



CLUSTERPRO MC
ApplicationMonitor 1.0 for Linux
利用の手引き

日本電気株式会社

2013年4月

はしがき

本書は CLUSTERPRO MC ApplicationMonitor 1.0 for Linux の機能と操作方法について説明したものです。

本書の構成は次のとおりです。

| 章 | タイトル | 内容 |
|---|------------------------------|--|
| 1 | ApplicationMonitor とは | ApplicationMonitor の概要と特長 |
| 2 | ApplicationMonitor のソフトウェア構成 | 監視プロセス、管理インタフェース、設定ファイルの概要 |
| 3 | ApplicationMonitor の設定 | ApplicationMonitor の設定方法、関連製品(CLUSTERPRO, サーバ管理基盤, ProcessSaver, Oracle Clusterware、Oracle Restart)と一緒に使用する場合の設定 |
| 4 | ApplicationMonitor の動作 | 起動時 / 停止時の動作、インスタンス、リスナ、表領域、ディスク領域、OCW 監視時の動作 |
| 5 | ApplicationMonitor の運用 | 運用手順 |
| A | コマンドリファレンス | 管理コマンド、構成情報管理サーバ、モニタ制御デーモン、リスナ監視モニタ、表領域監視モニタ、ディスク領域監視モニタ、CRS 監視モニタのリファレンス |

2012年10月 初版

2013年 4月 第2版

備考

(1) 本書は、以下のオペレーティングシステムに対応しています。

- ・ Red Hat Enterprise Linux 5.7 , 5.8 , 6.1 , 6.2 , 6.3
- ・ Oracle Linux 6.1 , 6.2 , 6.3

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

(2) 本書で説明している全ての機能は、プログラムプロダクトであり、次の表のプロダクト型番及びプロダクト名に対応しています。

| プロダクト型番 | プロダクト名 | プロダクトリリース |
|------------|--|-----------|
| UL4437-102 | CLUSTERPRO MC ApplicationMonitor 1.0 for Linux (1CPU ライセンス) | 1.0 |
| UL4437-112 | CLUSTERPRO MC ApplicationMonitor 1.0 for Linux (1CPU ライセンス)(他社機版) | 1.0 |
| UL4437-122 | CLUSTERPRO MC ApplicationMonitor 1.0 for Linux VM (1 ノードライセンス) | 1.0 |
| UL4437-132 | CLUSTERPRO MC ApplicationMonitor 1.0 for Linux VM (1 ノードライセンス)(他社機版) | 1.0 |
| UL4437-101 | CLUSTERPRO MC ApplicationMonitor CD 1.0 | 1.0 |

(3) 本書は次の表現を使用します。

大かっこ [] 省略可能なパラメータを表します。

山かっこ <> 任意の文字列を指定することを表します。

(4)

・ Linux は、Linus Torvalds 氏の米国及びその他の国における、登録商標または商標です。

・ CLUSTERPRO は、日本電気株式会社の登録商標です。

・ ORACLE は、米国 Oracle Corporation の登録商標です。

その他、本書に登場する会社名及び商品名は各社の商標または登録商標です。

目 次

| | | |
|---------|---|----|
| 第 1 章 | ApplicationMonitor とは..... | 1 |
| 1.1. | ApplicationMonitor の概要..... | 1 |
| 1.2. | ApplicationMonitor の特長..... | 3 |
| 1.2.1. | インスタンス障害の監視と障害時アクション | 4 |
| 1.2.2. | リスナ障害の監視と障害時アクション | 4 |
| 1.2.3. | データベース容量不足の監視..... | 4 |
| 1.2.4. | ローカルディスク容量不足の監視 | 5 |
| 1.2.5. | Oracle Clusterware、Oracle Restart 無応答障害の監視..... | 5 |
| 1.2.6. | システムステートの自動採取とスクリプトの実行..... | 5 |
| 1.2.7. | 障害検出時の運用方針の設定..... | 5 |
| 1.2.8. | AM の一括設定・一括変更..... | 5 |
| 1.2.9. | 監視・採取の一時中断・再開..... | 6 |
| 1.2.10. | 各モニタ状態、各モニタ設定の表示..... | 6 |
| 1.2.11. | 統計情報の採取..... | 6 |
| 1.2.12. | 設定ファイルのテンプレート作成機能..... | 6 |
| 1.2.13. | 複数インスタンス監視..... | 6 |
| 1.2.14. | 監視モニタ、採取モニタの単独起動..... | 7 |
| 1.2.15. | Oracle エラーのフィルタリング設定..... | 7 |
| 1.2.16. | DR 構成のバックアップサイト監視..... | 7 |
| 1.2.17. | OCW によるインスタンス/リスナ再起動処理の待ち合わせ..... | 7 |
| 第 2 章 | ApplicationMonitor のソフトウェア構成..... | 9 |
| 2.1. | 監視プロセス..... | 11 |
| 2.1.1. | モニタ制御デーモン(oramond) | 11 |
| 2.1.2. | インスタンス監視モニタ(instmond)..... | 12 |
| 2.1.3. | リスナ監視モニタ(lsnrmond) | 12 |
| 2.1.4. | 表領域監視モニタ(tschkmond) | 13 |
| 2.1.5. | ディスク領域監視モニタ(fdsmond) | 13 |
| 2.1.6. | リコンフィグ監視モニタ(altmond)..... | 13 |
| 2.1.7. | CRS 監視モニタ (crsmond)..... | 14 |
| 2.2. | 採取プロセス..... | 15 |
| 2.2.1. | 統計情報採取モニタ(statsmond) | 15 |
| 2.2.2. | システムステートダンプ採取デーモン(ssdump)..... | 15 |
| 2.3. | 管理インタフェース..... | 16 |

| | | |
|---------|---|----|
| 2.3.1. | 管理コマンド(oraadmin)..... | 16 |
| 2.3.2. | 構成情報管理サーバ(oraconfd) | 16 |
| 2.3.3. | 設定ファイルテンプレート作成コマンド(amctf)..... | 16 |
| 2.4. | 設定ファイル..... | 17 |
| 2.4.1. | 設定ファイル(oramond.conf) | 17 |
| 2.4.2. | 設定ファイルテンプレート(oramond.tmp) | 17 |
| 2.4.3. | 構成情報ファイル(oramond<構成番号>.bin) | 17 |
| 2.4.4. | CRS 監視設定ファイル(crsmond.conf)..... | 17 |
| 2.4.5. | RENS 連携用辞書ファイル (haam.dict 、 haam_stat.dict 、 haam_stat_map.dict) | 17 |
| 第3章 | ApplicationMonitor の設定..... | 19 |
| 3.1. | インスタンス監視表の作成..... | 20 |
| 3.2. | 設定ファイル(oramond.conf)の作成と適用 | 24 |
| 3.2.1. | 設定ファイルの記述例..... | 25 |
| 3.2.2. | コメントの記述形式 | 29 |
| 3.2.3. | パラメータの記述形式..... | 29 |
| 3.2.4. | ステートメントの記述形式 | 29 |
| 3.2.5. | ステートメントの階層..... | 40 |
| 3.2.6. | パラメータとステートメントの記述順 | 41 |
| 3.2.7. | パラメーター一覧..... | 43 |
| 3.2.8. | ログレベルの設定..... | 64 |
| 3.2.9. | ノード名の設定..... | 64 |
| 3.2.10. | 停止ノード選択ポリシーの設定 (RAC 構成時のみ) | 65 |
| 3.2.11. | 障害ノードの停止動作の設定..... | 65 |
| 3.2.12. | Oracle 構成プロセス停止の設定 | 66 |
| 3.2.13. | 監視モニタ制御の設定..... | 66 |
| 3.2.14. | インスタンス / リスナ障害時のグループ停止の設定 | 66 |
| 3.2.15. | リスナ障害時の障害情報採取の設定..... | 67 |
| 3.2.16. | OracleDBA 監査(audit)機能対応の設定..... | 67 |
| 3.2.17. | Oracle パスワードの扱いについて | 68 |
| 3.2.18. | Oracle ハング検知機能を利用した監視について (RAC 環境のみ) | 68 |
| 3.2.19. | パスワード認証環境での監視..... | 68 |
| 3.2.20. | 安定したリスナ監視の実現 | 69 |
| 3.2.21. | 統計情報採取の設定 | 69 |
| 3.3. | Oracle エラーのフィルタリング設定..... | 71 |
| 3.3.1. | フィルタリング設定ファイルの記述形式 | 72 |

