = /u01/app/grid
GRID_HOME        = /u01/app/21.3.0/grid
GRID_USER        = grid

# Oracle ハング検知機能設定
ORACLE_HANG_CHECK     = YES
CHECK_WAIT_EVENT      = YES

# ノード設定
Node node1 {
        ORACLE_SID     = sid1

        InstanceMonitor {
                OWNER_NAME     = SYS
        }

        ListenerMonitor LISTENER1 {
                MONITOR_CONTROL     = NO
                NET_SERVICE_NAME    = LSNR1
        }
}

# ノード設定
Node node2 {
        ORACLE_SID     = sid2

        InstanceMonitor {
                OWNER_NAME     = SYS
        }

        ListenerMonitor LISTENER2 {
                MONITOR_CONTROL     = NO
                NET_SERVICE_NAME    = LSNR2
        }
}
```

2 ノードで構成されたクラスタ環境で、PDB 障害の監視、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 21C (19C or 21C )
MONTYPE     = 21C

# Oracle 環境変数
ORACLE_BASE      = /u01/app/oracle
ORACLE_HOME      = /u01/app/oracle/product/21.3.0/dbhome_1
ORA_NLS          = /u01/app/oracle/product/21.3.0/dbhome_1/nls/data
SHLIB_PATH       = /u01/app/oracle/product/21.3.0/dbhome_1/lib
NLS_LANG         = japanese_japan.ja16euc
GRID_BASE        = /u01/app/grid
GRID_HOME        = /u01/app/21.3.0/grid
GRID_USER        = grid

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

# Oracle ハング検知機能設定
ORACLE_HANG_CHECK     = YES
CHECK_WAIT_EVENT      = YES

# インスタンス監視レベル
SQL_LEVEL     = UPDATE

# PDB 監視有無の設定
PLUGGABLE_DB_CHECK     = YES

# 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
}

# PDB 監視設定
PluggableDatabaseMonitor {
        LOG_LEVEL                   = 2
        MONITOR_USER                = oracle
        BOOT_TIMEOUT                = 10
        POLL_INTERVAL               = 90
        POLL_TIMEOUT                = 120
        POLL_DOWN_RETRY             = 0
        POLL_STALL_RETRY            = 0
        PDB_REOPEN_TIMEOUT          = 60

        PluggableDatabase PDB1 {
                SQL_LEVEL               = UPDATE
                OWNER_NAME              = SYS
                PDB_OVER_ACTION         = NEVER
```

```
                Table USER_TABLE1 {
                        TABLE_USER          = scott
#                       TABLE_PASSWORD      = "tiger"
                }
        }

        PluggableDatabase PDB2 {
                SQL_LEVEL               = UPDATE
                OWNER_NAME              = SYS
                PDB_OVER_ACTION         = NEVER

                Table USER_TABLE2 {
                        TABLE_USER          = scott
#                       TABLE_PASSWORD      = "tiger"
                }
        }
}

# 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
        GET_STATS               = YES
}

# 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
        GET_STATS               = 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
        POLL_INTERVAL       = 60
        POLL_TIMEOUT        = 60
        GET_DUMP            = YES
        DUMP_INTERVAL       = 30
        DUMP_COUNT          = 3
        DUMP_TIMEOUT        = 90
}

# ノード設定
Node node1 {
        ORACLE_SID      = sid1

        InstanceMonitor {
                OWNER_NAME      = SYS
                GET_STATS       = YES
```

```
        }

        TableSpaceMonitor TSCHKMOND1 {
                PDB_NAME            = PDB1

                TableSpace TABLESPACE1 {
                        USEABLE_EXTENT_BLOCK    = 0
                        FREE_SPACE              = 20
                }
        }

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

# ノード設定
Node node2 {
        ORACLE_SID     = sid2

        InstanceMonitor {
                OWNER_NAME     = SYS
                GET_STATS      = YES
        }

        TableSpaceMonitor TSCHKMOND2 {
                PDB_NAME            = PDB2

                TableSpace TABLESPACE2 {
                        USEABLE_EXTENT_BLOCK    = 0
                        FREE_SPACE              = 20
                }
        }

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

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     = 2

# モニタタイプ 21C (19C or 21C )
MONTYPE     = 21C

# Oracle 環境変数
ORACLE_BASE      = /u01/app/oracle
ORACLE_HOME      = /u01/app/oracle/product/21.3.0/dbhome_1
ORA_NLS          = /u01/app/oracle/product/21.3.0/dbhome_1/nls/data
SHLIB_PATH       = /u01/app/oracle/product/21.3.0/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
        }

        # 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     = 2

# モニタタイプ 21C (19C or 21C )
MONTYPE     = 21C

# Oracle 環境変数
ORACLE_BASE      = /u01/app/oracle
ORACLE_HOME      = /u01/app/oracle/product/21.3.0/dbhome_1
ORA_NLS          = /u01/app/oracle/product/21.3.0/dbhome_1/nls/data
SHLIB_PATH       = /u01/app/oracle/product/21.3.0/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 20 種類に分類します。

- 全ノードステートメント
- PluggableDatabaseMonitor ステートメント
- DataGuardSite ステートメント
- 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
- StatsMonitor ステートメント
- SystemStateDaemon ステートメント
- PluggableDatabase 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
- Destination ステートメント
- RedoApply ステートメント
- OdgBroker ステートメント
- Stats ステートメント

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、PluggableDatabaseMonitor 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]
[PluggableDatabaseMonitor 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]
Node 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 パラメーター覧」の全ノードステートメントを参照してください。

◆ **PluggableDatabaseMonitor ステートメント**

PluggableDatabaseMonitor ステートメントを記述することで、PDB 障害を監視する PDB 監視モニタ (pdbmond) 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:

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

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

注意: **PluggableDatabaseMonitor ステートメントを省略した場合、PDB 監視を行いません。**

注意: **PluggableDatabaseMonitor ステートメント内には、必ず PluggableDatabase ステートメントを記述してください。**

◆ **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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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 ステートメント内には、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 ステートメント内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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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＝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 ステートメント内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 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* | *リスナー監視モニタID*] {
[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。単一リスナー監視および冗長化リスナーの状態監視を行う場合に指定します。
リスナー監視モニタ ID は、ノード内でリスナー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。冗長化リスナーの接続監視を行う場合に指定します。
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* および*リスナー監視モニタID* を指定しない場合と指定する場合があり、*リスナー名* および*リスナー監視モニタ ID* を指定しない場合は、該当するノード内すべてのリスナー監視モニタの設定になります。*リスナー名* または*リスナー監視モニタID* 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 または*リスナー監視モニタID* 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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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* および*リスナー監視モニタ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 *リスナー名* および*リスナー監視モニタID* は指定できません。

注意: Node ステートメント内に *リスナー名* または*リスナー監視モニタ ID* 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、*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を起動します。
TableSpaceMonitor ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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 ステートメント内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 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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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、*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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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 Oracle のシステムステート・ダンプの採取は行われます。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。**

◆ PluggableDatabase **ステートメント**

PluggableDatabase ステートメントを記述することで、PDB 監視モニタ (pdbmond) が設定された内容で指定された PDB の障害を監視します。PluggableDatabase ステートメントには、指定した PDB を監視する際の PDB 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:

PluggableDatabase *PDB 名* {
[パラメータ名＝*パラメータ値*]
[Table ステートメント]
}

PDB 名 は、監視対象の PDB 名を記述します。半角で最大 30 文字まで指定できます。
PluggableDatabase ステートメントは、PluggableDatabaseMonitor ステートメント内にのみ記述できます。
PDB 名 を指定した PluggableDatabase ステートメントが複数ある場合は、指定した複数の PDB を監視します。
PluggableDatabase ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」の PluggableDatabase ステートメントを参照してください。

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

注意: *PDB 名* は省略できません。

注意: 重複する *PDB 名* は指定できません。

◆ **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) または PDB 監視モニタ (pdbmond) がデフォルトの監視と合わせ、ユーザー指定表を監視します。
Table ステートメントには、ユーザー指定表を監視する際のインスタンス監視モニタまたは PDB 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 (") で囲んでください。
Table ステートメントは、PluggableDatabase ステートメント内または Node ステートメント内の InstanceMonitor ステートメント内に記述できます。
表名 を指定した Table ステートメントが複数ある場合は、指定した複数のユーザー指定表を監視します。
Table ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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。**

注意: **マルチテナント構成で PDB 上のユーザー指定表を監視する場合は、PluggableDatabase ステートメント内に記述してください。**
非マルチテナント構成でユーザー指定表を監視する場合は、Node ステートメント内の InstanceMonitor ステートメント内に記述してください。

◆ **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 (") で囲んでください。
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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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、ASM 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
ASM	DISKGROUP_NAME パラメータで指定した ASM ディスクグループ

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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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(stbdbmond) 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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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 (stbdbmond) 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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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(stbdbmond) 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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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。

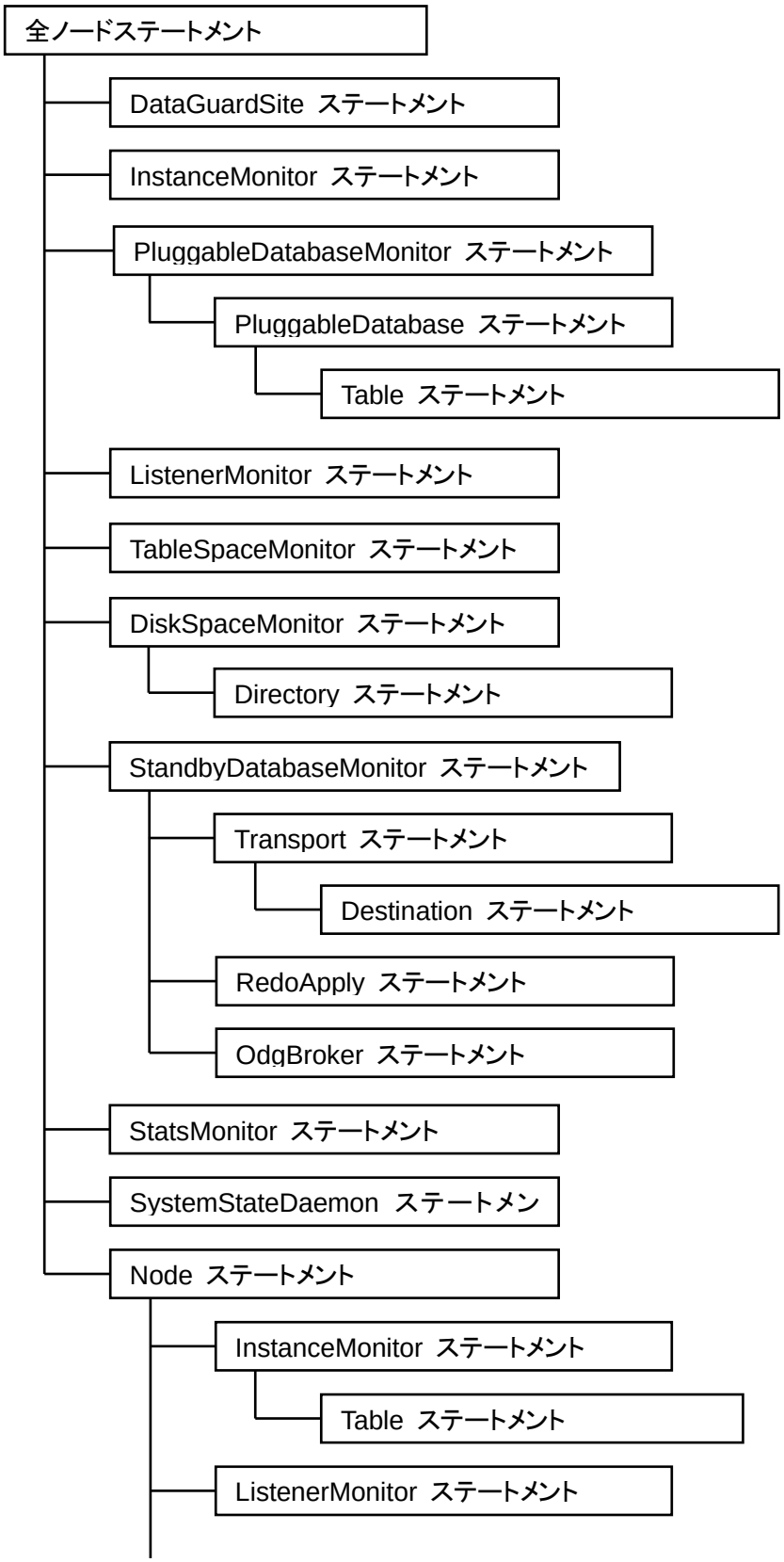
統計情報種別	採取対象
SGA_SPACE	SGA 情報

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 ステートメント内に記述できるパラメータの詳細は、「3.2.7 パラメーター覧」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。



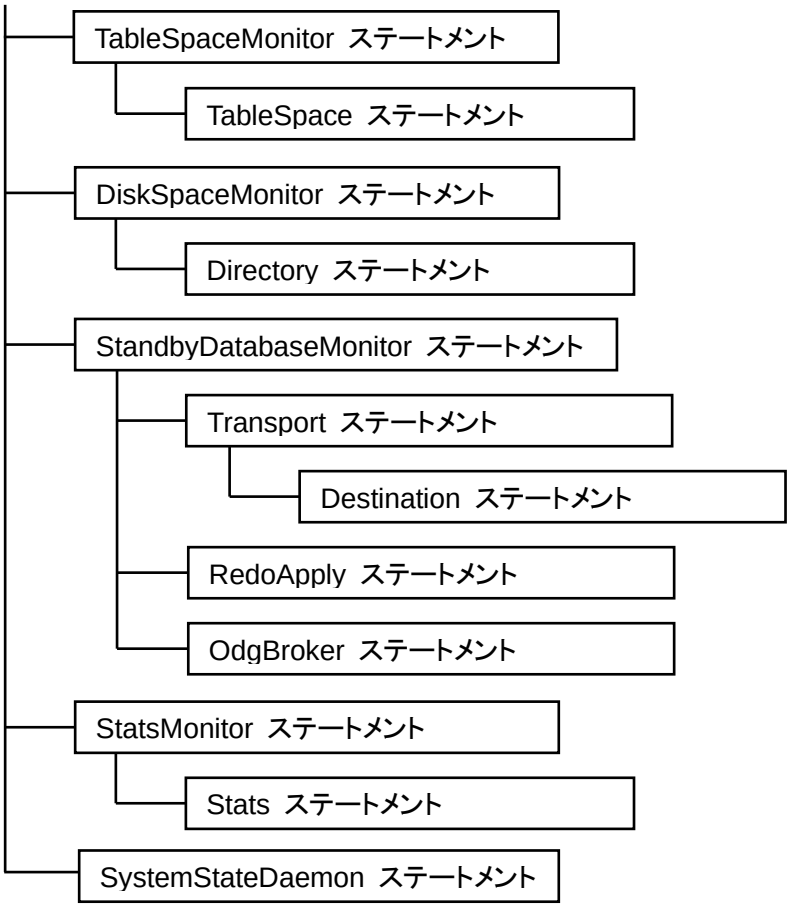


図 3-7 ステートメントの階層

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内で省略した場合、上位のステートメントで設定された値を適用します。また、下位のステートメント内で設定された場合、設定したステートメント以下のステートメントに適用します。

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
}
PluggableDatabaseMonitor ステートメント {
    PluggableDatabaseMonitor ステートメントのパラメータ
    PDB 名を指定した PluggableDatabase ステートメント {
        PluggableDatabase 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
        }
    }
}
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
            }
        }
        リスナー名およびリスナー監視モニタ ID 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
        }
        リスナー名またはリスナー監視モニタ ID 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。

3.2.7. パラメーター覧

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。

全ノードステートメント				
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, pdbmond, 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。	19C 21C		
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。 ※指定するポート番号は、OS のパラメータ ip_local_reserved_ports に設定することを推奨します。		△	×
	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。 ※指定するポート番号は、OS のパラメータ ip_local_reserved_ports に設定することを推奨します。		△	×
	25312	1024～65535		

全ノードステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
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。 ※指定するポート番号は、OS のパラメータ ip_local_reserved_ports に設定することを推奨します。		△	○
	25311	1024～65535		
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	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の処理が完了しない場合、システム障害として扱います。		○	○
	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（絶対パス）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 ステートメント内で記述する必要があります。		○	×
	なし	なし		

全ノードステートメント				
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_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(絶対パス)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 ステートメント内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(絶対パス)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 ステートメント内で記述する必要があります。		○	×
	なし	なし		
SHLIB_PATH	db ユーザー環境の 64bit 共有ライブラリのパス(絶対パス)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 ステートメント内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, pdbmond, lsnrmond, tschkmond, fdsmond, stbdbmond, 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(絶対パス)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(絶対パス)を指定してください。 ※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(絶対パス)を指定してください。 ※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 (crs_control.sh) の実効ユーザー名を指定します。 ※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		

全ノードステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
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, pdbmond, lsnrmond, tschkmond, fdsmond, stbdbmond, 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 (crs_control.sh) 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, pdbmond, lsnrmond, tschkmond, fdsmond, stbdbmond, 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います。		○	○
	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, pdbmond, tschkmond, fdsmond, stbdbmond, 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		
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, pdbmond, tschkmond, fdsmond, stbdbmond, 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。		○	○
	指定なし	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しません。 ※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。		○	○
	NO	YES NO		
MONITOR_CONTROL	モニタ制御デーモン (oramond) から監視モニタ (pdbmond, 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	YES NO		

全ノードステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
SCRIPT_USER	障害検出時に実行するシェルスクリプト（SCRIPT_NAME パラメータに設定したシェルスクリプト）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（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。		○	○
	指定なし	なし		
SCRIPT_TIMEOUT	障害検出時に実行するシェルスクリプト（SCRIPT_NAME パラメータに設定したシェルスクリプト）の実行時間の上限を秒で指定します。		○	○
	30 秒	1～3600		
ORACLE_HANG_CHECK	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。 ※RAC 構成の環境で指定が有効なパラメータです。 ※詳細は、「3.2.18 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		
CHECK_WAIT_EVENT	Oracle ハング検知機能によって検知されたハングアップ状態のプロセスが待機中の待機イベントの同一性を確認するかどうかを指定します。 ※RAC 構成の環境で指定が有効なパラメータです。 ※詳細は、「3.2.19 待機中の待機イベントの確認（RAC 構成のみ）」を参照してください。		○	○
	YES	YES NO		
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 (crs_control.sh) の実行時間の上限を秒で指定します。 ※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, tschkmond) 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、「3.2.20 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、「3.2.21 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		
SQL_LEVEL	監視モニタ (instmond, pdbmond) の監視方法を指定します。 ※詳細は、「3.2.22 インスタンス監視方法の設定」を参照してください。		○	○
	UPDATE	UPDATE SELECT NONE		

全ノードステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
PLUGGABLE_DB_CHECK	PDB 監視モニタ (pdbmond) が PDB を監視するかどうかを指定します。 ※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。 ※このパラメータは、PDB 監視モニタ (pdbmond) の設定に合わせて、ApplicationMonitor が自動的に値を調整する場合があります。 ※このパラメータは、すべての PDB 監視モニタ (pdbmond) の設定に共通です。 ※詳細は、「3.2.23 PDB 監視の設定（マルチテナント構成のみ）」を参照してください。		○	○
	YES	YES NO		
START_CHANGE_TIMEOUT	監視タイムアウトの自動変更を開始する時刻を 24 時間表記の”時”で指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」を参照してください。		○	○
	指定なし	0～23		
END_CHANGE_TIMEOUT	監視タイムアウトの自動変更を終了する時刻を 24 時間表記の”時”で指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」を参照してください。		○	○
	指定なし	0～23		
CHANGE_TIMEOUT_RATIO	監視タイムアウトを自動変更する際の倍率を指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」を参照してください。		○	○
	1	1～10		
CHANGE_TIMEOUT_LIMIT	監視タイムアウトの自動変更中に監視タイムアウトが発生した場合に、無応答障害として扱うかどうかを指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」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。

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

PluggableDatabaseMonitor ステートメント				
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	PDB 監視モニタ (pdbmond) のログの出力レベルを全 PDB 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。		○	○
	上位のステートメントに指定した値	0～3		
MONITOR_USER	PDB 監視モニタ (pdbmond) の実効ユーザー名を全 PDB 共通に指定します。		○	×
	上位のステートメントに指定した値	なし		
BOOT_TIMEOUT	PDB 監視モニタ (pdbmond) の起動を確認するまでの待ち時間を秒で全 PDB 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います。		○	○
	上位のステートメントに指定した値	1～3600		
ORACLE_USER	PDB 監視モニタ (pdbmond) が Oracle に接続するための Oracle ユーザー名を全 PDB 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値	なし		
ORACLE_PASS	PDB 監視モニタ (pdbmond) が Oracle に接続するための Oracle ユーザーのパスワードを全 PDB 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値	なし		
SYSDBA_CONNECT	ORACLE_USER パラメータを指定した際、PDB 監視モニタ (pdbmond) が SYSDBA 権限で接続するかどうかを全 PDB 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値	YES NO		
SQL_LEVEL	PDB 監視モニタ (pdbmond) の監視方法を全 PDB 共通に指定します。 ※詳細は、「3.2.22 インスタンス監視方法の設定」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	UPDATE SELECT NONE		
OWNER_NAME	PDB 上に作成したインスタンス監視表のスキーマを全 PDB 共通に指定します。 ※このパラメータを PluggableDatabaseMonitor ステートメント内で省略した場合、すべての PluggableDatabase ステートメント内で記述する必要があります。 ※詳細は、「3.2.25 インスタンス監視表のスキーマの設定」を参照してください。		○	○
	なし	なし		
POLL_INTERVAL	PDB を監視する間隔を秒で全 PDB 共通に指定します。		○	○
	90 秒	5～2678400		
POLL_TIMEOUT	PDB のストールと判断するまでの上限時間を秒で全 PDB 共通に指定します。		○	○
	120 秒	1～86400		
POLL_DOWN_RETRY	PDB のダウン検出時、PDB を再確認する回数を全 PDB 共通に指定します。		○	○
	0 回	0～100		

PluggableDatabaseMonitor ステートメント				
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_STALL_RETRY	PDB のストール検出時、PDB を再確認する回数を全 PDB 共通に指定します。		○	○
	0 回	0～100		
SERVICE_DOWN	PDB の障害時、CLUSTERPRO へ障害を通知するかどうかを指定します。 ※詳細は、「3.2.28 インスタンス／PDB／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		
PDB_OVER_ACTION	PDB の障害検出時、PDB 監視モニタ (pdbmond) の動作を全 PDB 共通に指定します。 ※詳細は、「3.2.26 PDB 障害検出時の動作の設定（マルチテナント構成のみ）」を参照してください。		○	○
	NEVER	NEVER REOPEN CLOSE		
PDB_REOPEN_TIMEOUT	障害を検出した PDB の再オープンを行う場合、PDB 再オープンシェルスクリプト (am_pdb_reopen.sh) の実行時間の上限を秒で全 PDB 共通に指定します。		○	○
	60 秒	1～3600		
COMMAND_NAME	PDB 障害検出時に実行するコマンドを絶対パスで全 PDB 共通に指定します。 ※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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	指定なし	なし		
UNOPEN_PDB_DOWN	PDB の CLOSE 状態を障害として扱うかどうかを指定します。 ※詳細は、「3.2.27 PDB クローズ検出時の動作の設定（マルチテナント構成のみ）」を参照してください。		○	○
	NO	YES NO		
PDB_OPEN_TIMEOUT	PDB のオープン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。PDB 監視モニタ (pdbmond) 起動時、このパラメータに指定された秒数を経過した際、監視を開始します。		○	○
	60 秒	0～3600		
MONITOR_CONTROL	モニタ制御デーモン (oramond) から PDB 監視モニタ（pdbmond）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。		○	○
	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。

※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 ステートメント内に記述可能なパラメータは、以下のとおりです。

DataGuardSite ステートメント				
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。

Node ステートメント内に記述可能なパラメータは、以下のとおりです。

Node ステートメント				
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(絶対パス)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を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(絶対パス)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(絶対パス)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(絶対パス)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, fdsmond, stbdbmond, 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(絶対パス)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, stbdbmond, 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 (crs_control.sh) 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, stbdbmond, 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います。		○	○
	上位のステートメントに指定した値	1～3600		

Node ステートメント				
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 (instmond, tschkmond, fdsmond, stbdbmond, 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 (instmond, tschkmond, fdsmond, stbdbmond, 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しません。 ※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		
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(SCRIPT_NAME パラメータに設定したシェルスクリプト)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		
SCRIPT_NAME	障害検出時に実行するシェルスクリプトのパス(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 (instmond, lsnrmond, tschkmond) 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ノードごとに指定します。 ※詳細は、「3.2.20 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		
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、「3.2.21 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		

Node ステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
START_CHANGE_TIMEOUT	監視タイムアウトの自動変更を開始する時刻を 24 時間表記の”時”で指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」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	0～23		
END_CHANGE_TIMEOUT	監視タイムアウトの自動変更を終了する時刻を 24 時間表記の”時”で指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」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	0～23		
CHANGE_TIMEOUT_RATIO	監視タイムアウトを自動変更する際の倍率を指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」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	1～10		
CHANGE_TIMEOUT_LIMIT	監視タイムアウトの自動変更中に監視タイムアウトが発生した場合に、無応答障害として扱うかどうかを指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」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 ステートメント内に記述可能なパラメータは、以下のとおりです。

InstanceMonitor ステートメント				
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います。		○	○
	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		
OWNER_NAME	インスタンス監視表のスキーマを指定します。 ※詳細は、「3.2.25 インスタンス監視表のスキーマの設定」を参照してください。		×	○
	省略できません。	なし		
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、ApplicationMonitor が自動的に値を調整する場合があります。		○	○
	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		

InstanceMonitor ステートメント				
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_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	0～100		
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	0～100		
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、「3.2.28 インスタンス／PDB／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 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 Clusterware／Oracle Restart 環境で指定が有効なパラメータです。		○	○
	60 秒	0～3600		
UP_DOWN_RETRY	インスタンスを繰り返し障害と判断する、監視結果エラーの累積回数を指定します。 ※詳細は、「3.2.29 インスタンスの繰り返し障害の監視の設定」を参照してください。		○	○
	5 回	0～10		
UPTIME_COUNT	インスタンスを安定稼働状態と判断する、監視結果正常の連続回数を指定します。 ※詳細は、「3.2.29 インスタンスの繰り返し障害の監視の設定」を参照してください。		○	○
	1 回	1～60		
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、「3.2.20 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。

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

ListenerMonitor ステートメント				
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	リスナー監視モニタ (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	リスナー監視モニタ (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 (crs_control.sh) 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	リスナー監視モニタ (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います。		○	○
	上位のステートメントに指定した値	1～3600		
MONITOR_CONTROL	モニタ制御デーモン (oramond) からリスナー監視モニタ (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		
LSNR_CHECK_MODE	リスナー監視モニタ (lsnrmond) の監視方式を指定します。 ※詳細は、「3.2.32 リスナー監視方法の設定」を参照してください。		○	×
	PING	PING PING_NO_RESTART STATUS		
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、「3.2.28 インスタンス／PDB／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		
RETRY_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。 ※冗長化リスナーの接続監視で指定が有効なパラメータです。		○	○
	5 回	0～999		
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。 ※単一リスナー監視および冗長化リスナーの状態監視で指定が有効なパラメータです。		○	○
	5 回	0～999		
RESTART_INTERVAL	リスナーの障害時、リスナーの再起動を行うまでの待ち合わせ時間を秒で指定します。 ※単一リスナー監視および冗長化リスナーの状態監視で指定が有効なパラメータです。		○	○
	3 秒	1～10		

ListenerMonitor ステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
NET_SERVICE_NAME	リスナーのネット・サービス名を指定します。 ※単一リスナー監視および冗長化リスナーの接続監視で指定が有効なパラメータです。		×	×
	ListenerMonitor ステートメントにリスナー名またはリスナー監視モニタ ID を指定し、単一リスナーの監視または冗長化リスナーの接続監視を行う場合は、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。 ※単一リスナー監視および冗長化リスナーの接続監視で指定が有効なパラメータです。		○	○
	${ORACLE_HOME}/bin/tnsping	なし		
LSNR_CMD_TIMEOUT	ApplicationMonitor によるリスナー再起動を行う場合、リスナー再起動シェルスクリプト (lsnr_control.sh) の実行時間の上限を秒で指定します。 ※単一リスナー監視および冗長化リスナーの状態監視で指定が有効なパラメータです。		○	○
	90 秒	1～3600		
RESTART_TIMEOUT	障害を検出したリスナーを再起動するためにリスナーに対して kill コマンド実行後、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。 ※ApplicationMonitor による再起動を行う場合、リスナー再起動後にインスタンスがリスナーにエンドポイントを登録するまでに最大 60 秒かかります。 また、Oracle Clusterware／Oracle Restart による再起動を行う場合、Oracle Clusterware／Oracle Restart のデフォルト設定でリスナー再起動までに最大 60 秒かかります。 そのためリスナーを動的構成にした場合、このパラメータは、60 秒以上に設定することを推奨します。 ※単一リスナー監視および冗長化リスナーの状態監視で指定が有効なパラメータです。		○	○
	60 秒	0～3600		
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、「3.2.30 リスナー再起動方法の設定」を参照してください。 ※単一リスナー監視および冗長化リスナーの状態監視で指定が有効なパラメータです。		○	○
	YES	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（絶対パス）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（絶対パス）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		

ListenerMonitor ステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
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（絶対パス）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（絶対パス）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	リスナー監視モニタ (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（絶対パス）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	リスナーを繰り返し障害と判断する、監視結果異常の累積回数を指定します。 ※詳細は、「3.2.31 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	リスナーを安定稼動状態と判断する、監視結果正常の連続回数を指定します。 ※詳細は、「3.2.31 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	リスナー監視モニタ (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、「3.2.20 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。

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またはリスナー監視モニタ ID、およびリスナー監視モニタ (lsnrmond) 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: 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。

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。

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

TableSpaceMonitor ステートメント				
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います。		○	○
	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。 ※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。		○	○
	指定なし	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		

TableSpaceMonitor ステートメント				
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	表領域の最大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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	指定なし	なし		
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、「3.2.20 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 (tschkmond) 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。

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

DiskSpaceMonitor ステートメント				
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います。		○	○
	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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	指定なし	なし		

DiskSpaceMonitor ステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
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		
FREE_SPACE_WARNING	警告を出力する空き容量率のしきい値を割合で指定します。0を指定した場合、警告を出力しません。		○	○
	0 %	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。

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

StandbyDatabaseMonitor ステートメント				
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 ステートメント内の 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います。		○	○
	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しません。 ※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		
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		

StandbyDatabaseMonitor ステートメント				
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_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 (stbdbmond) 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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	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。

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

StatsMonitor ステートメント				
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います。		○	○
	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		
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、「3.2.20 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 (statsmond) 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。

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

SystemStateDaemon ステートメント				
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います。		○	○
	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		
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	Oracle のシステムステート・ダンプを採取するかどうかを指定します。		○	○
	YES	YES NO		
DUMP_INTERVAL	Oracle のシステムステート・ダンプを採取する間隔を秒で指定します。		○	○
	30 秒	1～60		
DUMP_COUNT	Oracle のシステムステート・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 Oracle のシステムステート・ダンプのレベルを指定します。		○	○
	10	0～266		

SystemStateDaemon ステートメント				
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	Oracle のシステムステート・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の SystemStateDaemon ステートメント内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。

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

PluggableDatabase ステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
SQL_LEVEL	PDB 監視モニタ (pdbmond) の監視方法を指定します。 ※詳細は、「3.2.22 インスタンス監視方法の設定」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	UPDATE SELECT NONE		
OWNER_NAME	PDB 上に作成したインスタンス監視表のスキーマを指定します。 ※詳細は、「3.2.25 インスタンス監視表のスキーマの設定」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		
POLL_INTERVAL	PDB 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	5～2678400		
POLL_TIMEOUT	PDB 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		
POLL_DOWN_RETRY	PDB のダウン検出時、PDB 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	0～100		
POLL_STALL_RETRY	PDB のストール検出時、PDB 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	0～100		
SERVICE_DOWN	PDB の障害時、CLUSTERPRO へ障害を通知するかどうかを指定します。 ※詳細は、「3.2.28 インスタンス／PDB／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。		○	○
	上位のステートメントに指定した値	YES NO NO_DOWN NO_STALL		
PDB_OVER_ACTION	PDB の障害検出時、PDB 監視モニタ (pdbmond) の動作を指定します。 ※詳細は、「3.2.26 PDB 障害検出時の動作の設定（マルチテナント構成のみ）」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	NEVER REOPEN CLOSE		
PDB_REOPEN_TIMEOUT	障害を検出した PDB の再オープンを行う場合、PDB 再オープンシェルスクリプト (am_pdb_reopen.sh) の実行時間の上限を秒で指定します。		○	○
	上位のステートメントに指定した値	1～3600		
COMMAND_NAME	PDB 障害検出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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	上位のステートメントに指定した値	なし		

PluggableDatabase ステートメント				
パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
UNOPEN_PDB_DOWN	PDB の CLOSE 状態を障害として扱うかどうかを指定します。 ※詳細は、「3.2.27 PDB クローズ検出時の動作の設定（マルチテナント構成のみ）」を参照してください。		○	○
	NO	YES NO		
PDB_OPEN_TIMEOUT	PDB のオープン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。PDB 監視モニタ (pdbmond) 起動時、このパラメータに指定された秒数を経過した際、監視を開始します。		○	○
	60 秒	0～3600		
MONITOR_CONTROL	モニタ制御デーモン(oramond）から PDB 監視モニタ（pdbmond）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。		○	○
	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。

※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。

Table ステートメント内に記述可能なパラメータは、以下のとおりです。

Table ステートメント				
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。 ※PluggableDatabase ステートメント内の Table ステートメント内には指定できません。 ※詳細は、「3.2.15 Oracle パスワードの扱い」を参照してください。		×	○
	InstanceMonitor ステートメント内に 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		

※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 ステートメント内に記述可能なパラメータは、以下のとおりです。

TableSpace ステートメント				
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	表領域の最大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、最大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		
USEABLE_EBLOCK_WARNING	警告を出力する最大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、警告を出力しません。。		○	○
	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		
FREE_SPACE_WARNING	警告を出力する空き容量率のしきい値を割合で指定します。0 を指定した場合、警告を出力しません。		○	○
	0 %	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、「3.2.20 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。 ※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。

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

Directory ステートメント				
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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	指定なし	なし		
DISKGROUP_NAME	ASM ディスクグループ名を指定します。種別が ASM の場合に指定可能です。		○	○
	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		
FREE_SPACE_WARNING	警告を出力する空き容量率のしきい値を割合で指定します。0 を指定した場合、警告を出力しません。		○	○
	0 %	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 ステートメント内に記述可能なパラメータは、以下のとおりです。

Transport ステートメント				
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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	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ニタ (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。

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

Destination ステートメント				
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・サイトへ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。		×	○
	Destination ステートメントを指定した場合は、省略できません。	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。		○	○
	上位のステートメントに指定した値	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。		○	○
	上位のステートメントに指定した値	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。		○	○
	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。

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

RedoApply ステートメント				
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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	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ニタ (stbdbmond) 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ニタ (stbdbmond) 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ニタ (stbdbmond) 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。

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

OdgBroker ステートメント				
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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	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 ステートメント内に記述可能なパラメータは、以下のとおりです。

Stats ステートメント				
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、「3.2.20 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 に出力す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 で出力す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 で出力す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 で出力す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。

出力す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。

出力するメッセージの詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 2.9 for Linux メッセージ一覧』を参照してください。

注意: **StandbyDatabaseMonitor ステートメント内の LOG_LEVEL パラメータは、スタンバイデータベース監視モニタ (stbdbmond) のみの設定パラメータです。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 ステートメント内の 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 アドレスを使用します。

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。

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。

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。**
また、ノード名に紐づく IP アドレスを変更した場合は、APPLY リクエストによる構成情報ファイルの再配布を行ってください。

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を選択します。
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を ApplicationMonitor 停止ノードとして選択します。

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 として扱います。 他に ApplicationMonitor が生存しているノードが存在しない場合、NEVER として扱います。

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。

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・ファイル認証の環境で監視するには、ORACLE_USER パラメータおよび ORACLE_PASS パラメータに SYSDBA 権限が付与されたユーザー、または SYS ユーザーを設定してください。

なお、SYS ユーザーを設定した場合は、SYSDBA_CONNECT パラメータを YES に設定してください。

注意: マルチスレッド構成の場合、オペレーティング・システム認証による接続はできません。ORACLE_USER パラメータおよび ORACLE_PASS パラメータを設定し、パスワード・ファイル認証による接続を行ってください。

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 PDB 監視モニタ(pdbmond)、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、一時停止、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しないため、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、監視および採取を行います。

単独起動モニタの詳細は、「3.4 PDB監視モニタの単独起動」、「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 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。

Oracle ハング検知機能で検知されたハングアップ状態のプロセスが待機中の待機イベントを確認して、より詳細な障害判定を行う場合は、CHECK_WAIT_EVENT パラメータを指定してください。

CHECK_WAIT_EVENT パラメータの詳細は、「3.2.19 待機中の待機イベントの確認（RAC 構成のみ）」を参照してください。

3.2.19. 待機中の待機イベントの確認（RAC 構成のみ）

Oracle ハング検知機能の監視結果によって検知されたハングアップ状態のプロセスが待機中の待機イベントの同一性を確認するかどうかを設定するには、CHECK_WAIT_EVENT 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。

CHECK_WAIT_EVENT=YES	待機イベントの同一性を確認します。
CHECK_WAIT_EVENT=NO	待機イベントの同一性を確認しません。

YES に設定した場合、同一待機イベントによる待機が継続している状態を Oracle バックグラウンドプロセスのストール障害と判断します。

異なる待機イベントによる待機が継続している状態（待機イベントが切り替わりながら待機が継続している状態）は、Oracle バックグラウンドプロセスのストール障害と判断しません。

NO に設定した場合、同一待機イベントによる待機かどうかにかかわらず、待機イベントによる待機が継続している状態を Oracle バックグラウンドプロセスのストール障害と判断します。

ApplicationMonitor 2.0 以前と同一のストール判断を行う場合は、NO に設定してください。

注意： ORACLE_HANG_CHECK パラメータを YES に設定した場合に指定が有効なパラメータです。

3.2.20. 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 またはリスナー監視モニタ ID ・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 :リスナー名またはリスナー監視モニタ ID
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します。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

```
2020/3/13 08:23:40,rac1,1
2020/3/13 08:25:10,rac1,1
2020/3/13 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,リスナー名またはリスナー監視モニタ ID,監視 TAT

出力イメージ

```
2020/3/13 08:23:40,LISTENER,1
2020/3/13 08:24:40,LISTENER,1
2020/3/13 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

```
2020/3/13 08:23:40,USERS,152,1600,9
2020/3/13 09:23:40,USERS,152,1600,9
2020/3/13 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

```
2020/3/13 08:23:40, SGA_SPACE,345529792.00,603979776.00,57.00
2020/3/13 09:23:40, SGA_SPACE,345529792.00,603979776.00,57.00
2020/3/13 10:23:40, SGA_SPACE,345529792.00,603979776.00,57.00
```

3.2.21. 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。

3.2.22. インスタンス監視方法の設定

インスタンス監視時、SYSTEM 表領域上のオブジェクトに対する監視方法を設定するには、SQL_LEVEL パラメータを指定します。

◆ **非マルチテナント構成の場合**

非マルチテナント構成の場合、インスタンスの監視方法は、全ノードステートメントの SQL_LEVEL パラメータに指定します。値ごとの監視方法は、以下のとおりです。

SQL_LEVEL=UPDATE	V$ 表の参照、SYSTEM 表領域上のオブジェクトの更新および参照を行います。
SQL_LEVEL=SELECT	V$ 表の参照、SYSTEM 表領域上のオブジェクトの参照を行います。 SYSTEM 表領域上のオブジェクトの更新は行いません。
SQL_LEVEL=NONE	V$ 表の参照を行います。 SYSTEM 表領域上のオブジェクトの更新および参照は行いません。

インスタンス監視動作の詳細は、「4.2 インスタンス監視時の動作」を参照してください。

注意: スタンバイ・インスタンスの監視を行う場合は、UPDATE を設定してください。

◆ **マルチテナント構成の場合**

マルチテナント構成の場合、CDB の監視方法は、全ノードステートメントの SQL_LEVEL パラメータに指定します。値ごとの監視方法は、以下のとおりです。

SQL_LEVEL=UPDATE	V$ 表の参照、SYSTEM 表領域上のオブジェクトの更新および参照を行います。
SQL_LEVEL=SELECT	V$ 表の参照、SYSTEM 表領域上のオブジェクトの参照を行います。 SYSTEM 表領域上のオブジェクトの更新は行いません。
SQL_LEVEL=NONE	V$ 表の参照を行います。 SYSTEM 表領域上のオブジェクトの更新および参照は行いません。

CDB 監視動作の詳細は、「4.2 インスタンス監視時の動作」を参照してください。

注意: スタンバイ・インスタンスの監視を行う場合は、UPDATE を設定してください。

PDB の監視を行う場合、PDB の監視方法は、全ノードステートメント、

PluggableDatabaseMonitor ステートメント内または PluggableDatabase ステートメント内の SQL_LEVEL パラメータに指定します。値ごとの監視方法は、以下のとおりです。

SQL_LEVEL=UPDATE	V$ 表の参照、SYSTEM 表領域上のオブジェクトの更新および参照を行います。
SQL_LEVEL=SELECT	V$ 表の参照、SYSTEM 表領域上のオブジェクトの参照を行います。 SYSTEM 表領域上のオブジェクトの更新は行いません。
SQL_LEVEL=NONE	PDB の監視は行いません。

また、PluggableDatabaseMonitor ステートメント内および PluggableDatabase ステートメント内の SQL_LEVEL パラメータは、全ノードステートメントの SQL_LEVEL パラメータに設定した値によって指定できる値が異なります。指定できる値は、以下のとおりです。

全ノードステートメントの値	PluggableDatabaseMonitor ステートメント内および PluggableDatabase ステートメント内に指定できる値
UPDATE	UPDATE／SELECT／NONE
SELECT	SELECT／NONE
NONE	NONE

PDB 監視動作の詳細は、「4.3 PDB 監視時の動作」を参照してください。

3.2.23. PDB 監視の設定（マルチテナント構成のみ）

PDB の監視を設定するには、PLUGGABLE_DB_CHECK パラメータを指定します。PLUGGABLE_DB_CHECK パラメータに指定できる値は、以下のとおりです。

PLUGGABLE_DB_CHECK=YES	PDB の監視を行います。
PLUGGABLE_DB_CHECK=NO	PDB の監視を行いません。

注意： PDB の監視を行うには、PLUGGABLE_DB_CHECK パラメータの設定以外に、PluggableDatabaseMonitor ステートメントおよび PluggableDatabase ステートメントの設定を行ってください。

注意： 監視対象の PDB の監視方法（SQL_LEVEL パラメータ）をすべて NONE に設定した場合、PLUGGABLE_DB_CHECK パラメータの値は NO として扱います。

3.2.24. 監視タイムアウトの時間帯での自動変更の設定

監視タイムアウトの時間帯での自動変更を設定するには、START_CHANGE_TIMEOUT パラメータ、END_CHANGE_TIMEOUT パラメータ、CHANGE_TIMEOUT_RATIO パラメータおよび CHANGE_TIMEOUT_LIMIT パラメータを指定します。

START_CHANGE_TIMEOUT パラメータに設定された時刻から、END_CHANGE_TIMEOUT パラメータに設定された時刻の間は、監視タイムアウトは各監視の POLL_TIMEOUT パラメータに設定した値に CHANGE_TIMEOUT_RATIO パラメータに設定した値を乗じた値に変更されます。

深夜 23 時から翌朝 5 時まで監視タイムアウトを 5 倍に延長する場合の設定は、以下のとおりです。

```
START_CHANGE_TIMEOUT=23
END_CHANGE_TIMEOUT=5
CHANGE_TIMEOUT_RATIO=5
```

監視タイムアウトの自動変更中に監視タイムアウトが発生した場合に、無応答障害として扱うかどうかを設定するには、CHANGE_TIMEOUT_LIMIT 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。

CHANGE_TIMEOUT_LIMIT=YES	無応答障害として扱います。
CHANGE_TIMEOUT_LIMIT=NO	無応答障害として扱いません。

注意： START_CHANGE_TIMEOUT パラメータと END_CHANGE_TIMEOUT パラメー

タの一方だけを指定することはできません。
また、START_CHANGE_TIMEOUT パラメータと END_CHANGE_TIMEOUT パラメータに同じ値を設定することはできません。

注意: 以下の監視は、自動変更の対象外です。

- **Oracle Clusterware ／ Oracle Restart の監視**
- **リスナー監視**
- **ASM リスナー監視**
- **SCAN リスナー監視**

3.2.25. インスタンス監視表のスキーマの設定

「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。

また、PDB の監視を行う場合は、InstanceMonitor ステートメント内の OWNER_NAME パラメータに加えて、PluggableDatabaseMonitor ステートメント内または PluggableDatabase ステートメント内の OWNER_NAME パラメータの指定が必須です。

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

3.2.26. PDB 障害検出時の動作の設定（マルチテナント構成のみ）

PDB 障害検出時の動作を設定するには、PDB_OVER_ACTION 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。

PDB_OVER_ACTION=NEVER	障害を検出した PDB を再オープン／クローズせず、監視を継続します。
PDB_OVER_ACTION=REOPEN	障害を検出した PDB を再オープンし、監視を継続します。
PDB_OVER_ACTION=CLOSE	障害を検出した PDB をクローズし、監視を継続します。

REOPEN を設定した場合、再オープン前と同様のオープン・モードで再オープンします。

3.2.27. PDB クローズ検出時の動作の設定（マルチテナント構成のみ）

PDB クローズ検出時の動作を設定するには、UNOPEN_PDB_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。

UNOPEN_PDB_DOWN=YES	障害として扱います。
UNOPEN_PDB_DOWN=NO	障害として扱いません。

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

インスタンス／PDB／リスナーの障害時、再確認および再起動を行っても復旧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、PDB 監視モニタ (pdbmond)、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。

3.2.29. インスタンスの繰り返し障害の監視の設定

メモリがひっ迫した場合など、インスタンスの状態が正常と異常を繰り返す可能性があります。

インスタンスが正常と異常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。

監視結果エラーの累積回数は、障害を検出した初回時（UP の状態から DOWN を検出した際）に加算します。監視結果エラーの累積回数が 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、インスタンス監視の結果が 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、監視結果エラーの累積回数をクリアします。

なお、以下の場合にも、監視結果エラーの累積回数と監視結果 UP の連続回数がクリアされます。

- 動的な設定変更（apply リクエスト）実行時
- RAC 構成のいずれかのノードでのインスタンス障害検出時
- RAC 構成のいずれかのノードでのモニタ制御デーモン (oramond) の起動または停止

3.2.30. 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: LSNR_CHECK_MODE パラメータに PING_NO_RESTART を指定した場合は、TARGET_RESTART パラメータの設定にかからわず、ApplicationMonitor によるリスナーの再起動は行われません。

3.2.31. 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。

監視結果異常の累積回数は、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）に加算します。監視結果異常の累積回数が 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、監視結果異常の累積回数をクリアします。

なお、モニタ制御デーモン (oramond) に制御されたリスナー監視モニタ (lsnrmond) では、以下の場合にも、監視結果異常の累積回数と監視結果 UP の連続回数がクリアされます。

- 動的な設定変更（apply リクエスト）実行時
- RAC 構成のいずれかのノードでのインスタンス障害検出時

3.2.32. リスナー監視方法の設定

リスナー監視時、監視対象リスナーの構成に応じた監視方法を設定するには、LSNR_CHECK_MODE パラメータを指定します。

LSNR_CHECK_MODE=PING	単一リスナーの監視
LSNR_CHECK_MODE=PING_NO_RESTART	冗長化リスナーの接続監視
LSNR_CHECK_MODE=STATUS	冗長化リスナーの状態監視

PING に設定した場合、NET_SERVICE_NAME パラメータに指定されたネット・サービス名を使用してリスナーの状態を監視します。障害検出時は、TARGET_RESTART パラメータの設定に従い監視対象リスナーの再起動を行います。

単一リスナーの監視を行う場合に指定します。なお、単一リスナー監視の監視方式は、旧リリースのリスナー監視と同等です。

PING_NO_RESTART に設定した場合、NET_SERVICE_NAME パラメータに指定されたネット・サービス名を使用してリスナーの状態を監視します。障害検出時は、監視対象リスナーの再起動は行わずにリトライ動作を行います。

インスタンスに対して冗長化されたリスナーの監視を行う場合に指定します。

STATUS に設定した場合、Oracle の lsnrctl コマンドを用いてリスナーの状態を監視します。障害検出時は、TARGET_RESTART パラメータの設定に従い監視対象リスナーの再起動を行います。

インスタンスに対して冗長化されたリスナーの再起動を行う場合に指定します。

TARGET_RESTART パラメータの詳細は、「3.2.30 リスナー再起動方法の設定」を参照してください。

3.2.33. 表領域監視、ローカルディスク監視のしきい値設定

表領域、ローカルディスク監視では 通知を判断するしきい値を 2 段階に設定できます。
各しきい値は以下のパラメータで設定できます。

監視する現在の測定値が USEABLE_EBLOCK_WARNING、FREE_SPACE_WARNING パラメータを下回ると警告を示す syslog 出力をします。
さらに測定値が下がり、USEABLE_EXTENT_BLOCK、FREE_SPACE パラメータ を下回ると、警告出力のかわりに設定した下限値を下回ることを示す syslog 出力に加え任意のコマンドを実行します。

TableSpace ステートメント

USEABLE_EXTENT_BLOCK	最大使用可能エクステントサイズの最終判断の syslog 出力をするしきい値を設定します。 指定すると任意のコマンドを実行します。
USEABLE_EBLOCK_WARNING	最終判断よりも先に syslog に警告出力するしきい値を設定します。 コマンド実行は行いません。 デフォルト値としては 0 が設定され、警告出力はされません。

TableSpace、DiskSpaceMonitor、Directory ステートメント

FREE_SPACE	空き容量率の最終判断の syslog 出力をするしきい値を設定します。 指定すると任意のコマンドを実行します。
FREE_SPACE_WARNING	最終判断よりも先に syslog に警告出力するしきい値を設定します。 コマンド実行は行いません。 デフォルト値としては 0 が設定され、警告出力はされません。

3.3. 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を設定してください。

```
SCRIPT_NAME = /opt/HA/AM/bin/oradiag
```

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します。

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します。
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します。
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します。
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します。

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

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。

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。

3.4. PDB 監視モニタの単独起動

PDB 監視モニタ (pdbmond) 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。

PDB 監視モニタを単独で起動する場合は、以下の環境変数の指定が必要です。

単独起動モニタ用起動スクリプト (start_alone_monitor.sh)に環境変数を設定します。

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

PDB 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。

注意: PDB 監視モニタは Oracle 21c の環境でのみ単独起動が可能です。

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、以下の環境変数の指定が必要です。

単独起動モニタ用起動スクリプト (start_alone_monitor.sh) を使用して単独起動する場合は、単独起動モニタ用起動スクリプトに環境変数を設定します。

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、以下の環境変数の指定が必要です。

単独起動モニタ用起動スクリプト (start_alone_monitor.sh) を使用して単独起動する場合は、単独起動モニタ用起動スクリプトに環境変数を設定します。

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、以下の環境変数の指定が必要です。

単独起動モニタ用起動スクリプト (start_alone_monitor.sh) を使用して単独起動する場合は、単独起動モニタ用起動スクリプトに環境変数を設定します。

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、Oracle Clusterware／Oracle Restart のデーモンの状態を表示するコマンド、CRS 監視モニタ (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は、/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、必ずシステムの環境に応じた指定を行ってください。

CRS 監視モニタの詳細は、「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          = 21C           # 19C, 21C
CRS_STAT                = "/u01/app/21.3.0/grid/bin/crsctl check crs"
#CRS_STAT               = "/u01/app/12.1.0/grid/bin/crsctl check crs"
CRS_CHECK_MODE          = ALL
LOG_LEVEL               = 2
POLL_INTERVAL           = 90
POLL_TIMEOUT            = 180
RETRY_COUNT             = 3
HALT_METHOD             = KILL
WAIT_INTERVAL           = 180
SERVICE_DOWN            = NO
#CUSTOM_USER            = root
#CUSTOM_CMD             = "/u01/app/19.3.0/grid/bin/crsctl stop crs"
CRS_CHECK_CLUSTER       = YES
NODE_NAME               = nx1
NODE_NAME               = nx2
```

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、「3.9.4 パラメーター覧」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。	19C 21C		
CRS_STAT	Oracle が提供する Oracle Clusterware／Oracle Restart のデーモン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 Oracle Clusterware／Oracle Restart のデーモン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		
CRS_CHECK_MODE	監視対象デーモンを指定します。 ※Oracle Clusterware の環境で指定が有効なパラメータです。 ※詳細は、「3.9.6 監視対象デーモンの設定（Oracle Clusterware 環境のみ）」を参照してください。		○	○
	ALL	ALL OHAS		
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	Oracle Clusterware／Oracle Restart のデーモンを監視する間隔を秒で指定します。		○	○
	90 秒	10～86400		
POLL_TIMEOUT	Oracle Clusterware／Oracle Restart のデーモン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、Oracle Clusterware／Oracle Restart のデーモンを再確認する回数を指定します。		○	○
	3 回	0～24		
HALT_METHOD	障害検出時、障害デーモンの再起動の待ち合わせ動作を指定します。 ※詳細は、「3.9.7 障害デーモン再起動の待ち合わせ動作の設定」を参照してください。		○	○
	KILL	KILL NONE DOWN		
WAIT_INTERVAL	障害検出時、init(8)／Oracle Clusterware による障害デーモンの再起動を待ち合わせる時間を秒で指定します。		○	○
	180 秒	60～86400		

パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
SERVICE_DOWN	障害検出時、init(8)／Oracle Clusterware による障害デーモンの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、「3.9.8 Oracle Clusterware／Oracle Restart 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		
CRS_CHECK_CLUSTER	CRS 監視モニタ(crsmond)間での通信により Oracle Clusterware の障害をクラスタ全体で統合的に判断するかを指定します。 ※詳細は、「3.9.9 障害検出時のクラスタ全体での統合的判断の設定」を参照してください。		○	×
	YES	YES NO		
NODE_NAME	監視対象の Oracle Clusterware 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。 CRS_CHECK_CLUSTER が YES のときは必ず指定します。		×	×
	なし	1023 文字以内		
SERVICE_PORT	CRS 監視モニタ(crsmond)間での通信に使用するポート番号を指定します。モニタ制御デーモン(oramond)が使用するポート番号と違う番号を指定する必要があります。 ※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。 ※指定するポート番号は、OS のパラメータ ip_local_reserved_ports に設定することを推奨します。		○	×
	25313	1024～65535		
HALT_CONTROL_OPTION	障害ノード上の CRS 監視モニタ(crsmond)の停止動作を指定します。 ※詳細は、「3.9.10 障害ノード上の CRS監視モニタの停止動作の設定」を参照してください。		○	○
	ALWAYS	ALWAYS OUTLIVE NEVER		

※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. Oracle Clusterware／Oracle Restart のデーモン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、Oracle Clusterware／Oracle Restart のデーモンの状態表示用コマンドの指定が必須です。Oracle Clusterware／Oracle Restart のデーモン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、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。

環境	設定するコマンド
Oracle Restart	crsctl check has
Oracle Clusterware	crsctl check crs

3.9.6. 監視対象デーモンの設定（Oracle Clusterware 環境のみ）

Oracle Clusterware 環境では、監視対象デーモンを指定できます。

監視対象デーモンを設定するには、CRS_CHECK_MODE 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。

CRS_CHECK_MODE=ALL	以下の Oracle Clusterware のデーモンを監視します。 ・Oracle 高可用性サービス・デーモン（以後 ohasd と記載します） ・クラスタ・レディ・サービス (CRS) ・デーモン (crsd) ・クラスタ同期サービス (CSS) ・デーモン（以後 ocssd と記載します） ・イベント・マネージャ (EVM) ・デーモン (evmd)
CRS_CHECK_MODE=OHAS	Oracle Clusterware のデーモンの ohasd のみ監視します。

3.9.7. 障害デーモン再起動の待ち合わせ動作の設定

Oracle Clusterware／Oracle Restart の障害時、再確認を行っても復旧できなかった場合、障害デーモンの再起動の待ち合わせ動作を設定するには、HALT_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_METHOD=KILL	障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対して kill コマンドを実行し、init(8)／Oracle Clusterware による障害デーモンの再起動を待ち合わせます。 待ち合わせ後、監視を再開します。
HALT_METHOD=NONE	障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対して kill コマンドを実行せず、init(8)／Oracle Clusterware による障害デーモンの再起動を待ち合わせます。 待ち合わせ後、監視を再開します。
HALT_METHOD=DOWN	障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対する kill コマンドの実行、init(8)／Oracle Clusterware による障害デーモンの再起動の待ち合わせを行わず、CLUSTERPRO への障害通知を行うか判断します。

注意： KILL に設定した場合、ocssd プロセスに対しては、障害を検出しても kill コマンドを実行しません。

3.9.8. Oracle Clusterware／Oracle Restart 障害時の CLUSTERPRO への障害通知の設定

Oracle Clusterware／Oracle Restart の障害時、再確認および再起動を行っても復旧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。

3.9.9. 障害検出時のクラスタ全体での統合的判断の設定

障害検出時にクラスタ全体での監視結果の統合的な判断をするかどうかを設定するには、CRS_CHECK_CLUSTER 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。

CRS_CHECK_CLUSTER=YES	障害検出時にクラスタ全体での監視結果の統合的な判断をします。
CRS_CHECK_CLUSTER=NO	障害検出時の監視結果の判断はノードごとにそれぞれ行います。MC2.8 以前の動作です。

注意: クラスタ全体での監視結果の統合的な判断をする場合、各ノードの設定ファイルは同一の内容にする必要があります。

3.9.10. 障害ノード上の CRS 監視モニタの停止動作の設定

障害が発生したノード上の CRS 監視モニタ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	CRS 監視モニタを停止します。
HALT_CONTROL_OPTION=NEVER	CRS 監視モニタを停止しません。
HALT_CONTROL_OPTION=OUTLIVE	他に CRS 監視モニタが生存しているノードが存在する場合、ALWAYS として扱います。 他に CRS 監視モニタが生存しているノードが存在しない場合、NEVER として扱います。

3.10. ASM 監視設定ファイル (asmmond.conf) の作成

ApplicationMonitor で ASM インスタンス障害の監視を行う場合は、ASM 監視設定ファイルを作成する必要があります。ASM 監視設定ファイルには、監視対象とする ASM インスタンスが動作する Oracle に関する情報(ORACLE_SID, ORACLE_BASE, ORACLE_HOME, ORA_NLS10, LD_LIBRARY_PATH, NLS_LANG)、監視モニタの動作設定などを定義します。

ASM 監視設定ファイルのサンプルは、/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。

ASM 監視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（LD_LIBRARY_PATH に相当）および NLS_LANG パラメータは、Oracle の環境に依存しますので、必ず環境を確認した上で指定してください。

ASM 監視モニタ (asmmond) 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 ASM 監視設定ファイルは、/etc/opt/HA/AM/conf/asmmond.conf として置いてください。

注意: ASM 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。

3.10.1. ASM 監視設定ファイルの記述例

ASM 監視設定ファイル (asmmond.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。

```
# asmmond.conf

MONITOR_USER            = grid
ORACLE_SID              = "+ASM1"
ORACLE_BASE             = /u01/app/grid
ORACLE_HOME             = /u01/app/21.3.0/grid
ORA_NLS                 = /u01/app/21.3.0/grid/nls/data
SHLIB_PATH              = /u01/app/21.3.0/grid/lib
#NLS_LANG               = japanese_japan.ja16euc
#LOG_LEVEL              = 2
#POLL_INTERVAL          = 90
#POLL_TIMEOUT           = 120
#POLL_DOWN_RETRY        = 0
#POLL_STALL_RETRY       = 0
#HALT_METHOD            = KILL
#RESTART_COUNT          = 5
#WAIT_INTERVAL          = 180
#SERVICE_DOWN           = YES
```

3.10.2. 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。

3.10.3. パラメータの記述形式

パラメータは、ASM 監視モニタ (asmmond) 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、「3.10.4 パラメーター覧」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。

3.10.4. パラメーター覧

ASM 監視設定ファイル (asmmond.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		
MONITOR_USER	ASM 監視モニタ (asmmond) の実効ユーザー名を指定します。		○	×
	grid	なし		
ORACLE_SID	ASM インスタンスの識別子を指定します。		×	×
	省略できません。	なし		
ORACLE_BASE	Oracle Grid Infrastructure をインストールしたディレクトリ構成の基本となるディレクトリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
ORACLE_HOME	Oracle Grid Infrastructure をインストールしたディレクトリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
ORA_NLS	Oracle Grid Infrastructure の言語、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のパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
SHLIB_PATH	Oracle Grid Infrastructure の 64bit 共有ライブラリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
NLS_LANG	ASM 監視モニタ (asmmond) 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。		○	×
	AMERICAN_AMERICA.US7ASCII	なし		
TNS_ADMIN	Oracle Grid Infrastructure の Oracle Net Services 構成が格納されているディレクトリのパス(絶対パス)を指定します。		○	×
	指定なし	なし		
LOG_LEVEL	ASM 監視モニタ (asmmond) 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		
ORACLE_USER	ASM 監視モニタ (asmmond) 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、「3.10.5 パスワード・ファイル認証での接続」を参照してください。		○	○
	指定なし	なし		
ORACLE_PASS	ASM 監視モニタ (asmmond) 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、「3.10.5 パスワード・ファイル認証での接続」を参照してください。		○	○
	指定なし	なし		
POLL_INTERVAL	ASM インスタンスを監視する間隔を秒で指定します。		○	○
	90 秒	5～2678400		
POLL_TIMEOUT	ASM 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。		○	○
	120 秒	1～86400		
POLL_DOWN_RETRY	ASM インスタンスのダウン検出時、ASM インスタンスを再確認する回数を指定します。		○	○
	0 回	0～100		
POLL_STALL_RETRY	ASM インスタンスのストール検出時、ASM インスタンスを再確認する回数を指定します。		○	○
	0 回	0～100		

パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
HALT_METHOD	障害検出時、ASM インスタンスを強制停止するかどうかを指定します。 ※詳細は、「3.10.6 ASM インスタンス強制停止の設定」を参照してください。		○	○
	KILL	KILL NONE		
RESTART_COUNT	ASM インスタンスの障害時、ASM 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。		○	○
	5 回	0～999		
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 (crs_control.sh) の実行時間の上限を秒で指定します。		○	○
	90 秒	1～3600		
WAIT_INTERVAL	障害検出時、ASM インスタンスの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	ASM 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 ASM 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、「3.10.7 ASM インスタンス障害時の 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		
COMMAND_NAME	障害検出時に実行するコマンドの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。 ※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。 ※コマンドのオプションやシェルスクリプトの引数の指定はできません。		○	○
	指定なし	なし		
START_CHANGE_TIMEOUT	監視タイムアウトの自動変更を開始する時刻を 24 時間表記の”時”で指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」を参照してください。		○	○
	指定なし	0～23		
END_CHANGE_TIMEOUT	監視タイムアウトの自動変更を終了する時刻を 24 時間表記の”時”で指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」を参照してください。		○	○
	指定なし	0～23		
CHANGE_TIMEOUT_RATIO	監視タイムアウトを自動変更する際の倍率を指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」を参照してください。		○	○
	1	1～10		
CHANGE_TIMEOUT_LIMIT	監視タイムアウトの自動変更中に監視タイムアウトが発生した場合に、無応答障害として扱うかどうかを指定します。 ※詳細は、「3.2.24 監視タイムアウトの時間帯での自動変更の設定」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。

※2 ASM 監視モニタ (asmmond) 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。

3.10.5. 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 ASM インスタンスを監視するには、ORACLE_USER パラメータ、ORACLE_PASS パラメータを指定します。

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。

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。

3.10.6. ASM インスタンス強制停止の設定

ASM インスタンスの障害時、再確認を行っても復旧しなかった場合、ASM インスタンスを強制停止するかどうかを設定するには、HALT_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_METHOD=KILL	ASM インスタンスの 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 ASM インスタンス再起動を待ち合わせます。 待ち合わせ後、監視を再開します。
HALT_METHOD=NONE	ASM インスタンスの SMON プロセスに対して kill コマンドを実行せず、Oracle Clusterware／Oracle Restart による ASM インスタンス再起動を待ち合わせます。 待ち合わせ後、監視を再開します。

3.10.7. ASM インスタンス障害時の CLUSTERPRO への障害通知の設定

ASM インスタンスの障害時、再確認および再起動を行っても復旧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。

3.11. ASM 監視モニタ起動スクリプト (start_asmmond.sh) の設定

ApplicationMonitor で ASM インスタンスの障害監視を行う場合は、ASM 監視設定ファイル (asmmond.conf) と ASM 監視モニタ起動スクリプトを設定する必要があります。ASM 監視モニタ起動スクリプトには、監視対象の Oracle のバージョン (ORACLE_VERSION)、Oracle に関する情報(ORACLE_SID, ORACLE_BASE, ORACLE_HOME, LD_LIBRARY_PATH, ORA_NLS10, NLS_LANG) を定義します。

ASM 監視モニタ起動スクリプトは、/etc/opt/HA/AM/lbin ディレクトリにインストールされます。ASM 監視モニタ起動スクリプトに記述する ORACLE_VERSION パラメータおよび Oracle に関する情報は、環境に依存しますので、必ず環境を確認した上で指定してください。

ASM 監視モニタ起動スクリプトの詳細は、「6 コマンドリファレンス」を参照してください。

注意: **設定した ASM 監視モニタ起動スクリプトは、/etc/opt/HA/AM/lbin ディレクトリ内に置くと、アップデート時などに上書きや削除が行われる可能性があります。**
設定した ASM 監視モニタ起動スクリプトは、バックアップすることを推奨します。

3.11.1. ASM 監視モニタ起動スクリプトの記述例

ASM 監視モニタ起動スクリプト (start_asmmond.sh) は、コメント、パラメータ、複数の処理によって構成されたテキストファイルです。

以下のように、スクリプトファイルの先頭部分に、パラメータおよび Oracle 環境変数の設定を記述します。

```
#!/bin/sh

# Set PATH to reference the appropriate directories.
PATH=${PATH}:/etc/opt/HA/AM/lbin

#
# Please set parameter for your Oracle system.
#
ORACLE_VERSION=21C              # 19C, 21C
export ORACLE_SID=”+ASM1”
export ORACLE_BASE=/u01/app/grid
export ORACLE_HOME=/u01/app/21.3.0/grid
export LD_LIBRARY_PATH=${ORACLE_HOME}/lib
export ORA_NLS10=${ORACLE_HOME}/nls/data
export NLS_LANG=AMERICAN_AMERICA.US7ASCII
GRID_USER=

            ＜以下省略＞
```

3.11.2. パラメーター覧

ASM 監視モニタ起動スクリプト (start_asmmond.sh) に設定するパラメータは、以下のとおりです。

パラメータ名	意味	
	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。	19C 21C
GRID_USER	Oracle ASM クラスタののモードを確認するために実行される asmcmd コマンドの、実効ユーザーを指定します。指定しない場合は、スクリプト実効ユーザーを使用します。 ※スクリプト実効ユーザーで asmcmd コマンドが実行できない場合に指定します。	
	指定なし	なし

3.11.3. Oracle 環境変数一覧

ASM 監視モニタ起動スクリプト (start_asmmond.sh) には、以下の Oracle 環境変数を必ず設定してください。

ORACLE_SID	ASM インスタンスの識別子
ORACLE_BASE	Oracle Grid Infrastructure をインストールした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 Grid Infrastructure をインストールしたホームディレクトリ
LD_LIBRARY_PATH	Oracle Grid Infrastructure の 64bit 共有ライブラリのパス
ORA_NLS10	Oracle Grid Infrastructure の言語、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	ASM 監視モニタ (asmmond) 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

3.12. ASM リスナー監視設定ファイル (alsnrmond.conf) の作成

ApplicationMonitor で ASM リスナー障害の監視を行う場合は、ASM リスナー監視設定ファイルを作成する必要があります。ASM リスナー監視設定ファイルには、監視対象とする ASM リスナーが動作する Oracle に関する情報(ORACLE_BASE, ORACLE_HOME, ORA_NLS10, LD_LIBRARY_PATH, NLS_LANG)、監視モニタの動作設定などを定義します。

ASM リスナー監視設定ファイルのサンプルは、/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。

ASM リスナー監視設定ファイルに記述する 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(LD_LIBRARY_PATH に相当)および NLS_LANG パラメータは、Oracle の環境に依存しますので、必ず環境を確認した上で指定してください。

注意: 作成した ASM リスナー監視設定ファイルは、/etc/opt/HA/AM/conf/alsnrmond.conf として置いてください。

注意: ASM リスナー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。

注意: ASM リスナー監視モニタは、モニタ制御デーモン (oramond) により制御されません。ASM リスナー監視モニタを使用する場合は、Oracle Clusterware へのリソース登録を行ってください。

3.12.1. ASM リスナー監視設定ファイルの記述例

ASM リスナー監視設定ファイル (alsnrmond.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。

```
# alsnrmond.conf

ORACLE_BASE          = /u01/app/grid
ORACLE_HOME          = /u01/app/21.3.0/grid
ORA_NLS              = /u01/app/21.3.0/grid/nls/data
SHLIB_PATH           = /u01/app/21.3.0/grid/lib
#NLS_LANG            = japanese_japan.ja16euc
#MONITOR_USER        = grid
#LOG_LEVEL           = 2
#POLL_INTERVAL       = 60
#POLL_TIMEOUT        = 90
```

3.12.2. 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。

3.12.3. パラメータの記述形式

パラメータは、ASM リスナー監視モニタ (alsnrmond) 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、「3.12.4 パラメーター覧」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。

3.12.4. パラメーター覧

ASM リスナー監視設定ファイル (alsnrmond.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_BASE	Oracle Grid Infrastructure をインストールしたディレクトリ構成の基本となるディレクトリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
ORACLE_HOME	Oracle Grid Infrastructure をインストールしたディレクトリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
ORA_NLS	Oracle Grid Infrastructure の言語、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のパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
SHLIB_PATH	Oracle Grid Infrastructure の 64bit 共有ライブラリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
NLS_LANG	ASM リスナー監視モニタ (alsnrmond) 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。		○	×
	AMERICAN_AMERICA.US7ASCII	なし		
MONITOR_USER	ASM リスナー監視モニタ (alsnrmond) の実効ユーザー名を指定します。		○	×
	grid	なし		
LOG_LEVEL	ASM リスナー監視モニタ (alsnrmond) 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	ASM 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	ASM リスナー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。		○	○
	90 秒	1～86400		
LSNRCTL	Oracle の lsnrctl コマンドのコマンド名を絶対パスで指定します。		○	○
	${ORACLE_HOME}/bin/lsnrctl	なし		
LSNR_CMD_TIMEOUT	リスナー再起動シェルスクリプト (lsnr_control.sh) による ASM リスナーの強制停止を待ち合わせる時間の上限を秒で指定します。		○	○
	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 ASM リスナー監視モニタ (alsnrmond) 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。

3.13. SCAN リスナー監視設定ファイル (slsnrmond.conf) の作成

ApplicationMonitor で SCAN リスナー障害の監視を行う場合は、SCAN リスナー監視設定ファイルを作成する必要があります。SCAN リスナー監視設定ファイルには、監視対象とする SCAN リスナーが動作する Oracle に関する情報(ORACLE_BASE, ORACLE_HOME, ORA_NLS10, LD_LIBRARY_PATH, NLS_LANG)、監視モニタの動作設定などを定義します。

SCAN リスナー監視設定ファイルのサンプルは、/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。

SCAN リスナー監視設定ファイルに記述する 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(LD_LIBRARY_PATH に相当)および NLS_LANG パラメータは、Oracle の環境に依存しますので、必ず環境を確認した上で指定してください。

注意: 作成した SCAN リスナー監視設定ファイルは、/etc/opt/HA/AM/conf/slsnrmond.conf として置いてください。

注意: SCAN リスナー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。

注意: SCAN リスナー監視モニタは、モニタ制御デーモン (oramond) により制御されません。SCAN リスナー監視モニタを使用する場合は、Oracle Clusterware へのリソース登録を行ってください。

3.13.1. SCAN リスナー監視設定ファイルの記述例

SCAN リスナー監視設定ファイル (slsnrmond.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。

```
# slsnrmond.conf

ORACLE_BASE          = /u01/app/grid
ORACLE_HOME          = /u01/app/21.3.0/grid
ORA_NLS              = /u01/app/21.3.0/grid/nls/data
SHLIB_PATH           = /u01/app/21.3.0/grid/lib
#NLS_LANG            = japanese_japan.ja16euc
#MONITOR_USER        = grid
#LOG_LEVEL           = 2
#POLL_INTERVAL       = 60
#POLL_TIMEOUT        = 90
#RESTART_TIMEOUT     = 60
```

3.13.2. 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。

3.13.3. パラメータの記述形式

パラメータは、SCAN リスナー監視モニタ (slsnrmond) 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、「3.13.4 パラメーター覧」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。

3.13.4. パラメーター覧

SCANリスナー監視設定ファイル (slsnrmond.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_BASE	Oracle Grid Infrastructure をインストールしたディレクトリ構成の基本となるディレクトリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
ORACLE_HOME	Oracle Grid Infrastructure をインストールしたディレクトリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
ORA_NLS	Oracle Grid Infrastructure の言語、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のパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
SHLIB_PATH	Oracle Grid Infrastructure の 64bit 共有ライブラリのパス(絶対パス)を指定します。		×	×
	省略できません。	なし		
NLS_LANG	SCAN リスナー監視モニタ (slsnrmond) 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。		○	×
	AMERICAN_AMERICA.US7ASCII	なし		
MONITOR_USER	SCAN リスナー監視モニタ (slsnrmond) の実効ユーザー名を指定します。		○	×
	grid	なし		
LOG_LEVEL	SCAN リスナー監視モニタ (slsnrmond) 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	SCAN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	SCAN リスナー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。		○	○
	90 秒	1～86400		
RESTART_INTERVAL	SCAN リスナーの障害時、SCAN リスナーの再起動を行うまでの待ち合わせ時間を秒で指定します。		○	○
	3 秒	1～10		
RESTART_TIMEOUT	障害を検出した SCAN リスナーを再起動するために、SCAN リスナーに対して kill コマンド実行後、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 Clusterware のデフォルト設定で SCANリスナー再起動までに最大 60 秒かかります。そのため、SCAN リスナーを動的構成とした場合、このパラメータは、60 秒以上に設定することを推奨します。		○	○
	60 秒	0～3600		
GET_STATS	SCANリスナー監視モニタ (slsnrmond)で統計情報の採取を行うかどうかを指定します。 ※詳細は、「3.13.5 統計情報採取の設定」を参照してください。		○	○
	NO	YES NO		

パラメータ名	意味		※1	※2
	省略時のデフォルト値	値の範囲		
LSNRCTL	Oracle の lsnrctl コマンドのコマンド名を絶対パスで指定します。		○	○
	${ORACLE_HOME}/bin/lsnrctl	なし		
LSNR_CMD_TIMEOUT	リスナー再起動シェルスクリプト (lsnr_control.sh) による SCAN リスナーの強制停止を待ち合わせる時間の上限を秒で指定します。		○	○
	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 SCANリスナー監視モニタ (slsnrmond) 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。

3.13.5. 統計情報採取の設定

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。

採取する情報は、以下のとおりです。

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/slsnrmond/data/<SCAN リスナー名>_1.dat
一世代前の情報	/var/opt/HA/AM/slsnrmond/data/<SCAN リスナー名>_2.dat
二世代前の情報	/var/opt/HA/AM/slsnrmond/data/<SCAN リスナー名>_3 .dat

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します。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。

出力フォーマット：情報採取日時, SCAN リスナー名, TAT

出力イメージ

```
2020/3/13 08:23:40,LISTENER_SCAN1,1
2020/3/13 08:24:40,LISTENER_SCAN1,1
2020/3/13 08:25:40,LISTENER_SCAN1,1
```

3.14. 単独起動モニタ用起動スクリプト (start_alone_monitor.sh) の設定

単独起動モニタ用起動スクリプトを使用して、 PDB 監視モニタ(pdbmond)、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) を単独起動することができます。単独起動モニタ用起動スクリプトには、監視対象の Oracle のバージョン(ORACLE_VERSION) 、 Oracle に関する情報 (ORACLE_SID, ORACLE_BASE, ORACLE_HOME, LD_LIBRARY_PATH, ORA_NLS10, NLS_LANG) を定義します。

単独起動モニタ用起動スクリプトは、/etc/opt/HA/AM/lbin ディレクトリにインストールされます。単独起動モニタ用起動スクリプトに記述する ORACLE_VERSION パラメータおよび Oracle に関する情報は、環境に依存しますので、必ず環境を確認した上で指定してください。

単独起動モニタ用起動スクリプトの詳細は、「6 コマンドリファレンス」を参照してください。

注意: **設定した 単独起動モニタ用起動スクリプトトは、/etc/opt/HA/AM/lbin ディレクトリ内に置くと、アップデート時などに上書きや削除が行われる可能性があります。**
設定した 単独起動モニタ用起動スクリプトは、バックアップすることを推奨します。

3.14.1. 単独起動モニタ用起動スクリプトの記述例

単独起動モニタ用起動スクリプト (start_alone_monitor.sh) は、コメント、パラメータ、複数の処理によって構成されたテキストファイルです。

以下のように、スクリプトファイルの先頭部分に、パラメータおよび Oracle 環境変数の設定を記述します。

```
#!/bin/sh

# Set PATH to reference the appropriate directories.
PATH=${PATH}:/etc/opt/HA/AM/lbin

#
# Please set parameter for your Oracle system.
#
ORACLE_VERSION=21C              # 19C, 21C
export ORACLE_SID=”sid1”
export ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=/u01/app/oracle/product/21.3.0/dbhome_1
export LD_LIBRARY_PATH=${ORACLE_HOME}/lib
export ORA_NLS10=${ORACLE_HOME}/nls/data
export NLS_LANG=AMERICAN_AMERICA.US7ASCII


        ＜以下省略＞
```

3.14.2. パラメータ一覧

単独起動モニタ用起動スクリプト (start_alone_monitor.sh) に設定するパラメータは、以下のとおりです。

パラメータ名	意味	
	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。	19C 21C

3.14.3. Oracle 環境変数一覧

単独起動モニタ用起動スクリプト (start_alone_monitor.sh) には、以下の Oracle 環境変数を必ず設定してください。

ORACLE_SID	Oracle のインスタンスの識別子
ORACLE_BASE	Oracle をインストールした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 をインストールしたホームディレクトリ
LD_LIBRARY_PATH	Oracle の 64bit 共有ライブラリのパス
ORA_NLS10	Oracle の言語、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	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

3.15. 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 (inst.conf、lsnr.conf、tschk.conf、fds.conf、stbdb.conf、asm.conf、alsnr.conf、slsnr.conf、stats.conf、ssdump.conf) 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しています。

デフォルト設定の詳細は、「3.15.2 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) および PDB 監視モニタ (pdbmond) 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
ASM 監視モニタ (asmmond) の フィルタリング設定ファイル	/etc/opt/HA/AM/conf/asm.conf
ASM リスナー監視モニタ（alsnrmond）の フィルタリング設定ファイル	/etc/opt/HA/AM/conf/alsnr.conf
SCAN リスナー監視モニタ（slsnrmond）の フィルタリング設定ファイル	/etc/opt/HA/AM/conf/slsnr.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ーモン (ssdump) のフィルタリング設定ファイル	/etc/opt/HA/AM/conf/ssdump.conf

注意：　フィルタリング設定ファイルの格納先およびファイル名は変更しないでください。

注意：　設定したフィルタリング設定ファイルは、/etc/opt/HA/AM/conf ディレクトリ内に置くと、アップデート時などに上書きや削除が行われる可能性があります。
設定したフィルタリング設定ファイルは、バックアップすることを推奨します。

3.15.1. 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 (inst.conf、lsnr.conf、tschk.conf、fds.conf、stbdb.conf、asm.conf、alsnr.conf、slsnr.conf、stats.conf、ssdump.conf) 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。

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
-56955   #quarantined plan used.
```

◆ 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、設定された内容を使用します。

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、デフォルト設定を使用します。

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。

◆ 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。

3.15.2. 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)、PDB 監視モニタ (pdbmond) および ASM 監視モニタ (asmmond) 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、PDB 監視モニタおよび ASM 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えました。
ORA-00019	最大セッション・ライセンスを超えました。
ORA-00020	最大プロセス数(string)を超えました。
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 制限を超えました。
ORA-03113	通信チャ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えました。
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:内部制限を超えました。
ORA-01033	Oracle の初期化またはシャットダウン中です。
ORA-01109	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-01219	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。
ORA-56955	隔離プランが使用されています。

リスナー監視モニタ (lsnrmond) 、ASM リスナー監視モニタ(alsnrmond)および SCAN リスナー監視モニタ(slsnrmond)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、ASM リスナー監視モニタおよび SCAN 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-01033	Oracle の初期化またはシャットダウン中です。
ORA-56955	隔離プランが使用されています。

スタンバイデータベース監視モニタ (stbdbmond) 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。
ORA-56955	隔離プランが使用されています。

3.16. Oracle Clusterware へのリソース登録

ApplicationMonitor で ASM リスナーの障害監視を行う場合、SCAN リスナーの障害監視を行う場合、または CLUSTERPRO が使用できない環境で Oracle Clusterware と HA クラススタソフトとして連携する場合には、Oracle Clusterware へのリソース登録が必要です。

ApplicationMonitor の監視プロセス (モニタ制御デーモン (oramond) または単独起動の監視モニタ)を Oracle Clusterware のリソースとして登録することで、Oracle Clusterware からのインスタンス、PDB、ASM インスタンス、リスナー、ASM リスナー、および SCAN リスナーの起動/停止に合わせて、監視を行うことが可能となります。

3.16.1. リソースの登録方法

リソースの登録方法は、以下のとおりです。

1 crsctl コマンドのパスの取得

以下のシェルスクリプトを実行して、Oracle Clusterware との連携に使用する crsctl コマンドのパスを取得します。

シェルスクリプトは、RAC を構成するすべてのノードで、root ユーザーで実行してください。

「2 連携用リソースの作成」の前に一度だけ実行してください。

```
# /opt/HA/AM/bin/init/get_grid_path.sh <XXX>
```

XXX: grid ユーザー名を指定します。省略はできません。

crsctl コマンドの絶対パスのみの 1 行が出力されることを確認してください。

```
# /opt/HA/AM/bin/init/get_grid_path.sh grid
/u01/app/21.3.0/grid/bin/crsctl
```

【注意】 2 行以上表示される場合は、vi エディタなどを使って、/etc/opt/HA/AM/lbin/.grid_path ファイルが crsctl コマンドの絶対パスのみの 1 行となるよう修正を行って下さい。

2　連携用リソースの作成

Oracle Clusterware と連携させる監視プロセスごとに連携用のリソースを作成します。
監視プロセスごとの連携用リソース名は以下のようになります。

モニタ制御デーモン	mcam_oramond_<構成番号>.app
PDB 監視モニタ	mcam_pdbmond_<構成番号>_<監視対象 PDB 名>.app
リスナー監視モニタ	mcam_lsnrmond_<構成番号>_<識別子>.app <識別子> ： リスナー名またはリスナー監視モニタ ID
表領域監視モニタ	mcam_tschkmond_<構成番号>_<表領域監視モニタ ID>.app
ディスク領域監視モニタ	mcam_fdsmond_<構成番号>.app
統計情報採取モニタ	mcam_statsmond_<構成番号>_<統計情報採取モニタ ID>.app
CRS 監視モニタ	mcam_crsmond.app
ASM 監視モニタ	mcam_asmmond.app
ASM リスナー監視モニタ	mcam_alsnrmond_<モニタ番号>_<ASM リスナー名>.app
SCAN リスナー監視モニタ	mcam_slsnrmond_<SCAN リスナー名>.app

Oracle Clusterware 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。
シェルスクリプトは、RAC を構成するいずれかのノードで、root ユーザーで実行してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create <XXX> [<YYY …>]
```

XXX:　監視プロセス名を指定します。省略はできません。

YYY:　監視プロセスごとのオプションです。詳細はそれぞれの監視プロセスの説明を参照してください。

監視プロセスごとの連携用リソースの作成方法は以下のとおりです。

【注意】 マニュアル表記の都合、コマンドの実行イメージと出力メッセージを複数行で記載している箇所があります。
実際のコマンドは改行なしで実行してください。
「Succeeded in create of resource. resource name is '*xxxxxxxx*.app'.」のメッセージは 1 行で出力されます。

【注意】 監視対象のデータベース名は、crsctl status resource -t コマンドで出力されるデータベース・リソース名（ora.<データベース名>.db）のデータベース名の部分です。
データベース・リソース名が **ora.rac.db** の場合は、**rac** になります。

◆ モニタ制御デーモン用リソースの作成

モニタ制御デーモンを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create oramond <XXX> [<YYY>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in create of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create oramond rac
Succeeded in create of resource. resource name is 'mcam_oramond_0.app'.
```

◆ PDB 監視モニタ用リソースの作成

PDB 監視モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create pdbmond <XXX> <YYY>
  [<ZZZ>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：PDB 名を大文字で指定します。省略はできません。

ZZZ：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in create of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create pdbmond orcl PDB1
Succeeded in create of resource. resource name is
'mcam_pdbmond_0_PDB1.app'.
```

◆ リスナー監視モニタ用リソースの作成

リスナー監視モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create lsnrmond <XXX> [<YYY>]
```

XXX：監視方式に従って監視対象のリスナー名を指定します。省略はできません。
リスナーの監視方式が、単一リスナーの監視または冗長化リスナーの状態監視の場合は、以下のとおりに指定します。

lsnr <監視対象リスナー名>

リスナーの監視方式が、冗長化リスナーの接続監視の場合は、以下のとおりに指定します。

mon <リスナー監視モニタ ID> <監視対象リスナー名[,...]>
※監視対象リスナー名は、カンマ区切りで複数指定できます。

YYY：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in create of resource. resource name is ‘<連携用リソース名>’のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create lsnrmond LISTENER
Succeeded in create of resource. resource name is
 'mcam_lsnrmond_0_LISTENER.app'.
```

◆ 表領域監視モニタ用リソースの作成

表領域監視モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create tschkmond <XXX> <YYY>
  [<ZZZ>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：表領域監視モニタ ID を指定します。省略はできません。

ZZZ：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in create of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create tschkmond rac TSM1
Succeeded in create of resource. resource name is
'mcam_tschkmond_0_TSM1.app'.
```

◆ ディスク監視モニタ用リソースの作成

ディスク領域監視モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create fdsmond <XXX> [<YYY>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in create of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create fdsmond rac
Succeeded in create of resource. resource name is 'mcam_fdsmond_0.app'.
```

◆ 統計情報採取モニタ用リソースの作成

統計情報採取モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create statsmond <XXX> [<YYY>]
  [<ZZZ>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：統計情報採取モニタ ID を指定します。省略はできません。

ZZZ：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in create of resource. resource name is ‘<連携用リソース名>’のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create statsmond rac STS1
Succeeded in create of resource. resource name is
  'mcam_statsmond_0_STS1.app'.
```

◆ CRS 監視モニタ用リソースの作成

CRS 監視モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create crsmond
```

Succeeded in create of resource. resource name is ‘<連携用リソース名>’のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create crsmond
Succeeded in create of resource. resource name is 'mcam_crsmond.app'.
```

◆ ASM 監視モニタ用リソースの作成

ASM 監視モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create asmmond
```

Succeeded in create of resource. resource name is ‘<連携用リソース名>’のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create asmmond
Succeeded in create of resource. resource name is 'mcam_ asmmond.app'.
```

◆ ASM リスナー監視モニタ用リソースの作成

ASM リスナー監視モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create alsnrmond <XXX> [YYY]
```

XXX：監視対象の ASM リスナー名を指定します。省略はできません。

YYY：ASM リスナー監視モニタのモニタ番号を指定します。モニタ番号が 0 の場合は省略できます。

Succeeded in create of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create alsnrmond ASMNET1LSNR_ASM
Succeeded in create of resource. resource name is
 'mcam_alsnrmond_0_ASMNET1LSNR_ASM.app'.
```

◆ SCAN リスナー監視モニタ用リソースの作成

SCAN リスナー監視モニタを Oracle Clusterware と連携させるためリソースを作成します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create slsnrmond [<XXX>]
```

XXX：監視対象の SCAN リスナーが使用するネットワーク番号を指定します。ネットワーク番号が 1 の場合は省略できます。

Succeeded in create of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh create slsnrmond
Succeeded in create of resource. resource name is
 'mcam_slsnrmond_LISTENER_SCAN1.app '.
Succeeded in create of resource. resource name is
 'mcam_slsnrmond_LISTENER_SCAN2.app '.
Succeeded in create of resource. resource name is
 'mcam_slsnrmond_LISTENER_SCAN3.app '.
```

3.16.2. リソースとして登録した監視プロセスの操作方法

リソースとして登録した監視プロセスの操作方法は、以下のとおりです。

1　起動と停止

Oracle Clusterware へリソースとして登録した監視プロセスは、監視対象の Oracle のリソースと連動して Oracle Clusterware から起動、停止されるため、通常は個別に起動、停止する必要はありません。

ただし、リソースとして登録した直後は停止した状態となります。監視を開始するには、登録した連携用リソースを手動で起動するか、監視対象の Oracle のリソースを再起動してください。

個別に起動または停止する場合は、 crsctl start resource コマンドまたは crsctl stop resource コマンドを使用してください。

2　一時停止と再開

監視および採取の一時停止と再開は、ApplicationMonitor の管理コマンド (oraadmin) の suspend 要求または resume 要求で行います。

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】 監視および採取の一時停止中も、 crsctl status resource コマンドで表示されるステータスは ONLINE, STABLE となります。

3　動作状態の確認

監視および採取の動作状態の確認は、ApplicationMonitor の管理コマンド (oraadmin) の show 要求で行います。

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。

3.16.3. リソースの削除方法

リソースの削除方法は、以下のとおりです。

1　連携用リソースの削除

Oracle Clusterware と連携させる監視プロセスごとに連携用のリソースを削除します。

Oracle Clusterware 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。
削除するリソースはあらかじめ停止しておく必要があります。
シェルスクリプトは、RAC を構成するいずれかのノードで、root ユーザーで実行してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop <XXX> [<YYY …>]
```

XXX:　監視プロセス名を指定します。省略はできません。

YYY:　監視プロセスごとのオプションです。詳細はそれぞれの監視プロセスの説明を参照してください。

監視プロセスごとの連携用リソースの削除方法は以下のとおりです。

【注意】 マニュアル表記の都合、コマンドの実行イメージと出力メッセージを複数行で記載している箇所があります。
実際のコマンドは改行なしで実行してください。
「Succeeded in drop of resource. resource name is '*xxxxxxxx*.app'.」の出力メッセージは 1 行で出力されます。

【注意】 監視対象のデータベース名は、crsctl status resource -t コマンドで出力されるデータベース・リソース名（ora.<データベース名>.db）のデータベース名の部分です。
データベース・リソース名が **ora.rac.db** の場合は、**rac** になります。

◆ モニタ制御デーモン用リソースの削除

モニタ制御デーモンを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop oramond <XXX> [<YYY>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in drop of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop oramond rac
Succeeded in drop of resource. resource name is 'mcam_oramond_0.app'.
```

◆ PDB 監視モニタ用リソースの削除

PDB 監視モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop pdbmond <XXX> <YYY>
  [<ZZZ>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：PDB 名を大文字で指定します。省略はできません。

ZZZ：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in drop of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop pdbmond orcl PDB1
Succeeded in drop of resource. resource name is
'mcam_pdbmond_0_PDB1.app'.
```

◆ リスナー監視モニタ用リソースの削除

リスナー監視モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop lsnrmond <XXX> [<YYY>]
```

XXX：監視方式に従って監視対象のリスナー名を指定します。省略はできません。
記述形式は、「3.16.1 リソースの登録方法」と同じです。

YYY：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in drop of resource. resource name is ‘<連携用リソース名>’のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop lsnrmond LISTENER
Succeeded in drop of resource. resource name is
'mcam_lsnrmond_0_LISTENER.app'.
```

◆ 表領域監視モニタ用リソースの削除

表領域監視モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop tschkmond <XXX> <YYY>
  [<ZZZ>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：表領域監視モニタ ID を指定します。省略はできません。

ZZZ：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in drop of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop tschkmond rac TSM1
Succeeded in drop of resource. resource name is
'mcam_tschkmond_0_TSM1.app'.
```

◆ ディスク監視モニタ用リソースの削除

ディスク領域監視モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop fdsmond <XXX> [<YYY>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in drop of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop fdsmond rac
Succeeded in drop of resource. resource name is ' mcam_fdsmond_0.app'.
```

◆ 統計情報採取モニタ用リソースの削除

統計情報採取モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop statsmond <XXX> [<YYY>]
  [<ZZZ>]
```

XXX：監視対象のデータベース名を指定します。省略はできません。

YYY：統計情報採取モニタ ID を指定します。省略はできません。

ZZZ：構成番号（COMPONENT_ID パラメータ）を指定します。構成番号が 0 の場合は省略できます。

Succeeded in drop of resource. resource name is ‘<連携用リソース名>’のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop statsmond rac STS1
Succeeded in drop of resource. resource name is
  ' mcam_statsmond_0_STS1.app'.
```

◆ CRS 監視モニタ用リソースの削除

CRS 監視モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop crsmond
```

Succeeded in drop of resource. resource name is ‘<連携用リソース名>’のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop crsmond
Succeeded in drop of resource. resource name is ' mcam_crsmond.app'.
```

◆ ASM 監視モニタ用リソースの削除

ASM 監視モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop asmmond
```

Succeeded in drop of resource. resource name is ‘<連携用リソース名>’のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop asmmond
Succeeded in drop of resource. resource name is ' mcam_ asmmond.app'.
```

◆ ASM リスナー監視モニタ用リソースの削除

ASM リスナー監視モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop alsnrmond <XXX> [<YYY>]
```

XXX：監視対象の ASM リスナー名を指定します。省略はできません。

YYY：ASM リスナー監視モニタのモニタ番号を指定します。モニタ番号が 0 の場合は省略できます。

Succeeded in drop of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop alsnrmond ASMNET1LSNR_ASM
Succeeded in drop of resource. resource name is
 'mcam_alsnrmond_0_ASMNET1LSNR_ASM.app'.
```

◆ SCAN リスナー監視モニタ用リソースの削除

SCAN リスナー監視モニタを Oracle Clusterware と連携させるためリソースを削除します。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop slsnrmond [<XXX>]
```

XXX：監視対象の SCAN リスナーが使用するネットワーク番号を指定します。ネットワーク番号が 1 の場合は省略できます。

Succeeded in drop of resource. resource name is '<連携用リソース名>'のメッセージが出力されることを確認してください。

```
# /opt/HA/AM/bin/init/init_ocw_resource.sh drop slsnrmond
Succeeded in drop of resource. resource name is
 'mcam_slsnrmond_LISTENER_SCAN1.app '.
Succeeded in drop of resource. resource name is
 'mcam_slsnrmond_LISTENER_SCAN2.app '.
Succeeded in drop of resource. resource name is
 'mcam_slsnrmond_LISTENER_SCAN3.app '.
```

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 からフェイルオーバ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)、PDB 監視モニタ (pdbmond)、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、※3）します。