| | | |
|---------|---|-----|
| 3.3.2. | フィルタリングのデフォルト設定 | 73 |
| 3.4. | 情報採取スクリプト(oradiag)の設定..... | 74 |
| 3.5. | CLUSTERPRO の設定..... | 75 |
| 3.6. | リスナ監視モニタの単独起動..... | 77 |
| 3.7. | 表領域監視モニタの単独起動..... | 79 |
| 3.8. | ディスク領域監視モニタの単独起動..... | 79 |
| 3.9. | 統計情報採取モニタの単独起動 | 80 |
| 3.10. | ProcessSaver との併用 | 80 |
| 3.11. | CRS 監視設定ファイル(crsmond.conf)の作成 | 81 |
| 3.11.1. | CRS 監視設定ファイルの記述例..... | 81 |
| 3.11.2. | コメントの記述形式 | 82 |
| 3.11.3. | パラメータの記述形式..... | 82 |
| 3.11.4. | パラメーター一覧..... | 83 |
| 3.12. | OCW と ApplicationMonitor の監視対象設定について | 85 |
| 3.12.1. | OCW 監視の構築例 | 87 |
| 3.12.2. | CRS グループ | 88 |
| 3.12.3. | RAC グループ..... | 89 |
| 3.12.4. | LISTENER グループ | 89 |
| 第 4 章 | ApplicationMonitor の動作 | 91 |
| 4.1. | ApplicationMonitor の起動と停止..... | 91 |
| 4.2. | インスタンス監視時の動作..... | 93 |
| 4.3. | リスナ監視時の動作..... | 100 |
| 4.4. | 表領域監視時の動作..... | 105 |
| 4.5. | ディスク領域監視時の動作..... | 106 |
| 4.6. | 統計情報採取時の動作 | 107 |
| 4.7. | システムステートの採取 | 108 |
| 4.8. | スクリプトの実行 | 109 |
| 4.9. | OCW 監視時の動作..... | 110 |
| 第 5 章 | ApplicationMonitor の運用 | 115 |
| 5.1. | Oracle ハング検知機能の停止 (RAC 環境のみ) | 115 |
| 5.2. | ApplicationMonitor の設定..... | 116 |
| 5.3. | ApplicationMonitor の設定確認 | 118 |
| 5.4. | ApplicationMonitor の設定配布 | 119 |
| 5.5. | ApplicationMonitor の Oracle グループへの組み込み..... | 120 |
| 5.6. | Oracle グループの起動..... | 122 |
| 5.7. | 動作中の ApplicationMonitor のステータス確認 | 123 |

| | | |
|-------|---------------------------------------|-----|
| 5.8. | 監視間隔の調整..... | 125 |
| 5.9. | ApplicationMonitor 動作中の設定パラメータ変更..... | 126 |
| 5.10. | Oracle グループのメンテナンス..... | 128 |
| 5.11. | ApplicationMonitor による障害の検出..... | 132 |
| A. | コマンドリファレンス..... | 133 |

第1章 ApplicationMonitor とは

1.1. ApplicationMonitor の概要

現用待機構成の場合：

ApplicationMonitor (以下、AM と略す) は Oracle 11g (11.1.0、11.2.0) のインスタンス障害、リスナ障害、データベースの容量不足、ローカルディスク容量不足を監視します。

なお、Oracle11gR2 環境の Oracle Restart を監視することができます。監視機能は、AM 本体の設定とは別の設定が必要となります。監視の設定についての詳細は、「3.11 CRS 監視設定ファイル(crsmond.conf)の作成」を参照してください。

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することができます。さらに、サーバ管理基盤 (Resource Event Notification Service 以下、RENS と略す) と連携することで、これまで RENS で行われていたハードウェア監視機能などと統一されたインタフェースでのログ監視 (障害情報管理) やクラスタ連携が行えます。

RENS との連携機能は Express5800/A1000 シリーズのみの対応となります。

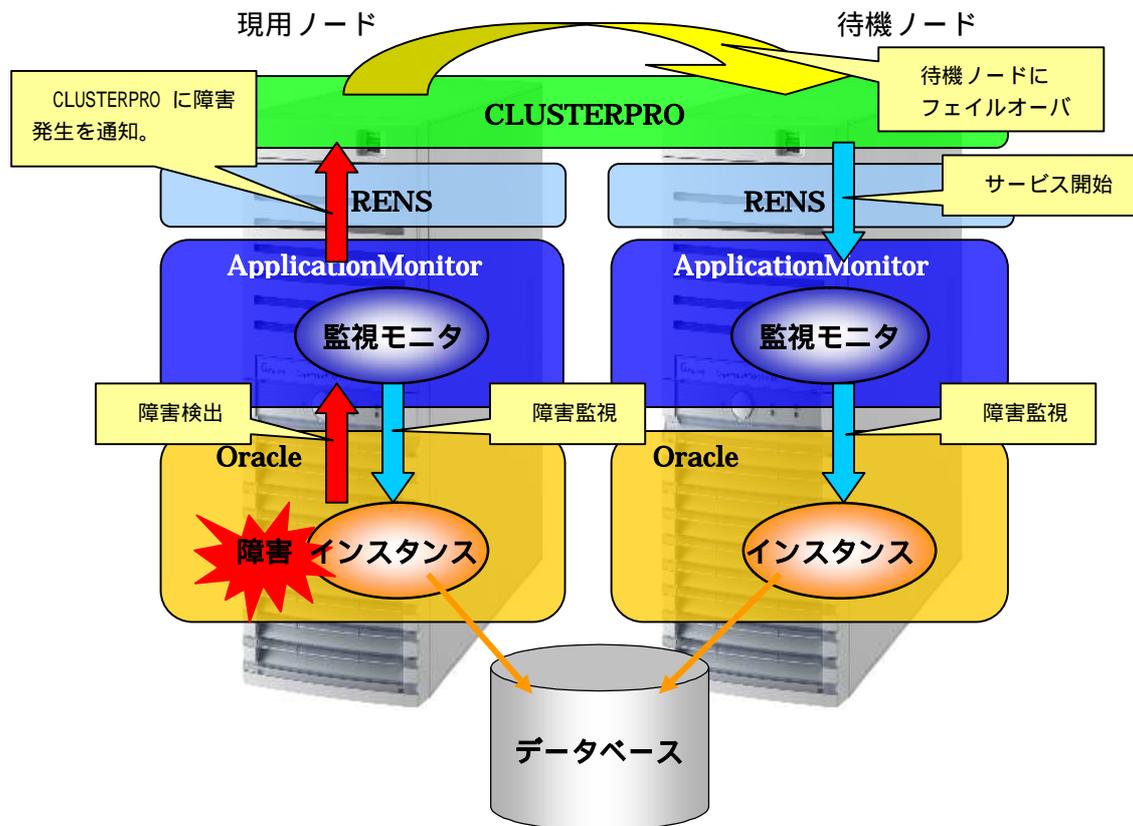


図 1-1 ApplicationMonitor の概要 (現用待機構成)

RAC 構成の場合：

AM は Oracle Real Application Clusters (11.1.0、 11.2.0)(以下、 RAC と略す) のインスタンス障害、リスナ障害、データベースの容量不足、ローカルディスク容量不足を監視します。また、Oracle 監視機能を統合しており、Oracle11g (11.1.0、 11.2.0) の監視も可能です。監視の設定についての詳細は「 5.2 ApplicationMonitor の設定」を参照してください。

なお、Oracle Clusterware の監視をすることができます。監視機能は、AM 本体の設定とは別の設定が必要となります。監視の設定についての詳細は、「 3.11 CRS 監視設定ファイル(crsmond.conf)の作成」を参照してください。

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することができます。さらに、RENS と連携することで、これまで RENS で行われていたハードウェア監視機能などと統一されたインタフェースでのログ監視 (障害情報管理) やクラスタ連携が行えます。

RENS との連携機能は Express5800/A1000 シリーズのみの対応となります。

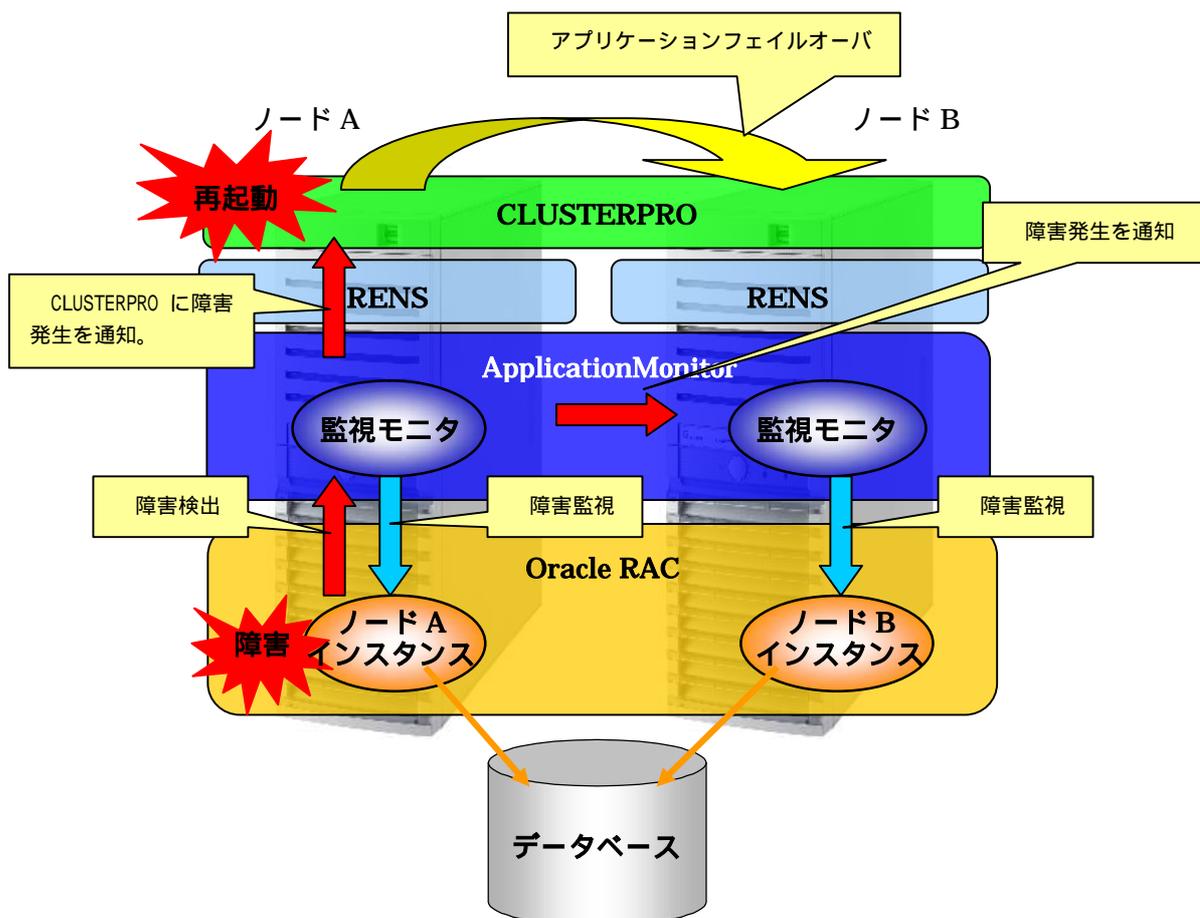


図 1-2 ApplicationMonitor の概要 (RAC 構成)