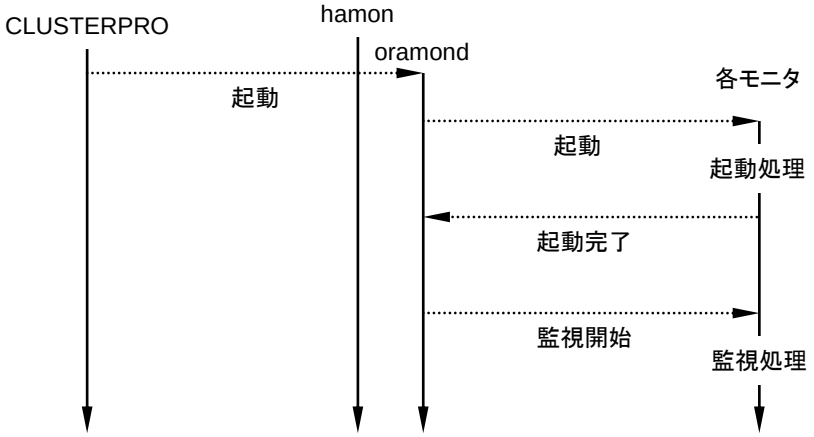


図 4-1 各モニタ起動時の動作

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、「4.7 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。

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、モニタ制御デーモンは、監視モニタの障害として扱います。

障害検出時、モニタ制御デーモン停止のタイミングで、CLUSTERPRO 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 (crsmond)、ASM 監視モニタ (asmmond) の CLUSTERPRO 連携時の動作についても同様です。

モニタ制御デーモンは、管理コマンド (oraadmin) から STOP リクエストを実行、または 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。

（※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。

（※3） 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た場合は、各モニタは起動すると(oramond<構成番号>.bin) を読み込み、設定にしたがって監視および採取を行います。
単独起動モニタの詳細は、「3.2.17 監視モニタ制御の設定」を参照してください。

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（※2）、ユーザー指定表の参照（※3）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。

（※2） SYSTEM 表領域上のオブジェクトの更新／参照の指定は、SQL_LEVEL パラメータで設定できます。

（※3） マルチテナント構成の場合、CDB 上のユーザー指定表の監視はできません。

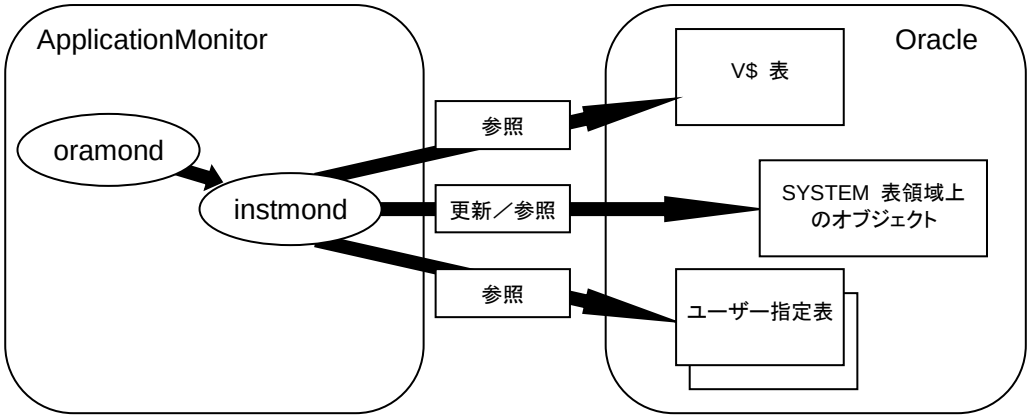


図 4-2 インスタンス監視モニタの概要

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。

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    = 90
    GET_STATS        = YES
}
```

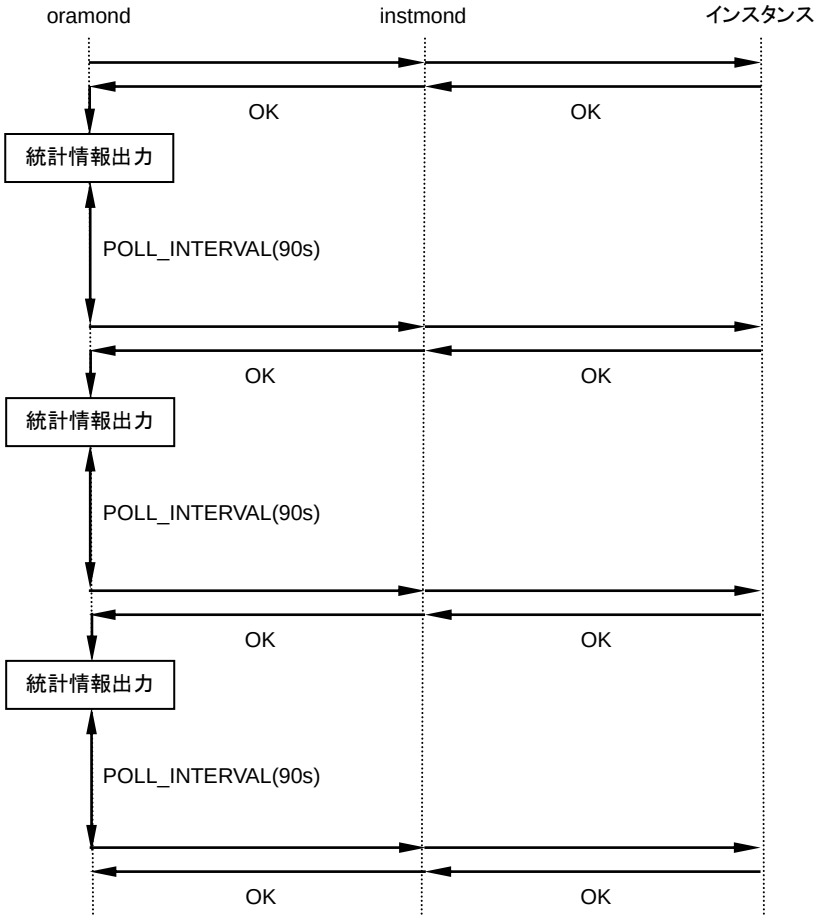


図 4-3 インスタンス監視時の動作

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）

```
HALT_CONTROL_OPTION      = ALWAYS

InstanceMonitor {
    POLL_INTERVAL          = 90
    POLL_DOWN_RETRY        = 2
    SERVICE_DOWN           = YES
    RESTART_COUNT          = 1
    RESTART_TIMEOUT        = 60
}
```

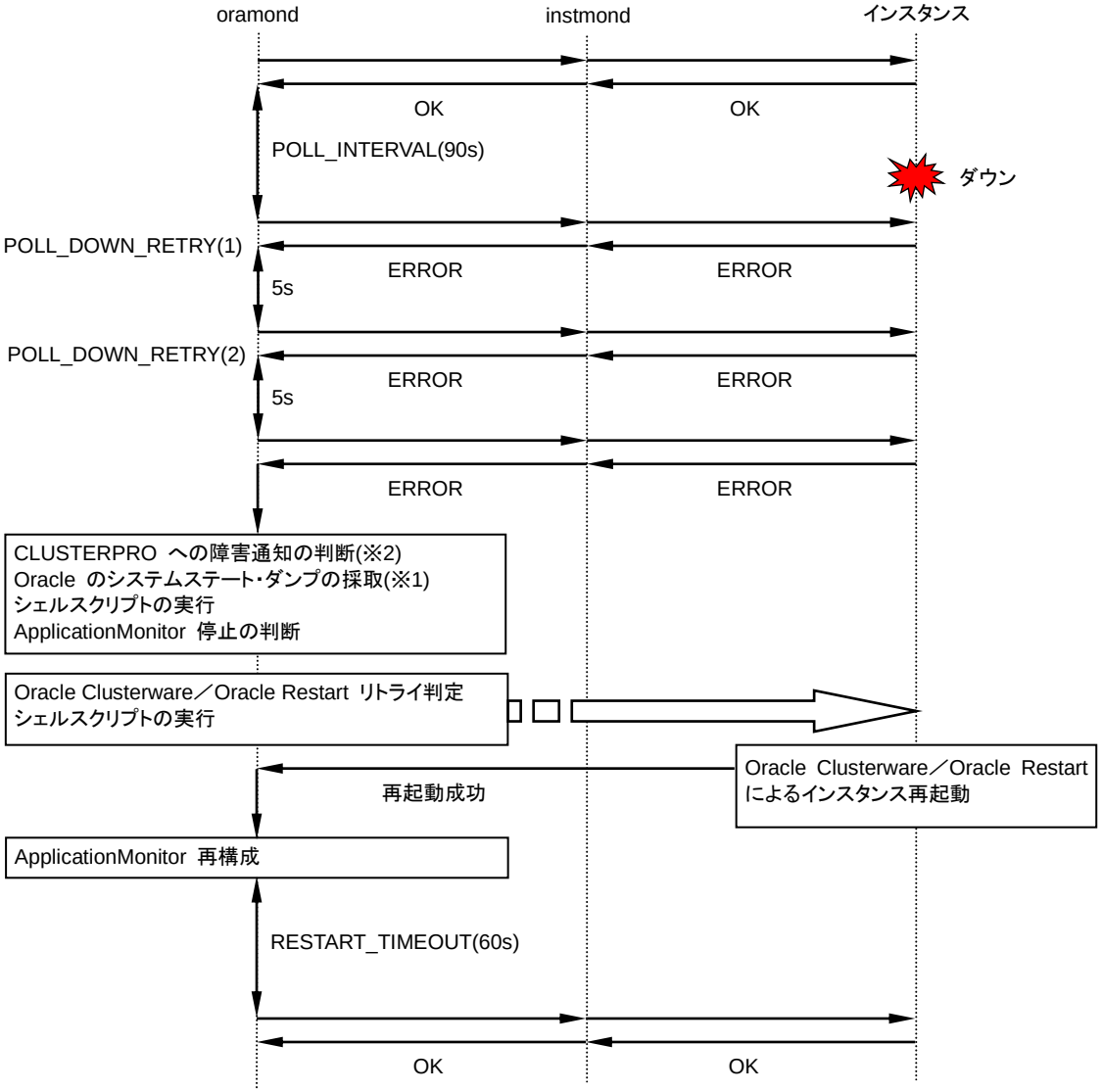


図 4-4 インスタンス監視エラー時の動作（Oracle Clusterware／Oracle Restart によってインスタンスが再起動された場合）

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、監視を継続します。

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 へノード内の障害を通知するかの判断、Oracle のシステムステート・ダンプの採取（※1）およびシェルスクリプトを実行し、情報採取を行います。

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 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（※1）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、「4.13 Oracle のシステムステート・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、「4.14 シェルスクリプトの実行時の動作」を参照してください。

インスタンス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（※2）。

なお、インスタンス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 (oramond) 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（※3）。

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 を停止すると判断し、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプト (crs_control.sh) を実行します。

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 を停止しないと判断し、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトを実行せずに ApplicationMonitor を再構成し、監視を継続します。

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、「4.15 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 の再構成後、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） 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 構成の非障害ノードでのみ行われます。

（※2） 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。

（※3） Oracle Clusterware／Oracle Restart によるインスタンスの再起動が失敗した場合、監視を継続します。

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          = 90
    POLL_DOWN_RETRY        = 2
    SERVICE_DOWN           = YES
    RESTART_COUNT          = 1
}
```

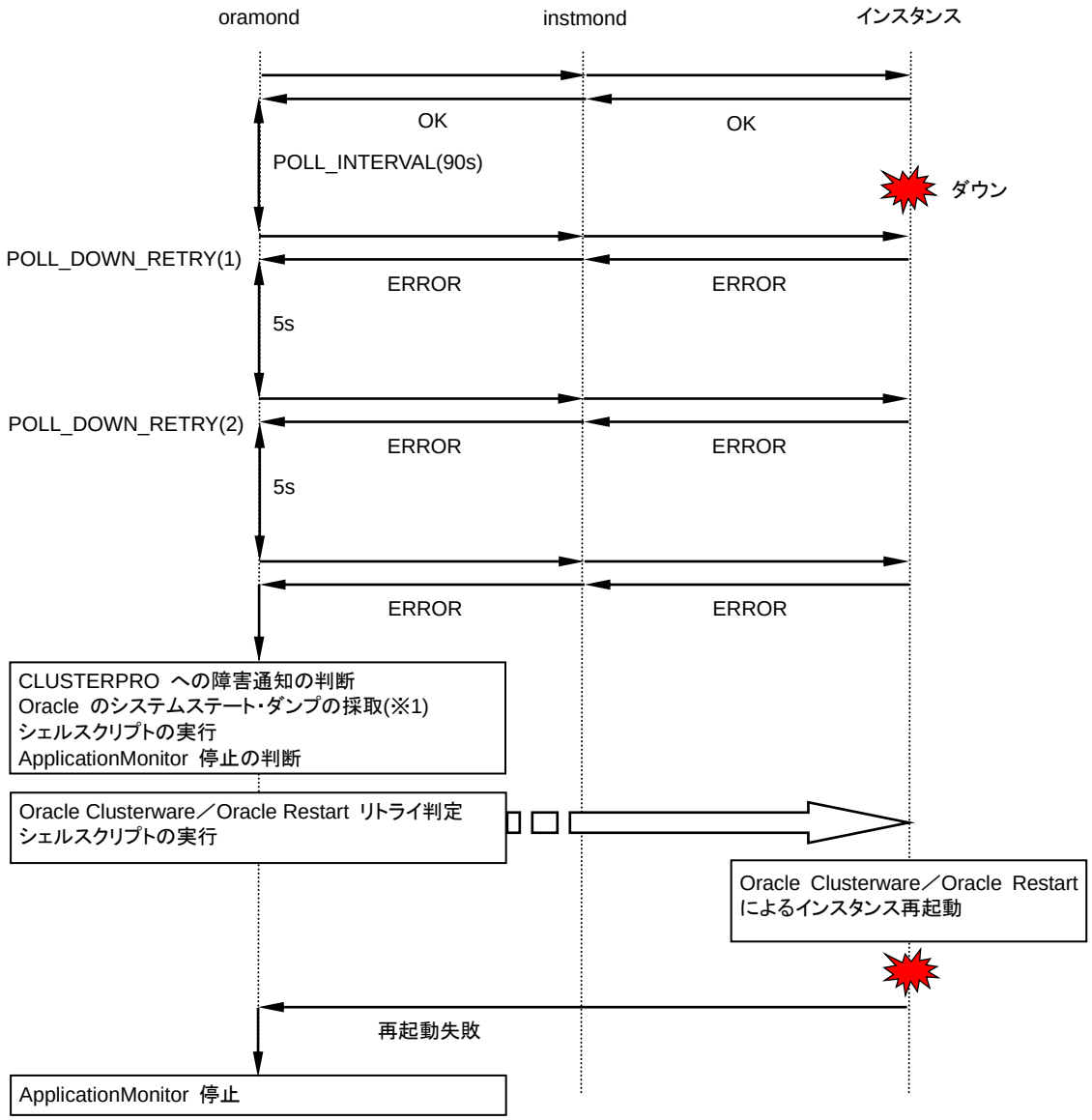


図 4-5 インスタンス監視エラー時の動作（Oracle Clusterware／Oracle Restart によってインスタンスが再起動されなかった場合）

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、障害ノード上の ApplicationMonitor を停止し、CLUSTERPRO へ障害を通知します。

（※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 構成の非障害ノードでのみ行われます。

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）

```
HALT_CONTROL_OPTION     = ALWAYS

InstanceMonitor {
    POLL_INTERVAL          = 90
    POLL_STALL_RETRY       = 2
    SERVICE_DOWN           = YES
    RESTART_COUNT          = 1
    RESTART_TIMEOUT        = 60
}
```

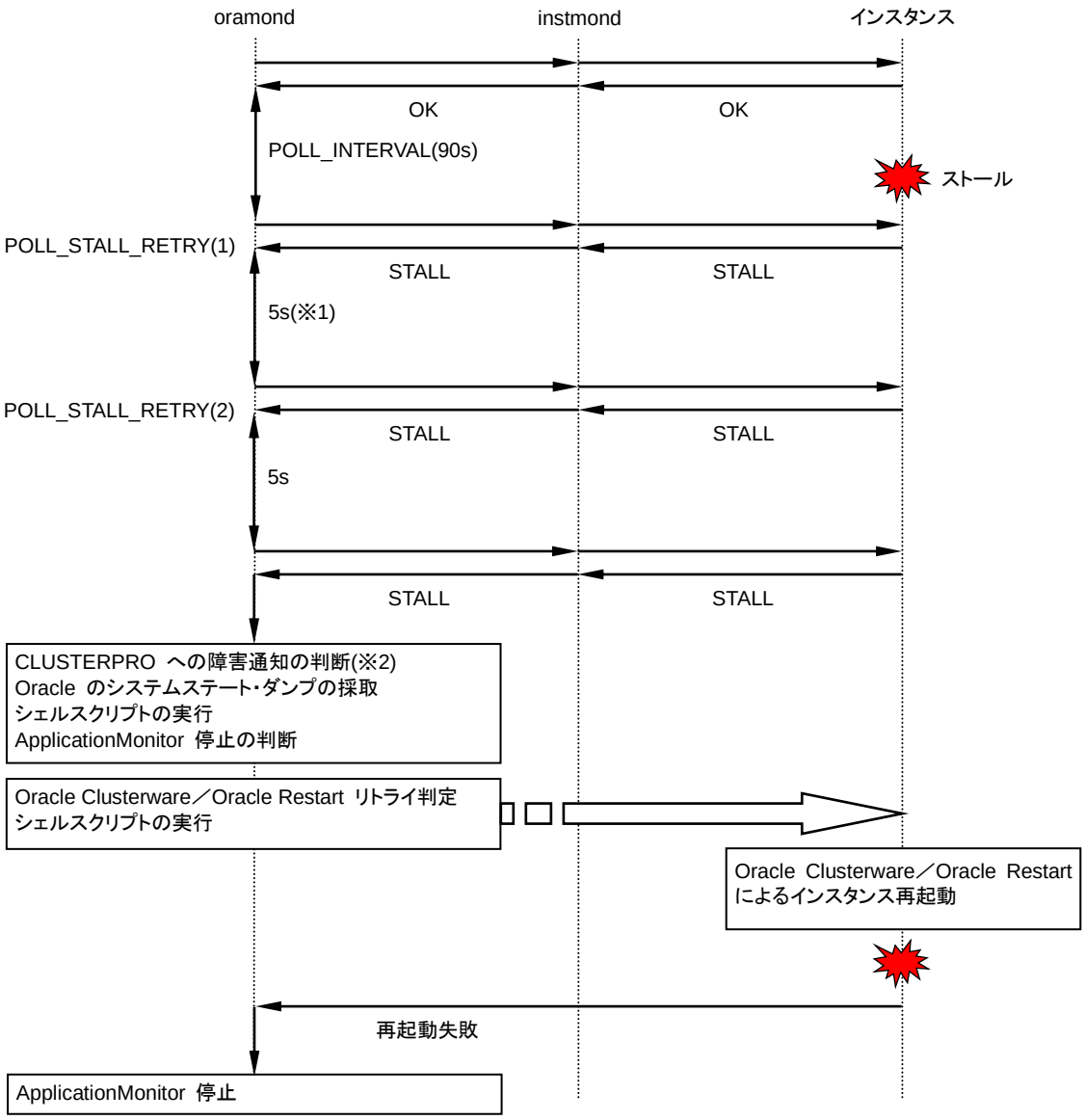


図 4-6 Oracle バックグラウンドプロセスのストール時の動作

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）

```
HALT_CONTROL_OPTION     = ALWAYS

InstanceMonitor {
    POLL_INTERVAL          = 90
    POLL_TIMEOUT           = 120
    POLL_STALL_RETRY       = 1
    SERVICE_DOWN           = NO
    RESTART_COUNT          = 1
}
```

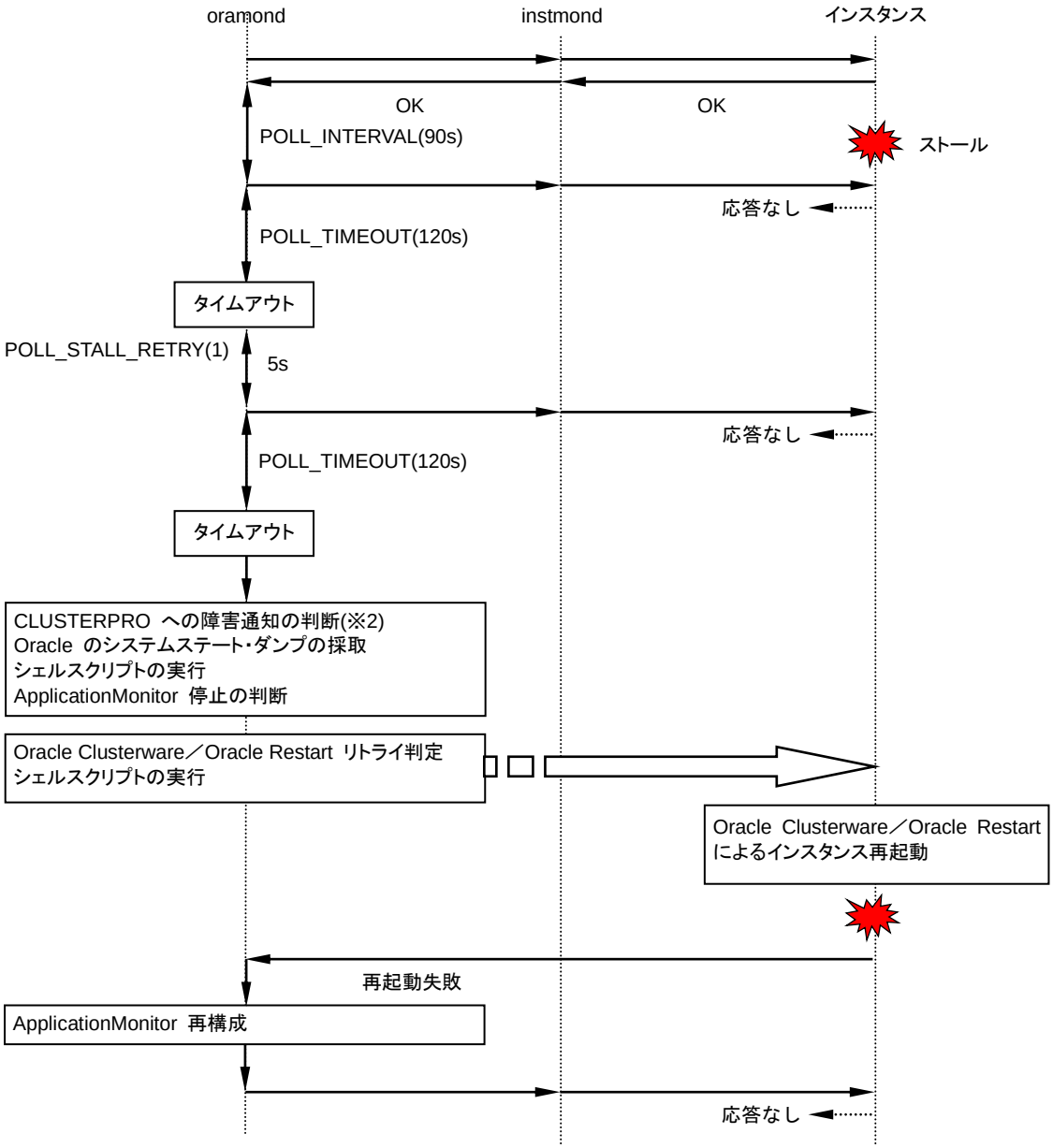


図 4-7 インスタンス監視タイムアウト時の動作

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、監視を継続します。

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を実行し、情報採取を行います。

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、「4.13 Oracle のシステムステート・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、「4.14 シェルスクリプトの実行時の動作」を参照してください。

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（※2）。

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、監視を継続します（※3）。

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 を停止すると判断し、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトを実行します。

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 を停止しないと判断し、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトを実行せずに ApplicationMonitor を再構成し、監視を継続します。

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、「4.15 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 の再構成後、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。

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、障害ノード上の ApplicationMonitor を停止し、CLUSTERPRO へ障害を通知します。

（※1） ORACLE_HANG_CHECK パラメータを YES に設定すると、Oracle ハング検知機能の結果を確認するため、20 秒要する場合があります。

（※2） 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。

（※3） Oracle Clusterware／Oracle Restart によるインスタンスの再起動が失敗した場合、監視を継続します。

4.3. PDB 監視時の動作

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

（※1） SYSTEM 表領域上のオブジェクトの更新／参照の指定は、SQL_LEVEL パラメータで設定できます。

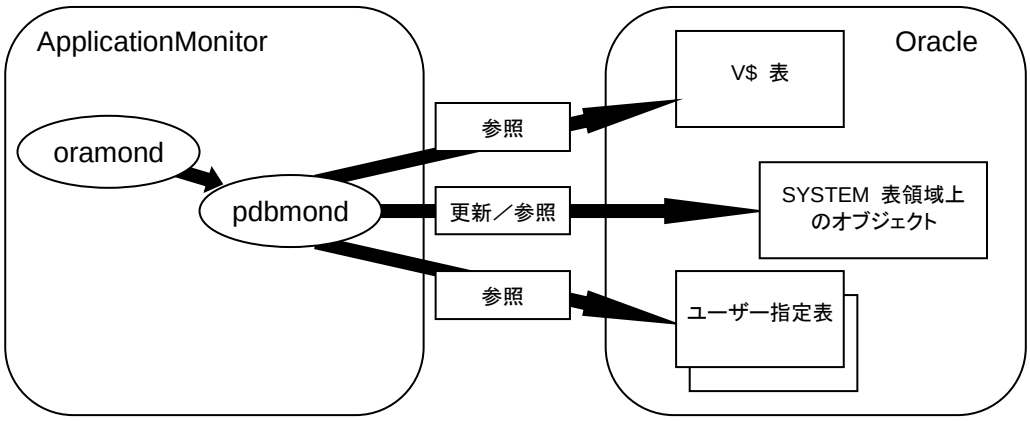


図 4-8 PDB 監視モニタの概要

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_TIME_CHK[0～n]	テーブル	PDB 監視モニタのアクセス時間を書き込むオブジェクト（PDB ごとに異なるオブジェクトを作成します）。

PDB の監視は、以下のような動作になります。

（例）

```
PluggableDatabaseMonitor {
    PluggableDatabase PDB {
        POLL_INTERVAL           = 90
        PDB_OPEN_TIMEOUT        = 60
        UNOPEN_PDB_DOWN         = NO
    }
}
```

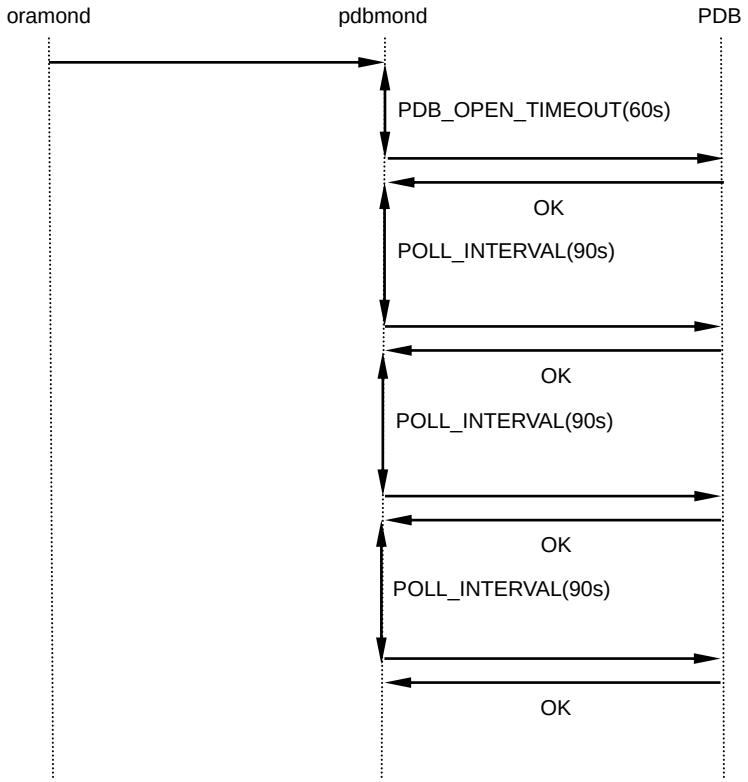


図 4-9 PDB 監視時の動作

PDB 監視モニタは、POLL_INTERVAL パラメータに設定した間隔で PDB にアクセスします。

なお、監視対象の PDB がマウント状態になった場合、UNOPEN_PDB_DOWN パラメータに NO が設定されていると、PDB 監視モニタは監視を中断します。監視対象の PDB がオープンされた際、監視を再開します。

UNOPEN_PDB_DOWN パラメータに YES が設定されていると、PDB 監視モニタは、PDB 障害と判断し、エラーを検出した場合の動作を行います。

また、OPEN_PDB_TIMEOUT パラメータで、監視対象 PDB の起動に合わせて、監視を開始するタイミングを調整することができます。

PDB の監視でエラーを検出した場合、以下のような動作になります。

（例）

```
PluggableDatabaseMonitor {
    PluggableDatabase PDB {
        MONITOR_USER          = oracle
        POLL_INTERVAL         = 90
        POLL_DOWN_RETRY       = 2
        SERVICE_DOWN          = NO
        PDB_OVER_ACTION       = NEVER
        PDB_OPEN_TIMEOUT      = 0
    }
}
```

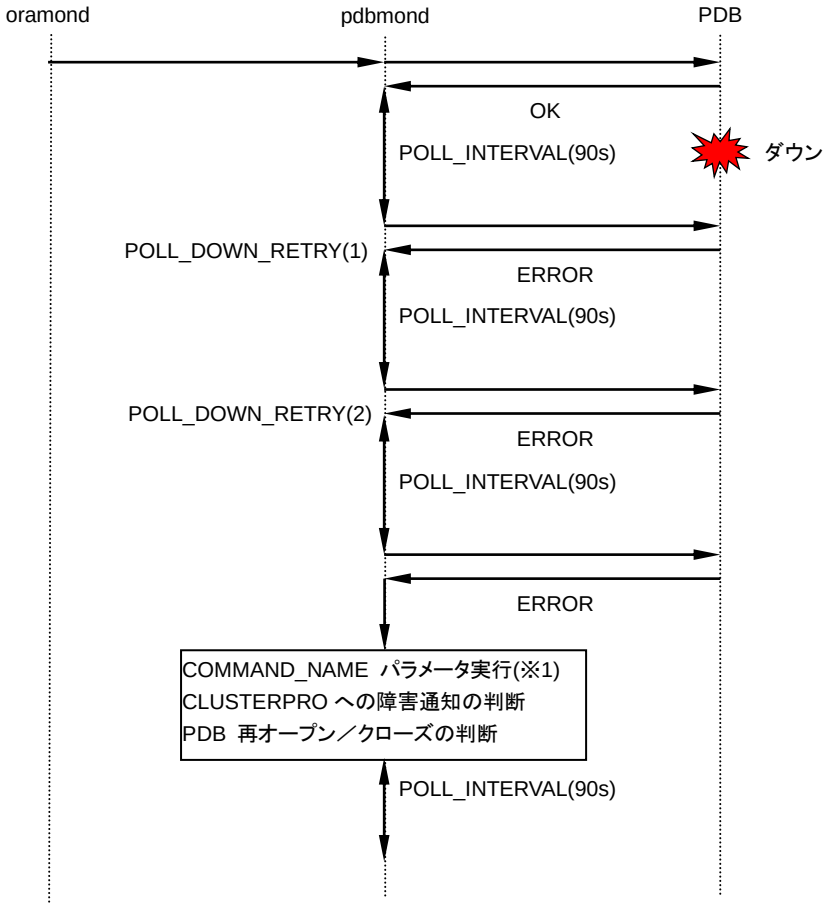


図 4-10 PDB 監視エラー時の動作

PDB 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 PDB の再確認を行います。再確認中に PDB 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。

POLL_DOWN_RETRY パラメータに設定した回数の PDB の再確認を行っても、PDB 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の実行（※1）および SERVICE_DOWN パラメータの設定により、CLUSTERPRO への障害通知を行うかの判断を行

います。

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 COMMAND_NAME パラメータに設定したコマンドの実行および CLUSTERPRO への障害通知の判断を行います。

CLUSTERPRO への障害通知を行わない場合には、PDB_OVER_ACTION パラメータの設定により、障害を検出した PDB を再オープンまたはクローズするかの判断を行います。

なお、CLUSTERPRO への障害通知を行う場合には、 PDB の再オープン／クローズの判断以降の処理は行わずに、インスタンス監視の障害検出時の Oracle のシステムステート・ダンプの採取以降の処理を実行します。

PDB_OVER_ACTION パラメータに NEVER が設定されていると、障害を検出した PDB の再オープン／クローズは行いません。

なお、PDB_OVER_ACTION パラメータに REOPEN が設定されていると、PDB 再オープンシェルスクリプト (am_pdb_reopen.sh) を実行して障害を検出した PDB の再オープンを行います。また、PDB_OVER_ACTION パラメータに CLOSE が設定されていると、PDB 再オープンシェルスクリプトを実行して障害を検出した PDB のクローズを行います。

PDB 再オープンシェルスクリプトの詳細は、「4.16 PDB 再オープンシェルスクリプトの動作」を参照してください。

障害を検出した PDB を再オープン／クローズしなかった場合は、POLL_INTERVAL パラメータに設定した間隔で PDB の監視を継続します。

（※1） MONITOR_USER パラメータに設定されたユーザーで実行します。

PDB 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）

```
PluggableDatabaseMonitor {
    PluggableDatabase PDB {
        MONITOR_USER          = oracle
        POLL_INTERVAL         = 90
        POLL_TIMEOUT          = 120
        POLL_STALL_RETRY      = 1
        SERVICE_DOWN          = NO
        PDB_OVER_ACTION       = REOPEN
        PDB_OPEN_TIMEOUT      = 0
    }
}
```

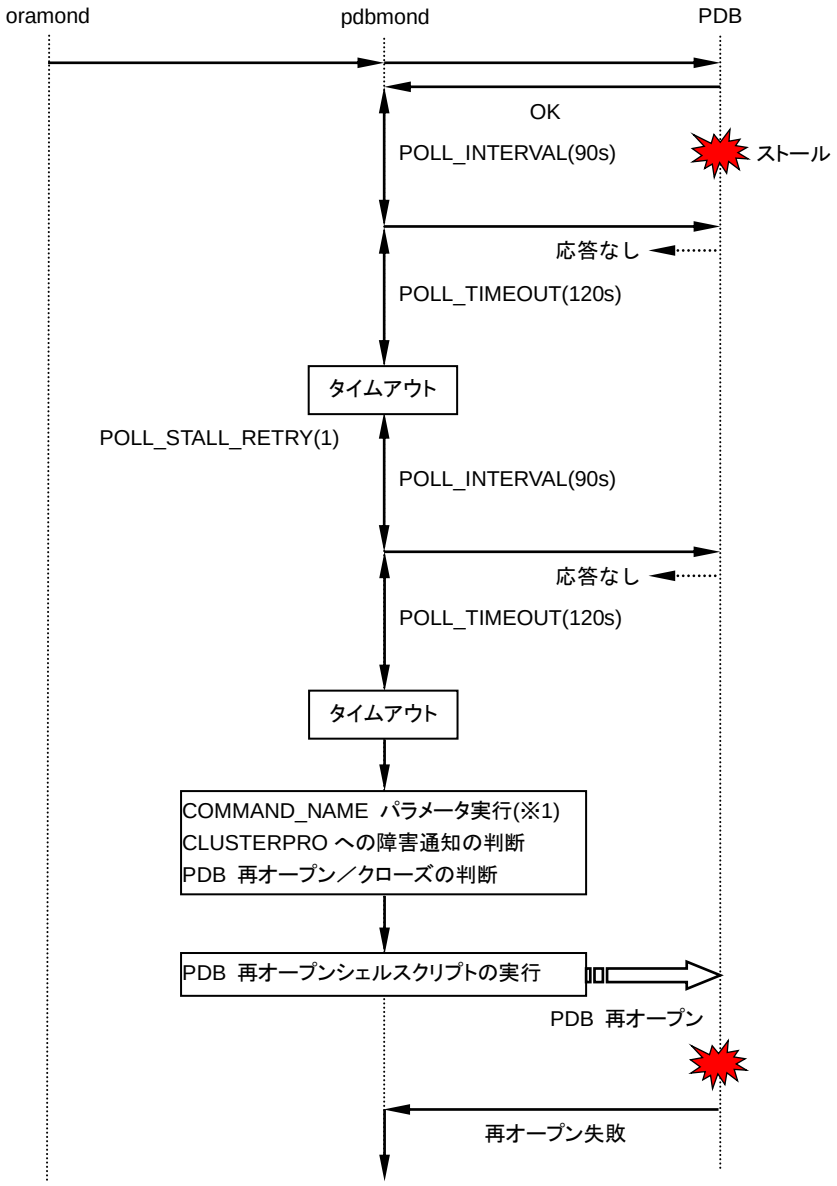


図 4-11 PDB 監視タイムアウト時の動作

PDB 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 PDB の再確認を行います。再確認中に PDB 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。

POLL_STALL_RETRY パラメータに設定した回数の PDB の再確認を行っても、PDB へのアクセスがタイムアウトする場合は、COMMAND_NAME パラメータに設定したコマンドの実行（※1）および SERVICE_DOWN パラメータの設定により、CLUSTERPRO への障害通知を行うかの判断を行います。

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 COMMAND_NAME パラメータに設定したコマンドの実行および CLUSTERPRO への障害通知の判断を行います。

CLUSTERPRO への障害通知を行わない場合には、PDB_OVER_ACTION パラメータの設定により、障害を検出した PDB を再オープン／クローズするかの判断を行います。

なお、CLUSTERPRO への障害通知を行う場合には、 PDB の再オープン／クローズの判断以降の処理は行わずに、インスタンス監視の障害検出時の Oracle のシステムステート・ダンプの採取以降の処理を実行します。

PDB_OVER_ACTION パラメータに REOPEN が設定されていると、PDB 再オープンシェルスクリプトを実行して障害を検出した PDB の再オープンを行います。

なお、PDB_OVER_ACTION パラメータに CLOSE が設定されていると、PDB 再オープンシェルスクリプトを実行して障害を検出した PDB のクローズを行います。

また、PDB_OVER_ACTION パラメータに NEVER が設定されていると、PDB 再オープンシェルスクリプトを実行せずに、POLL_INTERVAL パラメータに設定した間隔で PDB の監視を継続します。

PDB 再オープンシェルスクリプトの詳細は、「4.16 PDB 再オープンシェルスクリプトの動作」を参照してください。

PDB 再オープンシェルスクリプトで PDB の再オープンを実行後は、PDB 再オープンの結果にかかわらず監視を継続します。

（※1） MONITOR_USER パラメータに設定されたユーザーで実行します。

4.4. リスナー監視時の動作

4.4.1. 単一リスナーの監視

リスナー監視モニタ (lsnrmond) は、単一リスナーの監視時、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。tnspingコマンド実行時、NET_SERVICE_NAME パラメータに設定されたネット・サービス名を使用します。NET_SERVIEC_NAME パラメータには、1つのリスナーで構成されたネット・サービス名を指定してください。

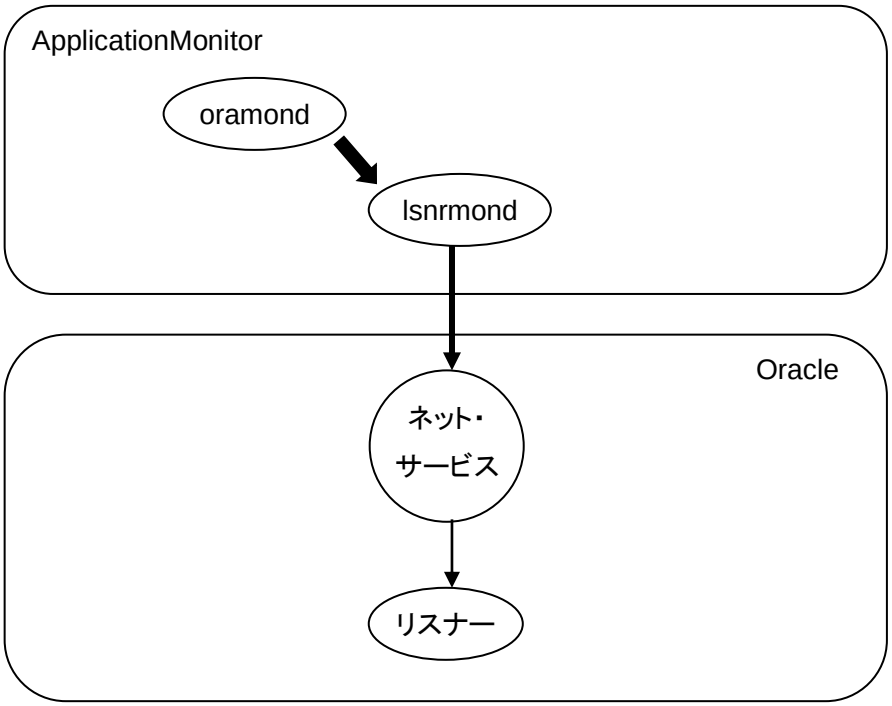


図 4-12 リスナー監視モニタ(単一リスナ監視)の概要

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
}
```

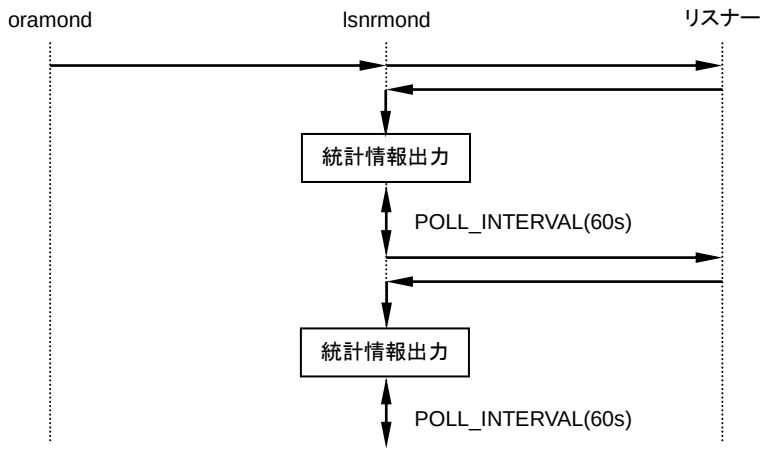


図 4-13 リスナー監視時の動作

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。

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          = 5
    RESTART_INTERVAL       = 3
    NET_SERVICE_NAME       = LSNR
    RESTART_TIMEOUT        = 60
    TARGET_RESTART         = YES
}
```

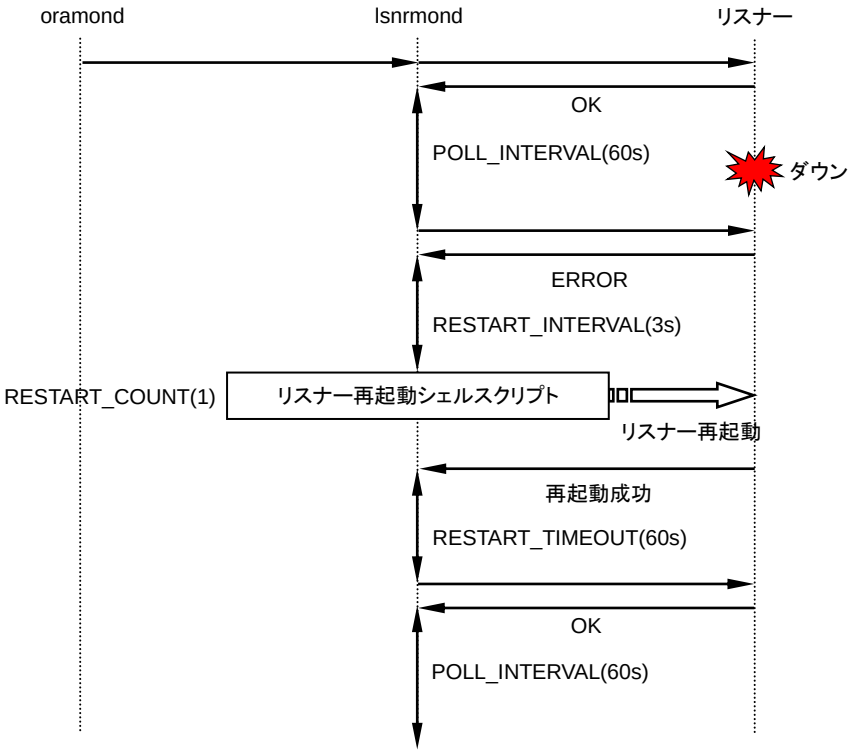


図 4-14 リスナー監視エラー時の動作（ApplicationMonitor によってリスナーが再起動した場合）

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再起動シェルスクリプト (lsnr_control.sh) を実行します。

リスナー再起動シェルスクリプトの詳細は、「4.17 リスナー再起動シェルスクリプトの動作」を参照してください。

リスナーが再起動した場合は、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
    TARGET_RESTART         = YES
}
```

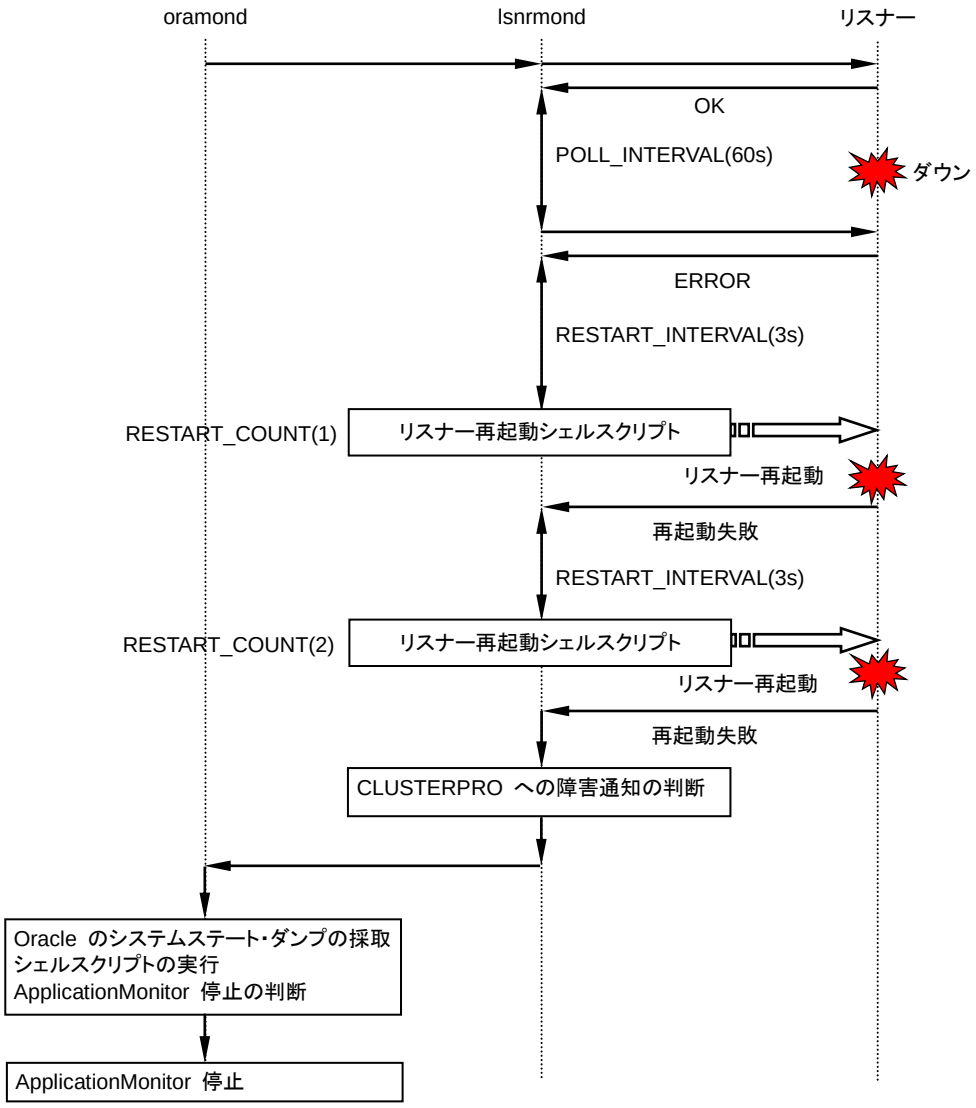


図 4-15 リスナー監視エラー時の動作（ApplicationMonitor によるリスナー再起動が失敗した場合）

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 へノード内の障害を通知するか判断します。

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を行います。

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。

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、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。

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 (oramond) 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。

Oracle のシステムステート・ダンプの採取動作の詳細は、「4.13 Oracle のシステムステート・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、「4.14 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          = 5
    RESTART_INTERVAL       = 3
    NET_SERVICE_NAME       = LSNR
    RESTART_TIMEOUT        = 60
    TARGET_RESTART         = NO
}
```