1.2. ApplicationMonitor の特長

AM は以下の特長・機能を持ちます。

【監視機能】

- ・ Oracle バックグラウンドプロセス単位でのインスタンス障害の監視
- ・ リスナ障害の監視
- ・ 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
- ・ Oracle Clusterware、Oracle Restart 無応答障害の監視

【障害時アクション】

- ・ 障害検出時、CLUSTERPRO へ障害を通知し、グループをフェイルオーバ
- ・ 障害検出時に Oracle / RAC のシステムステートを自動採取
- ・ 障害検出時に任意のスクリプトを実行
- ・ リスナ障害時にリスナを再起動
- ・ データベース容量不足時にコマンドを実行
- ・ ローカルディスク容量不足時にコマンドを実行
- ・ 障害検出時にインスタンス / リスナの再起動を待ち合わせることが可能
- ・ 障害検出時、DOWN / STALL 毎にフェイルオーバするかどうかを指定可能

【設定・運用・メンテナンス時の支援機能】

- ・ 障害検出時の運用方針の設定 (**RAC 構成時のみの機能**)
- ・ 監視対象ノード内のいずれかのノードで一括設定・一括変更が可能な設定機能
- ・ 監視の一時中断・再開
- ・ 監視状態、監視設定の表示
- ・ 最大監視応答時間 (ターンアラウンドタイム) の表示
- ・ 統計情報の採取
- ・ 設定ファイルテンプレート作成機能

【さまざまな運用形態に対応した構成】

- ・ 同一ノード上で複数のインスタンスを監視できる複数インスタンス監視
- ・ リスナ障害、データベース容量不足、ローカルディスク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
- ・ IPv6 対応

第 1 章 ApplicationMonitor とは

- ・ Oracle の DR 構成のバックアップサイトの監視に対応

それぞれの特長・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

AM はインスタンスに接続し、一定時間毎にデータベースの参照 / 更新などの処理を、Oracle バックグラウンドプロセス単位で行うことでインスタンス障害監視を行います。また、障害監視の対象となる表名を指定することで特定の表で発生するインスタンス障害を監視します。

障害検出時、指定された回数のリトライを試みますが、リトライを行っても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。この時、AM は自身を停止することで CLUSTERPRO へ障害を通知し、グループをフェイルオーバーします。また、プロセス停止時に RENS を経由しての CLUSTERPRO への障害通知も行います。

SERVICE_DOWN パラメータの設定により、対象ノード上の AM 自身(oramond)を"停止する"または"停止しない"の選択ができます。

1.2.2. リスナ障害の監視と障害時アクション

AM は指定されたリスナ名に対するネットサービスの有無を一定時間毎に監視することでリスナ障害を監視します。

障害検出時、指定された回数のリスナの再起動を試みますが、指定回数内にリスナが回復しない場合には最終的にリスナの障害と判断します。この時、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、グループをフェイルオーバーします。また、プロセス停止時に RENS を経由しての CLUSTERPRO への障害通知も行いません。

障害検出時のリスナの再起動は設定により AM で行わずに、Oracle Clusterware、Oracle Restart による再起動を待ち合わせることも可能です。

SERVICE_DOWN パラメータの設定により、対象ノード上の AM 自身(oramond)を"停止する"または"停止しない"の選択ができます。

1.2.3. データベース容量不足の監視

AM は一定時間毎に表領域の最大使用可能エクステンツサイズと空き容量率を監視します。指定された閾値を下回った場合、AM の設定ファイルに指定されたコマンドを実行します。

1.2.4. ローカルディスク容量不足の監視

AM は一定時間毎に Oracle アーカイブログ等ローカルディスクの空き容量率を監視します。指定された閾値を下回った場合、AM の設定ファイルに指定されたコマンドを実行します。

1.2.5. Oracle Clusterware、Oracle Restart 無応答障害の監視

AM は一定時間毎に指定されたコマンドにより、Oracle Clusterware または Oracle Restart (以下、OCW と略す) の状態を監視します。

障害検出時、指定された回数のリトライを試みますが、指定回数内に状態が回復しない場合、指定されたコマンドを実行します。また、プロセス停止時に RENS を経由しての CLUSTERPRO への障害通知も行います。

SERVICE_DOWN パラメータの設定により、対象ノード上の監視モジュール自身(crsmond)を"停止する"または"停止しない"の選択ができます。

1.2.6. システムステートの自動採取とスクリプトの実行

AM は障害検出時 (CLUSTERPRO へ障害を通知する前) Oracle の障害解析に有益なシステムステートを自動で採取します。これにより、早急な問題解決に対応できます。また、AM の設定ファイルにスクリプトが指定されている場合、スクリプトの実行を行います (AM には障害解析に有益な情報採取スクリプトが標準で添付されています)。

1.2.7. 障害検出時の運用方針の設定

AM では、障害が発生したノード上の AM 自身に対して、"常に停止"、"停止しない"、"生存ノード数が1の時、停止しない"の3種類の設定ができ、障害検出時の運用方針を選択できます。また、一定時間内における複数ノード障害時に、"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。

1.2.8. AM の一括設定・一括変更

AM は現用ノードと待機ノードのいずれかのノードで1つの設定ファイルを作成することで、一括設定もしくは一括変更が可能です。

注意： OCW監視を行う場合は、監視するそれぞれのノードでCRS監視設定ファイルの設定を行う必要があります。

第 1 章 ApplicationMonitor とは

1.2.9. 監視・採取の一時中断・再開

AM のサスペンド機能を用いることにより、グループ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）が可能です。また、メンテナンス後、レジューム機能により監視及び採取を再開することができます。

注意： AMのモニタは、起動時にOracleのライブラリをリンクしており、suspend 機能にて一時停止を行った場合、Oracle DBに対するアクセスは停止しますが、モニタプロセス自体は動作しています。このため、Oracleパッチ適用の際は、AMの停止も必要になります。

1.2.10. 各モニタ状態、各モニタ設定の表示

AM の管理コマンド(oraadmin)により、各監視モニタ及び採取モニタの状態や現在の設定値を表示することで AM の状態を把握できます。また、各監視モニタの最大監視応答時間の表示により、POLL_TIMEOUT パラメータ（AM 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、障害検出からグループをフェイルオーバーさせるまでの時間短縮につながります。

1.2.11. 統計情報の採取

AM は一定時間毎にインスタンス監視及びリスナ監視の監視応答時間(ターンアラウンドタイム 以下、監視 TAT と略す)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（以下、SGA と略す）の情報を採取します。

GET_STATS パラメータの設定により、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。

1.2.12. 設定ファイルのテンプレート作成機能

Oracle の環境変数を自動で収集し、AM 本体の設定ファイルのテンプレートを作成することができます。

本機能を用いることで、簡易設定ファイルが容易に作成可能です。

1.2.13. 複数インスタンス監視

AM 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。それぞれのインスタンスのグループ（現用・待機もしくは 1 クラスタ構成で1つのグループ）毎に一意的構成番号（COMPONENT_ID パラメータ）を割り振り、AM をそれぞれのグループのリソースとして登録することで実現します。

1.2.14. 監視モニタ、採取モニタの単独起動

リスナ監視モニタ(lsnrmond)、表領域監視モニタ(tschkmond)、ディスク領域監視モニタ(fdsmond)及び統計情報採取モニタ(statsmond)は、MONITOR_CONTROL パラメータを NO に設定することで、それぞれ単独で起動を行うことができます。これにより、インスタンス監視とは別のグループのリソースとして登録することができ、より柔軟な構成が可能となります。

1.2.15. Oracle エラーのフィルタリング設定

AM の監視中に発生する Oracle エラー毎に、グループの切り替えを行う / 行わないをユーザの運用形態に合わせて設定が可能です。

1.2.16. DR 構成のバックアップサイト監視

AM は、Oracle Data Guard によるディザスタリカバリ構成に対応しています。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を行うことが可能です。

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。

1.2.17. OCW によるインスタンス / リスナ再起動処理の待ち合わせ

AM では、障害検出後の動作として Oracle の環境により OCW によるインスタンス / リスナ再起動処理の状況を待ち合わせるかどうかの選択が可能です。

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

第 1 章 ApplicationMonitor とは

第2章 ApplicationMonitor のソフトウェア構成

AM は以下の監視プロセス、管理インタフェース、設定ファイルから構成されます。

【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
- ・ リスナ監視モニタ(lsnrmond)
- ・ 表領域監視モニタ(tschkmond) 1
- ・ ディスク領域監視モニタ(fdsmond) 1
- ・ リコンフィグ監視モニタ(almond) 1
- ・ CRS 監視モニタ(crsmond) 2

【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)
- ・ 設定ファイルテンプレート作成コマンド(amctf)

【設定ファイル】

- ・ 設定ファイル(oramond.conf)
- ・ 設定ファイルテンプレート(oramond.tmp)
- ・ 構成情報ファイル(oramond<構成番号>.bin) 3
- ・ CRS 監視設定ファイル(crsmond.conf) 2

- 1 監視対象とする Oracle のバージョン(設定ファイルの MONTYPE パラメータの設定)により、実際に動作するプロセス名は、_11G または_11GR2 が末尾に付きますが、本文中では省略しています。
- 2 RAC 環境、Oracle Restart 環境が対象となります。
- 3 構成番号は、設定ファイルの COMPONENT_ID の値が自動で付加されます。COMPONENT_ID パラメータの記載を省略した時は、0 となります。

第2章 ApplicationMonitor のソフトウェア構成

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

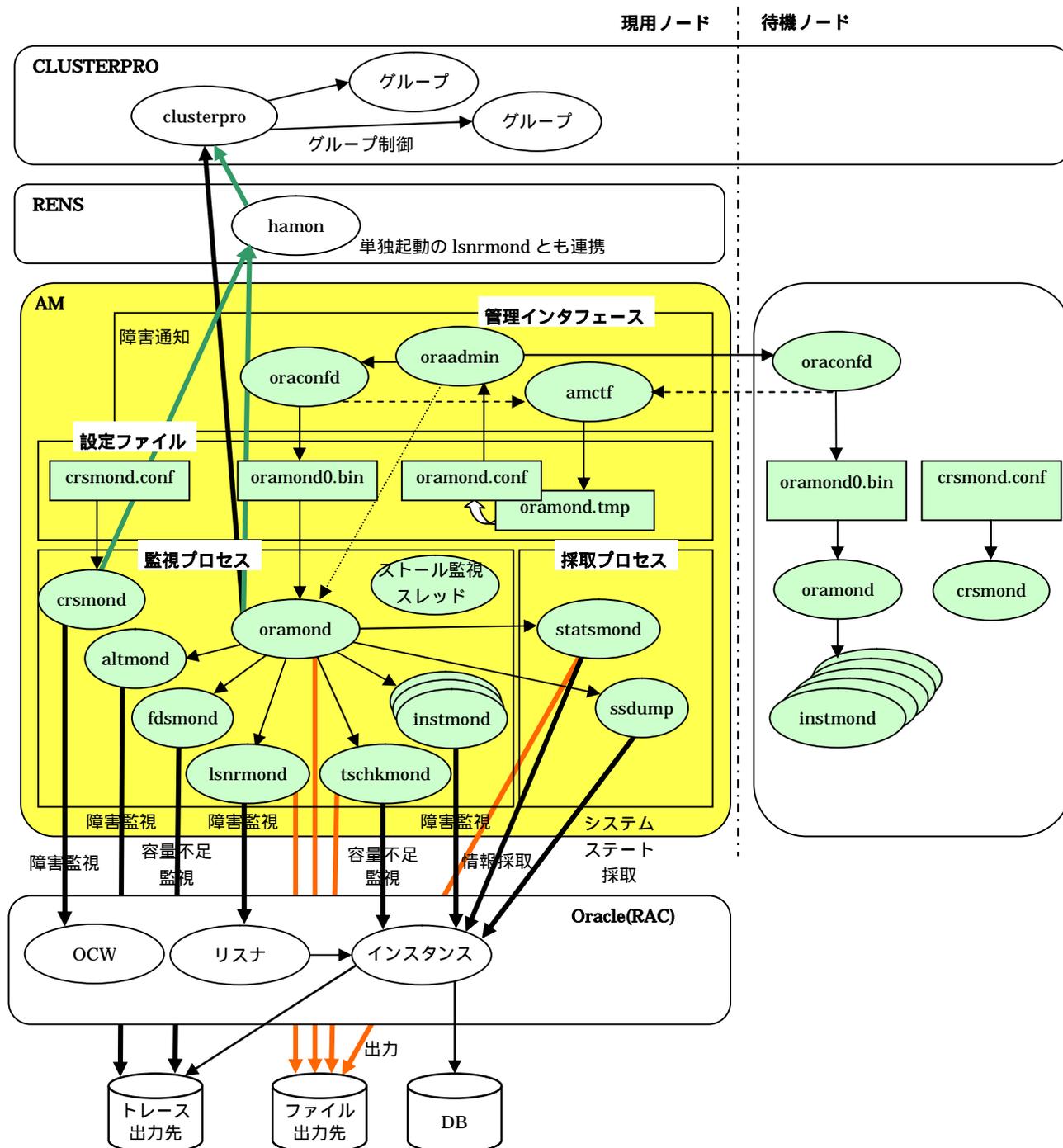


図 2-1 ApplicationMonitor のソフトウェア構成

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 AM の中心となるデーモンです。

インスタンス監視モニタ(instmond)、リスナ監視モニタ(lsnrmond)、表領域監視モニタ(tschkmond)、ディスク領域監視モニタ(fdsmond)、リコンフィグ監視モニタ(almond)、統計情報採取モニタ(statsmond)、システムステータス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 情報の採取及びファイルへの出力を行います。

現用待機構成時、モニタ制御デーモンはインスタンス監視モニタ、リスナ監視モニタと通信を行うことにより Oracle の状態を把握し、障害が発生した場合はシステムステータスダンプ採取デーモンへシステムステータスの採取を要求、同時にスクリプトの実行を行います。また、監視モニタとの通信が切断された場合は監視モニタの再起動を行い、Oracle の監視を継続します。

RAC 構成時、各ノード上で動作し、個々のノードでの Oracle 状態把握の機能と合わせ、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、互い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、全てのノード上でシステムステータスダンプを採取するようにシステムステータス採取デーモンへ指示を出し、同時にスクリプトの実行を行います。また、障害が発生したノードのうち停止するノードの選択を行います。

AM でのシステムステータスの採取及びスクリプト実行後は、自身を停止することで CLUSTERPRO へ障害を通知します。同時に RENS に対しても DOWN 通知を行い、RENS 経由でも CLUSTERPRO へ障害を通知します。

なお、AM は Oracle11gR2 の OCW 環境では、OCW での再起動処理を有効にしたままでも、OCW による Oracle 再起動動作結果も含め最終的な oramond 停止の判断をすることができます。

また、スクリプトの実行機能は、以下の値をスクリプトに第一引数として渡します。

| 値 | 値の意味 |
|-------|---|
| UP | RAC 構成の際、全ノードで障害情報採取を行った場合に、自ノード上のインスタンスが問題なく動作していた場合に指定されます。 |
| DOWN | 自ノード上のインスタンスが停止していた場合に指定されます。 |
| STALL | 自ノード上のインスタンスに無応答停止障害が発生していた場合に指定されます。 |
| ERROR | AM 自身が ERROR となり、監視の続行が不可能な場合に指定されます。 |

モニタ制御デーモンには、自身のスレッドを監視するスレッド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/参照することでスレッドの監視を行います。モニタ制御デーモンがスレッドして時刻情報が更新され

ないと、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。

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、ストールを誤検出する可能性があります。システム時間を変更する場合は、監視スレッド停止オプションを付与して監視を一時停止した上で行ってください。監視スレッドの停止を行った場合でも、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。

2.1.2. 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、一定時間毎に V\$表の参照、SYSTEM 表領域上の表の更新などの処理を Oracle バックグラウンドプロセス単位で行います。また、ユーザ指定表の参照を行います。インスタンス障害検出時はモニタ制御デーモン(oramond)に障害を通知します。

なお、データベースが読み取り専用アクセスでオープンされている場合、及びデータベースが MOUNT 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： DBAの認証方式でパスワード・ファイルを使用している場合は、SYSユーザによる監視を行うことができません。ORACLE_USER / ORACLE_PASSパラメータに任意のOracleユーザを指定して監視を行ってください。ORACLE_USER / ORACLE_PASSについては、「3.2.16 OracleDBA監査 (audit)機能対応の設定」、「3.2.19 パスワード認証環境での監視」を参照してください。

2.1.3. リスナ監視モニタ(lsnrmond)

リスナ監視モニタはリスナ障害を監視するデーモンです。

一定時間毎に Oracle の tnsping コマンドを用いてネットサービスの有無を確認します。また、監視 TAT 情報の採取及びファイルへの出力を行います。リスナ障害検出時はリスナの再起動を試みます。リスナの再起動に失敗した場合はモニタ制御デーモン(oramond)に障害を通知します。

なお、AM は Oracle11gR2 の OCW 環境では、OCW での再起動処理を有効にしたままでも、OCW による Oracle 再起動動作結果も含め最終的な lsnrmond 停止の判断をすることができます。