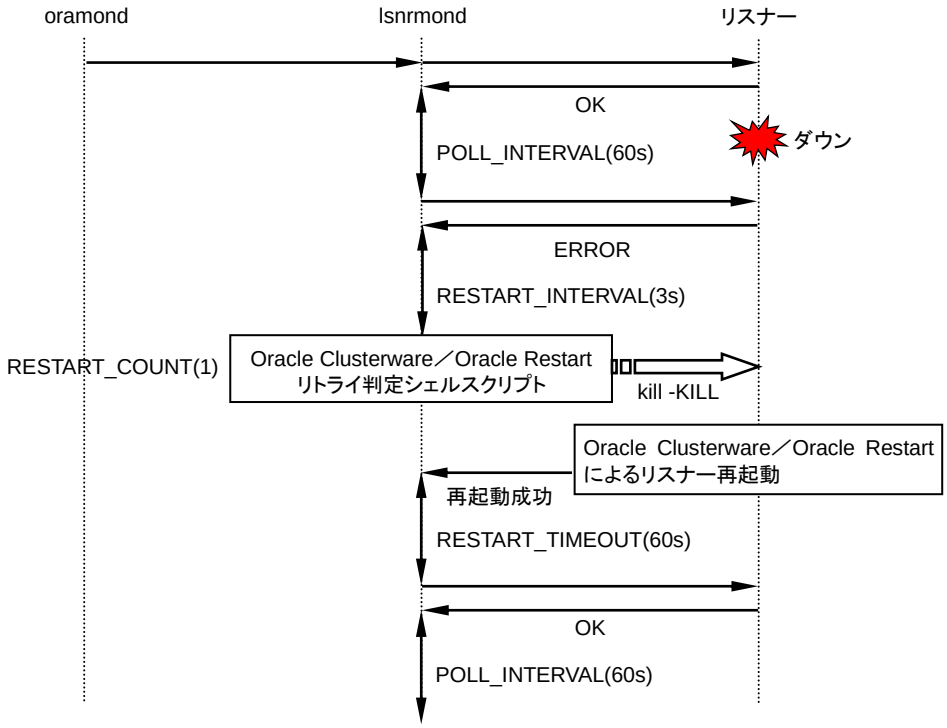


図 4-16 リスナー監視エラー時の動作（Oracle Clusterware／Oracle Restart によってリスナーが再起動された場合）

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、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプト (crs_control.sh) を実行し、Oracle Clusterware／Oracle Restart によるリスナーの再起動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、「4.15 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。

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。

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
    TARGET_RESTART         = NO
}
```

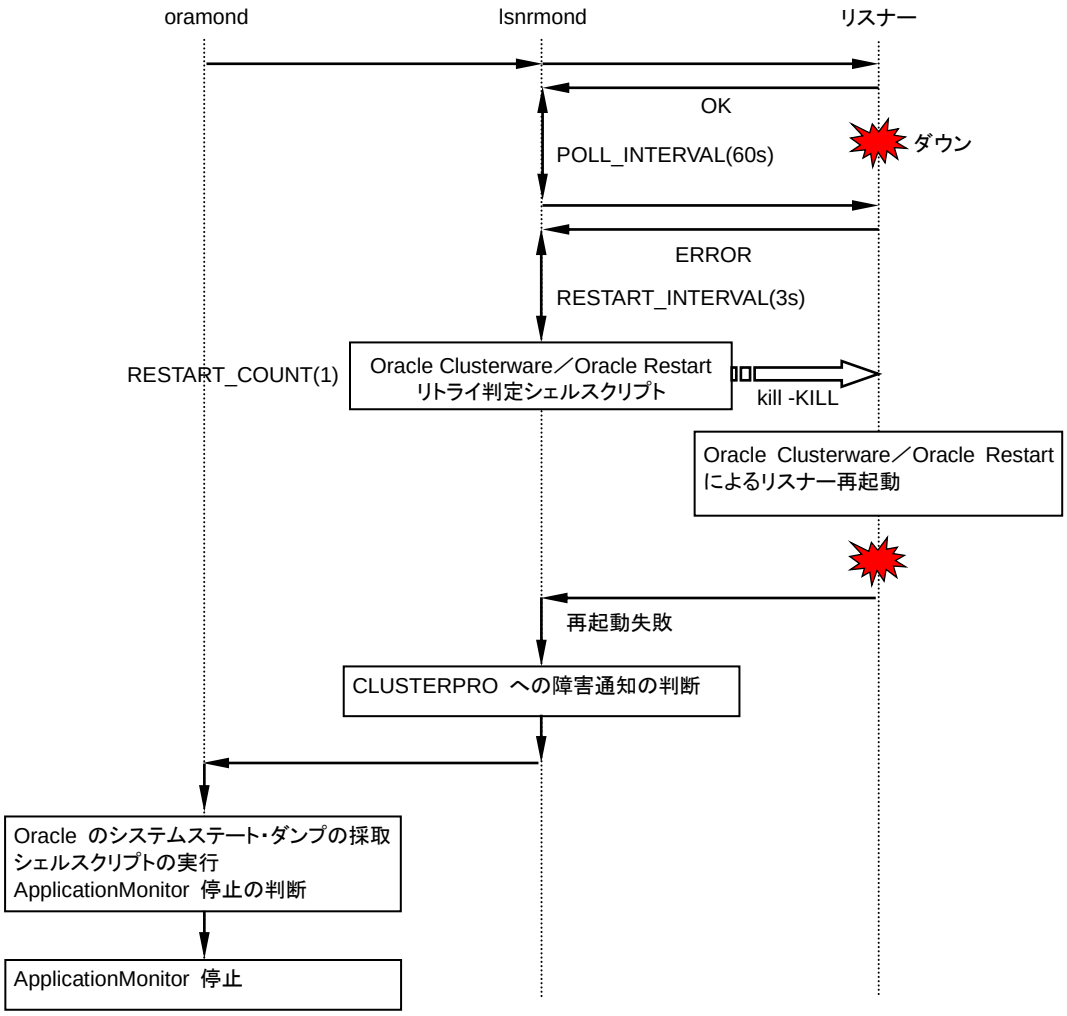


図 4-17 リスナー監視エラー時の動作（Oracle Clusterware／Oracle Restart によってリスナーが再起動されなかった場合）

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。

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を行います。

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。
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、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。

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。
Oracle のシステムステート・ダンプの採取動作の詳細は、「4.13 Oracle のシステムステート・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、「4.14 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、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          = 1
    RESTART_INTERVAL       = 3
    NET_SERVICE_NAME       = LSNR
    RESTART_TIMEOUT        = 60
    TARGET_RESTART         = YES
}
```

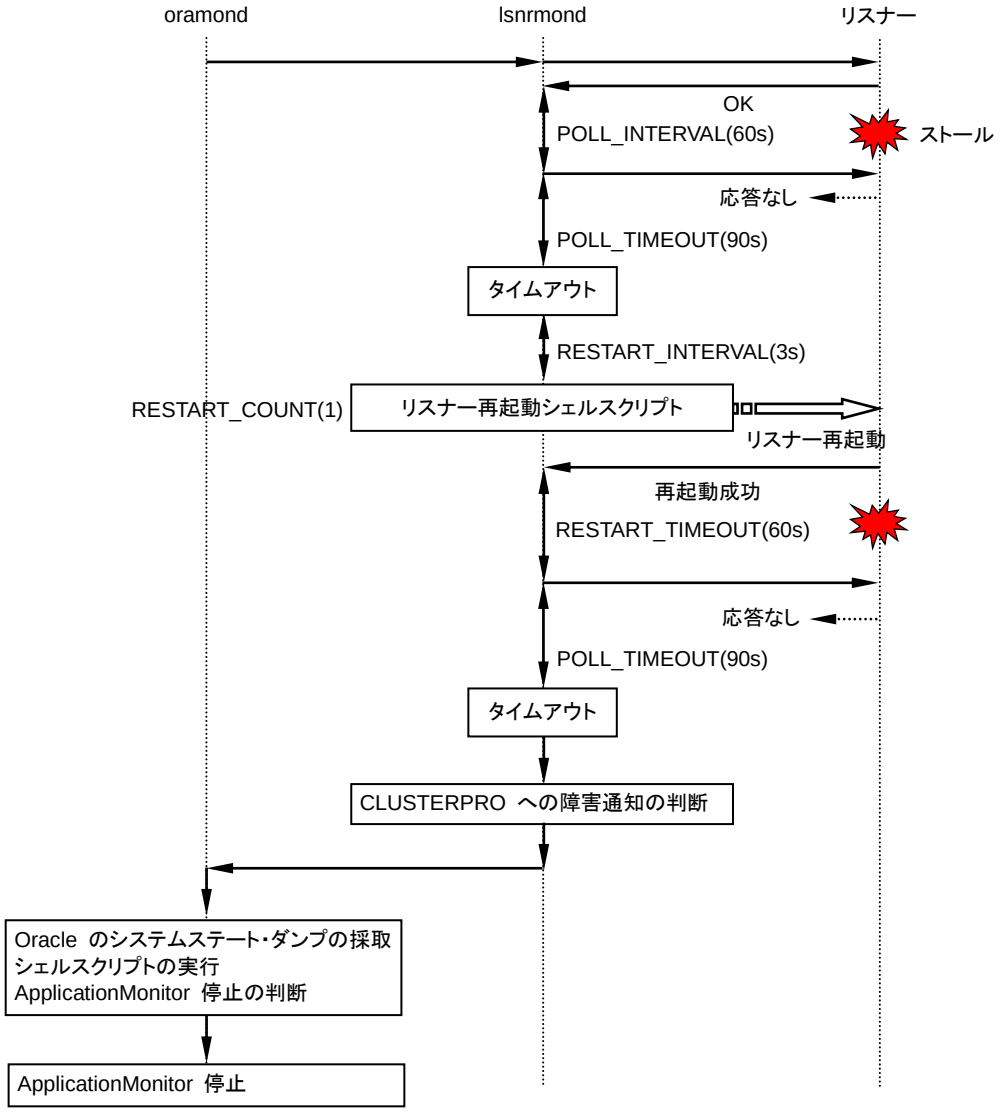


図 4-18 リスナー監視タイムアウト時の動作（ApplicationMonitor によるリスナー再起動が失敗した場合）

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再起動シェルスクリプトを実行します。

リスナー再起動シェルスクリプトの詳細は、「4.17 リスナー再起動シェルスクリプトの動作」を参照してください。

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。

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を行います。

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。

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、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。

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。

Oracle のシステムステート・ダンプの採取動作の詳細は、「4.13 Oracle のシステムステート・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、「4.14 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
    TARGET_RESTART         = NO
}
```

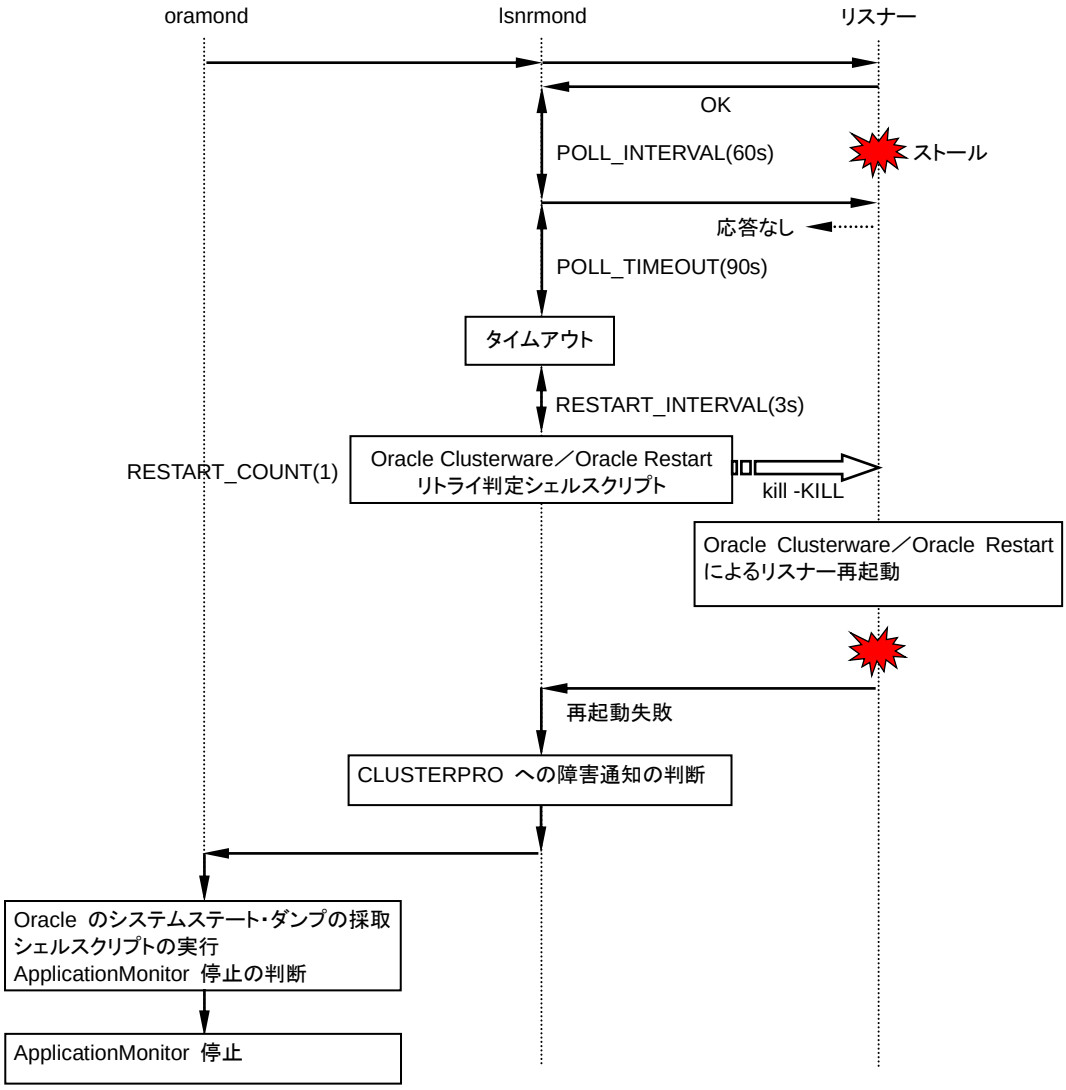


図 4-19 リスナー監視タイムアウト時の動作（Oracle Clusterware／Oracle Restart によってリスナーが再起動されなかった場合）

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。

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を行います。

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。

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、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。

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。

Oracle のシステムステート・ダンプの採取動作の詳細は、「4.13 Oracle のシステムステート・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、「4.14 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。

4.4.2. 冗長化リスナーの接続監視

リスナー監視モニタ (lsnrmond) は、冗長化リスナーの接続監視時、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。tnsping コマンド実行時、NET_SERVICE_NAME パラメータに設定されたネット・サービス名を使用します。NET_SERVIEC_NAME パラメータには、複数のリスナーで構成されたネット・サービス名を指定してください。

冗長化リスナーの接続監視では、障害検出時、監視対象リスナーの再起動は行われません。監視対象リスナーの再起動を行う場合は、冗長化リスナーの状態監視を併用してください。

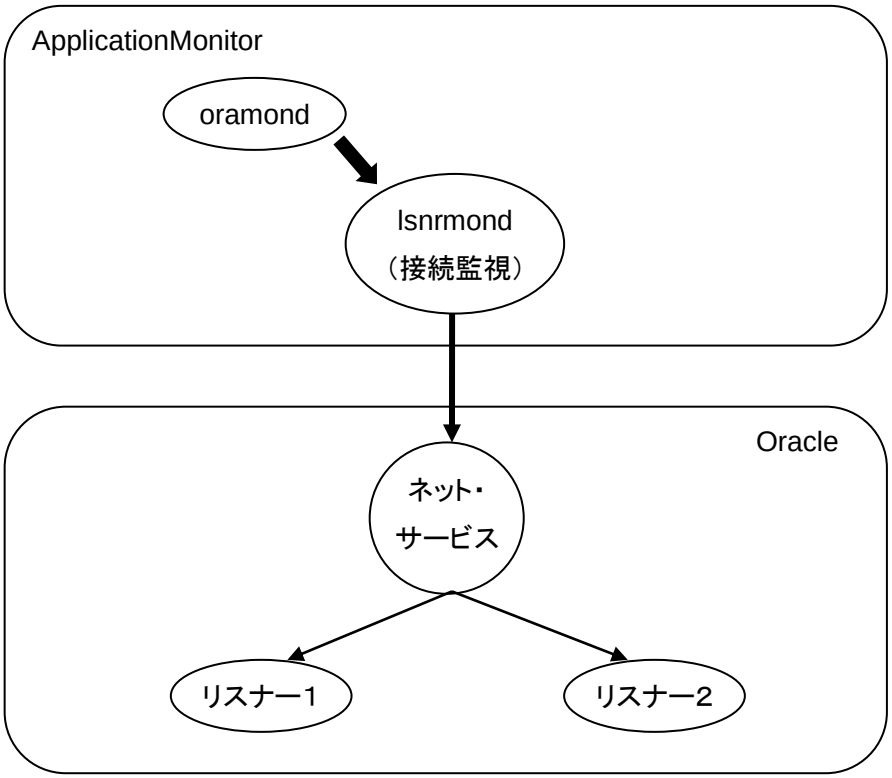


図 4-20 リスナー監視モニタ（冗長化リスナーの接続監視）の概要

冗長化リスナーの接続監視は、以下のような動作になります。

（例）

```
ListenerMonitor LISTENER_MULTI {
    LSNR_CHECK_MODE     = PING_NO_RESTART
    POLL_INTERVAL       = 60
    NET_SERVICE_NAME    = MULTI
    GET_STATS           = YES
}
```

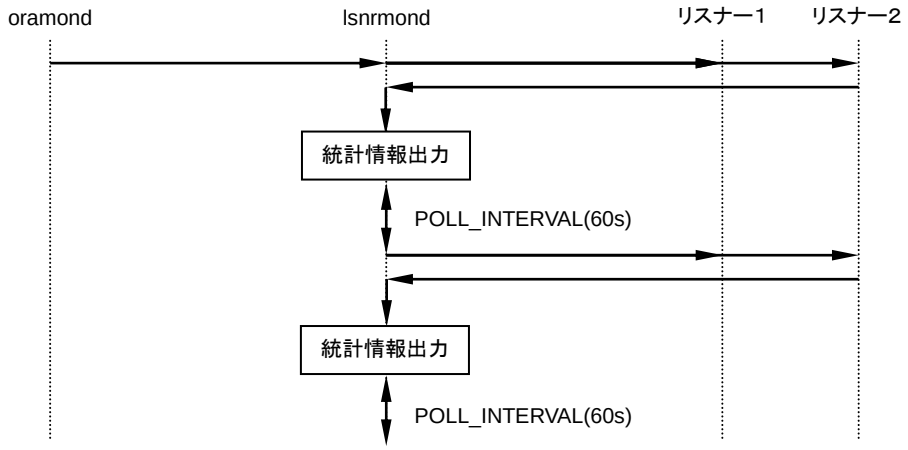


図 4-21 冗長化リスナー接続監視時の動作

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。

冗長化リスナーの接続監視でエラーを検出した場合、以下のような動作になります。

（例）

```
HALT_CONTROL_OPTION     = ALWAYS

ListenerMonitor LISTENER_MULTI {
    LSNR_CHECK_MODE     = PING_NO_RESTART
    POLL_INTERVAL       = 60
    SERVICE_DOWN        = YES
    RETRY_COUNT         = 1
    NET_SERVICE_NAME    = MULTI
}
```

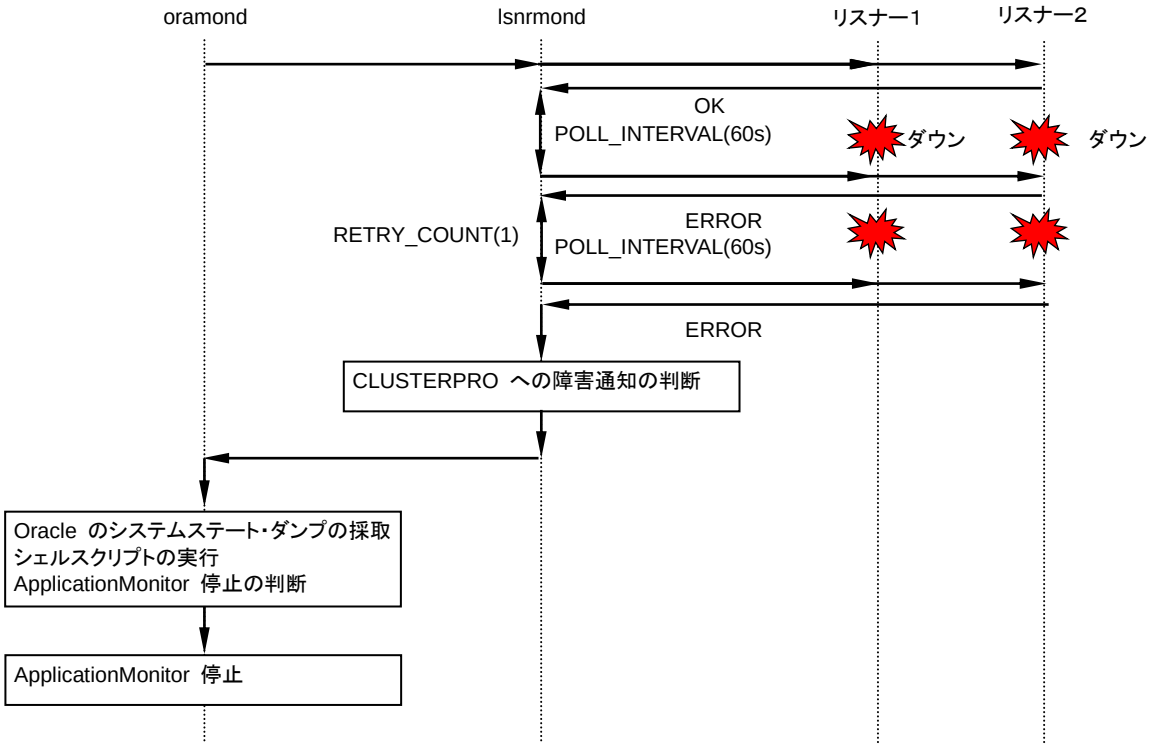


図 4-22 冗長化リスナー接続監視エラー時の動作

冗長化リスナーの接続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リトライを行います。設定した回数のリトライを行ってもリスナーの状態が改善されない場合は、SERVICE_DOWN パラメータの設定により、CLUSTERPRO への通知判定および ApplicationMonitor 停止判定を行います。

SERVICE_DOWN パラメータに YES または NO_STALL が設定されていると、CLUSTERPRO へ障害を通知します。

冗長化リスナーの接続監視では、監視対象リスナーの再起動は行われません。

冗長化リスナーの接続監視で 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、以下のような動作になります。

（例）

```
HALT_CONTROL_OPTION     = ALWAYS

ListenerMonitor LISTENER_MULTI {
    LSNR_CHECK_MODE     = PING_NO_RESTART
    POLL_INTERVAL       = 60
    POLL_TIMEOOUT       = 90
    SERVICE_DOWN        = YES
    RETRY_COUNT         = 1
    NET_SERVICE_NAME    = MULTI
}
```

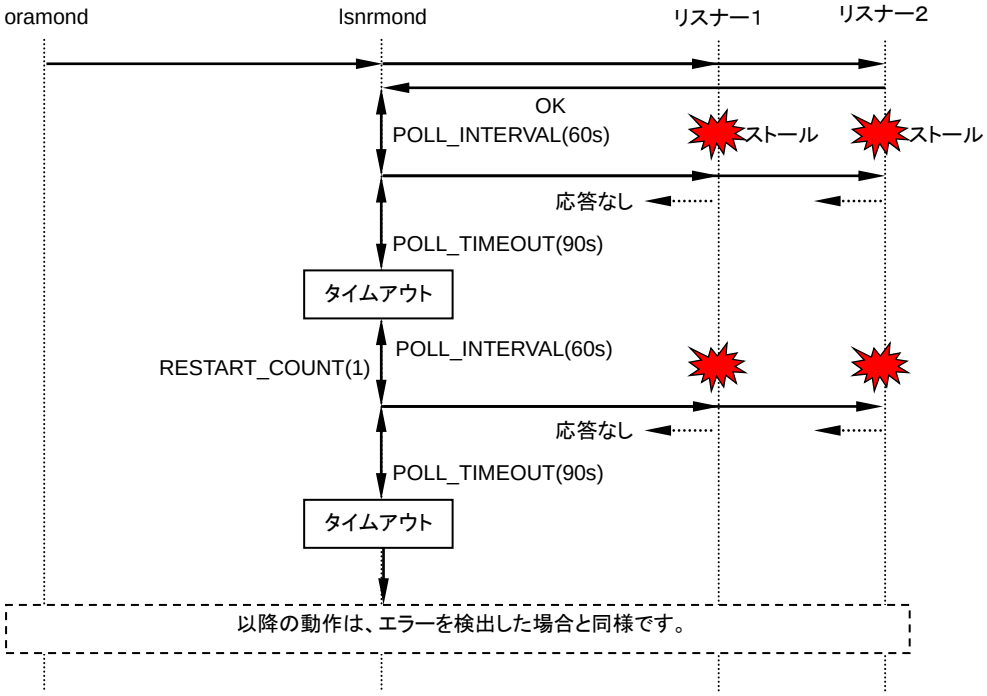


図 4-23 冗長化リスナー接続監視タイムアウト時の動作

冗長化リスナーの接続監視がPOLL_TIMEOUTパラメータに設定した時間内に完了しない場合、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リトライを行います。設定した回数のリトライを行ってもリスナーの状態が改善されない場合は、SERVICE_DOWN パラメータの設定により、CLUSTERPRO への通知判定および ApplicationMonitor 停止判定を行います。

SERVICE_DOWN パラメータに YES または NO_DOWN が設定されていると、CLUSTERPRO へ障害を通知します。

冗長化リスナーの接続監視では、監視対象リスナーの再起動は行われません。

4.4.3. 冗長化リスナーの状態監視

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

冗長化リスナーの状態監視では、ネット・サービスは使用しません。

冗長化リスナーの接続監視では、監視対象リスナーの再起動は行われません。冗長化リスナーの状態監視を併用することで、冗長構成された各リスナーの再起動を行うことができます。

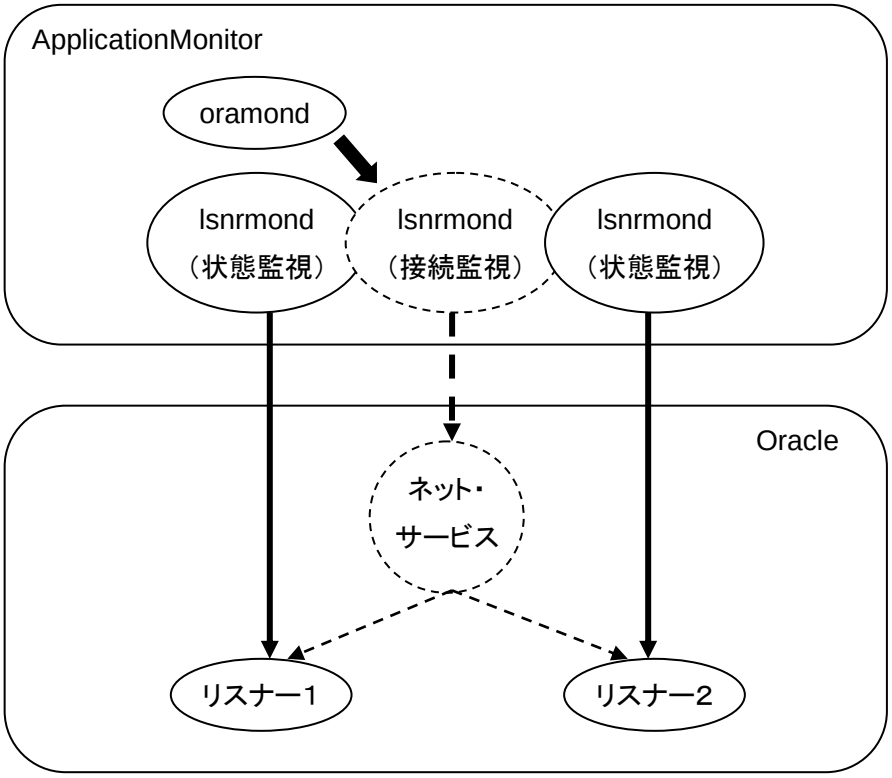


図 4-24 リスナー監視モニタ（冗長化リスナーの状態監視）の概要

冗長化リスナーの状態監視は、以下のような動作になります。

（例）

```
ListenerMonitor LISTENER_1 {
    LSNR_CHECK_MODE     = STATUS
    POLL_INTERVAL       = 60
    GET_STATS           = YES
}
```

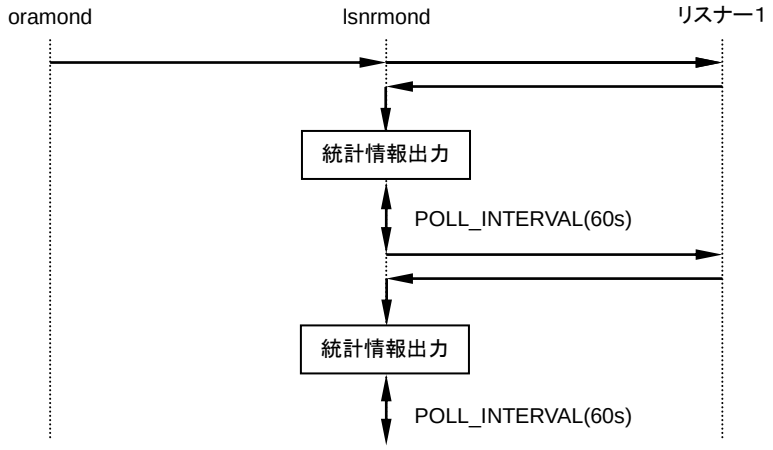


図 4-25 冗長化リスナー接続監視時の動作

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。

冗長化リスナーの状態監視でエラーを検出した場合、以下のような動作になります。

（例）

```
HALT_CONTROL_OPTION     = ALWAYS

ListenerMonitor LISTENER_1 {
    LSNR_CHECK_MODE     = STATUS
    SERVICE_DOWN        = NO
    POLL_INTERVAL       = 60
    RESTART_COUNT       = 1
    RESTART_INTERVAL    = 3
    RESTART_TIMEOUT     = 60
    TARGET_RESTART      = YES
}
```

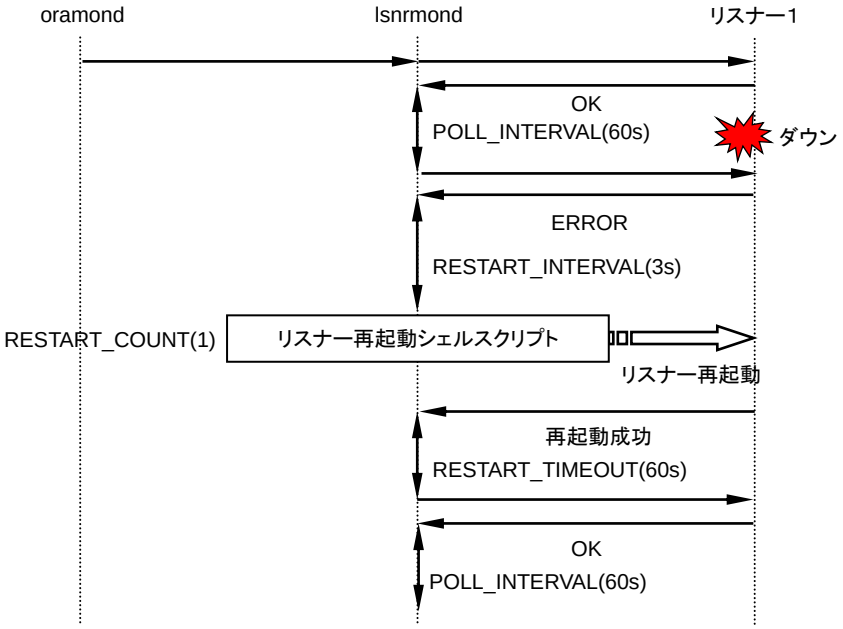


図 4-26 冗長化リスナー状態監視エラー時の動作

冗長化リスナーの状態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トライを行います。

リトライ時、TARGET_RESTART パラメータの値が YES に設定されていると、リスナー再起動シェルスクリプト (lsnr_control.sh) を実行します。

なお、TARGET_RESTART パラメータの値が NO に設定されていると、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプト (crs_control.sh) を実行し、Oracle Clusterware／Oracle Restart によるリスナーの再起動を待ち合わせます。

リスナーが再起動した場合は、RESTART_TIMEOUT パラメータに設定した時間の待ち合わせを行い、監視を再開します。

リスナー再起動シェルスクリプトの詳細は、「4.17 リスナー再起動シェルスクリプトの動作」を参照してください。

また、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトの詳細は、「4.15 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。

SERVIE_DOWN パラメータに NO または、NO_DOWN を設定することで、冗長構成された何れかのリスナーで障害が発生しても、CLUSTERPRO への障害通知を行わず、監視を継続することができます。

冗長化リスナーの状態監視で 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、以下のような動作になります。

（例）

```
HALT_CONTROL_OPTION     = ALWAYS

ListenerMonitor LISTENER_1 {
    LSNR_CHECK_MODE     = STATUS
    POLL_INTERVAL       = 60
    POLL_TIMEOOUT       = 90
    SERVICE_DOWN        = NO
    RESTART_COUNT       = 1
    RESTART_INTERVAL    = 3
    RESTART_TIMEOUT     = 60
    TARGET_RESTART      = YES
}
```

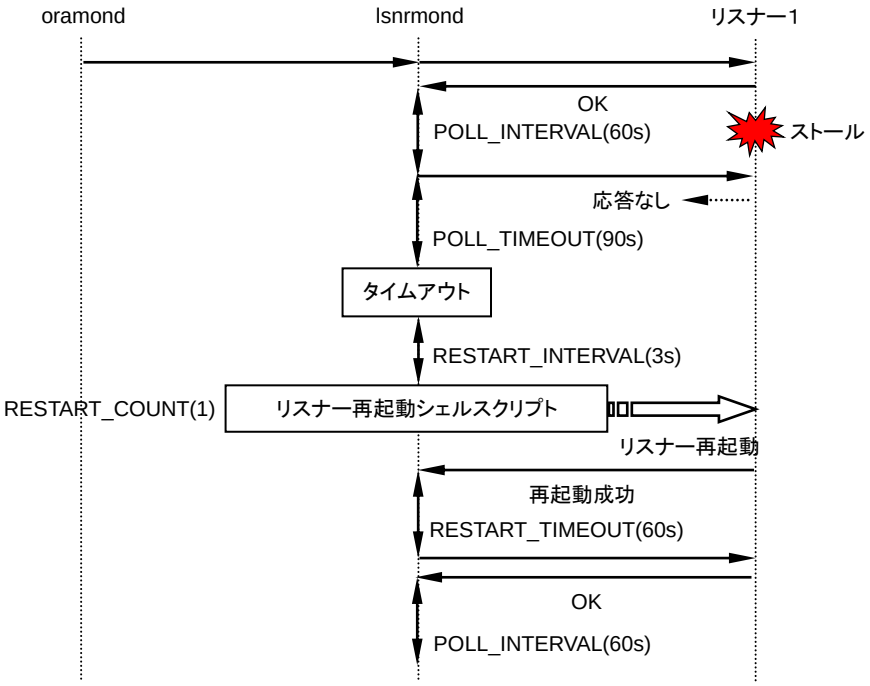


図 4-27 冗長化リスナー状態監視タイムアウト時の動作

冗長化リスナーの状態監視が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トライを行います。

リトライ時、TARGET_RESTARTパラメータの値がYESに設定されていると、リスナー再起動シェルスクリプト (lsnr_control.sh) を実行します。

なお、TARGET_RESTART パラメータの値が NO に設定されていると、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプト (crs_control.sh) を実行し、Oracle Clusterware／Oracle Restart によるリスナーの再起動を待ち合わせます。

リスナーが再起動した場合は、RESTART_TIMEOUT パラメータに設定した時間の待ち合わせを行い、監視を再開します。

リスナー再起動シェルスクリプトの詳細は、「4.17 リスナー再起動シェルスクリプトの動作」を参照してください。
また、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトの詳細は、「4.15 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。

SERVIE_DOWN パラメータに NO または、NO_STALL を設定することで、冗長構成された何れかのリスナーで障害が発生しても、CLUSTERPRO への障害通知を行わず、監視を継続することができます。

4.5. 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
    }
}
```

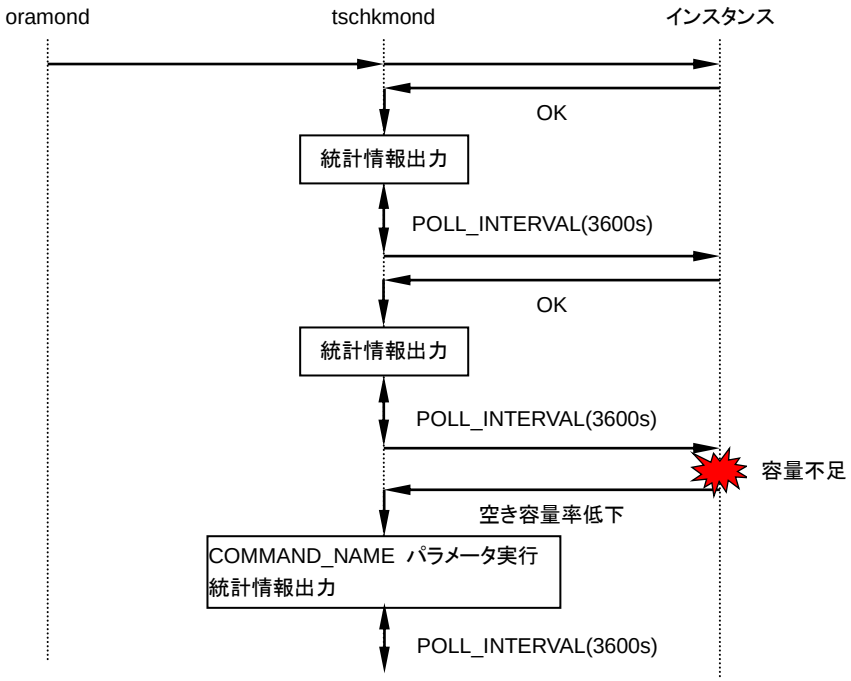


図 4-28 データベース領域の容量監視時の動作

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。

4.6. 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
}
```

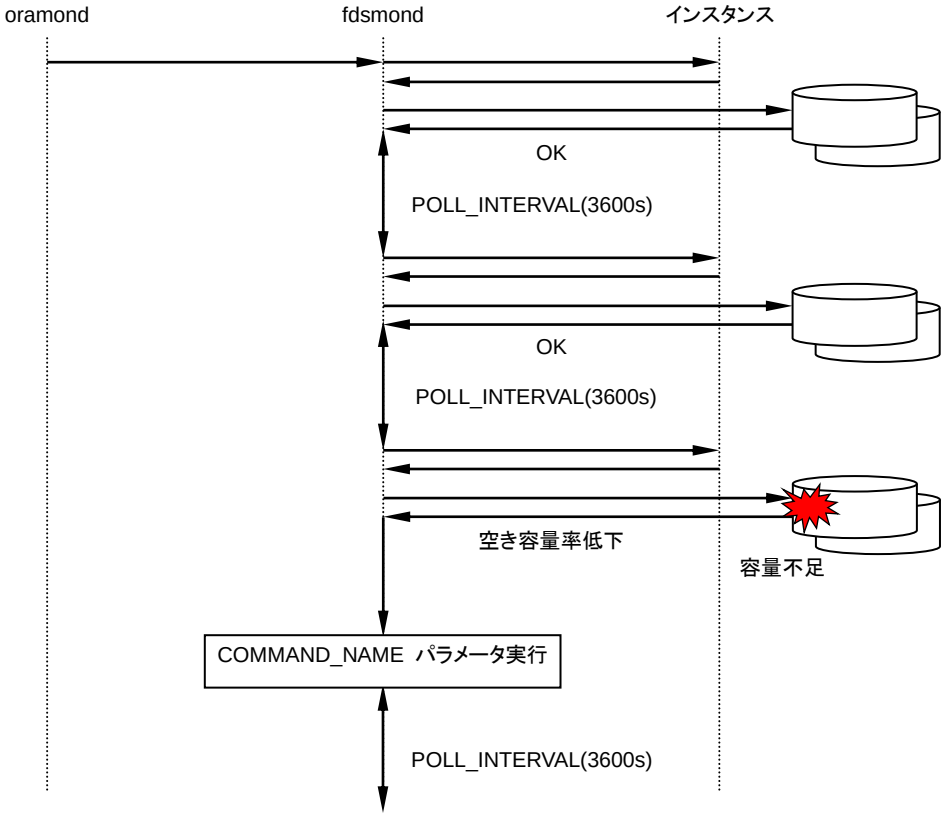


図 4-29 ローカルディスク領域の容量監視時の動作

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 ユーザーによる監視を行う（ORACLE_USER パラメータおよび 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。

4.7. 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"
        }
    }
}
```

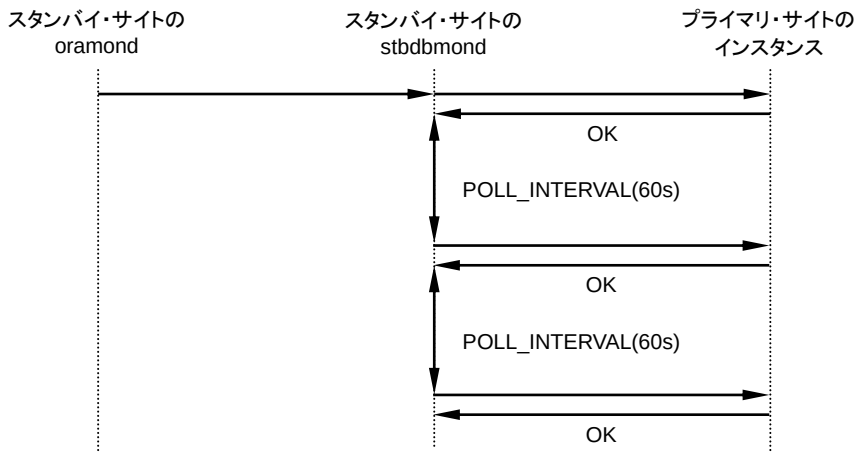


図 4-30 スタンバイ・インスタンスの REDO ログ転送状況監視時の動作
（一定時間経過による監視）

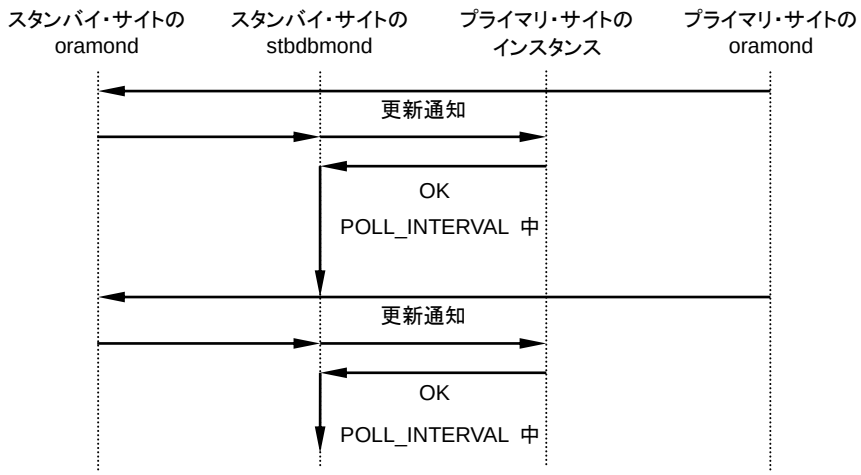


図 4-31 スタンバイ・インスタンスの REDO ログ転送状況監視時の動作
（更新通知による監視）

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。

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
    }
}
```

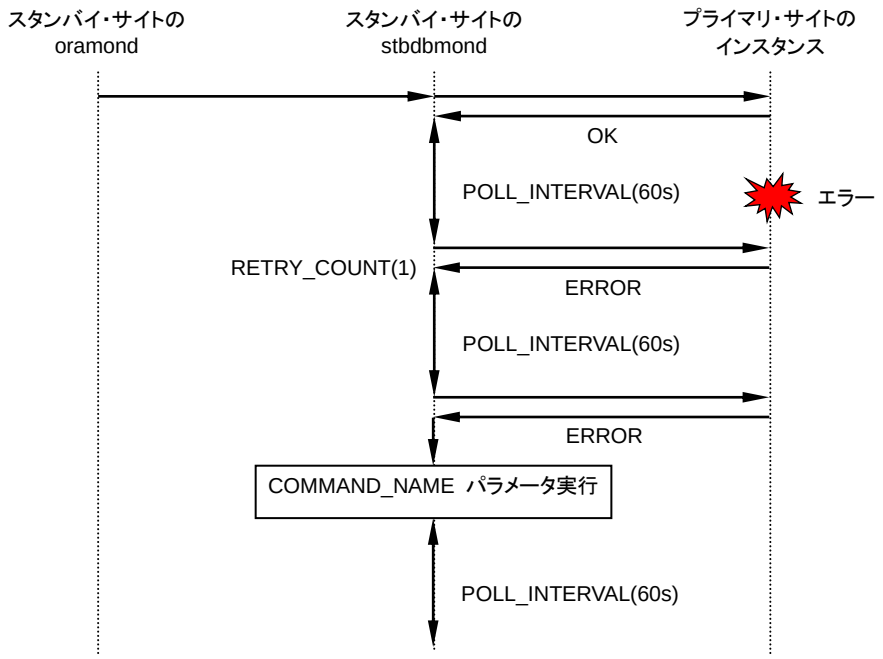


図 4-32 スタンバイ・インスタンスの REDO ログ転送状況監視エラー時の動作

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
    }
}
```