2.1.4. 表領域監視モニタ(tschkmond)

表領域監視モニタは表領域の容量不足を監視するデーモンです。

一定時間毎にインスタンスに接続し、表領域の最大使用可能エクステンツサイズと空き容量率を監視します。指定された閾値を下回った場合、あらかじめ設定された処理を実行します。また、表領域情報の採取及びファイルへの出力を行います。

なお、データベースが MOUNT 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： 表領域監視モニタは、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等) を使用して行ってください。

2.1.5. ディスク領域監視モニタ(fdsmond)

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

一定時間毎にインスタンスに接続し、設定されているユーザトレース出力先、バックグラウンドトレース出力先、コアファイル出力先及び監査証跡出力先を取得し、該当ディスクの空き容量率を監視します。また、アーカイブログモードで運用されている場合は、アーカイブログ出力先も監視対象とします。空き容量率が指定された閾値を下回った場合、あらかじめ設定された処理を実行します。

なお、データベースが MOUNT 状態の際、任意の Oracle ユーザによる監視を行う設定となっている(ORACLE_USER / ORACLE_PASS パラメータが設定されている)場合は、インスタンスに対する起動確認のみ行います。

2.1.6. リコンフィグ監視モニタ(altmond)

リコンフィグ監視モニタは Oracle が監視可能な状態か監視するデーモンです。

Oracle の起動時、ノード構成変更時などに一時的に Oracle にアクセスできない時間帯が発生します。この時間帯には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。

第 2 章 ApplicationMonitor のソフトウェア構成

2.1.7. CRS 監視モニタ (crsmond)

CRS 監視モニタは OCW デーモン(crsd/ohasd)の無応答障害を監視するデーモンです。

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

crsd/ohasd 障害検出時は、syslog にメッセージを出力します。また、crsd/ohasd プロセスが存在した場合、プロセスを強制終了することにより、init(8)による crsd/ohasd の再起動を促します。

また、CRS 監視設定ファイルの設定により、指定されたコマンドを実行します。

注意： CRS監視モニタはoramondにより制御されませんので、OCW監視を行う場合、別途起動する必要があります。

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。

一定時間毎に SGA 情報を採取します。採取した情報は、ファイルに出力を行います。

なお、データベースが MOUNT 状態の際、任意の Oracle ユーザによる監視を行う設定となっている(ORACLE_USER / ORACLE_PASS パラメータが設定されている)場合は、インスタンスに対する起動確認のみ行います。

2.2.2. 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)の要求によりシステムステートを採取するデーモンです。

システムステート採取デーモンは常にインスタンスとの接続状態を保ち、障害による接続エラーの場合にもシステムステートの採取を行えるようにしています。

なお、データベースが MOUNT 状態の際、任意の Oracle ユーザによる監視を行う設定となっている場合(ORACLE_USER / ORACLE_PASS パラメータが設定されている場合)は、インスタンスに対する起動確認のみ行います。

第 2 章 ApplicationMonitor のソフトウェア構成

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(oraadmin)は AM の設定、管理、状態表示を行うコマンドです。以下のリクエスト（コマンド）がサポートされています。

| リクエスト名 | 機能 |
|---------|--|
| show | 監視モニタ及び採取モニタの状態を表示します。 |
| suspend | 監視モニタ及び採取モニタのを一時中断（サスペンド）します。 |
| resume | 監視モニタ及び採取モニタのを再開（レジューム）します。 |
| check | 設定ファイル(oramond.conf)の書式及び設定内容が有効かチェックをします。ノード上で Oracle が動作中の場合、テーブル名、表領域名、ユーザ名、パスワードについてもチェックをします。 |
| apply | 設定ファイル(oramond.conf)から構成情報ファイル(oramond<構成番号>.bin)を生成し、各ノードに配布します。モニタ制御デーモン(oramond)が動作中の場合、直ちに設定内容が反映されます。 |
| delete | 構成情報ファイル(oramond<構成番号>.bin)を削除します。 |
| view | 構成情報ファイル(oramond<構成番号>.bin)の内容を表示します。 |

管理コマンドの詳細は「A 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(oraconfd)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。

構成情報管理サーバは管理コマンド(oraadmin)からのリクエスト（コマンド）に従い、構成情報ファイルの設定内容のチェック、ファイルの作成、ファイルの削除を行います。

2.3.3. 設定ファイルテンプレート作成コマンド(amctf)

設定ファイルテンプレート作成コマンド(amctf)は、対話形式もしくは、非対話形式で AM 本体の設定ファイルのテンプレートを作成するコマンドです。

指定された各ノード上で情報を収集し、監視対象の Oracle に応じた設定ファイルのテンプレートを作成します。

コマンドの詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 1.0 for Linux 設定ファイルテンプレート作成コマンド 利用の手引き』を参照してください。

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(oramond.conf)は、AM 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。説明の便宜のため、マニュアル上では設定ファイル(oramond.conf)としていますが、任意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については、「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. 設定ファイルテンプレート(oramond.tmp)

設定ファイルテンプレート(oramond.tmp)は、設定ファイルテンプレート作成コマンド(amctf)によって作成されるテンプレートファイルです。

設定ファイルテンプレートの詳細は、別冊の『CLUSTERPRO MC ApplicationMonitor 1.0 for Linux 設定ファイルテンプレート作成コマンド 利用の手引き』を参照してください。

2.4.3. 構成情報ファイル(oramond<構成番号>.bin)

構成情報ファイル(oramond<構成番号>.bin)は、設定ファイル(oramond.conf)から管理コマンド(oraadmin)及び構成情報管理サーバ(oraconfd)により生成されたバイナリファイルです。AM が動作する各ノード上に配布されます。構成番号は、設定ファイルのCOMPONENT_IDの番号が付加されます。

監視プロセス（モニタ制御デーモン、インスタンス監視モニタ、リスナ監視モニタ、表領域監視モニタ、ディスク領域監視モニタ、リコンフィグ監視モニタ）及び採取プロセス（統計情報採取モニタ、システムステータス採取デーモン）はこの構成情報ファイルを元に動作を決定します。

2.4.4. CRS 監視設定ファイル(crsmond.conf)

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

CRS 監視設定ファイルの詳細については、「3.11 CRS 監視設定ファイル(crsmond.conf)の作成」を参照してください。

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

haam.dict（イベント辞書ファイル）、haam_stat.dict（ステータス値定義ファイル）、haam_stat_map.dict（ステータス値マッピング定義ファイル）はRENS連携時に使用されるファイルです。AMインストール時に配置されます。このファイルはインストールしたままの状態で使用できます。

第 2 章 ApplicationMonitor のソフトウェア構成

第3章 ApplicationMonitor の設定

AM を利用するためには、以下の設定が必要となります。

- ・ インスタンス監視表の作成
- ・ 設定ファイル(oramond.conf)の作成と適用
- ・ Oracle エラーのフィルタリング設定
- ・ 情報採取スクリプト(oradiag)の設定
- ・ CLUSTERPRO の設定 (RENS の設定)
- ・ リスナ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
- ・ ProcessSaver との併用
- ・ CRS 監視設定ファイル(crsmond.conf)の作成
- ・ OCW と ApplicationMonitor の監視対象設定について

これらの設定方法について、順に説明します。

第3章 ApplicationMonitor の設定

3.1. インスタンス監視表の作成

AM 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表（インスタンス監視を行うための表）を作成する必要があります。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。

1 ビューの作成

監視を行うためのビューを作成します。製品に添付されているシェルを実行することで作成することができます。Oracle が起動した状態で、製品に添付されているシェルを実行してください。

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

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

2 監視用 Oracle 接続ユーザの作成

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

2.1 監視用プロファイルの作成

Oracle11g より、パスワードの有効期限がデフォルトで 180 日に設定されます。

パスワード有効期限切れにより、接続不可となる状況を防ぐため、パスワードの有効期限が無期限となるプロファイルを作成します。

Oracle が起動した状態で、製品に添付されているシェルを実行してください。

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

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

2.2 監視用 Oracle 接続ユーザの作成

OSDBA(dba)グループに所属する Oracle ユーザで Oracle に接続し、Oracle 接続ユーザを作成してください。

【注意】作成時、2.1 で作成した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> IDENTIFIED BY <Oracle 接続パスワード>
      DEFAULT TABLESPACE SYSTEM
      QUOTA UNLIMITED ON SYSTEM
      PROFILE HAAM_PROFILE;
```

3 監視表の作成

3.1 監視を行うための表を作成する

Oracle が起動した状態で、製品に添付されているシェルを実行してください。

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

【注意】SYS ユーザ以外のユーザで監視を行う場合は、2 で作成した Oracle 接続ユーザを指定してください。

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

XXX：監視対象ノード数

YYY：Oracle 接続ユーザ

【注意】監視表は、監視対象ノード数または Oracle 接続ユーザを変更する場合、再作成する必要があります。

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

Oracle 接続ユーザの確認方法及び出力例は以下のとおりです。

(USERNAME 列の値を指定してください)

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

USERNAME          DEFAULT_TABLESPACE  PROFILE
-----
SYS                SYSTEM              DEFAULT
SYSTEM            SYSTEM              DEFAULT
SCOTT              USERS              DEFAULT
HAAM              SYSTEM              HAAM_PROFILE
```

3.2 作成した監視表の確認

監視表作成時、確認処理が自動的に実施されます。

個別に確認を行う場合は、以下のシェルを実行してください。

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

【注意】SYS ユーザ以外のユーザで監視を行う場合は、2 で作成した Oracle 接続ユーザを指定してください。

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

XXX : 監視対象ノード数

YYY : Oracle 接続ユーザ

3.2.1 TABLE オブジェクトの確認

- ・監視対象ノード数分のレコードが検索されること。
- ・TABLESPACE_NAME 列が、SYSTEM であること。

出力例は以下のとおりです。

| TABLE_NAME | TABLESPACE_NAME |
|------------|-----------------|
| ----- | ----- |
| TIME_CHK0 | SYSTEM |
| TIME_CHK1 | SYSTEM |

3.2.2 SEQUENCE オブジェクトの確認

- ・監視対象ノード数分のレコードが検索されること。
- ・CYCLE_FLAG 列が、Y であること。
- ・CACHE_SIZE 列が、0 であること。
- ・FNO 列及び BNO 列が、作成した全ての SEQUENCE で同一であること。

出力例は以下のとおりです。

| OBJECT_NAME | OBJECT_TYPE | CYCLE_FLAG | CACHE_SIZE | FNO | BNO |
|-------------|-------------|------------|------------|-------|-------|
| ----- | ----- | ----- | ----- | ----- | ----- |
| TEST_SEQ0 | SEQUENCE | Y | 0 | 1 | 667 |
| TEST_SEQ1 | SEQUENCE | Y | 0 | 1 | 667 |

3.2.3 監視表の再作成

監視表が正しく作成されていない場合は、以下のシェルを実行して監視表を削除したのち、再度、監視表を作成してください。

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

【注意】SYS ユーザ以外のユーザで監視を行う場合は、2 で作成した Oracle 接続ユーザを指定してください。

```
$ /opt/HA/AM/bin/init/haam_init.sh drop table [<Oracle 接続ユーザ>]
```

4 権限の付与

SYS ユーザ以外のユーザで監視を行う場合、2 で作成した Oracle 接続ユーザに、監視に必要な権限を付与してください。製品に添付されているシェルを実行することで付与することができます。

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

```
$ /opt/HA/AM/bin/init/haam_init.sh grant <Oracle 接続ユーザ>
```

【注意】Oracle 接続ユーザを変更する場合は、権限を付与しなおす必要があります。

第 3 章 ApplicationMonitor の設定

3.2. 設定ファイル(oramond.conf)の作成と適用

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

AMをインストールすると、/etc/opt/HA/AM/conf/sampleディレクトリに設定ファイルのサンプルが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。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ようにしてください。

AMの設定ファイルに記述するORACLE_SID、ORACLE_BASE、ORACLE_HOME、ORA_NLS (ORA_NLS10に相当)、SHLIB_PATH、NLS_LANGパラメータはOracleの環境に依存しますので、環境を確認の上、必ず設定してください。

設定ファイルを作成後、設定内容を管理コマンド(oraadmin)の”-c apply”により適用を行います。設定ファイルは構成情報ファイル(oramond<構成番号>.bin)に変換され、各ノードの構成情報管理サーバ(oraconfd)に送信されます。この時、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)コマンド等により確認してください。

管理コマンドにより生成される構成情報ファイル(oramond<構成番号>.bin)のサイズは最大 31744 バイトになります。このサイズを超える設定をした場合、適用時にエラーになりますので注意してください。文字列パラメータ値の長さにより異なりますが、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「A コマンドリファレンス」を参照してください。

3.2.1. 設定ファイルの記述例

設定ファイルは、コメント、パラメータ、パラメータ群を囲むステートメントにより構成されたテキストファイルです。ダブルクォーテーション(")で囲まれた文字列以外は、全て半角英数文字で記述します。

Oracle のインスタンス監視、リスナ監視を 1つのグループとし、現用ノードと待機ノードで同一の設定をした場合、設定は以下のように記述します (例 1)。

(例 1) Oracle 11gR2 監視簡易設定例

```
# 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 11GR2 ( 11G or 11GR2 )
MONATYPE = 11GR2

#
# Oracle 環境変数
#
ORACLE_BASE = /oracle
ORACLE_HOME = /oracle/product/11.2.0
ORA_NLS      = /oracle/product/11.2.0/nls/data
SHLIB_PATH  = /oracle/product/11.2.0/lib
NLS_LANG    = japanese_japan.ja16euc

#
# 現用ノード設定
#
Node node1 {
    ORACLE_SID = c1db1

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

    ListenerMonitor LISTENER {
        NET_SERVICE_NAME = ORA11GR2
    }
}

#
# 待機ノード設定
#
Node node2 {
    ORACLE_SID = c1db1

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

    ListenerMonitor LISTENER {
        NET_SERVICE_NAME = ORA11GR2
    }
}
```

第 3 章 ApplicationMonitor の設定

2 ノードで構成されたクラスタ環境で、RAC のインスタンス監視、リスナ監視、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及び統計情報の採取、システムステータスの採取を行う場合、設定は以下のように記述します (例 2)。

例 1 のように、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。

(例 2) Oracle 11gR2 RAC 監視設定例

```
# 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 11GR2 ( 11G or 11GR2 )
MONATYPE = 11GR2

#
# Oracle 環境変数
#
ORACLE_BASE = /oracle/
ORACLE_HOME = /oracle/product/11.2.0
ORA_NLS = /oracle/product/11.2.0/ocommon/nls/admin/data
SHLIB_PATH = /oracle/product/11.2.0/lib
NLS_LANG = japanese_japan.ja16euc

GRID_BASE = /grid
GRID_HOME = /11.2.0/grid
GRID_USER = grid

# 障害時動作
HALT_CONTROL_POLICY = configured
HALT_CONTROL_OPTION = always
HALT_CONTROL_METHOD = none

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

#
# リスナ監視設定
#
ListenerMonitor {
    LOG_LEVEL = 2
    MONITOR_USER = oracle
    BOOT_TIMEOUT = 10
    MONITOR_CONTROL = yes
    TNSPING = /oracle/product/11.2.0/bin/tnsping
    LSNRCTL = /etc/opt/HA/AM/lbin/lsnr_control.sh
    POLL_INTERVAL = 60
    POLL_TIMEOUT = 90
    SERVICE_DOWN = no
    RESTART_COUNT = 5
    RESTART_INTERVAL = 3
}

#
# データベース容量不足監視設定
#
TableSpaceMonitor {
    LOG_LEVEL = 2
    MONITOR_USER = oracle
    BOOT_TIMEOUT = 10
    MONITOR_CONTROL = yes
}
```

3.2 設定ファイル(oramond.conf)の作成と適用

```
POLL_INTERVAL = 3600
POLL_TIMEOUT = 60
# COMMAND_NAME =
}

#
# 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
}

#
# システムステート採取設定
#
SystemStateDaemon {
    LOG_LEVEL = 2
    MONITOR_USER = oracle
    BOOT_TIMEOUT = 10
    GET_DUMP = yes
    POLL_INTERVAL = 60
    POLL_TIMEOUT = 60
    DUMP_INTERVAL = 30
    DUMP_COUNT = 3
    DUMP_TIMEOUT = 90
}

#
# ノード設定
#
Node node1 {
    ORACLE_SID = clustdb1

    InstanceMonitor {
        GET_STATS = yes
        Table USER_TABLE1 {
            TABLE_USER = scott
            TABLE_PASSWORD = tiger
        }
    }

    ListenerMonitor LISTENER1 {
        NET_SERVICE_NAME = LSNR1
        GET_STATS = yes
    }

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

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

第3章 ApplicationMonitor の設定

```
#
# ノード設定
#
Node node2 {
  ORACLE_SID = clustdb2

  InstanceMonitor {
    GET_STATS = yes
    Table USER_TABLE2 {
      TABLE_USER      = scott
      TABLE_PASSWORD  = tiger
    }
  }

  ListenerMonitor LISTENER2 {
    NET_SERVICE_NAME = LSNR2
    GET_STATS        = yes
  }

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

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

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、AMの動作に必要な値を数値または文字列で指定します。パラメータ値に全角文字、スペース、タブを含める場合は、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。文字列内にダブルクォーテーション(")及び改行コードを使用することはできません。

1つのパラメータにつき1行(改行コードを含め1023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。各種パラメータについては、「3.2.7 パラメータ一覧」を参照してください。

記述形式：

パラメータ名 = パラメータ値[, パラメータ値]

3.2.4. ステートメントの記述形式

ステートメントは、パラメータをそれぞれの設定毎に分類するために使用します。ステートメントは次の12種類があります。

- ・全ノードステートメント
- ・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
- ・StatsMonitor ステートメント
- ・SystemStateDaemon ステートメント
- ・Table ステートメント
- ・TableSpace ステートメント
- ・Directory ステートメント
- ・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以下に説明します。

・ 全ノードステートメント

他のどのステートメントにも含まれないパラメータ群が全ノードステートメントです。全ノードステートメントに記述したパラメータ及びステートメントは、全てのノードに共通な設定となります。

全ノードステートメントには、全ノードステートメント用パラメータ、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、StatsMonitor ステートメント、SystemStateDaemon ステートメント、Node 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できるパラメータについては、「3.2.7 パラメーター一覧」の全ノードステートメントを参照してください。