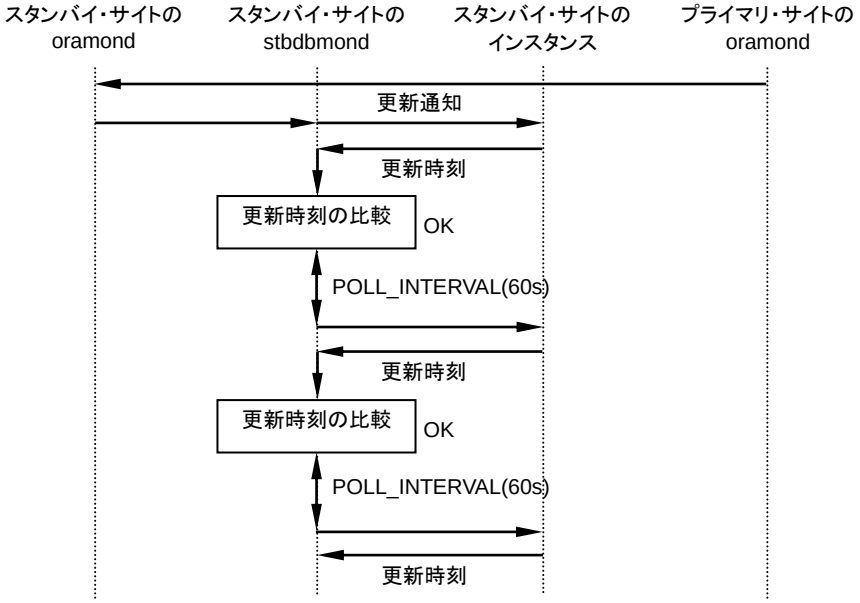


図 4-33 スタンバイ・インスタンスの REDO ログ適用状況監視時の動作
（一定時間経過による監視）

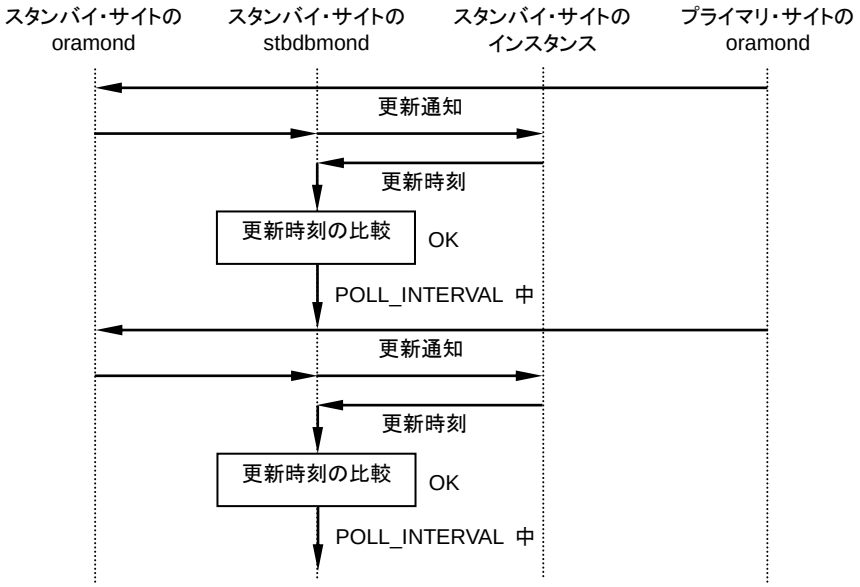


図 4-34 スタンバイ・インスタンスの REDO ログ適用状況監視時の動作
（更新通知による監視）

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
    }
}
```

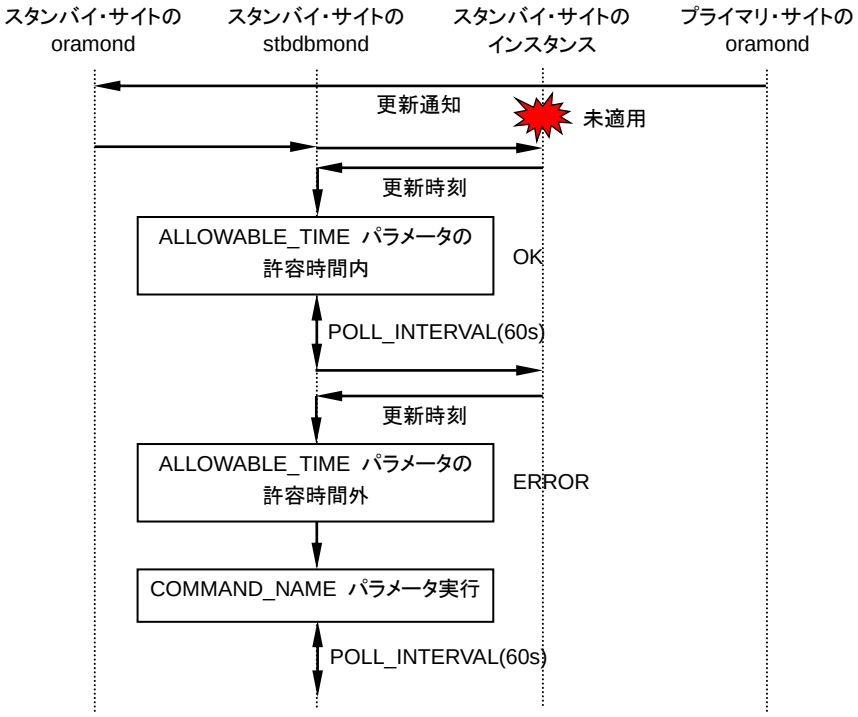


図 4-35 スタンバイ・インスタンスの REDO ログ適用状況監視エラー時の動作

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、監視を継続します。

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。

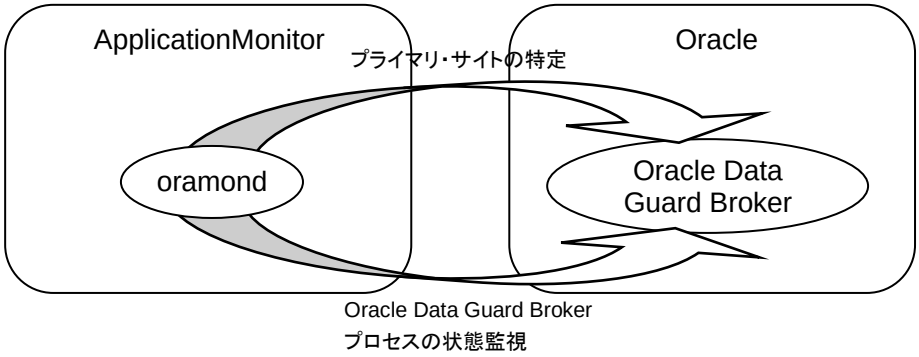


図 4-36 スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視時の概要

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、以下のような動作になります。

（例）

```
StandbyDatabaseMonitor {
    OdgBroker {
        POLL_INTERVAL     = 60
    }
}
```

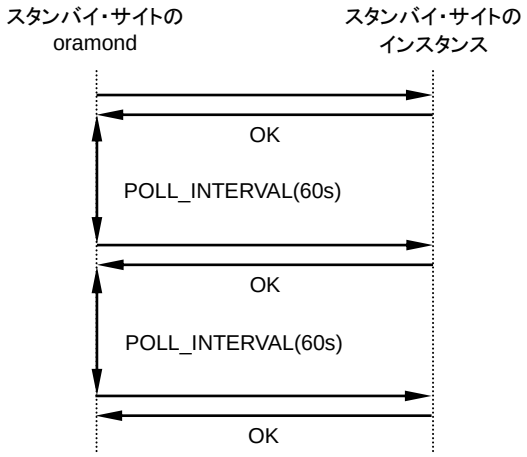


図 4-37 スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視時の動作（一定時間経過による監視）

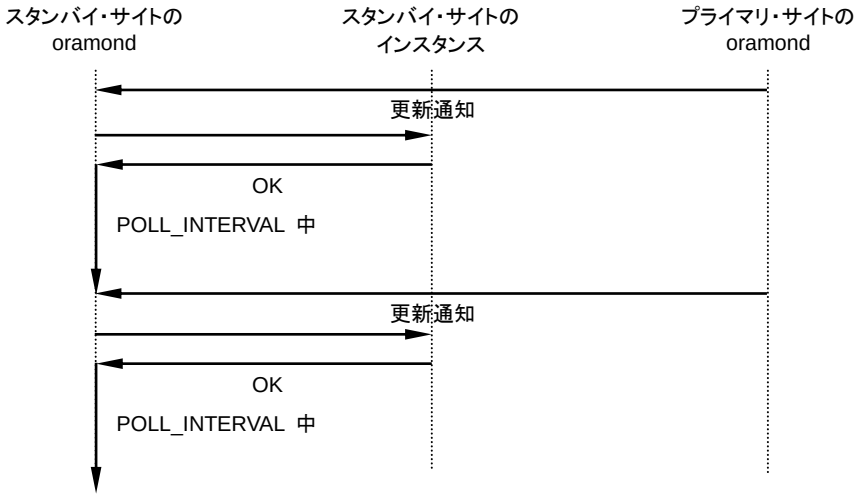


図 4-38 スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視時の動作（更新通知による監視）

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。

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
    }
}
```

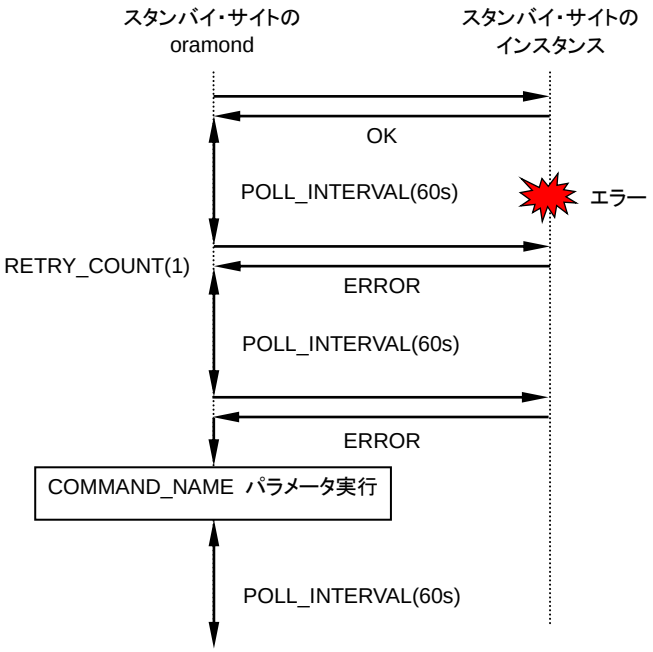


図 4-39 スタンバイ・インスタンスの Oracle Data Guard Broker プロセスの状態監視エラー時の動作

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。

4.8. 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、Oracle Clusterware／Oracle Restart のデーモンの状態を監視します。

なお、CRS_CHECK_MODE パラメータ（※1）に OHAS が設定されていると Oracle Clusterware のデーモンの ohasd のみ監視します。

（※1） Oracle Clusterware 環境でのみ設定が有効です。

Oracle Clusterware／Oracle Restart の監視は、以下のような動作になります。

（例）

```
ORACLE_VERSION     = 21C
CRS_STAT           = "/u01/app/21.3.0/grid/bin/crsctl check crs"
CRS_CHECK_MODE     = ALL
POLL_INTERVAL      = 90
```

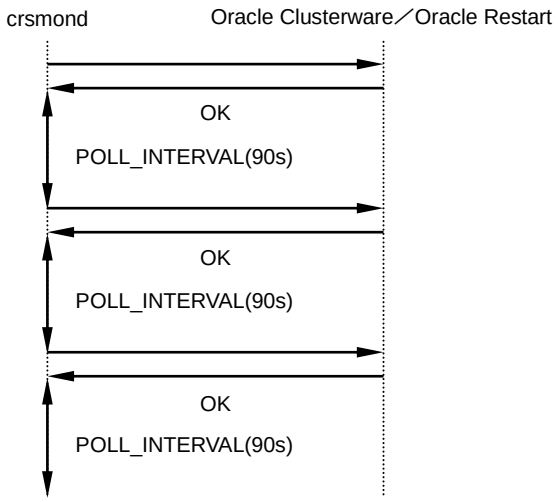


図 4-40 Oracle Clusterware／Oracle Restart 監視時の動作

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      = 21C
CRS_STAT            = "/u01/app/21.3.0/grid/bin/crsctl check crs"
POLL_INTERVAL       = 90
RETRY_COUNT         = 2
WAIT_INTERVAL       = 180
SERVICE_DOWN        = NO
```

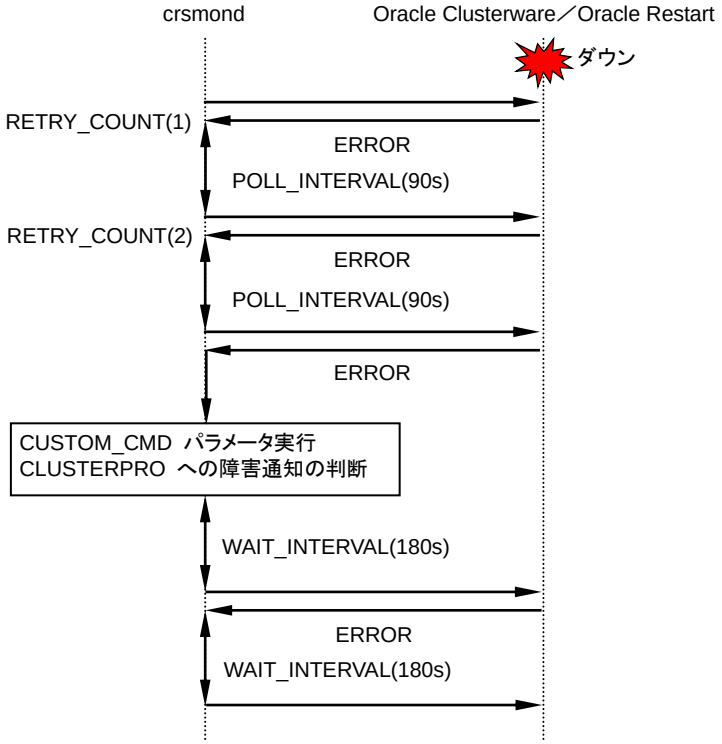


図 4-41 CRS 監視モニタ起動エラー時の動作

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 パラメータに設定した回数の Oracle Clusterware／Oracle Restart のデーモン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 Oracle Clusterware／Oracle Restart のデーモンの状態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、SERVICE_DOWN パラメータの設定により、コマンドの実行および CLUSTERPRO へ障害を通知するか判断します。

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 パラメータに設定したコマンドの実行および CLUSTERPRO への障害通知の判断を行います。

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、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。
なお、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、WAIT_INTERVAL パラメータに設定した間隔で Oracle Clusterware／Oracle Restart のデーモンの状態確認を継続します。この動作は、Oracle Clusterware／Oracle Restart のデーモン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。

Oracle Clusterware／Oracle Restart の監視でエラーを検出し、init(8)／Oracle Clusterware による障害デーモン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）

```
ORACLE_VERSION     = 21C
CRS_STAT           = "/u01/app/21.3.0/grid/bin/crsctl check crs"
POLL_INTERVAL      = 90
RETRY_COUNT        = 2
HALT_METHOD        = KILL
WAIT_INTERVAL      = 180
```

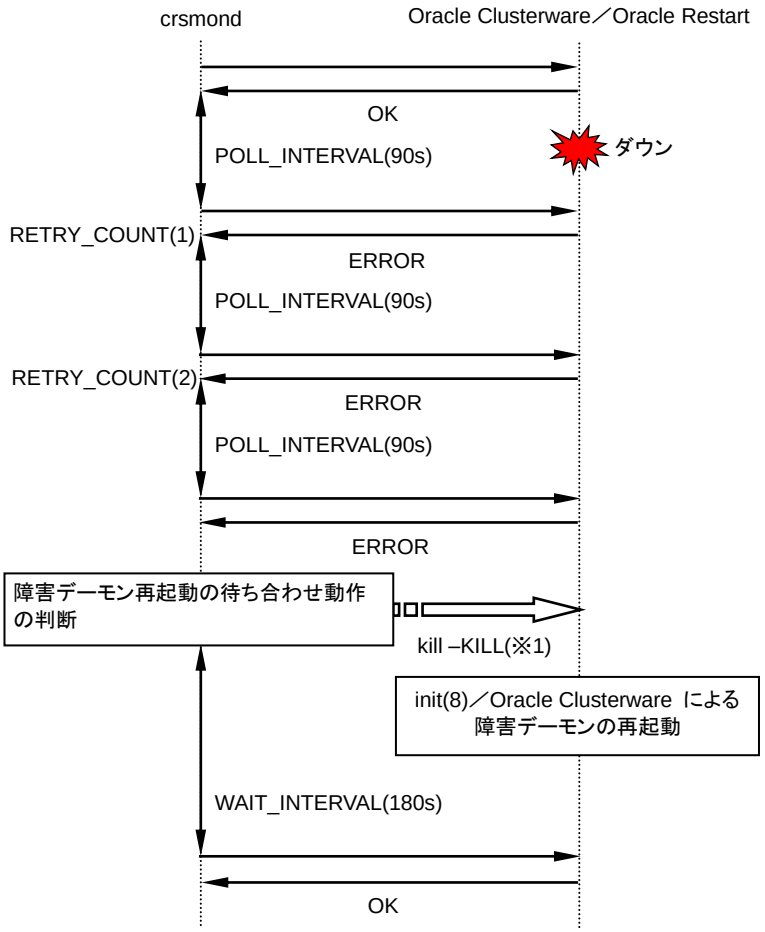


図 4-42 Oracle Clusterware／Oracle Restart 監視エラー時の動作（init(8)／Oracle Clusterware による障害デーモンの再起動が行われた場合）

Oracle Clusterware／Oracle Restart 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 Oracle Clusterware／Oracle Restart のデーモンの状態を再確認します。再確認中に Oracle Clusterware／Oracle Restart のデーモン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 Oracle Clusterware／Oracle Restart のデーモンの状態の再確認を行っても、Oracle Clusterware／Oracle Restart のデーモンの監視結果がエラーになる場合は、HALT_METHOD パラメータの設定により、障害デーモンの再起動の待ち合わせ動作を判断します。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに障害デーモンの再起動の待ち合わせ動作を判断します。

HALT_METHOD パラメータに KILL が設定されていると、障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対して kill コマンドを実行（※1）し、init(8)／Oracle Clusterware による障害デーモンの再起動を促します。

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。

（※1） ocssd プロセスに対しては、kill コマンドを実行しません。

Oracle Clusterware／Oracle Restart の監視でエラーを検出し、init(8)／Oracle Clusterware による障害デーモンの再起動が行われなかった場合、以下のような動作になります。

（例）

```
ORACLE_VERSION      = 21C
CRS_STAT            = "/u01/app/21.3.0/grid/bin/crsctl check crs"
POLL_INTERVAL       = 90
RETRY_COUNT         = 1
HALT_METHOD         = KILL
WAIT_INTERVAL       = 180
SERVICE_DOWN        = YES
```

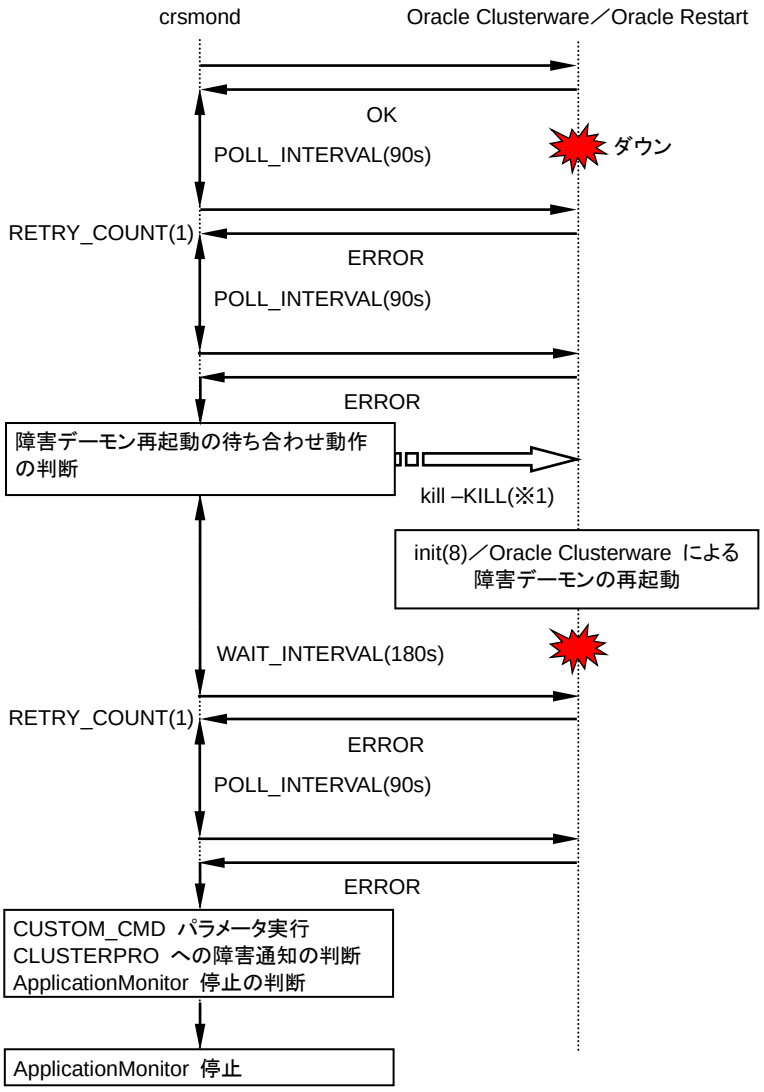


図 4-43 Oracle Clusterware／Oracle Restart 監視エラー時の動作
（init(8)／Oracle Clusterware による
障害デーモンの再起動が行われなかった場合）

kill コマンド実行（※1）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、

Oracle Clusterware／Oracle Restart のデーモン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 Oracle Clusterware／Oracle Restart のデーモンの状態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 Oracle Clusterware／Oracle Restart のデーモンの状態の再確認を行っても Oracle Clusterware／Oracle Restart のデーモンの監視結果がエラーになる場合は、CUSTOM_CMD パラメータ、SERVICE_DOWN パラメータの設定により、コマンドの実行および CLUSTERPRO へ障害を通知するか判断します。

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 パラメータに設定したコマンドの実行および CLUSTERPRO への障害通知の判断を行います。

Oracle Clusterware／Oracle Restart のデーモン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 へ障害を通知すると判断し、ApplicationMonitor を停止します。

なお、Oracle Clusterware／Oracle Restart のデーモン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、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。

CLUSTERPRO へ通知すると判断した場合、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 を停止すると判断します。

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 を停止しないと判断し監視を継続します。

（※1） ocssd プロセスに対しては、kill コマンドを実行しません。

Oracle Clusterware／Oracle Restart 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      = 21C
CRS_STAT            = "/u01/app/21.3.0/grid/bin/crsctl check crs"
POLL_INTERVAL       = 90
POLL_TIMEOUT        = 180
RETRY_COUNT         = 1
HALT_METHOD         = KILL
WAIT_INTERVAL       = 180
SERVICE_DOWN        = YES
```

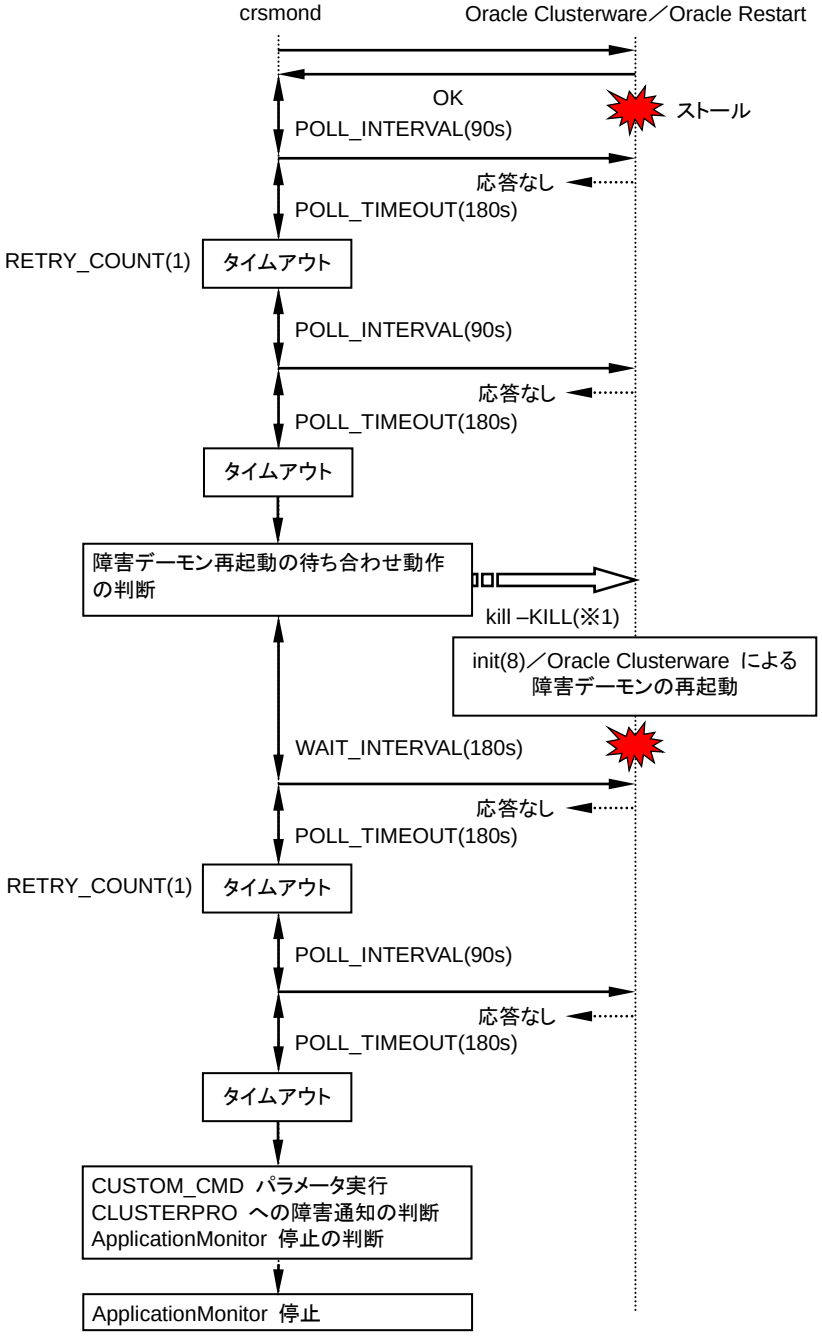


図 4-44 Oracle Clusterware／Oracle Restart 監視タイムアウト時の動作

Oracle Clusterware／Oracle Restart 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 Oracle Clusterware／Oracle Restart のデーモンの状態を再確認します。再確認中に Oracle Clusterware／Oracle Restart のデーモン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 Oracle Clusterware／Oracle Restart のデーモンの状態の再確認を行っても、Oracle Clusterware／Oracle Restart のデーモンの監視結果がストールとなる（タイムアウトする）場合は、HALT_METHOD パラメータの設定により、障害デーモンの再起動の待ち合わせ動作を判断します。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに障害デーモンの再起動の待ち合わせ動作を判断します。

HALT_METHOD パラメータに KILL が設定されていると、障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対して kill コマンドを実行（※1）し、init(8)／Oracle Clusterware による障害デーモンの再起動を促します。

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、Oracle Clusterware／Oracle Restart のデーモン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 Oracle Clusterware／Oracle Restart のデーモン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 Oracle Clusterware／Oracle Restart のデーモンの状態の再確認を行っても Oracle Clusterware／Oracle Restart のデーモンの監視結果がストールになる場合は、CUSTOM_CMD パラメータ、SERVICE_DOWN パラメータの設定により、コマンドの実行および CLUSTERPRO へ障害を通知するか判断します。

Oracle Clusterware／Oracle Restart のデーモン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、Oracle Clusterware／Oracle Restart のデーモン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、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。

CLUSTERPRO へ通知すると判断した場合、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

止すると判断します。

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 を停止しないと判断し監視を継続します。

（※1） ocssd プロセスに対しては、kill コマンドを実行しません。

Oracle Clusterware／Oracle Restart の監視で障害を検出し、障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対して kill コマンドを実行せず、init(8)／Oracle Clusterware による障害デーモンの再起動を待ち合わせる場合、以下のような動作になります。

（例）

```
ORACLE_VERSION      = 21C
CRS_STAT            = "/u01/app/21.3.0/grid/bin/crsctl check crs"
POLL_INTERVAL       = 90
RETRY_COUNT         = 1
HALT_METHOD         = NONE
WAIT_INTERVAL       = 180
```

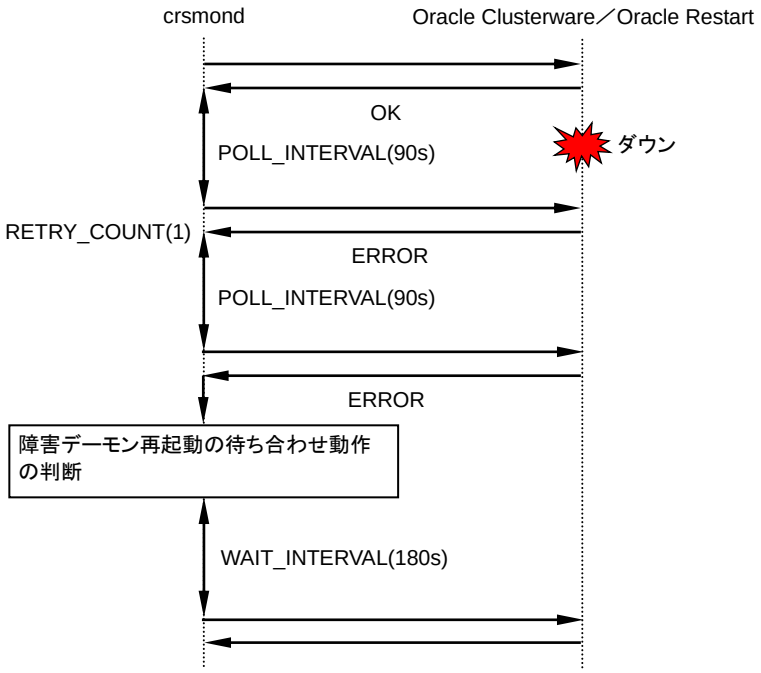


図 4-45 Oracle Clusterware／Oracle Restart 監視エラー時の動作（障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対して kill コマンドを実行せず、init(8)／Oracle Clusterware による障害デーモンの再起動を待ち合わせる場合）

RETRY_COUNT パラメータに設定した回数の Oracle Clusterware／Oracle Restart のデーモンの状態の再確認を行っても、Oracle Clusterware／Oracle Restart のデーモン の監視結果がエラーになる場合は、HALT_METHOD パラメータの設定により、障害デーモンの再起動の待ち合わせ動作を判断します。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに障害デーモンの再起動の待ち合わせ動作を判断します。

HALT_METHOD パラメータに NONE が設定されていると、障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対して kill コマンドを実行せず、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。

Oracle Clusterware／Oracle Restart の監視で障害を検出し、障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対する kill コマンドの実行、障害デーモンの再起動の待ち合わせを行わない場合、以下のような動作になります。

（例）

```
ORACLE_VERSION     = 21C
CRS_STAT           = "/u01/app/21.3.0/grid/bin/crsctl check crs"
POLL_INTERVAL      = 90
RETRY_COUNT        = 1
HALT_METHOD        = DOWN
SERVICE_DOWN       = YES
```

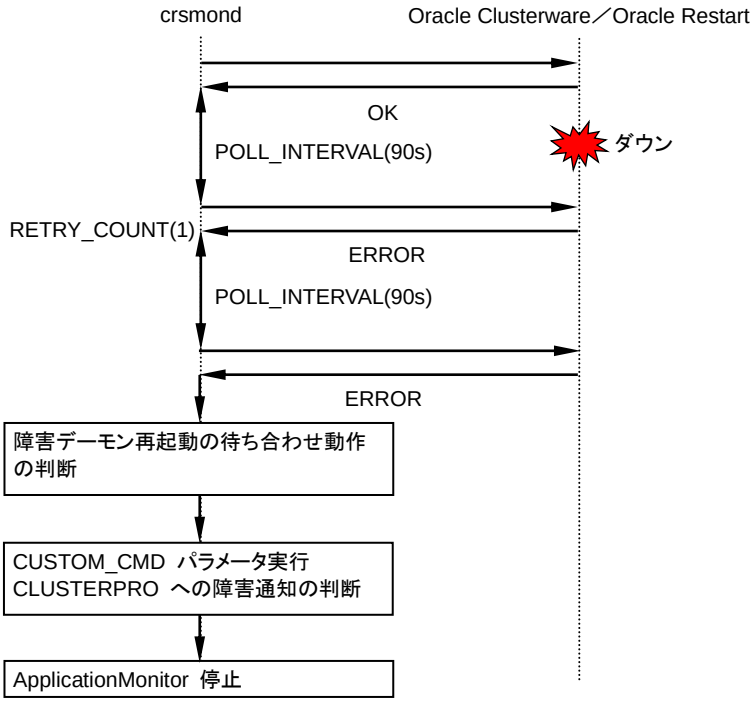


図 4-46 Oracle Clusterware／Oracle Restart 監視エラー時の動作（障害を検出した Oracle Clusterware／Oracle Restart のデーモンプロセスに対する kill コマンドの実行、障害デーモンの再起動の待ち合わせを行わない場合）

RETRY_COUNT パラメータに設定した回数の Oracle Clusterware／Oracle Restart のデーモンの状態の再確認を行っても、Oracle Clusterware／Oracle Restart のデーモンの監視結果がエラーになる場合は、HALT_METHOD パラメータの設定により、障害デーモンの再起動の待ち合わせ動作を判断します。

RETRY_COUNT パラメータに 0 が設定されている場合は、再確認せずに障害デーモンの再起動の待ち合わせ動作を判断します。

HALT_METHOD パラメータに DOWN が設定されていると、障害を検出した Oracle

Clusterware／Oracle Restart のデーモンプロセスに対して kill コマンドを実行および WAIT_INTERVAL パラメータに設定した時間の待ち合わせを行いません。

CUSTOM_CMD パラメータ、SERVICE_DOWN パラメータの設定により、コマンドの実行および CLUSTERPRO へ障害を通知するか判断します。

Oracle Clusterware／Oracle Restart のデーモン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 へ障害を通知すると判断し、ApplicationMonitor を停止します。

なお、Oracle Clusterware／Oracle Restart のデーモン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、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。

4.9. ASM インスタンス監視時の動作

ASM 監視モニタ (asmmond) は、常に単独起動モニタとして動作します。

ASM 監視モニタは、ASM 監視モニタ起動スクリプト (start_asmmond.sh) によって起動します。起動後に ASM 監視設定ファイル (asmmond.conf) の POLL_INTERVAL パラメータに設定した間隔で ASM インスタンスを監視します。ASM インスタンスの監視方法は、V$ 表の参照です。

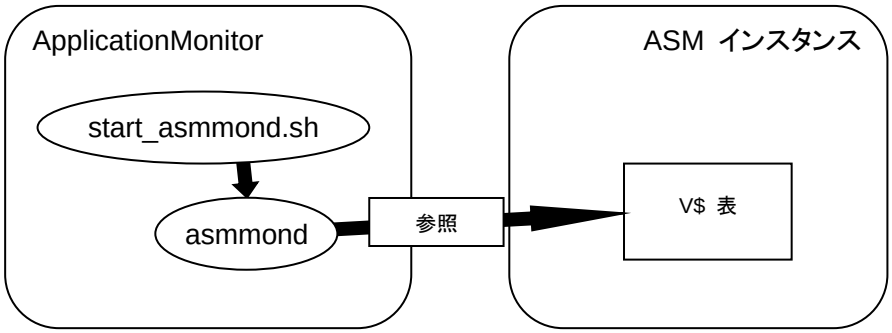


図 4-47 ASM 監視モニタの概要

Flex ASM 構成かどうかを自動的に判断し、Flex ASM 構成の場合は ASM インスタンスが起動しているノードでのみ監視を行います。

ASM インスタンスがシャットダウンされた場合は、ASM 監視モニタは停止します。その際、ASM 監視モニタ起動スクリプトは動作を継続し、ASM インスタンスの再起動を待ち合わせます。

ASM インスタンスが再起動された場合は、ASM 監視モニタを起動し監視を再開します。

ASM インスタンスの監視は、以下のような動作になります。

（例）

POLL_INTERVAL = 90

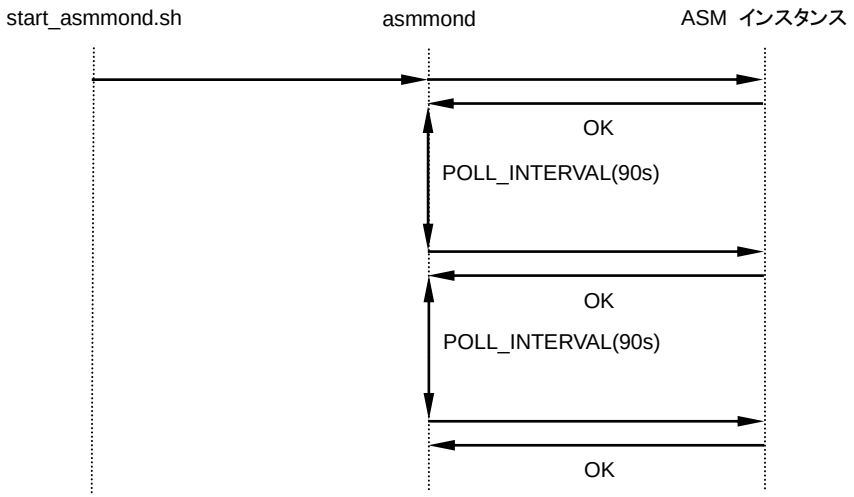


図 4-48 ASM インスタンス監視の動作

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

ASM 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 ASM 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）

POLL_INTERVAL = 90
POLL_DOWN_RETRY = 1
RESTART_COUNT = 5
WAIT_INTERVAL = 180
SERVICE_DOWN = YES

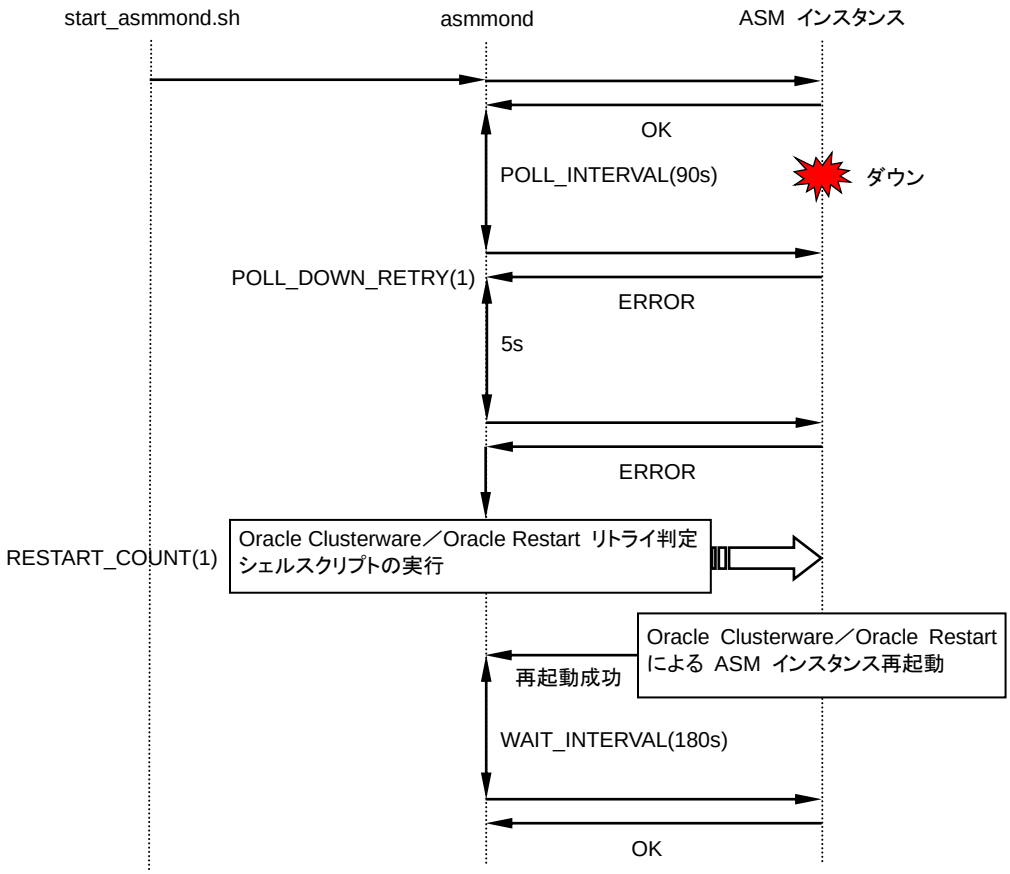


図 4-49 ASM インスタンス監視エラー時の動作
（Oracle Clusterware／Oracle Restart によって
ASM インスタンスが再起動された場合）

ASM 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 ASM インスタンスの再確認を行います。再確認中に ASM 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、監視を継続します。

POLL_DOWN_RETRY パラメータに設定した回数の ASM インスタンスの再確認を行っても、ASM インスタンス監視結果がエラーとなる場合は、Oracle Clusterware／Oracle Restart リトライ

判定シェルスクリプト (crs_control.sh) を実行します。

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 Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトを実行します。

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 ASM インスタンスの再起動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、「4.15 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。

ASM インスタンスが再起動された場合は、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。

ASM 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 ASM 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）

POLL_INTERVAL = 90
POLL_DOWN_RETRY = 0
RESTART_COUNT = 5
WAIT_INTERVAL = 180
SERVICE_DOWN = YES

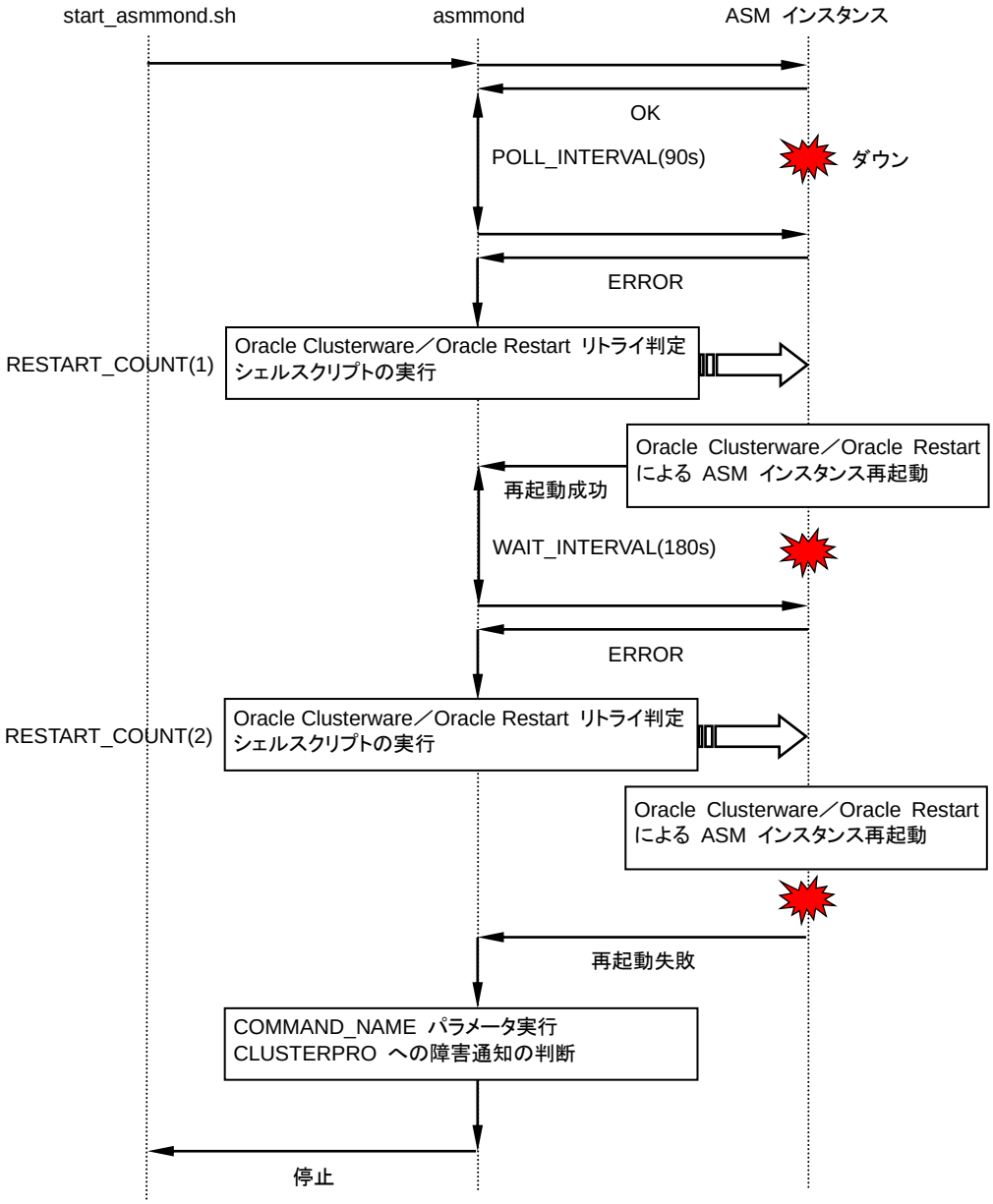


図 4-50 ASM インスタンス監視エラー時の動作
（Oracle Clusterware／Oracle Restart によって
ASM インスタンスが再起動されなかった場合）

RESTART_COUNT パラメータに設定した回数の ASM インスタンス再起動待ち合わせを行っても、ASM インスタンスが再起動されなかった場合は、COMMAND_NAME パラメータ、SERVICE_DOWN パラメータの設定により、コマンドの実行および CLUSTERPRO へ障害を通知するか判断します。

なお、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトが異常終了した（ASM インスタンス再起動失敗と判断された）場合は、RESTART_COUNT パラメータの回数内であっても、COMMAND_NAME パラメータに設定したコマンドの実行および CLUSTERPRO への障害通知の判断を行います。

ASM インスタンス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、ApplicationMonitor を停止します。

なお、ASM インスタンス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、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。