・ Node ステートメント

Node ステートメントには、ノード毎に設定するパラメータとステートメントを記述します。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]  
    [StatsMonitor ステートメント]  
    [SystemStateDaemon ステートメント]  
}
```

ノード名には、NODE_NAME パラメータに指定したノード名を指定します。ノード名は省略できません。

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

• InstanceMonitor ステートメント

InstanceMonitor ステートメントには、インスタンス監視モニタ(instmond)毎に設定するパラメータを記述します。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内に記述した場合は、全ノード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、そのノード内のインスタンス監視モニタの設定になります。

InstanceMonitor ステートメントに記述できるパラメータについては、「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 ステートメントには、リスナ監視モニタ (lsnrmond) 毎に設定するパラメータを記述します。記述形式は以下のとおりです。

記述形式：

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

ListenerMonitor ステートメントを全ノードステートメント内に記述した場合は、全ノードのリスナ監視モニタの設定になります。この場合、リスナ名は指定できません。

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、リスナ名を指定しない場合と指定する場合とがあり、リスナ名を指定しない場合は、そのノード内全てのリスナ監視モニタの設定になります。リスナ名を指定した場合は、該当するリスナ監視モニタの設定になります。ノード内で同じリスナ名を重複して指定することはできません。

ListenerMonitor ステートメントに記述できるパラメータについては、「3.2.7 パラメータ一覧」の ListenerMonitor ステートメントを参照してください。

注意： Node ステートメント内にリスナ名を指定した ListenerMonitor ステートメントがない場合は、該当ノードでリスナ監視モニタ (lsnrmond) の起動 (リスナ監視) を行いません。リスナ名を指定した ListenerMonitor ステートメントが複数ある場合は、それぞれのリスナ監視モニタが起動されます。

• TableSpaceMonitor ステートメント

TableSpaceMonitor ステートメントには、表領域監視モニタ(tschkmond)毎に設定するパラメータを記述します。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(tschkmond)を識別するための任意の文字列です。ノード内で同じ ID を指定することはできません。

TableSpaceMonitor ステートメントを全ノードステートメント内に記述した場合は、全ノードの表領域監視モニタの設定になります。この場合、表領域監視モニタ ID は指定できません。

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とがあり、表領域監視モニタ ID を指定しない場合は、そのノード内全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。

TableSpaceMonitor ステートメントに記述できるパラメータについては、「3.2.7 パラメータ一覧」の TableSpaceMonitor ステートメントを参照してください。

注意： 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、該当ノードで表領域監視モニタ(tschkmond)の起動(表領域監視)を行いません。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、それぞれの表領域監視モニタが起動されます。

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する必要があります。

• **DiskSpaceMonitor** ステートメント

DiskSpaceMonitor ステートメントには、ディスク領域監視モニタ(fdsmond)毎に設定するパラメータを記述します。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内に記述した場合は、全ノード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、そのノード内のディスク領域監視モニタの設定になります。

DiskSpaceMonitor ステートメントに記述できるパラメータについては、「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(fdsmond)の起動（ディスク領域監視）を行いません。

• StatsMonitor ステートメント

StatsMonitor ステートメントには統計情報採取モニタ(statsmond)毎に設定するパラメータを記述します。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(statsmond)を識別するための任意の文字列です。ノード内で同じ ID を指定することはできません。

StatsMonitor ステートメントを全ノードステートメント内に記述した場合は、全ノードの統計情報採取モニタの設定になります。この場合、*統計情報採取モニタ ID* は指定できません。

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とがあり、*統計情報採取モニタ ID* を指定しない場合は、そのノード内全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。

StatsMonitor ステートメントに記述できるパラメータについては、「3.2.7 パラメータ一覧」の StatsMonitor ステートメントを参照してください。

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、該当ノードで統計情報採取モニタ(statsmond)の起動 (SGA 情報の採取) を行いません。*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、それぞれの統計情報採取モニタが起動されます。

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する必要があります。

• **SystemStateDaemon ステートメント**

SystemStateDaemon ステートメントには、システムステートダンプ採取デーモン (ssdump) 毎に設定するパラメータを記述します。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内に記述した場合は、全ノード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、そのノード内のシステムステートダンプ採取デーモンの設定になります。

SystemStateDaemon ステートメントに記述できるパラメータについては、「3.2.7 パラメータ一覧」の SystemStateDaemon ステートメントを参照してください。

注意： SystemStateDaemon ステートメントを省略した場合でも、障害発生時のシステムステートの採取は、デフォルトの設定により行われます。明示的に採取をしない設定にする場合は、GET_DUMP パラメータを NO に設定してください。

• Table ステートメント

Table ステートメントには、表名毎に設定するパラメータを記述します。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]
}
```

表名は半角で最大 30 文字まで指定できます。表名に全角文字、スペース、タブ、半角小文字を含める場合は、ダブルクォーテーション(")で囲ってください。表名は省略できません。

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

Table ステートメントを記述することで、デフォルトの監視（独自表の更新）と合わせ、指定された表を定期的にチェックする監視を追加できます。

Table ステートメントに記述できるパラメータについては、「3.2.7 パラメータ一覧」の Table ステートメントを参照してください。

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

• TableSpace ステートメント

TableSpace ステートメントには、表領域名毎に設定するパラメータを記述します。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]
}
```

表領域名は半角で最大 30 文字まで指定できます。表領域名に全角文字、スペース、タブ、小文字を含める場合は、ダブルクォーテーション(")で囲ってください。表領域名は省略できません。

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。

TableSpace ステートメントに記述できるパラメータについては、「3.2.7 パラメータ一覧」の TableSpace ステートメントを参照してください。

• **Directory ステートメント**

Directory ステートメントには、Directory 種別毎に設定するパラメータを記述します。記述形式は以下のとおりです。

記述形式：

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

種別は、HOME、ARCHIVE、TRACE、ALERT、CORE、AUDIT のいずれかを指定します。種別を省略することはできません。

Directory ステートメントは、DiskSpaceMonitor ステートメント内にも記述できます。

Directory ステートメントに記述できるパラメータについては、「3.2.7 パラメータ一覧」の Directory ステートメントを参照してください。

注意： Directory ステートメントを省略した場合は、全ての領域について、DiskSpaceMonitor ステートメントで設定された値で監視が行われます。Directory ステートメントがある場合は、種別毎に設定された値で監視が行われます。

Directory ステートメントは、ディスク領域監視モニタの監視対象毎に、空き容量率、実行コマンドを変更する場合に記述します。指定できる種別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 |

• Stats ステートメント

Stats ステートメントには、SGA 情報の採取に設定するパラメータを記述します。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 を指定します。*統計情報種別*を省略することはできません。

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

Stats ステートメントに記述できるパラメータについては、「3.2.7 パラメータ一覧」の Stats ステートメントを参照してください。

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わすことができます。

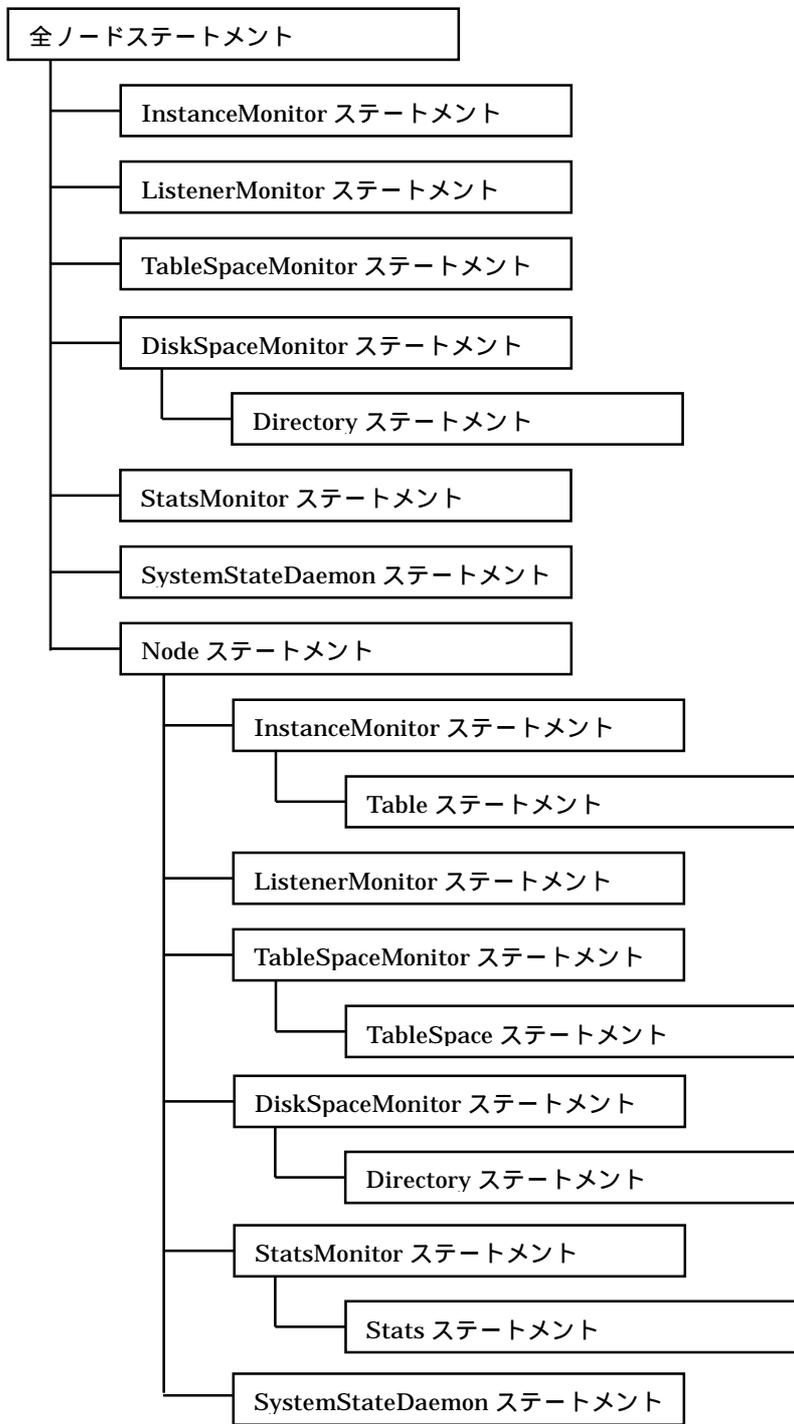


図 3-1 ステートメントの階層

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

3.2.6. パラメータとステートメントの記述順

設定ファイル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
InstanceMonitor ステートメント {
    InstanceMonitor ステートメントのパラメータ
}
ListenerMonitor ステートメント {
    ListenerMonitor ステートメントのパラメータ
}
TableSpaceMonitor ステートメント {
    TableSpaceMonitor ステートメントのパラメータ
}
DiskSpaceMonitor ステートメント {
    DiskSpaceMonitor ステートメントのパラメータ
    種別を指定した Directory ステートメント {
        Directory ステートメントのパラメータ
    }
}
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
    }
    Table ステートメント {
        Table ステートメントのパラメータ
    }
}
リスナ名を指定しない ListenerMonitor ステートメント {
    ListenerMonitor ステートメントのパラメータ
}
リスナ名を指定した ListenerMonitor ステートメント {
    ListenerMonitor ステートメントのパラメータ
}
表領域監視モニタ ID を指定しない TableSpaceMonitor ステートメント {
    TableSpaceMonitor ステートメントのパラメータ
}
表領域監視モニタ ID を指定した TableSpaceMonitor ステートメント {
    TableSpaceMonitor ステートメントのパラメータ
    TableSpace ステートメント {