ASM 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）

POLL_INTERVAL = 90
POLL_STALL_RETRY = 1
RESTART_COUNT = 1
SERVICE_DOWN = YES

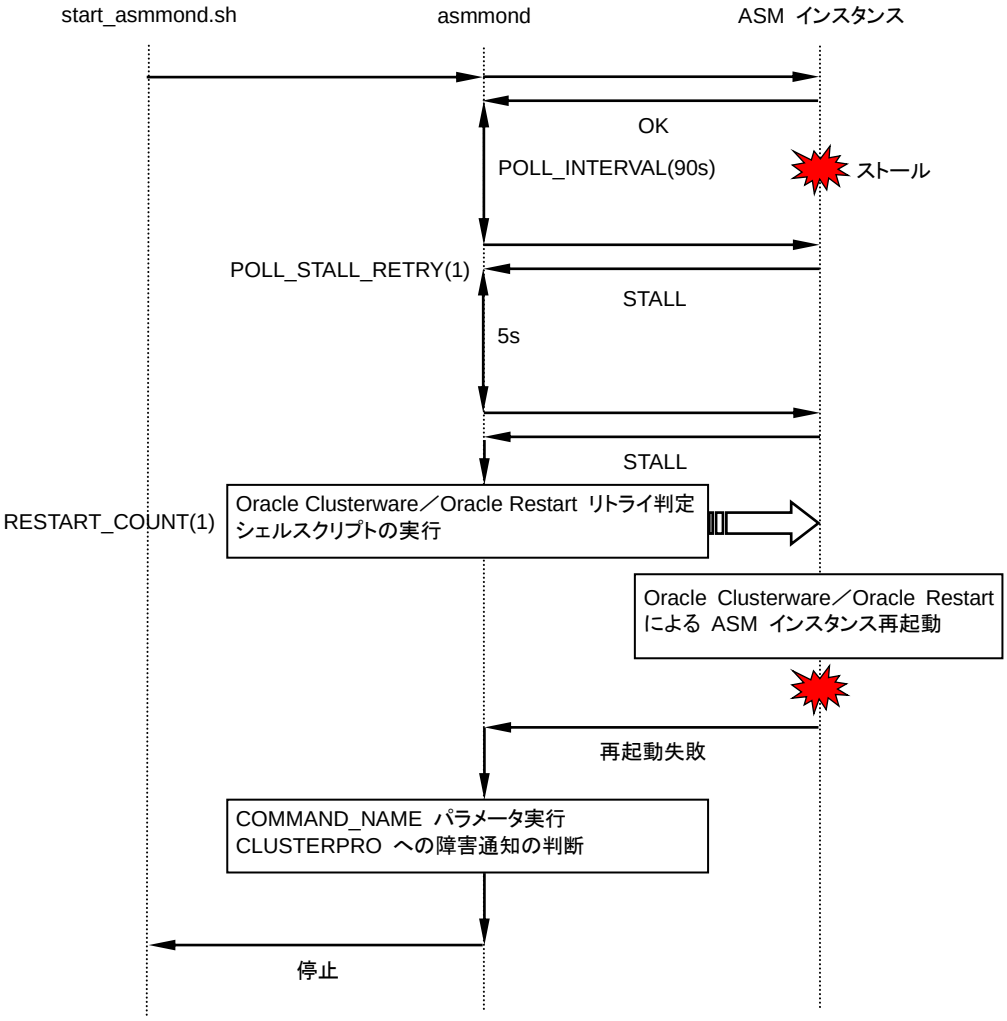


図 4-51 ASM インスタンス監視ストール時の動作
（Oracle バックグラウンドプロセスがストールした場合）

ASM 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）

POLL_INTERVAL = 90
POLL_TIMEOUT = 120
POLL_STALL_RETRY = 1
RESTART_COUNT = 1
WAIT_INTERVAL = 180
SERVICE_DOWN = NO

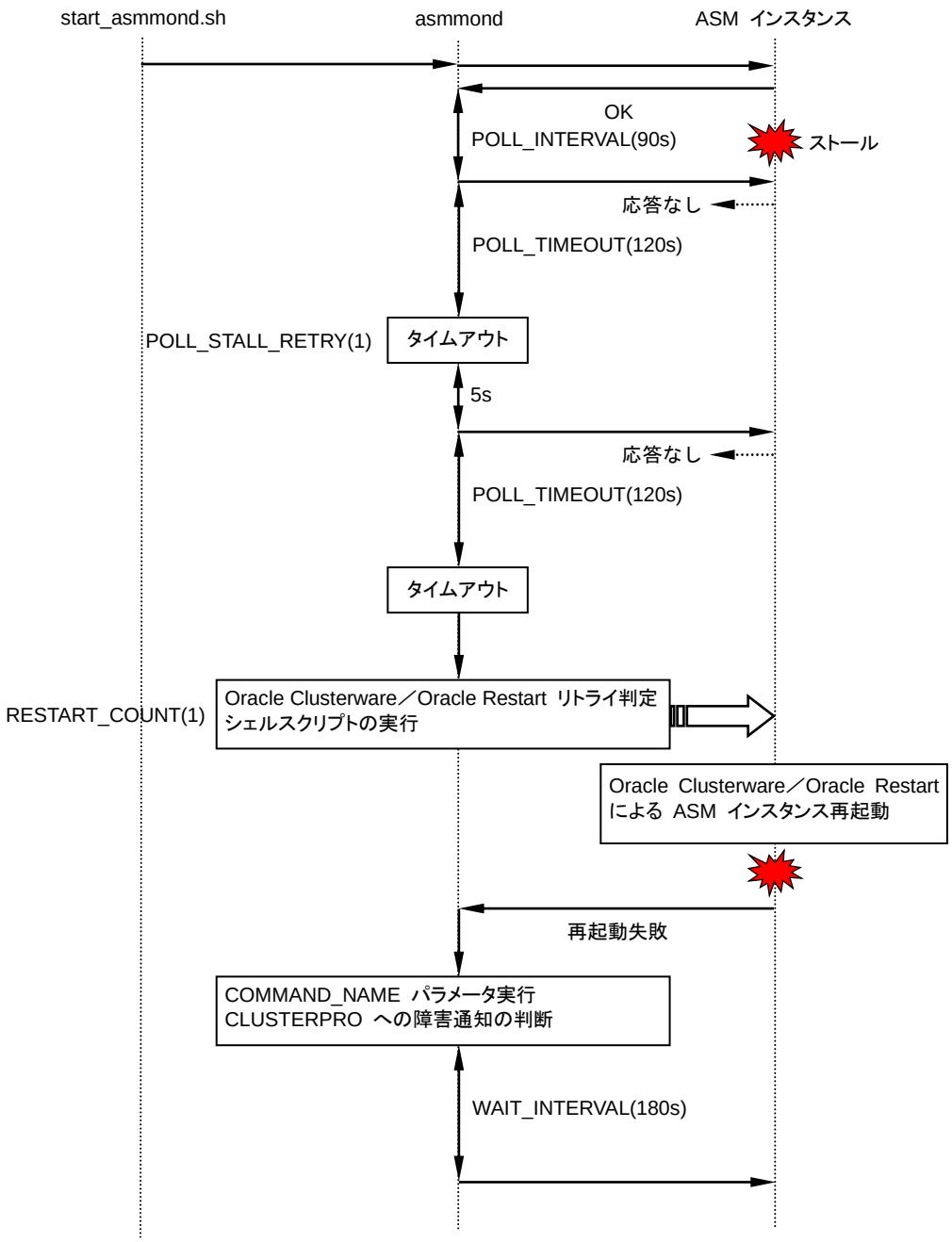


図 4-52 ASM インスタンス監視ストール時の動作
（タイムアウトした場合）

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 ASM 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 ASM 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、監視を継続します。

POLL_STALL_RETRY パラメータに設定した回数の ASM インスタンスの再確認を行っても、ASM 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 ASM インスタンスへのアクセスがタイムアウトする）場合は、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトを実行します。

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 Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトを実行します。

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 ASM インスタンスの再起動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、「4.15 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。

ASM インスタンスが再起動された場合は、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。

Oracle Clusterware／Oracle Restart による ASM インスタンスの再起動を待ち合わせても、ASM インスタンスが再起動されなかった場合は、COMMAND_NAME パラメータ、SERVICE_DOWN パラメータの設定により、コマンドの実行および CLUSTERPRO へ障害を通知するか判断します。

なお、RESTART_COUNT パラメータに 0 が設定されている場合は、Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトを実行せずに COMMAND_NAME パラメータに設定したコマンドの実行および CLUSTERPRO への障害通知の判断を行います。

ASM 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、ApplicationMonitor を停止します。

なお、ASM 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、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。

CLUSTERPRO へ障害を通知しないと判断した場合、WAIT_INTERVAL パラメータに設定した間隔で ASM インスタンスを監視します。この動作は、ASM インスタンス起動の確認または明示的に ASM 監視モニタを停止するまで繰り返します。

4.10. ASM リスナー監視時の動作

ASM リスナー監視モニタ（alsnrmond）は、ASM リスナー監視モニタ起動スクリプト（start_alsnrmond.sh）によって起動します。

ASM リスナー監視モニタ起動スクリプトは、監視対象の ASM リスナーが動作するノード上で、Oracle Clusterware のリソースとして起動されます（※1）。

（※1） ASM リスナー監視モニタは、Oracle Clusterware と連携する場合のみ使用可能です。

ASM リスナーの監視は、以下のような動作になります。

（例）

POLL_INTERVAL = 60

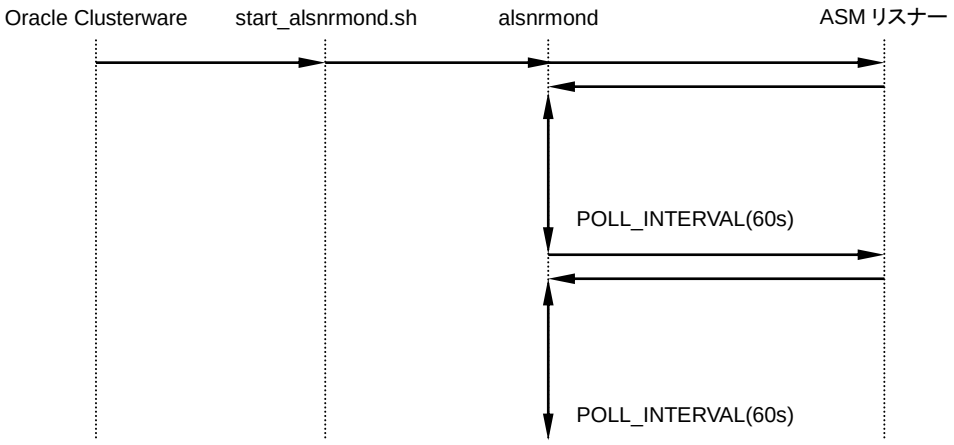


図 4-53 ASM リスナー監視時の動作

ASM 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 ASM リスナーを監視します。

ASM リスナーの監視でエラーを検出した場合、以下のような動作になります。

（例）

POLL_INTERVAL = 60

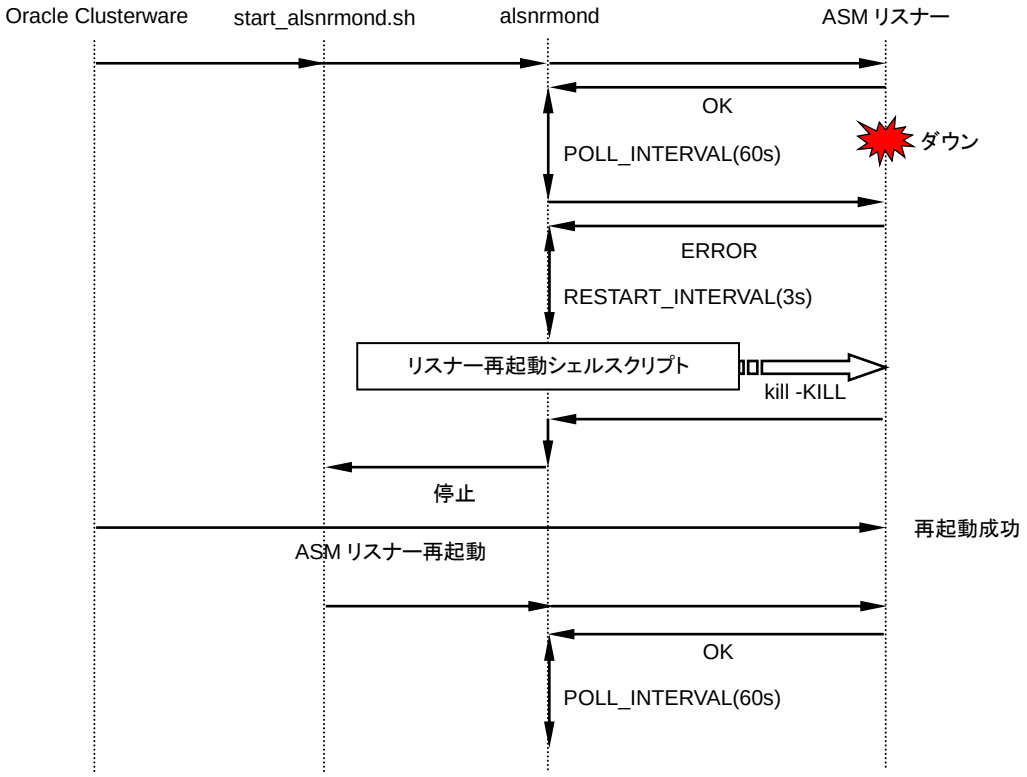


図 4-54 ASM リスナー監視エラー時の動作

ASM リスナー監視でエラーを検出した場合、リスナー再起動シェルスクリプト（lsnr_control.sh）を実行し、Oracle Clusterware による ASM リスナーの再起動を促して、ASM リスナー監視モニタは停止します。その際、ASM リスナー監視モニタ起動スクリプトは動作を継続し、ASM リスナーの再起動を待ち合わせます。

ASM リスナーが再起動された場合は、ASM リスナー監視モニタを起動し、監視を再開します。

リスナー再起動シェルスクリプトの詳細は、「4.17 リスナー再起動シェルスクリプトの動作」を参照してください。

ASM 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、以下のような動作になります。

（例）

POLL_INTERVAL = 60
POLL_TIMEOUT = 90

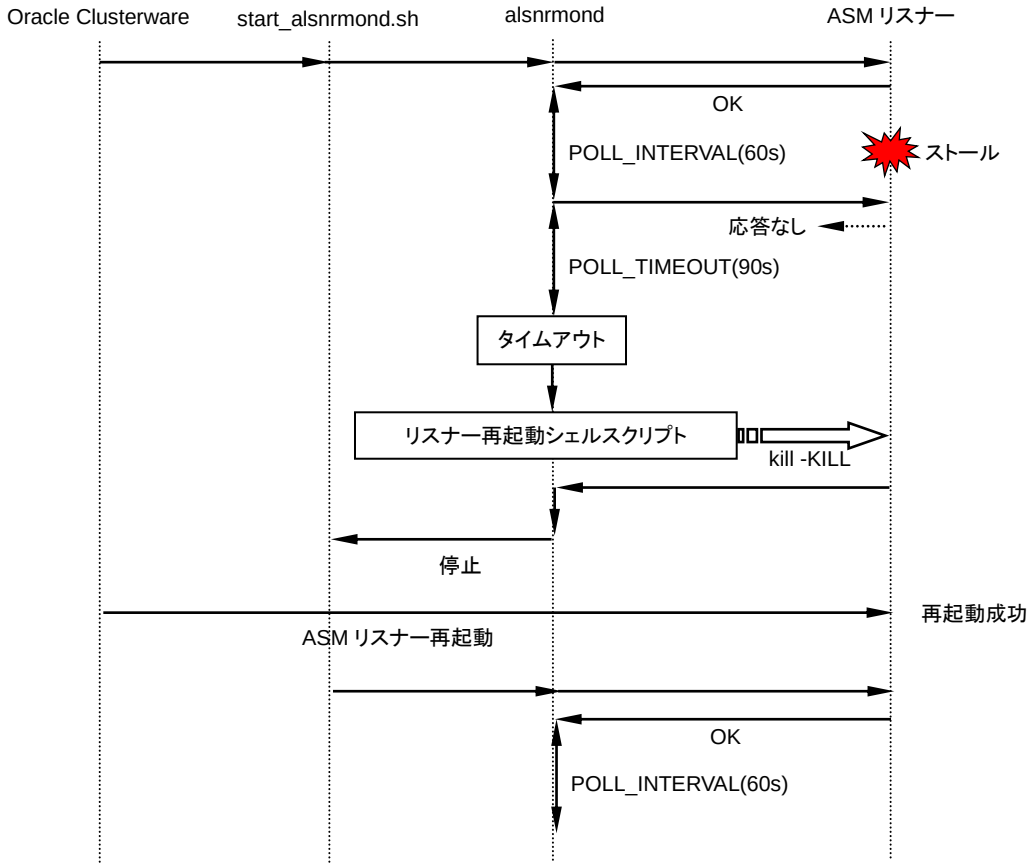


図 4-55 ASM リスナー監視タイムアウト時の動作

ASM 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、リスナー再起動シェルスクリプト（lsnr_control.sh）を実行し、Oracle Clusterware による ASM リスナーの再起動を促して、ASM リスナー監視モニタは停止します。その際、ASM リスナー監視モニタ起動スクリプトは動作を継続し、ASM リスナーの再起動を待ち合わせます。

ASM リスナーが再起動された場合は、ASM リスナー監視モニタを起動し、監視を再開します。

リスナー再起動シェルスクリプトの詳細は、「4.17 リスナー再起動シェルスクリプトの動作」を参照してください。

4.11. SCAN リスナー監視時の動作

SCAN リスナー監視モニタ（slsnrmond）は、監視対象の SCAN リスナーが動作するノード上で、Oracle Clusterware のリソースとして起動されます（※1）。

（※1） SCAN リスナー監視モニタは、Oracle Clusterware と連携する場合のみ使用可能です。

SCAN リスナーの監視は、以下のような動作になります。
（例）
POLL_INTERVAL = 60

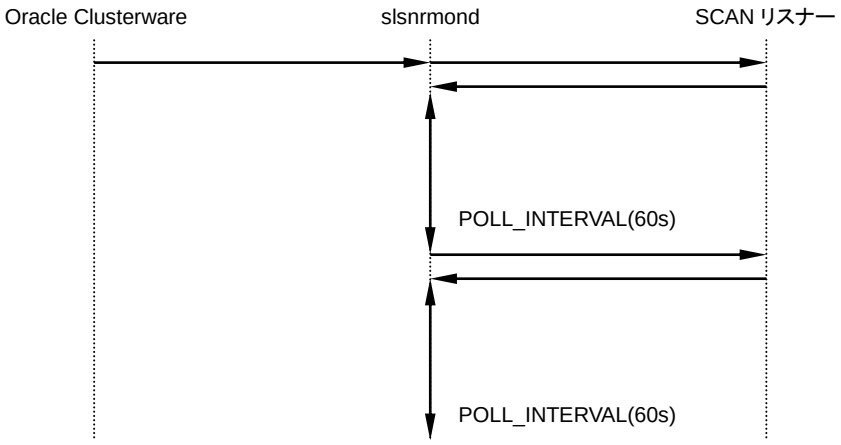


図 4-56 SCAN リスナー監視時の動作

SCAN 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 SCAN リスナーを監視します。

SCAN リスナーの監視でエラーを検出し、Oracle Clusterware によって SCAN 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）

POLL_INTERVAL = 60
RESTART_INTERVAL = 3
RESTART_TIMEOUT = 60

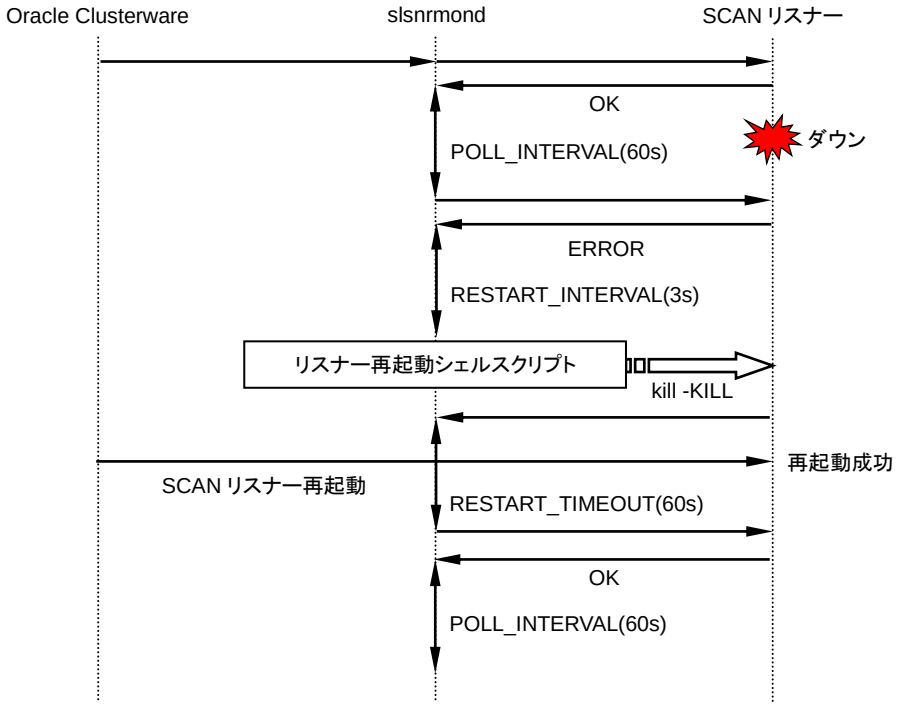


図 4-57 SCAN リスナー監視エラー時の動作（Oracle Clusterware によって SCAN リスナーが再起動された場合）

SCAN リスナー監視でエラーを検出した場合、リスナー再起動シェルスクリプト（lsnr_control.sh）を実行し、Oracle Clusterware による SCAN リスナーの再起動を促して待ち合わせます。

リスナー再起動シェルスクリプトの詳細は、「4.17 リスナー再起動シェルスクリプトの動作」を参照してください。

SCAN リスナーが再起動した場合は、RESTART_TIMEOUT パラメータに設定した時間の待ち合わせを行い、監視を再開します。

SCAN リスナーの監視でエラーを検出し、Oracle Clusterware によって SCAN 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）

POLL_INTERVAL = 60
RESTART_INTERVAL = 3

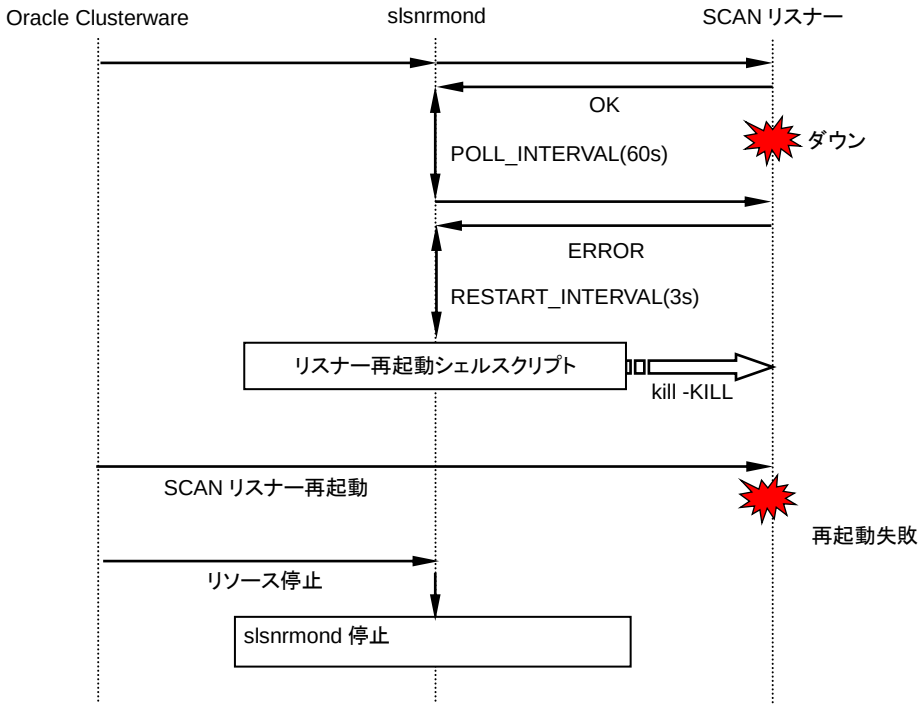


図 4-58 SCAN リスナー監視エラー時の動作（Oracle Clusterware によって SCAN リスナーが再起動されなかった場合）

Oracle Clusterware による SCAN リスナーの再起動を待ち合わせても、SCAN リスナーが再起動されなかった場合は、SCAN リスナー監視モニタは、Oracle Clusterware により停止されます。

SCAN 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 によって SCAN 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）

POLL_INTERVAL = 60
POLL_TIMEOUT = 90
RESTART_INTERVAL = 3

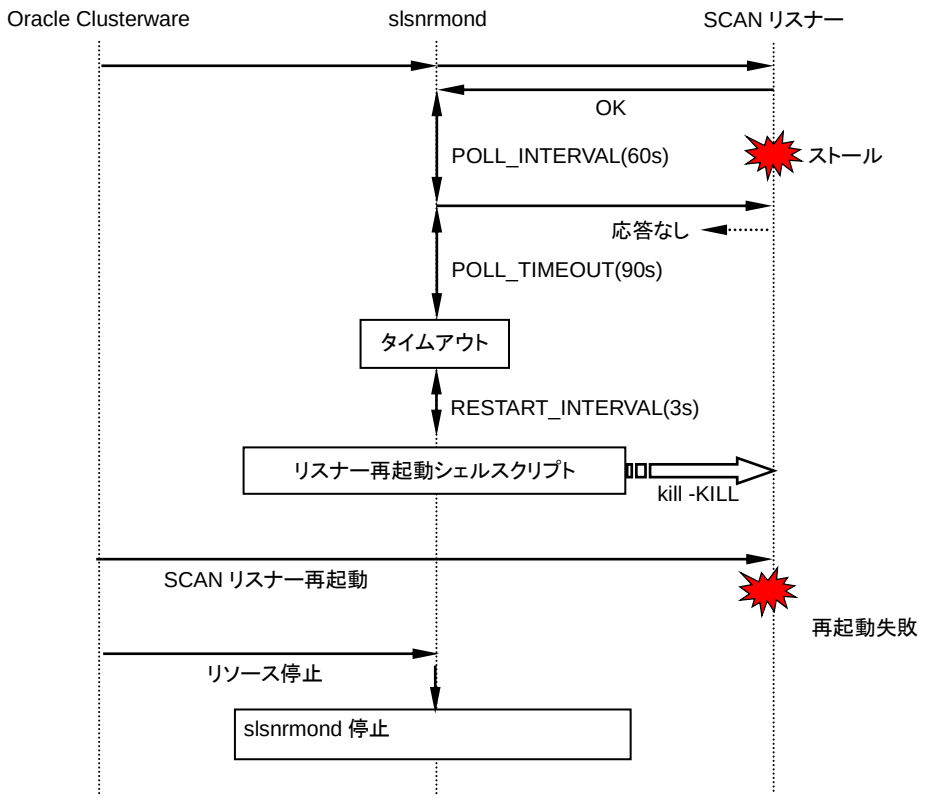


図 4-59 SCAN リスナー監視タイムアウト時の動作（Oracle Clusterware によって SCAN リスナーが再起動されなかった場合）

Oracle Clusterware による SCAN リスナーの再起動を待ち合わせても、SCAN リスナーが再起動されなかった場合は、SCAN リスナー監視モニタは、Oracle Clusterware により停止されます。

4.12. 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
    }
}
```

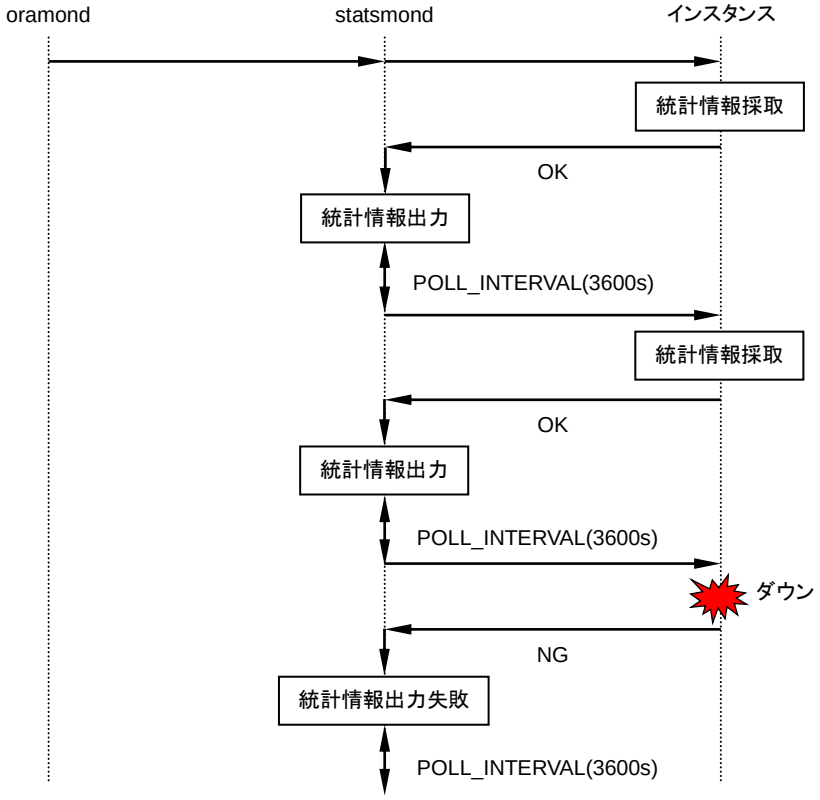


図 4-60 統計情報採取時の動作

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 パラメータおよび 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。

4.13. Oracle のシステムステート・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、Oracle のシステムステート・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、Oracle のシステムステート・ダンプの採取と同時に行います。

シェルスクリプトの実行動作の詳細は、「4.14 シェルスクリプトの実行時の動作」を参照してください。

Oracle のシステムステート・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
}
```

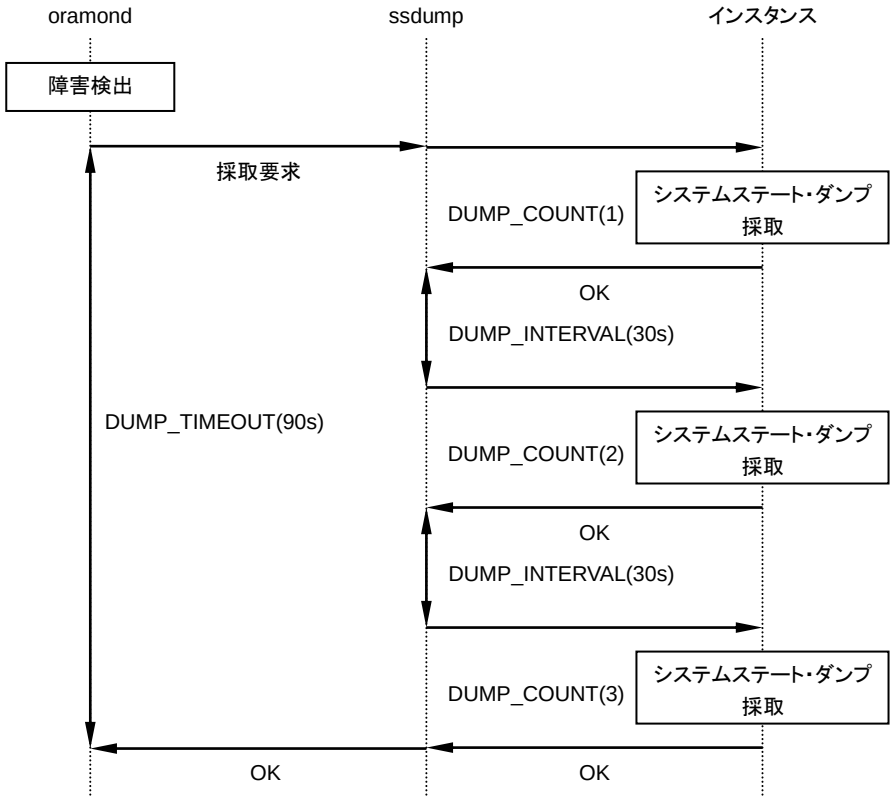


図 4-61 Oracle のシステムステート・ダンプ採取時の動作

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 Oracle のシステムステート・ダンプを採取します。Oracle のシステムステート・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 Oracle のシステムステート・ダンプの採取が完了しなかった場合、Oracle のシステムステート・ダンプの採取を停止します。

また、マルチテナント構成の場合は、CDB 上で Oracle のシステムステート・ダンプを採取します。

4.14. 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。

Oracle のシステムステート・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、「4.13 Oracle のシステムステート・ダンプ採取時の動作」を参照してください。

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。

4.15. 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、リスナー監視または ASM インスタンス監視で障害を検出した場合に、Oracle Clusterware／Oracle Restart によってインスタンス／リスナー／ASM インスタンスが再起動されるか判断し、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      = 2
}
```

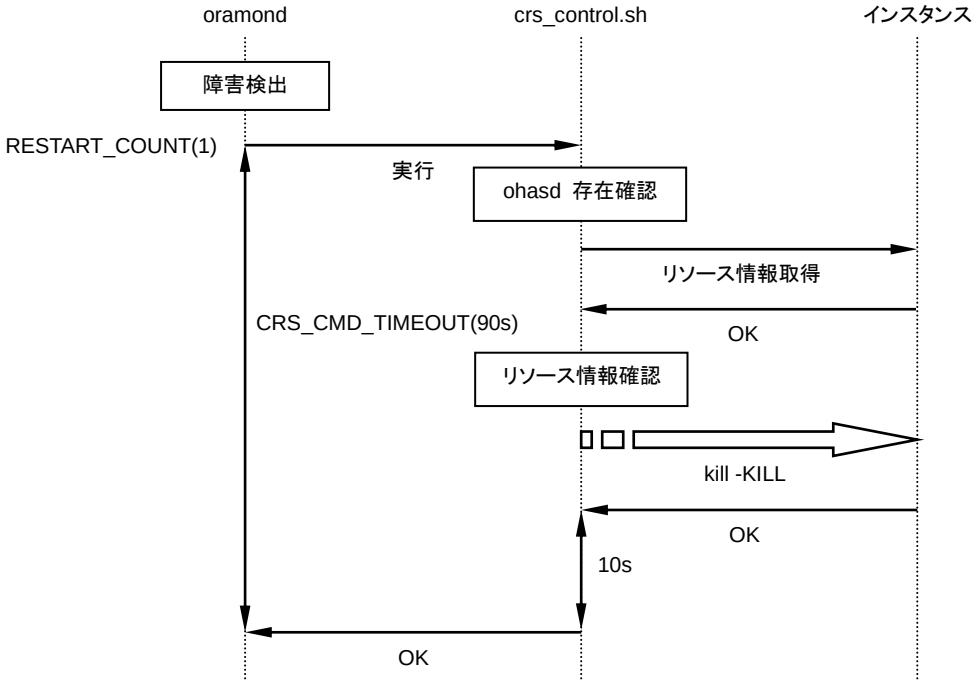


図 4-62 インスタンス監視時の Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトの動作

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、GRID_USER パラメータまたは MONITOR_USER パラメータに設定したユーザー（※1）で crsctl コマンドを実行し、リソース情報を取得します。

リソース情報からインスタンスの再起動が可能と判断した場合、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を促します。

10 秒待機後、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、リソース情報が取得できない、またはリソースがクリーンアップ中でインスタンス再起動の判断ができない場合は、再度リソース情報を取得します。

ohasd が存在しない、リソース情報からインスタンスが再起動できないと判断した、または SMON プロセスの KILL に失敗した場合、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、モニタ制御デーモンは、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） GRID_USER 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た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         = 5
    NET_SERVICE_NAME      = LSNR
    TARGET_RESTART        = NO
}
```

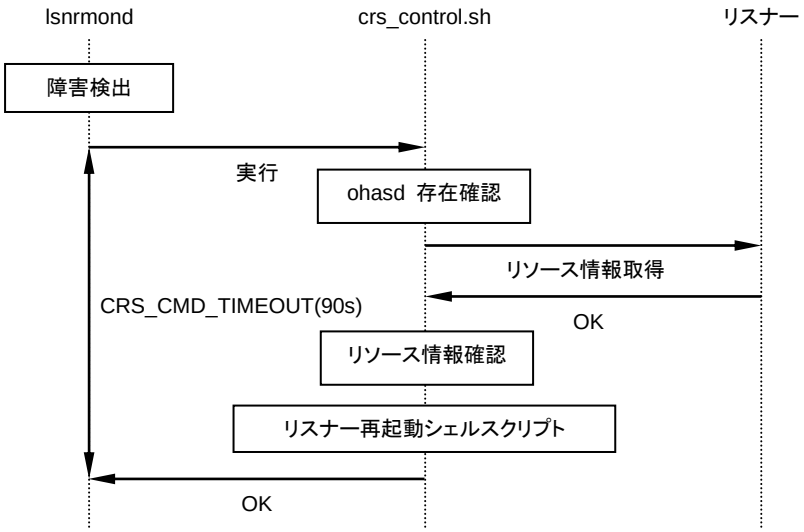


図 4-63 リスナー監視時の Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトの動作

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。

リソース情報からリスナーの再起動が可能と判断した場合、リスナー再起動シェルスクリプト (lsnr_control.sh) を実行し、リスナーを停止します。

リスナー再起動シェルスクリプトの詳細は、「4.17 リスナー再起動シェルスクリプトの動作」を参照してください。

その後、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、リソース情報が取得できない、またはリソースがクリーンアップ中でリスナー再起動の判断ができない場合は、再度リソース情報を取得します。

ohasd が存在しない、またはリソース情報からリスナーが再起動できないと判断した場合、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およびリスナー再起動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。

ASM 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）

```
MONITOR_USER        = grid
HALT_METHOD         = KILL
RESTART_COUNT       = 5
CRS_CMD_TIMEOUT     = 90
```

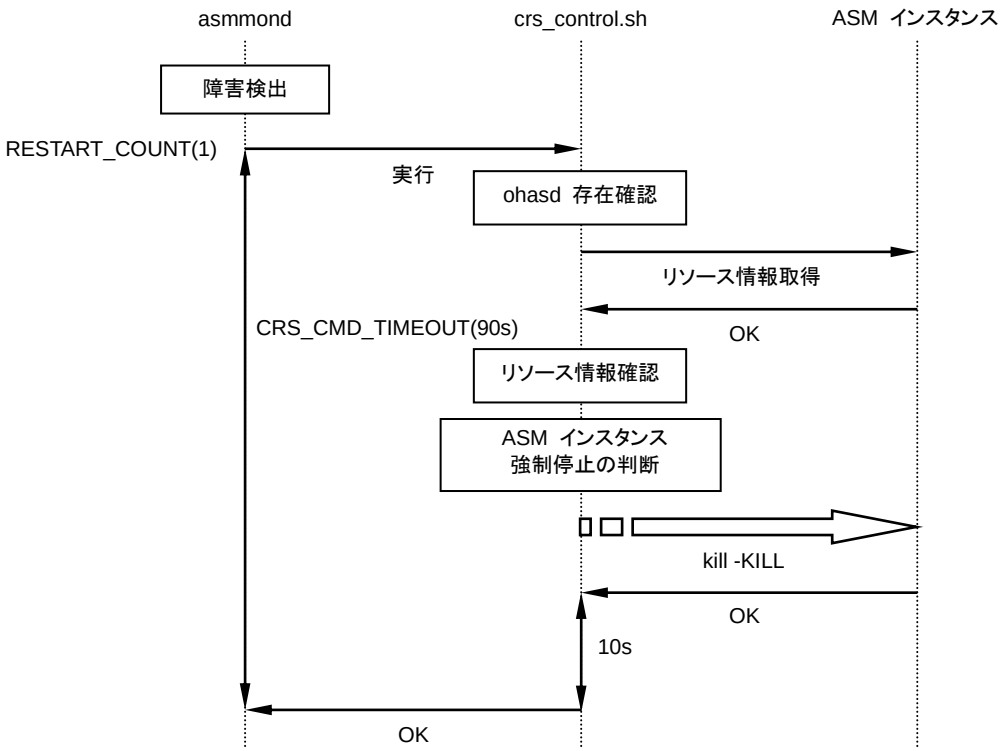


図 4-64 ASM インスタンス監視時の Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトの動作

ASM 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、ASM 監視モニタ (asmmond) 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。

リソース情報から ASM インスタンスの再起動が可能と判断した場合、HALT_METHOD パラメータの設定により、ASM インスタンスを強制停止するか判断します。

HALT_METHOD パラメータに KILL が設定されていると、ASM インスタンスの 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 ASM インスタンスの再起動を促します。

10 秒待機後、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、リソース情報が取得できない、またはリソースがクリーンアップ中で ASM インスタンス再起動の判断ができない場合は、再度リソース情報を取得します。

ohasd が存在しない、リソース情報から ASM インスタンスが再起動できないと判断した、または ASM インスタンスの SMON プロセスの KILL に失敗した場合、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、ASM 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。

ASM インスタンス監視時、ASM インスタンスの SMON プロセスに対して kill コマンドを実行せず、Oracle Clusterware／Oracle Restart による ASM インスタンスの再起動を待ち合わせる場合の 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）

MONITOR_USER = grid
HALT_METHOD = NONE
RESTART_COUNT = 5
CRS_CMD_TIMEOUT = 90

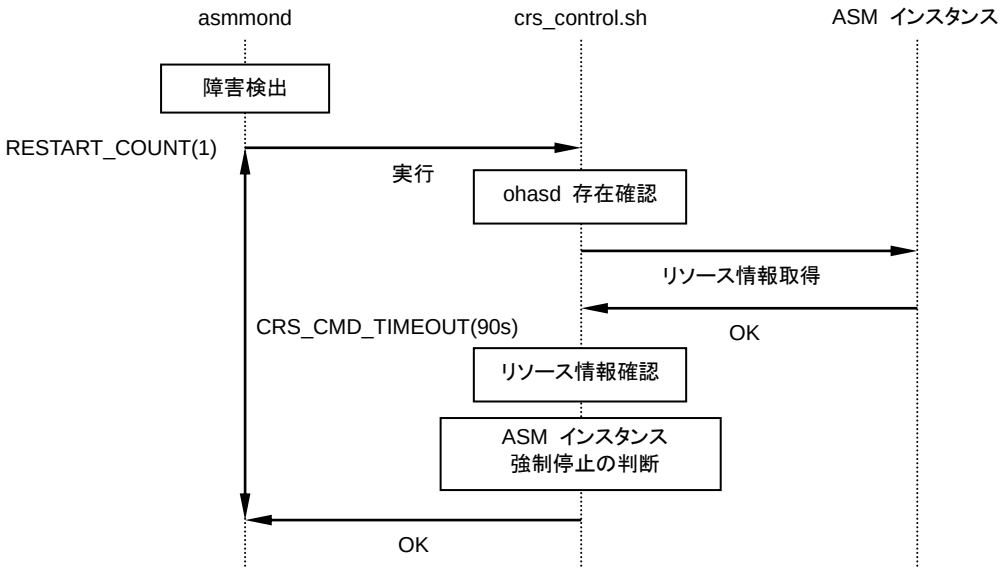


図 4-65 ASM インスタンス監視時の Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトの動作（ASM インスタンスの SMON プロセスに対して kill コマンドを実行せず、Oracle Clusterware／Oracle Restart による ASM インスタンス再起動を待ち合わせる場合）

ASM 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、ASM 監視モニタが 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。

リソース情報から ASM インスタンスの再起動が可能と判断した場合、HALT_METHOD パラメータの設定により、ASM インスタンスを強制停止するか判断します。

HALT_METHOD パラメータに NONE が設定されていると、ASM インスタンスの SMON プロセスに対して kill コマンドを実行せず、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、リソース情報が取得できない、またはリソースがクリーンアップ中で ASM インスタンス再起動の判断ができない場合は、再度リソース情報を取得します。

ohasd が存在しない、またはリソース情報から ASM インスタンスが再起動できないと判断した場合、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、ASM 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。

4.16. PDB 再オープンシェルスクリプトの動作

PDB 再オープンシェルスクリプト (am_pdb_reopen.sh) は、PDB 監視で障害を検出した場合に、PDB 監視モニタ (pdbmond) から実行され、監視対象の PDB の再オープンまたはクローズを行うシェルスクリプトです。

PDB のオープン・モードを取得し、再オープン前と同様のオープン・モードで再オープンします。
なお、PDB がマウント状態の場合は、再オープンを行いません。

PDB 監視モニタは、PDB_REOPEN_TIMEOUT パラメータに設定した時間までに PDB 再オープンシェルスクリプトの処理が完了しなかった場合、PDB 再オープンシェルスクリプトを停止します。

4.17. リスナー再起動シェルスクリプトの動作

リスナー再起動シェルスクリプト (lsnr_control.sh) は、リスナー監視で障害を検出した場合に、監視対象のリスナーを停止し、再起動を行うシェルスクリプトです。

ApplicationMonitor によるリスナーの再起動を行う場合のリスナー再起動シェルスクリプトは、以下のような動作になります。

（例）

```
ListenerMonitor LISTENER {
    MONITOR_USER          = oracle
    RESTART_COUNT         = 5
    NET_SERVICE_NAME      = LSNR
    LSNR_CMD_TIMEOUT      = 90
    TARGET_RESTART        = YES
}
```

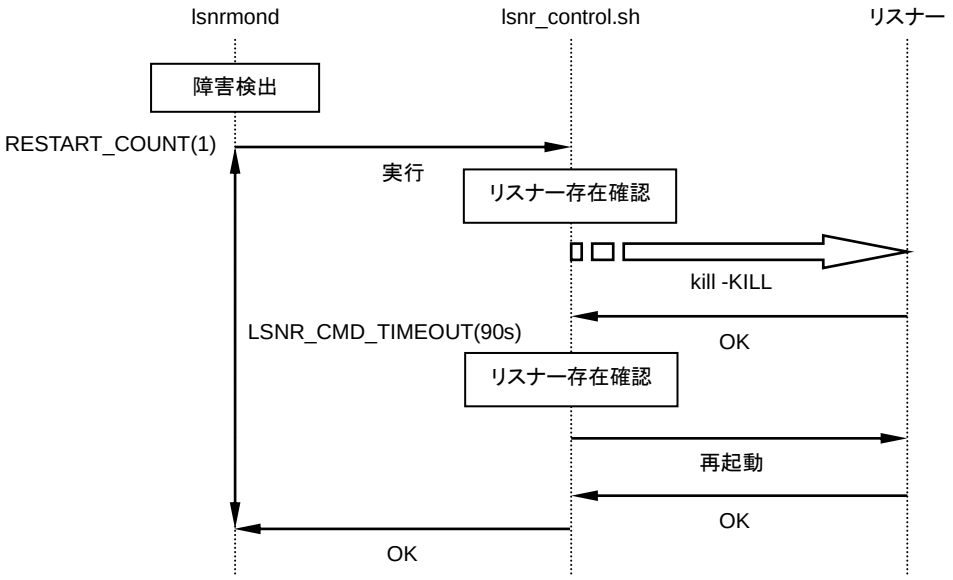


図 4-66 リスナー再起動シェルスクリプトの動作
（ApplicationMonitor によるリスナーの再起動）

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 以上の値が設定されていると、リスナー監視モニタ (lsnrmond) がリスナー再起動シェルスクリプトを実行します。

リスナー再起動シェルスクリプトは、ps コマンドを実行し、リスナーの存在確認を行います。

リスナーの存在確認後、リスナープロセスに対して kill コマンドを実行します。

その後、リスナーが存在していないことを確認します。存在していない場合、lsnrctl コマンドを実行してリスナーを再起動します。

リスナー再起動後、リスナー再起動シェルスクリプトは、正常終了します。

なお、kill コマンド実行後にリスナーが存在していた場合、再度リスナーの存在を確認します。

リスナーの KILL またはリスナーの再起動に失敗した場合、リスナー再起動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、LSNR_CMD_TIMEOUT パラメータに設定した時間までにリスナー再起動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。

Oracle Clusterware／Oracle Restart によるリスナーの再起動を行う場合は、以下のような動作になります。

（例）

```
ListenerMonitor LISTENER {
    MONITOR_USER          = oracle
    RESTART_COUNT         = 5
    NET_SERVICE_NAME      = LSNR
    TARGET_RESTART        = NO
}
```

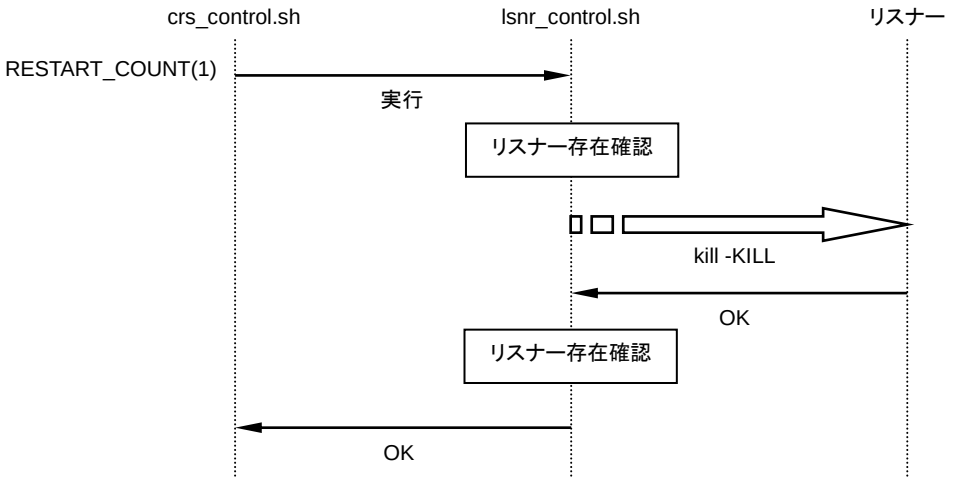


図 4-67 リスナー再起動シェルスクリプトの動作
（Oracle Clusterware／Oracle Restart によるリスナーの再起動）

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 (crs_control.sh) がリスナー再起動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の詳細は、「4.15 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。

リスナー再起動シェルスクリプトは、ps コマンドを実行し、リスナーの存在確認を行います。

リスナーの存在確認後、リスナープロセスに対して kill コマンドを実行します。

その後、リスナーが存在していないことを確認します。存在していない場合、リスナー再起動シェルスクリプトは、正常終了します。

なお、kill コマンド実行後にリスナーが存在していた場合、再度リスナーの存在を確認します。

リスナーの KILL に失敗した場合、リスナー再起動シェルスクリプトは、異常終了します。

Oracle ClusterwareによるASM リスナーの再起動を行う場合は、以下のような動作になります。
（例）
LSNR_CMD_TIMEOUT　= 90

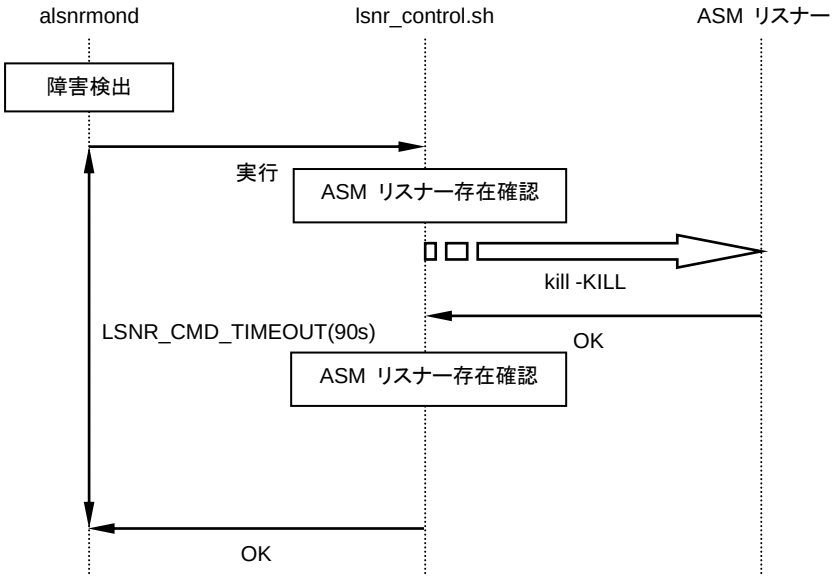


図 4-68 リスナー再起動シェルスクリプトの動作
（Oracle Clusterware による ASM リスナーの再起動）

ASM リスナー監視で障害を検出した場合、ASM リスナー監視モニタ (alsnrmond) がリスナー再起動シェルスクリプトを実行します。

リスナー再起動シェルスクリプトは、ps コマンドを実行し、ASM リスナーの存在確認を行います。
ASM リスナーの存在確認後、ASM リスナープロセスに対して kill コマンドを実行します。
その後、ASM リスナーが存在していないことを確認します。存在していない場合、リスナー再起動シェルスクリプトは、正常終了します。

なお、kill コマンド実行後に ASM リスナーが存在していた場合、再度 ASM リスナーの存在を確認します。
ASM リスナーの KILL に失敗した場合、リスナー再起動シェルスクリプトは、異常終了します。

Oracle Clusterware による SCAN リスナーの再起動を行う場合は、以下のような動作になります。

（例）

LSNR_CMD_TIMEOUT = 90

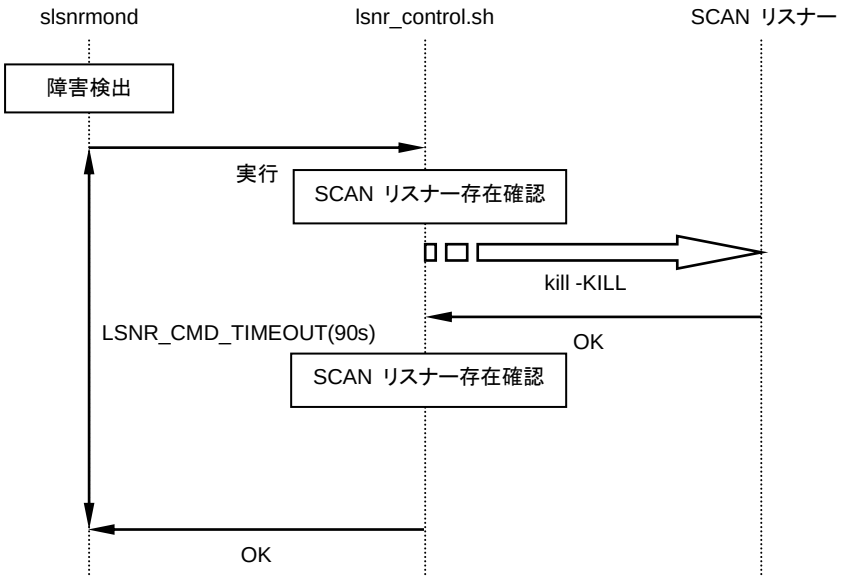


図 4-69 リスナー再起動シェルスクリプトの動作
（Oracle Clusterware による SCAN リスナーの再起動）

SCAN リスナー監視で障害を検出した場合、SCAN リスナー監視モニタ (slsnrmond) がリスナー再起動シェルスクリプトを実行します。

リスナー再起動シェルスクリプトは、ps コマンドを実行し、SCAN リスナーの存在確認を行います。

SCAN リスナーの存在確認後、SCAN リスナープロセスに対して kill コマンドを実行します。

その後、SCAN リスナーが存在していないことを確認します。存在していない場合、リスナー再起動シェルスクリプトは、正常終了します。

なお、kill コマンド実行後に SCAN リスナーが存在していた場合、再度 SCAN リスナーの存在を確認します。

SCAN リスナーの KILL に失敗した場合、リスナー再起動シェルスクリプトは、異常終了します。

4.18. Oracle Data Guard Broker による自動フェイルオーバ時の監視動作

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を検出し、自動フェイルオーバ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およびスイッチオーバを実行した場合は、フェイルオーバを検出できません。

Oracle Data Guard Broker による自動フェイルオーバが行われた場合、以下のような動作になります。

（例）

SWITCHOVER_TIMEOUT = 30

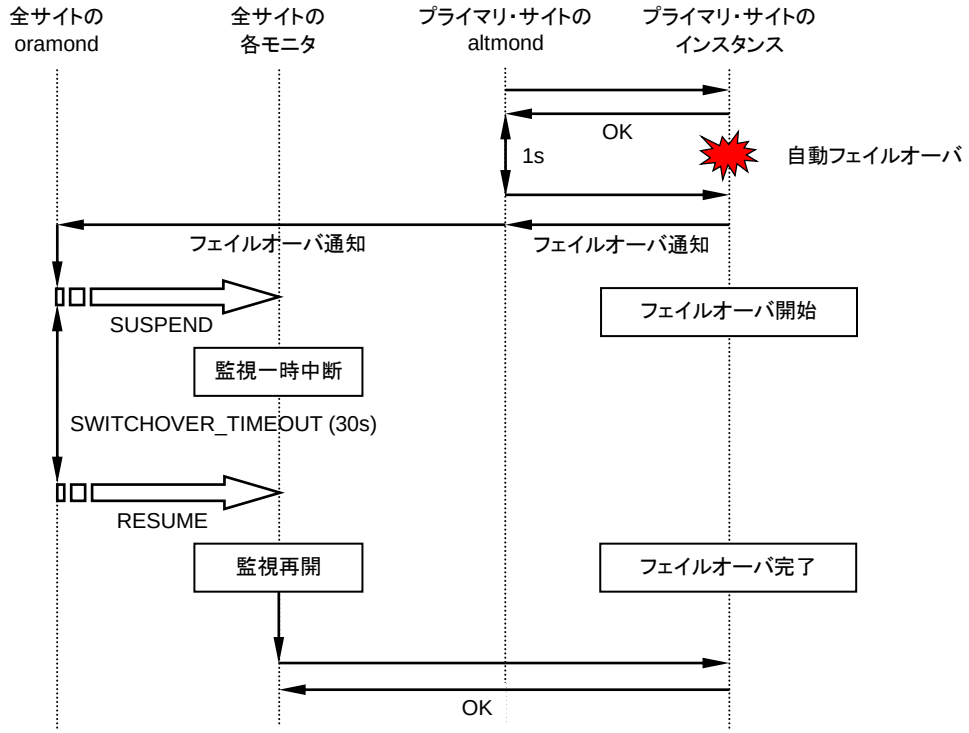


図 4-70 Oracle Data Guard Broker による自動フェイルオーバ時の監視動作

Oracle Data Guard Broker によって自動フェイルオーバが実行された場合、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、各モニタは監視および採取を一時停止します。

各モニタの一時停止完了後、自動フェイルオーバ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が完了しなかった場合、モニタ制御デーモンは、各モニタに対して RESUME 要求(監視再開要求)を送ります。この要求により、各モニタは監視および採取を再開します。

4.19. 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、監視および採取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
}
```

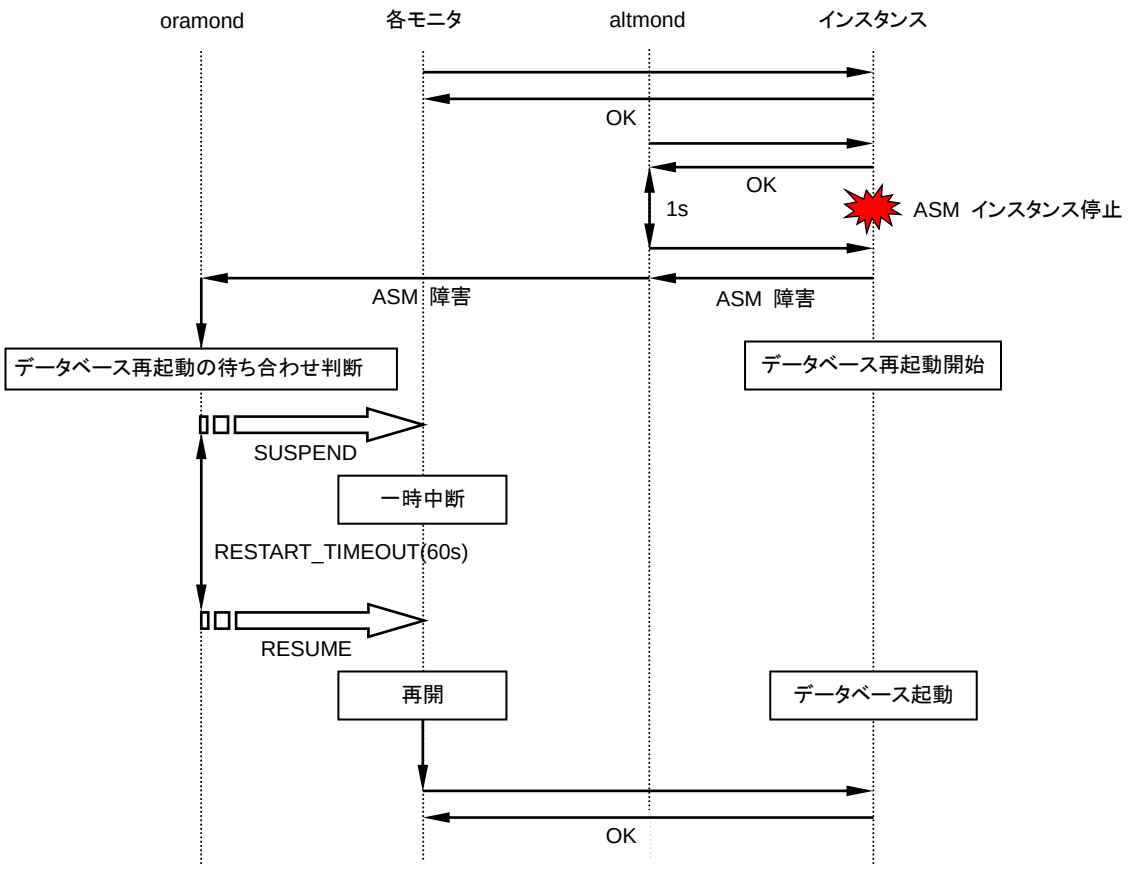


図 4-71 ASM インスタンス停止に伴うデータベースの再起動時の監視動作

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、各モニタは監視および採取を一時停止します。

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 ステートメント内の 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、各モニタは監視および採取を再開します。

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。

4.20. ライセンスチェックによる機能制限

ApplicationMonitor の各機能を使用するためには、コードワードの登録が必要です。

モニタ制御デーモン (oramond) と単独起動モニタは、起動時と日付が変わった後の最初の監視時に、ライセンスチェックを行います。

有効なコードワードが登録されずにインストールから 30 日が経過した場合には、機能を制限します。機能制限中も ApplicationMonitor の起動は可能ですが、障害の検出は行いません。

機能制限中は、ライセンスチェックのタイミングで、syslog に以下のメッセージを出力します。

Monitoring stop until activation succeeded.

また、管理コマンドの show リクエスト実行時に、機能制限中の監視モニタが動作中ノードには、「Monitoring stop until activation succeeded.」を出力します。

以下は、node1 で機能制限中の監視モニタが動作中の場合の出力イメージです。

```
Connect to monitor-control(master).
NODE: node1 RESUME
    monitors        name            m-status    t-status    interval    tat/limit   control
    ----------------------------------------------------------------------------------------------------------------
    CRS             n/a             ACTIVE      UNKNOWN     90          0/180       NO
    ASM             n/a             ACTIVE      UNKNOWN     90          1/120       NO
    INSTANCE        n/a             UNKNOWN     UNKNOWN     n/a         n/a         YES
    LISTENER        LISTENER1       ACTIVE      UNKNOWN     60          0/90        NO
    SSDUMP          n/a             UNKNOWN     n/a         n/a         n/a         YES
    PLUGGABLE       n/a             UNKNOWN     n/a         n/a         n/a         YES
Monitoring stop until activation succeeded.
NODE: node2 RESUME
    monitors        name            m-status    t-status    interval    tat/limit   control
    ----------------------------------------------------------------------------------------------------------------
    CRS             n/a             ACTIVE      UP          90          0/180       NO
    ASM             n/a             ACTIVE      UP          90          1/120       NO
    INSTANCE        n/a             ACTIVE      UP          90          1/120       YES
    LISTENER        LISTENER2       SUSPEND     UP          60          1/90        NO
    SSDUMP          n/a             ACTIVE      n/a         n/a         n/a         YES
    PLUGGABLE       n/a             ACTIVE      n/a         n/a         n/a         YES
```

機能制限は、有効なコードワードを登録し ApplicationMonitor を起動し直すことで、解除されます。

管理コマンドを使用して、ApplicationMonitor を起動したまま機能制限を解除することも可能ですが、すべてのモニタ制御デーモンと単独起動モニタに対して、機能制限解除の apply リクエストを実行する必要があります。

機能制限解除の具体的なコマンドについては、「6 コマンドリファレンス」を参照してください。

5. 製品の運用

ApplicationMonitor を運用するための手順を説明します。

本章では、CLUSTERPRO X SingleServerSafe(以降 CLUSTERPRO X SSS と記載します) を使用した、以下の構成を例に説明します。

なお、CLUSTERPRO X を使用する場合も、ApplicationMonitor の設定や CLUSTERPRO X との連携設定に違いありません。CLUSTERPRO X を使用する場合は、「CLUSTERPRO X SSS」を「CLUSTERPRO X」と読み替えてください。

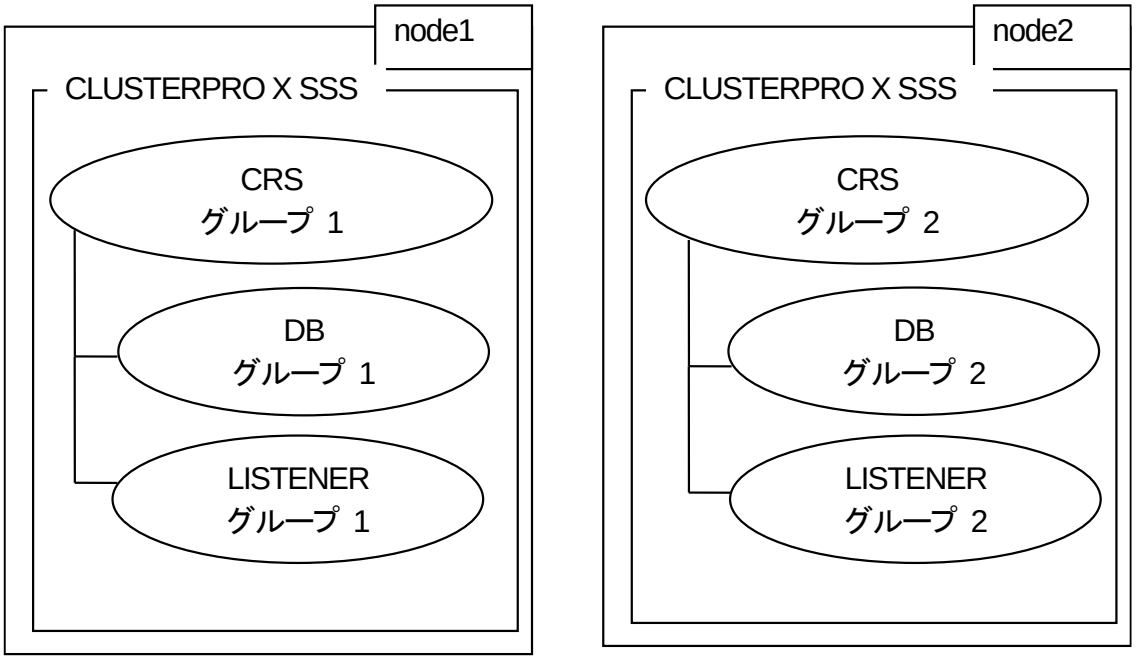


図 5-1 CLUSTERPRO X SSS を使用した RAC 構成例

CLUSTERPRO X SSS を使用した 2 ノード RAC 構成の環境です。

Oracle Clusterware／Oracle Restart を起動させ、その延長で ASM インスタンスを起動させるグループ（以後 CRS グループ と記載します）、インスタンスを起動させるグループ（以後 DB グループ と記載します）、DB グループとは別に仮想 IP を使用するリスナーを起動させるグループ（以後 LISTENER グループ と記載します）を作成しています。

CRS グループに CRS 監視モニタ (crsmond) および ASM 監視モニタ (asmmond)、DB グループにモニタ制御デーモン (oramond)、LISTENER グループに単独起動のリスナー監視モニタ (lsnrmond) を起動／停止するリソースを追加します。

Oracle Clusterware／Oracle Restart 構成で監視する場合は、以下の構成を推奨します。

- ◆ インスタンスは、Oracle Clusterware／Oracle Restart と ApplicationMonitor の両方から監視します。
- ◆ scan-vip 用リスナーは、Oracle Clusterware／Oracle Restart から監視します。
- ◆ scan-vip 用以外のリスナーは、Oracle Clusterware／Oracle Restart と ApplicationMonitor の両方から監視します。
- ◆ Oracle Clusterware／Oracle Restart は、ApplicationMonitor から監視します。
- ◆ ASM インスタンスは、Oracle Clusterware／Oracle Restart と ApplicationMonitor の両方から監視します。

Oracle Clusterware／Oracle Restart は、デフォルト設定の場合、インスタンス、ASM インスタンスおよびリスナーを監視し、監視対象の停止障害発生時に自動で再起動を行いますが、ApplicationMonitor は、Oracle Clusterware／Oracle Restart によるインスタンス、ASM インスタンスおよびリスナーの再起動が行われても監視を継続することができます。

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』および『Oracle Database 管理者ガイド』を参照してください。

5.1. 事前準備

以下の事前準備が必要です。

- ◆ CLUSTERPRO X SSS で CRS グループ 1,2 が作成されている(※1)。
- ◆ CLUSTERPRO X SSS で DB グループ 1,2 が作成されている(※2)。
- ◆ CLUSTERPRO X SSS で LISTENER グループ 1,2 が作成されている(※3)。
- ◆ CLUSTERPRO X SSS で上記のグループ内のすべての Oracle 起動用の EXEC リソースが起動する。

(※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 Grid Infrastructure をインストールしたホームディレクトリ＞/bin/crsctl disable has
```

(※2) Oracle Clusterware／Oracle Restart 起動の延長で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" に変更します。
環境に合わせて設定してください。

```
% srvctl modify database -d <database_name> -policy 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。

(※3) 仮想 IP を使用するリスナーとして、Oracle のデフォルトのリスナー LISTENER を監視対象としています。
各サーバの tnsnames.ora ファイルに、自サーバ用の仮想 IP アドレスを使用するネット・サービスを構成し、tnsping コマンドでそのネット・サービス名からアクセス可能なことを確認しておいてください。
ネット・サービスは、リスナーとネット・サービスが1対1となるように構成してください。
ApplicationMonitor のリスナー監視設定の NET_SERVICE_NAME パラメータには、そのネット・サービス名を指定します。
また、LISTENER グループ 1,2 の Oracle 起動用の EXEC リソースの開始スクリプトでは、自ノード用の仮想 IP が他サーバに配置されている場合には自サーバに再配置し、Oracle のデフォルトのリスナー LISTENER が停止している場合には起動してください。

CLUSTERPRO X SSS の設定などの詳細は、CLUSTERPRO X SSS のマニュアルを参照してください。

5.1.1. Oracle ハング検知機能の設定（RAC 構成のみ）

Oracle では、Oracle ハング検知機能にてハングが検知された場合に、インスタンスを停止する設定がデフォルトで有効になっています。

RAC 構成の環境で監視する場合は、インスタンスを停止する設定を無効にすることを推奨します。

インスタンスを停止する設定を無効にするには、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、事前準備は終了です。

5.2. ApplicationMonitor の設定

5.2.1. 設定ファイル (oramond.conf)

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。

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
  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
  OSDBA(dba) グループに所属する Oracle ユーザー名を指定してください。

➢ PluggableDatabaseMonitor ステートメント
PDB 障害の監視機能を使用する場合、PluggableDatabaseMonitor ステートメントを設定してください。

・ OWNER_NAME パラメータ

PDB 障害の監視機能を使用する場合、PDB 上に作成した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。

✧ PluggableDatabase ステートメント

PDB 障害の監視機能を使用する場合、PluggableDatabaseMonitor ステートメント内に PluggableDatabase ステートメントを設定してください。

設定した PluggableDatabase 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 PDB 名を指定してください。

● Table ステートメント

PDB 障害の監視機能と合わせて PDB 上の表の監視機能を使用する場合、PluggableDatabase ステートメント内に Table ステートメントを設定してください。

設定した Table ステートメント指示行に Oracle の設定を参照して、監視対象の表名を指定してください。

・ TABLE_USER パラメータ

PDB 障害の監視機能と合わせて PDB 上の表の監視機能を使用する場合、監視対象の表にアクセス可能な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設定してください。

設定した 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設定してください。

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 パラメータに指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 DB 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

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。

● Table ステートメント

非マルチテナント構成の環境で、インスタンス障害の監視機能と合わせて表の監視機能を使用する場合、InstanceMonitor ステートメント内に Table ステートメントを設定してください。

設定した Table ステートメント指示行に Oracle の設定を参照して、監視対象の表名を指定してください。

・ TABLE_USER パラメータ

非マルチテナント構成の環境で、インスタンス障害の監視機能と合わせて表の監視機能を使用する場合、監視対象の表にアクセス可能なユーザー名を指定してください。

・ TABLE_PASSWORD パラメータ

非マルチテナント構成の環境で、インスタンス障害の監視機能と合わせて表の監視機能を使用する場合、監視対象の表にアクセス可能なユーザー名のパスワードを指定してください。

✧ 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 ステートメントを設定してください。
設定した 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 を指定した TableSpaceMonitor ステートメント内に TableSpace ステートメントを設定してください。
設定した 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 ステートメントを設定してください。

✧ 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 ステートメントを設定してください。

● 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 ステートメントを設定してください。

□ 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 ステートメントを設定してください。
設定した 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・サイトへ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 ステートメント内の SITE_NODE パラメータに指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 ステートメントを設定してください。

- 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 ステートメントを設定してください。

✧ 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 ステートメントを設定してください。
設定した 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 を指定した StatsMonitor ステートメント内に Stats ステートメントを設定してください。
設定した 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。

5.2.2. CRS 監視設定ファイル (crsmond.conf)

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 監視設定ファイルを作成します。

```
# cd /etc/opt/HA/AM/conf
# cp sample/crsmond.conf.sample ./crsmond.conf
# vi crsmond.conf
```

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
  構成に合わせて、Oracle Clusterware／Oracle Restart のデーモン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。

5.2.3. ASM 監視設定ファイル (asmmond.conf)

ASM インスタンス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 ASM 監視設定ファイルを作成します。

```
# cd /etc/opt/HA/AM/conf
# cp sample/asmmond.conf.sample ./asmmond.conf
# vi asmmond.conf
```

ASM インスタンス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。

- MONITOR_USER パラメータ
  Oracle の OSASM グループに所属するユーザー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。

- ORACLE_SID パラメータ
  ASM 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。

- 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。

「3.10 ASM 監視設定ファイル (asmmond.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。

5.2.4. ASM 監視モニタ起動スクリプト (start_asmmond.sh)

ASM インスタンスの監視をする場合は、ASM 監視設定ファイル (asmmond.conf) のほかに ASM 監視モニタ起動スクリプトの設定が必要です。Oracle の設定を参照し、ASM 監視モニタ起動スクリプトを設定します。

```
# cd /etc/opt/HA/AM/lbin
# vi start_asmmond.sh
```

ASM 監視モニタ (asmmond) を起動するためには、以下の設定が必要です。

- 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。

- Oracle 環境変数 (ORACLE_SID, ORACLE_BASE, ORACLE_HOME, LD_LIBRARY_PATH, ORA_NLS10, NLS_LANG)
  Oracle の設定を参照し、各環境変数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。

設定した ASM 監視モニタ起動スクリプトは、バックアップすることを推奨します。

ASM 監視モニタ起動スクリプトの詳細は、「3.11 ASM 監視モニタ起動スクリプト (start_asmmond.sh) の設定」を参照してください。

以上で、ApplicationMonitor の設定は終了です。

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。

CRS グループ 1,2、DB グループ 1,2 および LISTENER グループ 1,2 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。

なお、現用待機構成の環境で、設定確認が正常に完了すると、以下の実行結果になります。

```
# /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.
#
```

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 2.9 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。**

注意: **ASM 監視設定ファイル (asmmond.conf) の設定確認は、管理コマンドでは行えません。ASM 監視設定ファイルの設定確認は、ASM 監視モニタ (asmmond) 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。

CRS グループ 1,2、DB グループ 1,2 および LISTENER グループ 1,2 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。

なお、現用待機構成の環境で、設定配布が正常に完了すると、以下の実行結果になります。

```
# /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=node2
Finished to check configuration-file.
Succeeded in apply configuration file.
#
```

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 2.9 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。

注意: ASM 監視設定ファイル (asmmond.conf) の設定配布は、管理コマンドでは行えません。各ノードで ASM 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 と CLUSTERPRO X SSS の連携

設定を終えた ApplicationMonitor と CLUSTERPRO X SSS を連携します。
図 5-2 は、PID モニタリソースを利用して連携する場合の構成例です。
連携は、CLUSTERPRO X SSS の GUI から設定します。

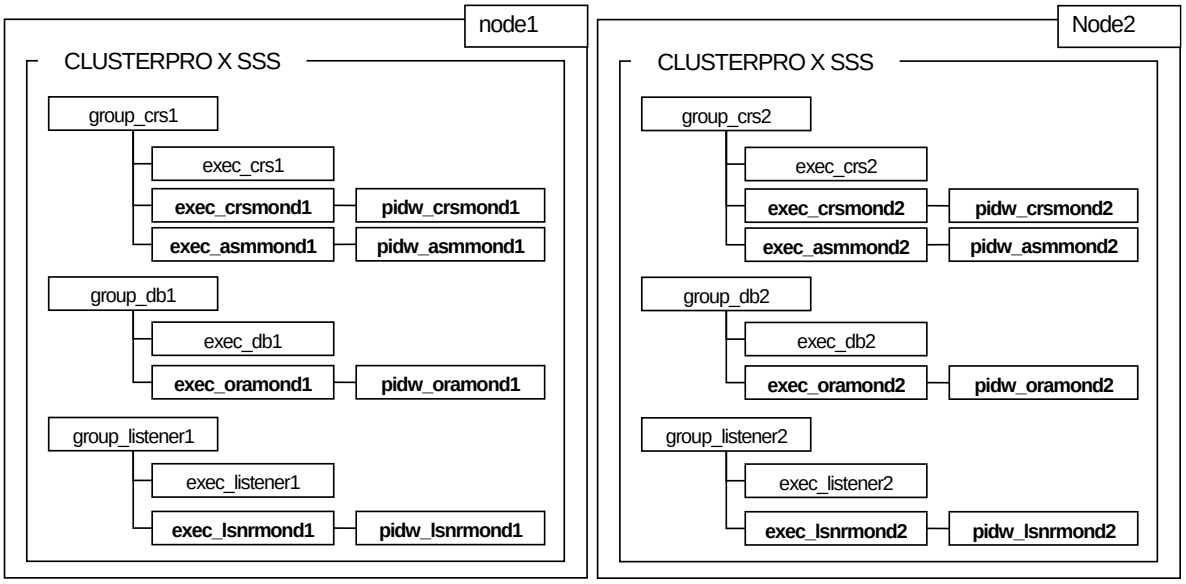


図 5-2 CLUSTERPRO X SSS の リソース構成例

以下は、各グループおよびリソースの設定名です。

グループおよびリソース	設定名
CRS グループ	group_crs1,2
DB グループ	group_db1,2
LISTENER グループ	group_listener1,2
Oracle Clusterware／Oracle Restart、ASM インスタンスを起動／停止させる EXEC リソース	exec_crs1,2
インスタンスを起動／停止させる EXEC リソース	exec_db1,2
仮想 IP を使用するリスナーを起動／停止させる EXEC リソース	exec_listener1,2
CRS 監視モニタ (crsmond) を起動／停止させる EXEC リソース	exec_crsmond1,2
ASM 監視モニタ (asmmond) を起動／停止させる EXEC リソース	exec_asmmond1,2
モニタ制御デーモン (oramond) を起動／停止させる EXEC リソース	exec_oramond1,2
単独起動のリスナー監視モニタ (lsnrmond) を起動／停止させる EXEC リソース	exec_lsnrmond1,2
CRS 監視モニタを監視する PID モニタリソース	pidw_crsmond1,2
ASM 監視モニタを監視する PID モニタリソース	pidw_asmmond1,2
モニタ制御デーモンを監視する PID モニタリソース	pidw_oramond1,2
単独起動のリスナー監視モニタを監視する PID モニタリソース	pidw_lsnrmond1,2

注意：　本ユーザーズガイドでは、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。
なお、操作手順や表示に関する説明は、CLUSTERPRO X SSS 3.3（内部バージョン 3.3.5-1）を使用した場合の例となります。
CLUSTERPRO X SSS の設定および表示などに関する詳細は、CLUSTERPRO X SSS 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

CRS グループ 1,2、DB グループ 1,2 および LISTENER グループ 1,2 に ApplicationMonitor を起動／停止するリソース（EXEC リソース）を追加します。

1 「exec_crsmond1」を追加します。
グループ「group_crs1」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 EXEC リソース名を入力します。
「名前」を「exec_crsmond1」とし、「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、「利用可能なリソース」から Oracle 起動用の EXEC リソースを選択します。
「exec_crs1」を選択し、「追加」をクリックします。
「依存するリソース」に「exec_crs1」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 X SSS 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。

各 EXEC リソースの開始スクリプトは、以下のように設定します。

EXEC リソース名	ApplicationMonitor を起動する処理
exec_crsmond1	/etc/opt/HA/AM/lbin/crsmond
exec_crsmond2	
exec_asmmond1	/etc/opt/HA/AM/lbin/start_asmmond.sh
exec_asmmond2	
exec_oramond1	/etc/opt/HA/AM/lbin/oramond
exec_oramond2	
exec_lsnrmond1	/etc/opt/HA/AM/lbin/lsnrmond -s LISTENER
exec_lsnrmond2	/etc/opt/HA/AM/lbin/lsnrmond -s LISTENER

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 「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。
CLUSTERPRO X SSS 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。

各 EXEC リソースの終了スクリプトは、以下のように設定します。

EXEC リソース名	ApplicationMonitor を停止する処理
exec_crsmond1	kill `cat /var/opt/HA/AM/crsmond/crsmond.lock`
exec_crsmond2	
exec_asmmond1	kill `cat /var/opt/HA/AM/asmmond/asmmond.lock`
exec_asmmond2	
exec_oramond1	kill `cat /var/opt/HA/AM/locks/oramond0.pid`
exec_oramond2	
exec_lsnrmond1	pid=`ps -ef | grep 'lsnrmond -s LISTENER' | grep -v grep | awk '{print $2}'` if [[${pid} != ""]] then kill ${pid} fi
exec_lsnrmond2	pid=`ps -ef | grep 'lsnrmond -s LISTENER' | grep -v grep | awk '{print $2}'` if [[${pid} != ""]] then kill ${pid} fi

注意: oramond.pid ファイルは、構成番号が自動的に付与されます。
oramond<構成番号>.pid
モニタ制御デーモンを停止する場合は、構成番号を指定してください。

ApplicationMonitor の停止処理は、起動モニタに対して SIGTERM を送るように終了スクリプトを設定してください。

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 「完了」をクリックし、「グループのリソース定義」を終了します。

7 同様の手順で exec_crsmond2 、exec_asmmond1,2 、exec_oramond1,2 、exec_lsnrmond1,2 を追加します。

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。

1 「pidw_crsmond1」を作成します。
「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」にモニタリソース名を入力します。
「名前」を「pidw_crsmond1」とし、「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 リソースを選択します。
「exec_crsmond1」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、グループを選択します。
「failover_crs1」を選択し、「最大再活性回数」を 0 回に設定します。
「完了」をクリックし、「モニタリソースの定義」を終了します。

5 同様の手順で pidw_crsmond2 、pidw_asmmond1,2 、pidw_oramond1,2 、pidw_lsnrmond1,2 を作成します。

5.5.3. 構成情報の反映

CLUSTERPRO X SSS の構成情報を 各サーバに反映します。

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 X SSS のマニュアルを参照してください。

CLUSTERPRO X SSS が停止状態の場合は、CLUSTERPRO X SSS を開始します。
CLUSTERPRO X SSS の GUI 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 CLUSTERPRO X SSS の GUI の「表示」メニューより、「操作モード」を選択します。

2 グループの Oracle 起動用 EXEC リソースが「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が「正常」であることを確認してください。

以上で、ApplicationMonitor と CLUSTERPRO X SSS の連携は終了です。

5.6. ApplicationMonitor の起動確認

フェイルオーバグループ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。

以下は、node 1 の出力例です。

```
…
oramond[6705]: 0:ApplicationMonitor 2.9
oramond[6705]: 0:Loading configuration file /etc/opt/HA/AM/conf/oramond0.bin.
oramond[6705]: 0:Logging level is 2.
oramond[6705]: 0:altmond[6708] started.
…
oramond[6705]: 0:Monitoring activity on local node is started.
oramond[6705]: 0:instmond[6729] sid1 started.
oramond[6705]: 0:altmond[6708] status is UP.
oramond[6705]: 0:ssdump[6731] started.
oramond[6705]: 0:tschkmond[6732] TSCHKMOND1 started.
oramond[6705]: 0:fdsmond[6733] started.
oramond[6705]: 0:statsmond[6734] STATSMOND1 started.
oramond[6705]: 0:pdbmond [6735] started.
oramond[6705]: 0:ssdump[6731] status is UP.
oramond[6705]: 0:fdsmond[6733] status is UP.
oramond[6705]: 0:statsmond[6734] STATSMOND1 status is UP.
oramond[6705]: 0:tschkmond[6732] TSCHKMOND1 status is UP.
oramond[6705]: 0:instmond[6729] sid1 status is UP.
oramond[6705]: 0:pdbmond [6735] 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。

ApplicationMonitor 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 によって出力するメッセージの詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 2.9 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 パラメーター覧」、「3.9.4 パラメーター覧」および「3.10.4 パラメーター覧」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 の起動確認は終了です。

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
    ASM           n/a           ACTIVE     UP         90         1/120       NO
    INSTANCE      n/a           ACTIVE     UP         90         1/120       YES
    LISTENER      LISTENER      ACTIVE     UP         60         0/90        NO
    TABLESPACE    TSCHKMOND1    ACTIVE     n/a        3600       n/a         YES
    DISKSPACE     n/a           ACTIVE     n/a        3600       n/a         YES
    STATS         STATSMOND1    ACTIVE     n/a        n/a        n/a         YES
    SSDUMP        n/a           ACTIVE     n/a        n/a        n/a         YES
    PLUGGABLE     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
    ASM           n/a           ACTIVE     UP         90         0/120       NO
    INSTANCE      n/a           ACTIVE     UP         90         1/120       YES
    LISTENER      LISTENER      ACTIVE     UP         60         1/90        NO
    TABLESPACE    TSCHKMOND2    ACTIVE     n/a        3600       n/a         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
    PLUGGABLE     n/a           ACTIVE     n/a        n/a        n/a         YES
#
```

監視モニタおよび採取モニタの動作状態 m-status が **"ACTIVE"**、監視対象の Oracle Clusterware／Oracle Restart、ASM インスタンス、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" で表示しているモニタは、単独起動モニタです。

なお、監視対象の PDB の状態は、"-P" オプションまたは "-a" オプションを付与することによって確認できます。

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。

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
        ASM             n/a             UNKNOWN         UNKNOWN         n/a             n/a             NO
        INSTANCE        n/a             ACTIVE          UP              90              4/120           YES
        LISTENER        LISTENER1       ACTIVE          UP              60              5/90            NO
        SSDUMP          n/a             ACTIVE          n/a             n/a             n/a             YES
        PLUGGABLE       n/a             ACTIVE          n/a             n/a             n/a             YES
NODE: node2 UNKNOWN
        monitors        name            m-status        t-status        interval        tat/limit       control
        ----------------------------------------------------------------------------------------------------------------------------------
        CRS             n/a             UNKNOWN         UNKNOWN         n/a             n/a             NO
        ASM             n/a             UNKNOWN         UNKNOWN         n/a             n/a             NO
        INSTANCE        n/a             UNKNOWN         UNKNOWN         n/a             n/a             YES
        LISTENER        LISTENER2       UNKNOWN         UNKNOWN         n/a             n/a             NO
        SSDUMP          n/a             UNKNOWN         n/a             n/a             n/a             YES
        PLUGGABLE       n/a             UNKNOWN         n/a             n/a             n/a             YES
#
```

Oracle が動作していないノードは、m-status を **"UNKNOWN"** で表示します。

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
    ASM             n/a             ACTIVE      UP          90          111/120      NO
    INSTANCE        n/a             ACTIVE      UP          90          67/120       YES
    LISTENER        LISTENER        ACTIVE      UP          60          35/90        NO
    TABLESPACE      TSCHKMOND1      ACTIVE      n/a         3600        n/a          YES
    DISKSPACE       n/a             ACTIVE      n/a         3600        n/a          YES
    STATS           STATSMOND1      ACTIVE      n/a         n/a         n/a          YES
    SSDUMP          n/a             ACTIVE      n/a         n/a         n/a          YES
    PLUGGABLE       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
    ASM             n/a             ACTIVE      UP          90          105/120      NO
    INSTANCE        n/a             ACTIVE      UP          90          65/120       YES
    LISTENER        LISTENER        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
    STATS           STATSMOND2      ACTIVE      n/a         n/a         n/a          YES
    SSDUMP          n/a             ACTIVE      n/a         n/a         n/a          YES
    PLUGGABLE       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 を表示します。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、現在の設定ですで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。

なお、監視対象の PDB の tat は、"-P" オプションまたは "-a" オプションを付与することによって確認できます。

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。

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 パラメータの設定値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。

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 pdb -P PDB1 –n node1
Success: pdbmond (node = node1, pdb_name = PDB1) is accepted (request = apply).
#
# /opt/HA/AM/bin/oraadmin -c apply -m lsnr -s LISTENER –n node1
Success: lsnrmond (node = node1, listener_name = LISTENER) is accepted (request = apply).
#
# /opt/HA/AM/bin/oraadmin -c apply -m lsnr -s LISTENER –n node2
Success: lsnrmond (node = node2, listener_name = LISTENER) is accepted (request = apply).
#
# /opt/HA/AM/bin/oraadmin -c apply -m crs -n node1
Success: crsmond (node = node1) is accepted (request = apply).
#
# /opt/HA/AM/bin/oraadmin -c apply -m crs -n node2
Success: crsmond (node = node2) is accepted (request = apply).
#
# /opt/HA/AM/bin/oraadmin -c apply -m asm -n node1
Success: asmmond (node = node1) is accepted (request = apply).
#
# /opt/HA/AM/bin/oraadmin -c apply -m asm -n node2
Success: asmmond (node = node2) 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 パラメーター覧」、「3.9.4 パラメーター覧」および「3.10.4 パラメーター覧」を参照してください。

注意: **リスナー監視モニタ (lsnrmond) 数、表領域監視モニタ (tschkmond) 数および統計情報採取モニタ (statsmond) 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 X SSS の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 X SSS の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ことを確認する。**

注意: **CLUSTERPRO X を使用する構成で、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。
以下の手順にしたがって設定を変更してください。

1. **CLUSTERPRO X のフェイルオーバグループを停止する。**
2. **フェイルオーバグループから、ApplicationMonitor を起動するリソースを外す。**
3. **フローティング IP を有効化するため、フェイルオーバ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. **フェイルオーバグループに ApplicationMonitor を起動する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ことを確認する。**

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で、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を障害と判断して CLUSTERPRO X SSS の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
    ASM             n/a             ACTIVE      UP          90          1/120        NO
    INSTANCE        n/a             UNKNOWN     UNKNOWN     n/a         n/a          YES
    LISTENER        LISTENER        ACTIVE      UP          60          0/90         NO
    TABLESPACE      TSCHKMOND1      UNKNOWN     n/a         n/a         n/a          YES
    DISKSPACE       n/a             UNKNOWN     n/a         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
    PLUGGABLE       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
    ASM             n/a             ACTIVE      UP          90          0/120        NO
    INSTANCE        n/a             UNKNOWN     UNKNOWN     n/a         n/a          YES
    LISTENER        LISTENER        ACTIVE      UP          60          1/90         NO
    TABLESPACE      TSCHKMOND2      UNKNOWN     n/a         n/a         n/a          YES
    DISKSPACE       n/a             UNKNOWN     n/a         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
    PLUGGABLE       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 pdb -P PDB1 -n node1
Success: pdbmond (node = node1, pdb_name = PDB1) is accepted (request = suspend).
#
# /opt/HA/AM/bin/oraadmin -c suspend -m lsnr -s LISTENER -n node1
Success: lsnrmond (node = node1, listener_name = LISTENER) is accepted (request = suspend).
#
# /opt/HA/AM/bin/oraadmin -c suspend -m lsnr -s LISTENER -n node2
Success: lsnrmond (node = node2, listener_name = LISTENER) 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).
#
# /opt/HA/AM/bin/oraadmin -c suspend -m asm -n node1
Success: asmmond (node = node1) is accepted (request = suspend).
#
# /opt/HA/AM/bin/oraadmin -c suspend -m asm -n node2
Success: asmmond (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/limi  control
                                                                      t
    ------------------------------------------------------------------------------------------------------------
    CRS           n/a           SUSPEND       UP            90        0/180     NO
    ASM           n/a           SUSPEND       UP            90        1/120     NO
    INSTANCE      n/a           UNKNOWN       UNKNOWN       n/a       n/a       YES
    LISTENER      LISTENER      SUSPEND       UP            60        0/90      NO
    TABLESPACE    TSCHKMOND1    UNKNOWN       n/a           n/a       n/a       YES
    DISKSPACE     n/a           UNKNOWN       n/a           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
    PLUGGABLE     n/a           UNKNOWN       n/a           n/a       n/a       YES
NODE: node2 SUSPEND
    monitors      name          m-status      t-status      interval  tat/limi  control
                                                                      t
    ------------------------------------------------------------------------------------------------------------
    CRS           n/a           SUSPEND       UP            90        0/180     NO
    ASM           n/a           SUSPEND       UP            90        0/120     NO
    INSTANCE      n/a           UNKNOWN       UNKNOWN       n/a       n/a       YES
    LISTENER      LISTENER      SUSPEND       UP            60        1/90      NO
    TABLESPACE    TSCHKMOND2    UNKNOWN       n/a           n/a       n/a       YES
    DISKSPACE     n/a           UNKNOWN       n/a           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
    PLUGGABLE     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。

一時停止した ApplicationMonitor は、Oracle に対するすべてのアクセスを抑止します。

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、すべての監視モニタおよび採取モニタを一時停止してください。
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。

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するには、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
    ASM           n/a           SUSPEND     UP          90         1/120       NO
    INSTANCE      n/a           ACTIVE      UP          90         1/120       YES
    LISTENER      LISTENER      SUSPEND     UP          60         0/90        NO
    TABLESPACE    TSCHKMOND1    ACTIVE      n/a         3600       n/a         YES
    DISKSPACE     n/a           ACTIVE      n/a         3600       n/a         YES
    STATS         STATSMOND1    ACTIVE      n/a         n/a        n/a         YES
    SSDUMP        n/a           ACTIVE      n/a         n/a        n/a         YES
    PLUGGABLE     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
    ASM           n/a           SUSPEND     UP          90         0/120       NO
    INSTANCE      n/a           ACTIVE      UP          90         1/120       YES
    LISTENER      LISTENER      SUSPEND     UP          60         1/90        NO
    TABLESPACE    TSCHKMOND2    ACTIVE      n/a         3600       n/a         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
    PLUGGABLE     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 pdb -P PDB1 -n node1
Success: pdbmond (node = node1, pdb_name = PDB1) is accepted (request = resume).
#
# /opt/HA/AM/bin/oraadmin -c resume -m lsnr -s LISTENER -n node1
Success: lsnrmond (node = node1, listener_name = LISTENER) is accepted (request = resume).
#
# /opt/HA/AM/bin/oraadmin -c resume -m lsnr -s LISTENER -n node2
Success: lsnrmond (node = node2, listener_name = LISTENER) 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).
#
# /opt/HA/AM/bin/oraadmin -c resume -m asm -n node1
Success: asmmond (node = node1) is accepted (request = resume).
#
# /opt/HA/AM/bin/oraadmin -c resume -m asm -n node2
Success: asmmond (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
      ASM          n/a          ACTIVE     UP         90         1/120       NO
      INSTANCE     n/a          ACTIVE     UP         90         1/120       YES
      LISTENER     LISTENER     ACTIVE     UP         60         0/90        NO
      TABLESPACE   TSCHKMOND1   ACTIVE     n/a        3600       n/a         YES
      DISKSPACE    n/a          ACTIVE     n/a        3600       n/a         YES
      STATS        STATSMOND1   ACTIVE     n/a        n/a        n/a         YES
      SSDUMP       n/a          ACTIVE     n/a        n/a        n/a         YES
      PLUGGABLE    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
      ASM          n/a          ACTIVE     UP         90         0/120       NO
      INSTANCE     n/a          ACTIVE     UP         90         1/120       YES
      LISTENER     LISTENER     ACTIVE     UP         60         1/90        NO
      TABLESPACE   TSCHKMOND2   ACTIVE     n/a        3600       n/a         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
      PLUGGABLE    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 は、起動時に Oracle 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 データベースに対するアクセスは停止しますが、ApplicationMonitor のプロセスは動作しているため、Oracle パッチ適用の際は、ApplicationMonitor が起動しているすべてのグループを停止してください。**

5.10.3. 監視対象 PDB の除外

管理コマンド (oraadmin) の suspend リクエストに "-P" オプションを付与することによって、監視中の PDB を監視対象から一時的に外すことができます。

PDB を監視対象から外すには、以下のように監視対象から外す PDB 名を指定し、suspend リクエストを実行してください。

```
# /opt/HA/AM/bin/oraadmin -c suspend -P PDB1
Connect to monitor-control(master).
Succeeded in suspend.
#
```

PDB 監視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 -P all
Connect to monitor-control(master).
NODE: node1 RESUME
   monitors        name        m-status   interval
   ----------------------------------------------------------------
   PLUGGABLE       n/a         ACTIVE     n/a
                   pdb_name                m-status     t-status     interval   tat/limit
                   ---------------------------------------------------------------------------------------
                   PDB1                    UNKNOWN      UNKNOWN      90         1/120
                   PDB2                    ACTIVE       UP           90         0/120
NODE: node2 RESUME
   monitors        name        m-status   interval
   ----------------------------------------------------------------
   PLUGGABLE       n/a         ACTIVE     n/a
                   pdb_name                m-status     t-status     interval   tat/limit
                   ---------------------------------------------------------------------------------------
                   PDB1                    UNKNOWN      UNKNOWN      90         1/120
                   PDB2                    ACTIVE       UP           90         0/120
#
```

監視対象から外した PDB 監視の状態が **"UNKNOWN"** に変わったことを確認してください。

なお、監視対象から外す PDB 名に "all" を指定することによって、すべての PDB を監視対象から一時的に外すことも可能です。

監視対象から外した PDB を監視対象に戻すには、監視対象から外した PDB 名を指定し、resume リクエストを実行してください。

```
# /opt/HA/AM/bin/oraadmin -c resume -P PDB1
Connect to monitor-control(master).
Succeeded in resume.
#
```

PDB 監視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 -P all
Connect to monitor-control(master).
NODE: node1 RESUME
   monitors        name        m-status   interval
   ----------------------------------------------------------------
   PLUGGABLE       n/a         ACTIVE     n/a
                   pdb_name                m-statu    t-status   interval   tat/limit
                                           s
                   ---------------------------------------------------------------------------
                   PDB1                    ACTIVE     UP         90         1/120
                   PDB2                    ACTIVE     UP         90         0/120
NODE: node2 RESUME
   monitors        name        m-status   interval
   ----------------------------------------------------------------
   PLUGGABLE       n/a         ACTIVE     n/a
                   pdb_name                m-statu    t-status   interval   tat/limit
                                           s
                   ---------------------------------------------------------------------------
                   PDB1                    ACTIVE     UP         90         1/120
                   PDB2                    ACTIVE     UP         90         0/120
#
```

監視対象に戻した PDB 監視の状態が **"ACTIVE"** に変わったことを確認してください。

なお、監視対象から外した PDB 名に "all" を指定することによって、すべての PDB を監視対象に戻すことも可能です。

注意: ”-P” オプションを付与して監視対象から一時的に外した PDB を監視対象に戻すには、”-P” オプションを付与する必要があります。

注意: ”-P” オプションを付与して PDB を監視対象から一時的に外した状態で、apply リクエストを実行した場合は、PDB を監視対象に戻します。

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。

以上で、ApplicationMonitor の一時停止と再開の説明は終了です。

5.11. フェイルオーバグループ停止原因の確認

フェイルオーバグループが異常終了した場合、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。

以下の出力は、DB グループが停止した例です。

```
…
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.
…
```

"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。

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 2.9 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を採取します。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。

CLUSTERPRO X SSS によって出力されるメッセージの詳細は、CLUSTERPRO X SSS のマニュアルを参照してください。

以上で、フェイルオーバグループ停止原因の確認は終了です。

5.12. 疑似障害モードを利用した動作確認

ApplicationMonitor では実際の DB 障害を発生させずに、障害検出時の動作を確認できます。
oraadmin コマンドで疑似障害モードへの移行/復帰、発生させる障害の種類を指示します。
oraadmin コマンドによる指示後、障害を検出したことを示す syslog 出力をします。
oraadmin コマンドによる指示後、自身を停止しクラスタとの連携動作を確認できます。
oraadmin コマンドによる指示後、監視対象の停止や再起動は行いません。

5.12.1. 疑似障害モードへの移行

ApplicationMonitor を疑似障害モードへ移行するには、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c test -T ON
Connect to monitor-control(master).
#
# /opt/HA/AM/bin/oraadmin -c test -A
node1: test mode is ON.
node2: test mode is ON.
Succeeded in show of test mode.
#
```

疑似障害モードから復帰するには -T OFF を指定します。

5.12.2. インスタンス監視のテスト

node1 上のインスタンス監視で DOWN 障害を検出したときの疑似動作を指示する例です。

```
# /opt/HA/AM/bin/oraadmin -c test -m inst -V DOWN -O YES -n node1
Connect to monitor-control(master).
Succeeded in test node1.
#
```

syslog の出力例です。

```
instmond_21C[656870]: [0] run_check: result = [ICK=ERROR(5), weight=0], [LMHB=UP(1), weight=9], [LMON=UP(1), weight=9], [LMS=UP(1), weight=9], [LMD0=UP(1), weight=9], [LCK0=UP(1), weight=9], [ALERT=UP(1), weight=9], [LGWR=UP(1), weight=7], [CKPT=UP(1), weight=8], [ARCn=UP(1), weight=8]
oramond[656706]: 0:instmond[656870] sid1 down retry over.
oramond[656706]: 0:instmond[656870] sid1 status is DOWN.
```

5.12.3. PDB 監視のテスト

node1 上の PDB 監視で STALL 障害を検出したときの疑似動作を指示する例です。

```
# /opt/HA/AM/bin/oraadmin -c test -m pdb -P PDB1 -V STALL -O YES
-n node1
Connect to monitor-control(master).
Succeeded in test node1.
#
```

syslog の出力例です。

```
pdbmond_21C[659269]: [0] [PDB1] down retry over.
pdbmond_21C[659269]: [0] [PDB1] configured as NEVER.
```

5.12.4. リスナー監視のテスト

node1 上のリスナー監視で DOWN 障害を検出したときの疑似動作を指示する例です。

```
# /opt/HA/AM/bin/oraadmin -c test -m lsnr -s LISTENER -V DOWN -N 2
-n node1
Connect to monitor-control(master).
Succeeded in test node1.
#
```

syslog の出力例です。

```
oramond[671978]: 0:lsnrmond[685596] LISTENER down retry. (1/5)
                :
oramond[671978]: 0:lsnrmond[685596] LISTENER down retry. (2/5)
                :
lsnrmond[685596]: [0] [LISTENER] Restarting timer has been set to 60
sec
                :
oramond[671978]: 0:lsnrmond[685596] LISTENER status reconfirmed
as UP by retry operation.
```

5.12.5. 表領域監視のテスト

node1 上の表領域監視で 最大使用可能エクステントのしきい値を下回ったときの疑似動作を指示する例です。

```
# /opt/HA/AM/bin/oraadmin -c test -m tschk -i TSCHKMOND1 -R
USERS -B 5 -n node1
Connect to monitor-control(master).
Succeeded in test node1.
#
```

syslog の出力例です。

```
tschkmond_21C[672213]: [0] [TSCHKMOND1] am_tschk_sense:
Current MAX extent size is lower than configuration
limit value. Current MAX extent blocks = 5, tablespace_name is
USERS.
tschkmond_21C[672213]: [0] [TSCHKMOND1] am_tschk_sense:
Succeeded in executing script in MAX extent check.
```

5.12.6. ローカルディスク監視のテスト

node1 上のローカルディスク監視で 空容量のしきい値を下回ったときの疑似動作を指示する例です。

```
# /opt/HA/AM/bin/oraadmin -c test -m fds -K ASM -G DATA -W 5 -n
node1
Connect to monitor-control(master).
Succeeded in test node1.
#
```

syslog の出力例です。

```
fdsmond_21C[672216]: [0] Current free space is lower than
configuration limit value.
Current free space = 5%, ASM disk group is DATA..
```

5.12.7. ASM インスタンス監視のテスト

node1 上の ASM インスタンス監視で DOWN 障害を検出したときの疑似動作を指示する例です。

```
# /opt/HA/AM/bin/oraadmin -c test -m asm -V DOWN -n node1
Connect to monitor-control(master).
Succeeded in test node1.
#
```

syslog の出力例です。

```
asmmond_21C[683055]: result = [ICK=DOWN(6)], [LMHB=UP(1)], [ALERT=UP(1)]
asmmond_21C[683055]: Retry over, so going down. status is DOWN(6)
```

5.12.8. Oracle Clusterware 監視のテスト

node1 上の Oracle Clusterware 監視で DOWN 障害を検出したときの疑似動作を指示する例です。

```
# /opt/HA/AM/bin/oraadmin -c test -m crs -V DOWN -n node1
Connect to monitor-control(master).
Succeeded in test node1.
#
```

syslog の出力例です。

```
crsmond[682877]: Detected an obstacle(5). Current retry count ( 1 / 1 ).
```

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、疑似障害モードを利用した動作確認の説明は終了です。

6. コマンドリファレンス

oraadmin

名　　前　oraadmin　ー　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 -P <*PDB 名*> [-P <*PDB 名*>]. . .　[-C <*component_id*>] [-p <*port_number*>]
-c show -P all [-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 –r [-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。省略した場合は 0 の指定になります。
<*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。省略した場合は 25310 の指定になります。
<*PDB 名*> は、監視対象の PDB 名を指定します。

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)、 ASM 監視モニタ (asmmond)、ASM リスナー監視モニタ（alsnrmond）および SCAN リスナー監視モニタ（slsnrmond）のモニタ情報も表示します。
show -P <*PDB 名*>	指定した監視対象の PDB の情報を表示します。
show -P all	すべての監視対象の PDB の情報を表示します。
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 -r	監視対象の SCAN リスナーの情報を表示します。
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) によって管理し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、CRS 監視モニタ、ASM 監視モニタおよび SCAN リスナー監視モニタは含みません。

-c suspend [-C <*component_id* >] [-n <*node_name*>] [-n <*node_name*>]. . .
-c suspend -P <*PDB 名*> [-P <*PDB 名*>]. . . [-C <*component_id* >] [-n <*node_name*>] [-n <*node_name*>]. . .
-c suspend -P all [-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。省略した場合は 0 の指定になります。
<*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。省略した場合は全ノードの指定になります。
<*PDB 名*> は、監視対象の PDB 名を指定します。

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 -P <*PDB 名*>	全ノードまたは指定したノードの指定した PDB を監視対象から一時的に外します。
suspend -P all	全ノードまたは指定したノードのすべての PDB を監視対象から一時的に外します。
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。

-P オプションを指定して PDB を監視対象から一時的に外した状態で、apply リクエストを実行した場合は、PDB を監視対象に戻します。

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しているモニタへの要求は行えません。
SCAN リスナー監視モニタを除く単独起動しているモニタへの要求では、ノード名の指定が必要です。

-c suspend -m pdb -P <*PDB 名*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>] [-M]
-c suspend -m lsnr -s <*リスナー名*>|<*リスナー監視モニタ ID*> [-C <*component_id*>] -n <*node_name*> ¥
[-p <*port_number*>] [-M]
-c suspend -m tschk -i <*表領域監視モニタ ID*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>] [-M]
-c suspend -m fds [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>] [-M]
-c suspend -m stats -i <*統計情報採取モニタ ID*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>] [-M]
-c suspend -m crs -n <*node_name*> [-p <*port_number*>] [-M]
-c suspend -m asm -n <*node_name*> [-p <*port_number*>] [-M]
-c suspend -m alsnr -n <*node_name*> [-k <*ASM リスナー監視モニタ番号*>] [-p <*port_number*>] [-M]
-c suspend –m slsnr [-s <*network-number*>] [-p <*port_number*>] [-M]

<*PDB 名*> は、監視対象の PDB 名を指定します。
<*リスナー名*> は、監視対象のリスナー名を指定します。
<*リスナー監視モニタ ID*> は、リスナー監視モニタの識別子を指定します。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*node_name*> は、監視対象のノード名を指定します。省略した場合は全ノードの指定になります。
<*port_number*> は、構成情報管理サーバのポート番号を指定します。省略した場合は 25310 の指定になります。
<*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。
<*ASM リスナー監視モニタ番号*> は、ASM リスナー監視モニタの識別番号を指定します。省略した場合は 0 の指定になります。
<*network_number*> は、監視対象の SCAN リスナーが使用するネットワーク番号を指定します。省略した場合は 1 の指定になります。

suspend -m pdb	単独起動している PDB 監視モニタを一時停止します。
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。
suspend -m asm	ASM 監視モニタを一時停止します。
suspend -m alsnr	ASM リスナー監視モニタを一時停止します。
suspend -m slsnr	SCAN リスナー監視モニタを一時停止します。
suspend -M	単独起動モニタ自身の監視スレッドを一時停止します。

/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し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、CRS 監視モニタ、ASM 監視モニタおよび SCAN リスナー監視モニタは含みません。

-c resume [-C <*component_id* >] [-n <*node_name*>] [-n <*node_name*>]. . .
-c resume -P <*PDB 名*> [-P <*PDB 名*>]. . . [-C <*component_id* >] [-n <*node_name*>] [-n <*node_name*>]. . .
-c resume -P all [-C <*component_id* >] [-n <*node_name*>] [-n <*node_name*>]. . .

<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*node_name*> は、監視対象のノード名を指定します。省略した場合は全ノードの指定になります。
<*PDB 名*> は、監視対象の PDB 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。
resume -P <*PDB 名*>	全ノードまたは指定したノードの指定した PDB を監視対象に戻します。
resume -P all	全ノードまたは指定したノードのすべての PDB を監視対象に戻します。

-P オプションを指定して監視対象から一時的に外した PDB を監視対象に戻すには、-P オプションを指定する必要があります。-P オプションを指定せずにモニタを再開しても、監視対象から外した PDB を監視対象に戻しません。
また、モニタが一時停止中の場合は、-P オプションを指定しても、監視対象から外した PDB を監視対象に戻しません。モニタを再開した時に監視対象に戻します。

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しているモニタへの要求は行えません。
SCAN リスナー監視モニタを除く単独起動しているモニタへの要求では、ノード名の指定が必要です。
単独起動モニタの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。

-c resume -m pdb -P <*PDB 名*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]
-c resume -m lsnr -s <*リスナー名*>|<*リスナー監視モニタ ID*> [-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*>]
-c resume -m asm -n <*node_name*> [-p <*port_number*>]
-c resume -m alsnr -n <*node_name*> [-k <*ASM リスナー監視モニタ番号*>] [-p <*port_number*>]
-c resume -m slsnr [-s <*network-number*>] [-p <*port_number*>]

<*PDB 名*> は、監視対象の PDB 名を指定します。
<*リスナー名*> は、監視対象のリスナー名を指定します。
<*リスナー監視モニタ ID*> は、リスナー監視モニタの識別子を指定します。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*node_name*> は、監視対象のノード名を指定します。省略した場合は全ノードの指定になります。
<*port_number*> は、構成情報管理サーバのポート番号を指定します。省略した場合は 25310 の指定になります。
<*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。
<*ASM リスナー監視モニタ番号*> は、ASM リスナー監視モニタの識別番号を指定します。省略した場合は 0 の指定になります。
<*network_number*> は、監視対象の SCAN リスナーが使用するネットワーク番号を指定します。省略した場合は 1 の指定になります。

resume -m pdb	単独起動している PDB 監視モニタを再開します。
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。

resume -m asm	ASM 監視モニタを再開します。
resume -m alsnr	ASM リスナー監視モニタを再開します。
suspend -m slsnr	SCAN リスナー監視モニタを一時停止します。

/opt/HA/AM/bin/oraadmin -c stop

以下の stop リクエストは、モニタ制御デーモンによって管理し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、CRS 監視モニタおよび ASM 監視モニタは含みません。
また、ノード名の指定が必要です。

-c stop [-C <*component_id*>] [-p <*port_number*>] -n <*node_name*>

<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*port_number*> は、構成情報管理サーバのポート番号を指定します。省略した場合は 25310 の指定になります。
<*node_name*> は、監視対象のノード名を指定します。

stop　　指定したノードのモニタ制御デーモンを停止します。

stop リクエストを実行してもモニタ制御デーモンを停止できない場合は、モニタ制御デーモンに対して SIGTERM を送信してください。

以下の stop 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しているモニタへの要求は行えません。
また、ノード名の指定が必要です。

-c stop -m pdb -P <*PDB 名*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]
-c stop -m lsnr -s <*リスナー名*>|<*リスナー監視モニタID*> [-C <*component_id*>] -n <*node_name*> ¥
[-p <*port_number*>]
-c stop -m tschk -i <*表領域監視モニタ ID*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]
-c stop -m fds [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]
-c stop -m stats -i <*統計情報採取モニタ ID*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]
-c stop -m crs -n <*node_name*> [-p <*port_number*>]
-c stop -m asm -n <*node_name*> [-p <*port_number*>]

<*PDB 名*> は、監視対象の PDB 名を指定します。
<*リスナー名*> は、監視対象のリスナー名を指定します。
<*リスナー監視モニタID*> は、リスナー監視モニタの識別子を指定します。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*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。省略した場合は 25310 の指定になります。
<*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。

stop -m pdb	単独起動している PDB 監視モニタを停止します。
stop -m lsnr	単独起動しているリスナー監視モニタを停止します。
stop -m tschk	単独起動している表領域監視モニタを停止します。
stop -m fds	単独起動しているディスク領域監視モニタを停止します。
stop -m stats	単独起動している統計情報採取モニタを停止します。
stop -m crs	CRS 監視モニタを停止します。
stop -m asm	ASM 監視モニタを停止します。

stop リクエストを実行しても単独起動しているモニタを停止できない場合は、単独起動しているモニタに対して SIGTERM を送信してください。

/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し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、CRS 監視モニタ、ASM 監視モニタ、ASM リスナー監視モニタおよび SCAN リスナー監視モニタは含みません。
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みます。

-c apply -f <*configure_file_name*> [-p <*port_number*>] [-F] [-E]
-c apply -L [-C <component_id>]

<*configure_file_name*> は、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。省略した場合は 25310 の指定になります。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。

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、フェイルオーバ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が可能です。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。

-L オプションは、構成情報の配布、適用は行わず、機能制限の解除を行います。

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。
モニタ制御デーモンで管理しているモニタへの要求は行えません。
SCAN リスナー監視モニタを除く単独起動しているモニタへの要求では、ノード名の指定が必要です。

-c apply -m pdb -P <*PDB 名*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>] [-L]
-c apply -m lsnr -s <*リスナー名*>|<*リスナー監視モニタID*> [-C <*component_id*>] -n <*node_name*> ¥
[-p <*port_number*>] [-L]
-c apply -m tschk -i <*表領域監視モニタ ID*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>] [-L]
-c apply -m fds [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>] [-L]
-c apply -m stats -i <*統計情報採取モニタ ID*> [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>] [-L]
-c apply -m crs -n <*node_name*> [-p <*port_number*>] [-L]
-c apply -m asm -n <*node_name*> [-p <*port_number*>] [-L]
-c apply -m alsnr -n <*node_name*> [-k <*ASM リスナー監視モニタ番号*>] [-p <*port_number*>] [-L]
-c apply -m slsnr [-s <*network-number*>] [-p <*port_number*>] [-L]

<*PDB 名*> は、監視対象の PDB 名を指定します。
<*リスナー名*> は、監視対象のリスナー名を指定します。
<*リスナー監視モニタID*> は、リスナー監視モニタの識別子を指定します。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*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。省略した場合は 25310 の指定になります。
<*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。
<*ASM リスナー監視モニタ番号*> は、ASM リスナー監視モニタの識別番号を指定します。省略した場合は 0 の指定になります。
<*network_number*> は、監視対象の SCAN リスナーが使用するネットワーク番号を指定します。省略した場合は 1 の指定になります。

apply -m pdb　　単独起動している PDB 監視モニタに設定内容を適用します。

	-L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。
apply -m lsnr	単独起動しているリスナー監視モニタに設定内容を適用します。 -L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。
apply -m tschk	単独起動している表領域監視モニタに設定内容を適用します。 -L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。
apply -m fds	単独起動しているディスク領域監視モニタに設定内容を適用します。 -L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。
apply -m stats	単独起動している統計情報採取モニタに設定内容を適用します。 -L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。
apply -m crs	CRS 監視モニタに設定内容を適用します。 -L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。
apply -m asm	ASM 監視モニタに設定内容を適用します。 -L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。
apply -m alsnr	ASM リスナー監視モニタに設定内容を適用します。 -L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。
apply -m slsnr	SCAN リスナー監視モニタに設定内容を適用します。 -L オプションが指定された場合は、設定内容の適用は行わず、機能制限の解除を行います。

/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、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。省略した場合は 25310 の指定になります。

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が可能です。

/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。省略した場合は 25310 の指定になります。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。

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。省略した場合は 25310 の指定になります。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。

view　　現在の構成情報を表示します。

/opt/HA/AM/bin/oraadmin -c test

以下の test 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 test -T <ON|OFF> [-C <*component_id*>] [-p <*port_number*>]
-c test -A [-C <*component_id*>] [-p <*port_number*>]

<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*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。省略した場合は 25310 の指定になります。

test -T <*ON/OFF*>	全ノードの疑似障害モードへの移行(ON)/復帰(OFF)を指定します、
test -A	現在の疑似障害モードを表示します。

以下の test リクエストは、モニタ制御デーモン (oramond) によって管理し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 PDB 監視モニタ、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、CRS 監視モニタおよび ASM 監視モニタは含みません。

-c test -m inst -V <DOWN|STALL> [-N <*error_max*>] [-O <*YES*|*NO*>] [-C <*component_id*>] [-n <*node_name*>]
-c test -m pdb -P <*PDB 名*> -V <DOWN|STALL> [-N <*error_max*>] [-O <*YES*|*NO*>] [-C <*component_id*>] [-n <*node_name*>]
-c test -m lsnr -s <*リスナー名*> -V <DOWN|STALL> [-N <*error_max*>] [-O <*YES*|*NO*>] [-C <*component_id*>] -n <*node_name*>
-c test -m tschk -i <*表領域監視モニタ ID*> -R <*表領域名*> [-P <*PDB 名*>] -V <DOWN|STALL> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*>
-c test -m tschk -i <*表領域監視モニタ ID* > -R <*表領域名*> [-P <*PDB 名*>] -B <*block_number*> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*>
-c test -m tschk -i <*表領域監視モニタ ID* > -R <*表領域名*> [-P < *PDB 名*>] -W <*free_space*> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*>
-c test -m fds -K <*種別*> [-G <*ASM ディスクグループ名*>] -V <DOWN|STALL> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*>
-c test -m fds -K <*種別*> [-G < *ASM ディスクグループ名*>] -W <*free_space*> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*>

<*error_max*> は、疑似障害回数を指定します。省略した場合は 1 の指定になります。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*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。省略した場合は 25310 の指定になります。
<*node_name*> は、監視対象のノード名を指定します。省略した場合は全ノードの指定になります。
<*PDB 名*> は、監視対象の PDB 名を指定します。
<*リスナー 名*> は、監視対象のリスナー名を指定します。
<*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。
<*表領域名*> は、監視対象の表領域名を指定します。
<*block_number*> は、最大使用可能エクステントブロック数を指定します。
<*free_space*> は、空き容量率を指定します。
<*種別*> は、監視対象のディスク種別を指定します。種別には HOME、ARCHIVE、TRACE、ALERT、CORE、AUDIT または ASM のいずれかを指定します。種別が ASM の場合は-G で ASM ディスクグループを指定します。
<*ASM ディスクグループ名*> は、監視対象の ASM ディスクグループ名を指定します。

test -m inst	インスタンス障害を発生させます。-V で障害の種類(DOWN/STALL)を指定します。-O で Oracle Clusterware による再起動の結果(YES:再起動成功/NO:失敗)を指定します。
test -m pdb	PDB 障害を発生させます。-V で障害の種類(DOWN/STALL)を指定します。-O で Oracle Clusterware による再起動の結果(YES/NO)を指定します。
test -m lsnr	リスナー障害を発生させます。-V で障害の種類(DOWN/STALL)を指定します。-O で Oracle Clusterware による再起動の結果(YES/NO)を指定します。
test -m tschk	表領域障害を発生させます。-W で表領域空き容量を指定します。-V で障害の種類(DOWN/STALL)を指定します。
test -m fds	ローカルディスク障害を発生させます。-W でローカルディスク空き容量を指定します。-V で障害の種類(DOWN/STALL)を指定します。

以下の test 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し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、ノード名の指定が必要です。

-c test -m pdb -P <*PDB 名*> -V <DOWN|STALL> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]

-c test -m lsnr -s <*リスナー名*> -V <DOWN|STALL> [-N <*error_max*>] [-O <*YES*|*NO*>] [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]

-c test -m tschk -i <*表領域監視モニタ ID*> -R <*表領域名*> [-P <*PDB 名*>] -V <DOWN|STALL> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]

-c test -m tschk -i <*表領域監視モニタ ID* > -R <*表領域名*> [-P <*PDB 名*>] -B <*block_number*> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]

-c test -m tschk -i <*表領域監視モニタ ID* > -R <*表領域名*> [-P < *PDB 名*>] -W <*free_space*> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]

-c test -m fds -K <*種別*> [-G <*ASM ディスクグループ名*>] -V <DOWN|STALL> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]

-c test -m fds -K <*種別*> [-G < *ASM ディスクグループ名*>] -W <*free_space*> [-N <*error_max*>] [-C <*component_id*>] -n <*node_name*> [-p <*port_number*>]

-c test -m asm -V <DOWN|STALL> [-N <*error_max*>] [-O <*YES*|*NO*>] -n <*node_name*> [-p <*port_number*>]

-c test -m crs -V <DOWN|STALL> [-N <*error_max*>] -n <*node_name*> [-p <*port_number*>]

<*error_max*> は、疑似障害回数を指定します。省略した場合は 1 の指定になります。
<*component_id*> は、構成番号を指定します。省略した場合は 0 の指定になります。
<*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。省略した場合は 25310 の指定になります。
<*node_name*> は、監視対象のノード名を指定します。
<*PDB 名*> は、監視対象の PDB 名を指定します。
<*リスナー 名*> は、監視対象のリスナー名を指定します。
<*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。
<*表領域名*> は、監視対象の表領域名を指定します。
<*block_number*> は、最大使用可能エクステントブロック数を指定します。
<*free_space*> は、空き容量率を指定します。
<*種別*> は、監視対象のディスク種別を指定します。種別には HOME、ARCHIVE、TRACE、ALERT、CORE、AUDIT または ASM のいずれかを指定します。種別が ASM の場合は-G で ASM ディスクグループを指定します。
<*ASM ディスクグループ名*> は、監視対象の ASM ディスクグループ名を指定します。

test -m pdb	PDB 障害を発生させます。-V で障害の種類(DOWN/STALL)を指定します。-O で Oracle Clusterware による再起動の結果(YES/NO)を指定します。
test -m lsnr	リスナー障害を発生させます。-V で障害の種類(DOWN/STALL)を指定します。-O で Oracle Clusterware による再起動の結果(YES/NO)を指定します。
test -m tschk	表領域障害を発生させます。-W で表領域空き容量を指定します。-V で障害の種類(DOWN/STALL)を指定します。
test -m fds	ローカルディスク障害を発生させます。またはローカルディスク空き容量を指定します。-V で障害の種類(DOWN/STALL)を指定します。
test -m asm	ASM インスタンス障害を発生させます。-V で障害の種類(DOWN/STALL)を指定します。-O で Oracle Clusterware による再起動の結果(YES/NO)を指定します。
test -m crs	Oracle Clusterware/Oracle Restart 障害を発生させます。-V で障害の種類(DOWN/STALL)を指定します。

test リクエストの対象となる監視機能はインスタン監視、PDB 監視、リスナー監視、表領域監視、ローカルディスク監視尾、ASM インスタンス監視、Oracle Clusterware/Oracle Restart 監視です。

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
        ASM             n/a             ACTIVE      UP          90          1/120        NO
        ASM LSNR        ASMLISTENER1    ACTIVE      UP          60          1/90         NO
        INSTANCE        n/a             ACTIVE      UP          90          1/120        YES
        LISTENER        LISTENER1       ACTIVE      UP          60          0/90         NO
        LISTENER        LISTENER3       ACTIVE      UP          60          1/90         YES
        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
        PLUGGABLE       PDB1            ACTIVE      UP          90          0/120        NO
        PLUGGABLE       n/a             ACTIVE      n/a         n/a         n/a          YES
        SCAN LSNR       NET1            n/a         n/a         n/a         n/a          NO
                        scan_listeners               m-status    t-status    Interval   tat/limit
                        -------------------------------------------------------------------------
                        LISTENER_SCAN1               ACTIVE      UP          60         1/90
                        LISTENER_SCAN2               ACTIVE      UP          60         1/90
                        LISTENER_SCAN3               UNKNOWN     n/a         n/a        n/a
NODE: node2 RESUME
        monitors        name            m-status    t-status    interval    tat/limit    control
        -----------------------------------------------------------------------------------------
        CRS             n/a             ACTIVE      UP          90          0/180        NO
        ASM             n/a             ACTIVE      UP          90          1/120        NO
        ASM LSNR        ASMLUSTENER1    ACTIVE      UP          60          1/90         NO
        INSTANCE        n/a             ACTIVE      UP          90          1/120        YES
        LISTENER        LISTENER2       SUSPEND     UP          60          1/90         NO
        LISTENER        LISTENER4       ACTIVE      UP          60          0/90         YES
        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
        PLUGGABLE       PDB2            ACTIVE      UP          90          0/120        NO
        PLUGGABLE       n/a             ACTIVE      n/a         n/a         n/a          YES
        SCAN LSNR       NET1            n/a         n/a         n/a         n/a          NO
                        scan_listeners               m-status    t-status    Interval   tat/limit
                        -------------------------------------------------------------------------
                        LISTENER_SCAN1               UNKNOWN     n/a         n/a        n/a
                        LISTENER_SCAN2               UNKNOWN     n/a         n/a        n/a
                        LISTENER_SCAN3               ACTIVE      UP          60         1/90
```

ノード単位に状態を出力します。

上記の例では、node1 および node2 がノード名です。

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を出力します。

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を出力します。
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 監視モニタ
	ASM	ASM 監視モニタ
	ASM LSNR	ASM リスナー監視モニタ
	INSTANCE	インスタンス監視モニタ
	LISTENER	リスナー監視モニタ
	TABLESPACE	表領域監視モニタ
	DISKSPACE	ディスク領域監視モニタ
	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
	PLUGGABLE	PDB 監視モニタ
	SCAN LSNR	SCAN リスナー監視モニタ
name	リスナー監視モニタ、ASM 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、ASM リスナー名、表領域名および採取対象の統計情報名を出力します。 SCAN リスナー監視モニタの場合、監視対象の SCAN リスナーが使用しているネットワーク番号を出力します。 単独起動の PDB 監視モニタの場合、監視対象の PDB 名を出力します。 上記以外は、"n/a" を出力します。	
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 PDB 監視モニタ(oramond により制御される)、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を出力します。 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)、PDB 監視モニタ(oramond により制御される)、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" を出力します。	
tat	Oracle へのアクセスが完了するまでにかかった時間の最大値を出力します。 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)、PDB 監視モニタ(oramond により制御される)、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" を出力します。	
limit	監視対象を無応答と判断するまでの時間を出力します。 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)、PDB 監視モニタ(oramond により制御される)、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" を出力します。	

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
scan_listeners	SCAN リスナー監視モニタの場合、監視対象の SCAN リスナー名を出力します。	

ファイル　/opt/HA/AM/bin/oraadmin

注　　意　管理コマンド（show 以外）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またはスイッチオーバを実施する場合は、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。フェイルオーバまたはスイッチオーバが完了後、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　ー　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 または /etc/systemd/system/haam_oraconfd.service 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　ー　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)、PDB 監視モニタ (pdbmond)、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。

また、スタンバイ・インスタンス監視を行う場合は、Oracle Data Guard Broker プロセスの状態を監視します。
障害検出時は、syslog にエラーメッセージを出力し、自身を停止することで、CLUSTERPRO へ障害を通知します。

-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。

pdbmond

名　　前　pdbmond_21C
　　ー　PDB 監視モニタ

形　　式　**/etc/opt/HA/AM/lbin/pdbmond_21C -P *PDB 名* [-C *構成番号*]**

機能説明　PDB 監視モニタは、PDB 障害を監視するためのデーモンです。
障害検出時は、syslog にエラーメッセージを出力し、自身を停止することで、モニタ制御デーモン (oramond) または CLUSTERPRO へ障害を通知します。

-P	設定ファイル (oramond.conf) に記述した PluggableDatabase ステートメントの PDB 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
```
pdbmond_21C -P PDB1
pdbmond_21C -P PDB1 -C 1
```

ファイル　/etc/opt/HA/AM/lbin/pdbmond_21C

注　　意　PDB 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。
PDB 監視モニタを単独起動する場合は、設定ファイルの PluggableDatabaseMonitor ステートメントまたは PluggableDatabase 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。
PDB 監視モニタを単独起動する場合は、Oracle 21c 環境である必要があります。
PDB 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

lsnrmond

名　　前　lsnrmond　ー　リスナー監視モニタ

形　　式　**/etc/opt/HA/AM/lbin/lsnrmond -s** ***リスナー名*** **|*****リスナー監視モニタ*****ID [-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。
障害検出時は、syslog にエラーメッセージを出力し、自身を停止することで、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またはリスナー監視モニタの識別子を指定します。
-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。

tschkmond

名　　前　tschkmond_19C, tschkmond_21C
　　ー　表領域監視モニタ

形　　式　**/etc/opt/HA/AM/lbin/tschkmond_19C -i *表領域監視モニタ ID* [-C *構成番号*]**
　　/etc/opt/HA/AM/lbin/tschkmond_21C -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_19C -i TSCHKMOND
tschkmond_19C -i TSCHKMOND -C 1
tschkmond_21C -i TSCHKMOND
tschkmond_21C -i TSCHKMOND -C 1
```

ファイル　/etc/opt/HA/AM/lbin/tschkmond_19C
　　/etc/opt/HA/AM/lbin/tschkmond_21C

注　　意　表領域監視モニタは、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。
　　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。
　　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

fdsmond

名　　前　fdsmond_19C, fdsmond_21C
　　－ ディスク領域監視モニタ

形　　式　**/etc/opt/HA/AM/lbin/fdsmond_19C [-C *構成番号*]**
　　/etc/opt/HA/AM/lbin/fdsmond_21C [-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_19C
　　fdsmond_19C -C 1
　　fdsmond_21C
　　fdsmond_21C -C 1

ファイル　/etc/opt/HA/AM/lbin/fdsmond_19C
　　/etc/opt/HA/AM/lbin/fdsmond_21C

注　　意　ディスク領域監視モニタは、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。
　　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。
　　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

statsmond

名　　前　statsmond_19C, statsmond_21C
　　ー　統計情報採取モニタ

形　　式　**/etc/opt/HA/AM/lbin/statsmond_19C -i *統計情報採取モニタ ID* [-C *構成番号*]**
　　/etc/opt/HA/AM/lbin/statsmond_21C -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_19C -i STATSMOND
statsmond_19C -i STATSMOND -C 1
statsmond_21C -i STATSMOND
statsmond_21C -i STATSMOND -C 1
```

ファイル　/etc/opt/HA/AM/lbin/statsmond_19C
　　/etc/opt/HA/AM/lbin/statsmond_21C

注　　意　統計情報採取モニタは、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。
　　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。
　　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

crsmond

名　　前　crsmond － CRS 監視モニタ

形　　式　**/etc/opt/HA/AM/lbin/crsmond**

機能説明　CRS 監視モニタは、Oracle Clusterware／Oracle Restart のデーモンを監視するためのデーモンです。
障害検出時にエラーメッセージを出力し、自身を停止することで、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/lbin/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。

start_asmmond.sh

名　　前　start_asmmond.sh　ー　ASM 監視モニタ起動スクリプト

形　　式　**/etc/opt/HA/AM/lbin/start_asmmond.sh**

機能説明　ASM 監視モニタ起動スクリプトは、ASM インスタンス障害を監視する ASM 監視モニタを起動するためのシェルスクリプトです。
障害検出時にエラーメッセージを出力し、自身を停止することで、CLUSTERPRO へ障害を通知します。

使用例　コマンド
start_asmmond.sh

ファイル　/etc/opt/HA/AM/lbin/start_asmmond.sh

注　　意　ASM 監視モニタ (asmmond) 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、ASM 監視設定ファイル (asmmond.conf) を設定する必要があります。
ASM 監視モニタは、ASM 監視設定ファイル名が asmmond.conf 以外の名称では動作しません。
ASM 監視設定ファイルには、ORACLE_SID パラメータ、ORACLE_BASE パラメータ、ORACLE_HOME パラメータ、ORA_NLS パラメータおよび SHLIB_PATH パラメータを設定する必要があります。
同一ノード上で複数の ASM 監視モニタを起動することはできません。

ASM 監視モニタを起動するには、ASM 監視モニタ起動スクリプトを実行する必要があります。
ASM 監視モニタ起動スクリプトは、root ユーザーのみ実行できます。
Oracle が提供する asmcmd コマンドを root ユーザーで実行できない場合は、GRID_USER パラメータを設定する必要があります。
ASM 監視モニタ起動スクリプトには、ORACLE_VERSION パラメータおよび次の環境変数を設定する必要があります。

ORACLE_SID	ASM インスタンスの識別子
ORACLE_BASE	Oracle Grid Infrastructure をインストールしたディレクトリ構成の基本となるディレクトリ
ORACLE_HOME	Oracle Grid Infrastructure をインストールしたディレクトリ
LD_LIBRARY_PATH	Oracle Grid Infrastructure の 64bit 共有ライブラリパス
ORA_NLS10	Oracle Grid Infrastructure の言語、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	ASM 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

start_alone_monitor.sh

名　　前　start_alone_monitor.sh　ー　単独起動モニタ用起動スクリプト

形　　式　**/etc/opt/HA/AM/lbin/start_alone_monitor.sh ¥**
{ tschk <monitor-id> | fds | stats <monitor-id> | pdb <pdb-name>}
/etc/opt/HA/AM/lbin/start_alone_monitor<構成番号>**.sh ¥**
{ tschk <monitor-id> | fds | stats <monitor-id> | pdb <pdb-name>}

機能説明　単独起動モニタ用起動スクリプトは、単独起動モニタを起動するためのシェルスクリプトです。

使用例　コマンド

```
start_alone_monitor.sh tschk TSM1
start_alone_monitor.sh fds
start_alone_monitor.sh stats STS1
start_alone_monitor.sh pdb PDB1
start_alone_monitor1.sh tschk TSM1
start_alone_monitor1.sh fds
start_alone_monitor1.sh stats STS1
start_alone_monitor1.sh pdb PDB1
```

ファイル　/etc/opt/HA/AM/lbin/start_alone_monitor.sh

注　　意　単独起動モニタ用起動スクリプトは、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。
複数の構成番号の構成で単独起動モニタを使用する場合には、それぞれの構成ごとの起動スクリプトを用意する必要があります。
その場合は、起動スクリプトをコピーし、ファイル名を start_alone_monitor<構成番号>.sh としてください。
PDB 監視モニタを起動する場合は、Oracle Clusterware のリソースとして登録してリソースを起動します。
start_alone_monitor.sh を直接実行して起動することはできません。

単独起動モニタ用起動スクリプトには、ORACLE_VERSION パラメータおよび次の環境変数を設定する必要があります。

ORACLE_SID	Oracle のインスタンス識別子
ORACLE_BASE	Oracle の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
LD_LIBRARY_PATH	db ユーザー環境の 64bit 共有ライブラリのパス
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	単独起動モニタが 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

索　引

A

alsnr.conf 36
alsnrmond 31
alsnrmond.conf 36, 167
altmond 28
ApplicationMonitor 停止 10
ASM インスタンス 1
ASM インスタンス監視時の動作 262
ASM インスタンス強制停止 164
ASM インスタンス障害時の CLUSTERPRO への障害通知 164
ASM インスタンス障害の監視 9
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 299
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 10, 139
ASM 監視設定ファイル 35, 160, 315
ASM 監視モニタ 30
ASM 監視モニタ起動スクリプト . 37, 165, 316, 369
ASM リスナー監視時の動作 272
ASM リスナー監視設定ファイル 36, 167
ASM リスナー監視モニタ 31
ASM リスナー監視モニタ起動スクリプト 37
ASM リスナー障害の監視 9
asm.conf 36
asmmond 30
asmmond.conf 35, 160, 315

C

CDB 23
CLUSTERPRO 1
CLUSTERPRO の連携 320
CLUSTERPRO へ通知する障害種別 13
CRS 監視設定ファイル 35, 153, 314
CRS 監視モニタ 29, 368
crsd 157
crsmond 29, 368
crsmond.conf 35, 153, 314

D

DataGuardSite ステートメント 71
DataGuardSite ステートメントのパラメータ 101
Destination ステートメント 85
Destination ステートメントのパラメータ 127
Directory ステートメント 83
Directory ステートメントのパラメータ 125
DiskSpaceMonitor ステートメント 76
DiskSpaceMonitor ステートメントのパラメータ . 113

E

evmd 157
EXEC リソースの追加 322

F

fds.conf 36
fdsmond 27
fdsmond_19C 366

I

inst.conf 36
InstanceMonitor ステートメント 73
InstanceMonitor ステートメントのパラメータ 105
instmond 23

L

ListenerMonitor ステートメント 74
ListenerMonitor ステートメントのパラメータ 107
lsnr.conf 36
lsnrmond 25, 364

N

Node ステートメント 72
Node ステートメントのパラメータ 102

O

ocssd 157
OdgBroker ステートメント 87
OdgBroker ステートメントのパラメータ 129
ohasd 157
oraadmin 34, 347
Oracle Clusterware 監視時の動作 248
Oracle Clusterware 障害の監視 8
Oracle Clusterware によるインスタンス／リスナー／ASM インスタンス再起動の待ち合わせ 9
Oracle Clusterware／Oracle Restart 障害時の CLUSTERPRO への障害通知 158
Oracle Clusterware／Oracle Restart のデーモンの状態表示用コマンド 157
Oracle Clusterware／Oracle Restart リトライ判定シェルスクリプトの動作 283
Oracle Clusterwareへのリソース登録 181
Oracle Data Guard Broker による自動フェイルオーバの待ち合わせ 10
Oracle Data Guard Broker による自動フェイルオーバ時の監視動作 297
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 8
Oracle DBA 監査 (audit) 機能 134
Oracle Restart 監視時の動作 248
Oracle Restart 障害の監視 8
Oracle Restart によるインスタンス／リスナー／ASM インスタンス再起動の待ち合わせ 9
Oracle 環境変数 166, 175
Oracle 構成プロセス停止 133
Oracle のシステムステート・ダンプ採取時の動作 280
Oracle パスワード 134
Oracle ハング検知機能 5, 136, 307
oraconfd 35, 361
oradiag 149
oramond 21, 362
oramond.conf 35, 58, 308
oramond<構成番号>.bin 35

P

PDB 1
PDB 監視時の動作 208
PDB 監視の設定 142
PDB 監視モニタ 24
PDB 監視モニタの単独起動 150
PDB クローズ検出時の動作 144
PDB 再オープンシェルスクリプトの動作 291
PDB 障害検出時の動作 143
PDB 障害の監視 5
PDB 名 80
pdbmond 24
pdbmond_21C 363
PDB監視モニタ 363
PID モニタリソースの作成 324
PluggableDatabase ステートメント 80
PluggableDatabase ステートメントのパラメータ 121
PluggableDatabaseMonitor ステートメント 70
PluggableDatabaseMonitor ステートメントのパラメータ 99

R

RAC 構成 2
RAC 固有プロセスのストール監視機能 5
REDO ログ適用状況によるスタンバイ・インスタンスの監視 7
REDO ログ転送状況によるスタンバイ・インスタンスの監視 7
RedoApply ステートメント 86
RedoApply ステートメントのパラメータ 128

S

SCAN リスナー監視設定ファイル 36
SCAN リスナー監視モニタ..... 32
SCANリスナー監視時の動作 275
SCANリスナー監視設定ファイル 170
SCANリスナー障害の監視..... 9
SGA 情報 10
slsnr.conf..... 36
slsnrmond 32
slsnrmond.conf 36, 170
ssdump 33
ssdump.conf 36
StandbyDatabaseMonitor ステートメント..... 77
StandbyDatabaseMonitor ステートメントのパラメータ 115
start_alone_monitor.sh 37, 174, 370
start_alsnrmond.sh 37
start_asmmond.sh 37, 165, 316, 369
Stats ステートメント 88
Stats ステートメントのパラメータ 130
stats.conf 36
statsmond 33
statsmond_19C 367
StatsMonitor ステートメント 78
StatsMonitor ステートメントのパラメータ 117
stbdb.conf 36
stbdbmond 28
SystemStateDaemon ステートメント 79
SystemStateDaemon ステートメントのパラメータ 119

T

Table ステートメント 81
Table ステートメントのパラメータ 123
TableSpace ステートメント..... 82
TableSpace ステートメントのパラメータ..... 124
TableSpaceMonitor ステートメント..... 75
TableSpaceMonitor ステートメントのパラメータ111
Transport ステートメント 84
Transport ステートメントのパラメータ..... 126
tschk.conf 36
tschkmond 26
tschkmond_19C 365

あ

アラートログ監視モニタ 28

い

一時停止 11, 334
一括設定 11
インスタンス監視時の動作 197
インスタンス監視表 40
インスタンス監視表のスキーマ 143
インスタンス監視方法..... 140
インスタンス監視モニタ 23
インスタンス障害時の CLUSTERPRO への障害通知 144
インスタンス障害の監視.....5
インスタンスの繰り返し障害 145

う

運用 303
運用統合製品との連携 12

か

概要 1
監視 TAT 情報 10
監視応答時間..... 10
監視対象デーモン..... 157
監視タイムアウトの時間帯での自動変更 ... 12, 142
監視表の一括作成..... 55
監視プロセス 21
監視モニタ制御..... 135
管理インタフェース 34
管理コマンド 34, 347

き

疑似障害モード 14, 343
起動確認 326
起動用スクリプト..... 37
機能制限 301
共通ユーザー..... 47

け

現用待機構成 1

こ

構成情報管理サーバ 35, 361
構成情報ファイル 35
コメント..... 67, 154, 161, 168, 171

さ

再開..... 11, 334, 337
採取プロセス..... 33
最大監視応答時間の表示 12
サイト間通信 ID..... 71, 85

し

シェルスクリプト..... 9
シェルスクリプトの実行時の動作 282
しきい値設定..... 148
システム・グローバル領域 10
システムステート・ダンプ..... 9
システムステートダンプ採取デーモン 33
事前準備 305
自動フェイルオーバ..... 10
種別..... 83
障害検出時のクラスタ全体での統合的判断の設定
..... 159
障害デーモン再起動の待ち合わせ動作..... 158
障害ノード上の ApplicationMonitor の停止動作
..... 133
障害ノード上の CRS監視モニタの停止動作の設定
..... 159
状態の表示..... 12
冗長化リスナーの状態監視時の動作 230
冗長化リスナーの接続監視時の動作 226
情報採取スクリプト..... 149
除外 340

す

スキーマ 50
スタンバイ・インスタンス監視時の動作..... 238
スタンバイ・インスタンス監視の設定 132
スタンバイデータベース監視モニタ 28
ステートメント 68
ステートメントの階層 89

せ

設定 39, 308
設定確認 317
設定の表示 12
設定配布 319
設定ファイル 35, 58, 308
設定ファイルのテンプレート作成..... 12
全ノードステートメント 69
全ノードステートメントのパラメータ..... 93

そ

ソフトウェア構成..... 15

た

待機イベント 5, 136
単一リスナー監視時の動作..... 214
単独監視 13
単独起動モニタ用起動スクリプト 37, 174, 370
単独採取 13

て

ディザスタ・リカバリ構成..... 13
停止ノード選択ポリシー..... 133
ディスク領域監視モニタ..... 27, 366
ディスク領域監視モニタの単独起動..... 151
データベース領域の容量監視時の動作..... 236

データベース領域の容量不足の監視..... 6

と

統計情報採取時の動作..... 279
統計情報採取モニタ..... 33, 367
統計情報採取モニタ ID 78
統計情報採取モニタの単独起動..... 152
統計情報種別 88
統計情報の採取 10, 137, 173
動作..... 195
動作中の状態確認..... 327
動作中の設定パラメータの調整..... 329
動作中の設定パラメータの変更..... 331
特長..... 3

ね

ネット・サービス名..... 110

の

ノード名..... 132

は

パスワード・ファイル認証 135, 164
パラメータ ..67, 93, 154, 155, 161, 162, 166, 168, 169, 171, 175

ひ

表名..... 81
表領域監視モニタ 26, 365
表領域監視モニタ ID..... 74, 75
表領域監視モニタの単独起動 151
表領域名 82

ふ

ファスト・スタート・フェイルオーバ 10
フィルタリング設定 13, 176
フィルタリング設定ファイル 36, 177
フィルタリングのデフォルト設定 179
フェイルオーバグループ停止原因の確認..... 342
複数インスタンス監視..... 13

も

モニタ制御デーモン..... 21, 362

ゆ

ユーザー指定表.....5

ら

ライセンスチェック 301

り

リスナー監視方式の設定 147
リスナー監視モニタ 25, 364
リスナー監視モニタの単独起動 150
リスナー再起動シェルスクリプトの動作 292
リスナー再起動方法..... 146
リスナー障害時の CLUSTERPRO への障害通知 144
リスナー障害の監視.....5
リスナーの繰り返し障害 146
リスナー名 74, 110

ろ

ローカル・ユーザー 47
ローカルディスク領域の容量監視時の動作 237
ローカルディスク領域の容量不足の監視 6
ログレベル 131

CLUSTERPRO
MC ApplicationMonitor 2.9 for Linux
ユーザーズガイド

2024 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）

P

© NEC Corporation 2012-2024

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