```

第 3 章 ApplicationMonitor の設定

```
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
  }
}
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
  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
}
}
```

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場合がありますので注意してください。正しく設定ができたかどうかは、管理コマンド(oraadmin)の”-c check”オプションで確認をしてください。

oraadmin コマンドの詳細は「A コマンドリファレンス」を参照してください。

3.2.7. パラメーター一覧

全ノードステートメントに記述可能なパラメータは以下のとおりです。

| パラメータ名 | 意味 | 1 | 2 |
|---------------------|---|---|---|
| LOG_LEVEL | ログの出力レベルを指定します。指定できる値の範囲は0~3の整数値です。省略時は2が設定されます。 LOG_LEVELについては、「3.2.8 ログレベルの設定」を参照してください。 | | |
| COMPONENT_ID | 構成番号を指定します。指定できる値の範囲は0~9の整数値です。省略時は0が設定されます。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毎に別の構成番号を割り当ててください。 | | × |
| MONTYPE | 監視対象のOracleのバージョンを指定します。11Gまたは11GR2を指定できます。 このパラメータは省略できません。 | × | × |
| NODE_NAME | Oracleが動作するノード名またはIPアドレスを指定します。 このパラメータは省略できません。 NODE_NAMEについては、「3.2.9 ノード名の設定」を参照してください。 | × | × |
| 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。指定できる値の範囲は1024~65535の整数値です。省略時は25312が設定されます。ADMIN_PORTパラメータと違う番号を指定することも可能です。 複数の構成(構成番号)が存在する場合は設定ファイル毎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、競合しない番号を設定してください。 | | × |
| 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。指定できる値の範囲は1024~65535の整数値です。省略時は25312が設定されます。SERVICE_PORTパラメータと違う番号を指定することも可能です。 複数の構成(構成番号)が存在する場合は設定ファイル毎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、競合しない番号を設定してください。 | | × |
| HALT_CONTROL_POLICY | 停止するノードの選択ポリシーを指定します。FIRST_FAILEDまたはCONFIGUREDを指定できます。省略時はFIRST_FAILEDに設定されます。 HALT_CONTROL_POLICYについては、「3.2.10 停止ノード選択ポリシーの設定(RAC構成時のみ)」を参照してください。 RAC構成時のみ設定が有効となります。 | | |
| HALT_CONTROL_OPTION | 障害ノードの停止動作を指定します。ALWAYS、NEVER、OUTLIVEを指定できます。省略時はALWAYSに設定されます。 HALT_CONTROL_OPTIONについては、「3.2.11 障害ノードの停止動作の設定」を参照してください。 | | |
| HALT_CONTROL_METHOD | ノード停止時にOracleの構成プロセスに対しkillコマンドを送るかどうかを指定します。KILLまたはNONEを指定できます。省略時はNONEに設定されます。 HALT_CONTROL_METHODについては、「3.2.12 Oracle構成プロセス停止の設定」を参照してください。 | | |
| ORACLE_BASE | dbユーザの環境変数ORACLE_BASEを指定します。 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、全てのNodeステートメントで記述する必要があります。 | | × |

第 3 章 ApplicationMonitor の設定

| パラメータ名 | 意味 | 1 | 2 |
|-----------------|--|---|---|
| ORACLE_HOME | db ユーザの環境変数 ORACLE_HOME を指定します。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、全ての Node ステートメントで記述する必要があります。 | | × |
| ORA_NLS | db ユーザの環境変数 ORA_NLS10 の値を指定します。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、全ての Node ステートメントで記述する必要があります。 | | × |
| SHLIB_PATH | db ユーザ環境の 64bit 共有ライブラリのパスを指定します。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、全ての Node ステートメントで記述する必要があります。 | | × |
| NLS_LANG | db ユーザの環境変数 NLS_LANG(言語)を指定します。省略時は AMERICAN_AMERICA.US7ASCII に設定されます。 | | × |
| TNS_ADMIN | db ユーザの環境変数 TNS_ADMIN を指定します。省略時は環境変数を設定しません。 | | × |
| GRID_BASE | grid ユーザの環境変数 ORACLE_BASE を指定します。省略時は環境変数を設定しません。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| GRID_HOME | grid ユーザの環境変数 ORACLE_HOME を指定します。省略時は環境変数を設定しません。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| GRID_USER | 障害検出時に実行する OCW リトライ判定用スクリプトの実効ユーザ名を指定します。省略時は実行ユーザ名を設定しません。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| MONITOR_USER | 監視モニタ (instmond, lsnrmond, tschmond, fdsmond, altmond) 及び採取モニタ (statsmond, ssdump) の実効ユーザ名を指定します。省略時は oracle に設定されます。 | | × |
| BOOT_TIMEOUT | 監視モニタ (instmond, lsnrmond, tschmond, fdsmond, altmond) 及び採取モニタ (statsmond, ssdump) の起動を確認するまでの待ち時間を秒で指定します。この待ち時間内に起動できない場合システム障害として扱われます。指定できる値の範囲は 1~3600 です。省略時は 10 秒に設定されます。 | | |
| ORACLE_USER | 監視モニタ (instmond, tschmond, fdsmond, altmond) 及び採取モニタ (statsmond, ssdump) で監視接続を行うための Oracle ユーザ名を指定します。省略時は Oracle ユーザ名を設定しません。 ORACLE_USER については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| ORACLE_PASS | 監視モニタ (instmond, tschmond, fdsmond, altmond) 及び採取モニタ (statsmond, ssdump) で監視接続を行うための Oracle ユーザのパスワードを指定します。省略時は Oracle ユーザのパスワードを設定しません。 ORACLE_PASS については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| SYSDBA_CONNECT | ORACLE_USER を指定した際、SYSDBA 権限で接続するかどうかを YES または NO で指定します。省略時は NO が設定されます。 SYSDBA_CONNECT については、「3.2.19 パスワード認証環境での監視」を参照してください。 | | |
| MONITOR_CONTROL | モニタ制御デーモン (oramond) から監視モニタ (lsnrmond, tschmond, fdsmond) 及び採取モニタ (statsmond) の制御を行うかどうかを YES または NO で指定します。省略時は YES が設定されます。 MONITOR_CONTROL については、「3.2.13 監視モニタ制御の設定」を参照してください。 | | |

3.2 設定ファイル(oramond.conf)の作成と適用

| パラメータ名 | 意味 | 1 | 2 |
|-------------------|---|---|---|
| SCRIPT_USER | 障害検出時に実行するスクリプトの実効ユーザ名を指定します。存在しないユーザ名を指定した場合はエラーとなります。省略時は " root " が設定されます。 | | |
| SCRIPT_NAME | 障害検出時に実行するスクリプトのパス名(絶対パス)を指定します。存在しないパス名を指定した場合はエラーとなります。省略時はシェルスクリプトの実行を行いません。 必ず実行権が付いていることを確認してください。 | | |
| SCRIPT_TIMEOUT | 障害検出時に実行するスクリプトの実行時間の上限を秒で指定します。指定できる値の範囲は 1~3600 の整数です。省略時は 30 秒が設定されます。 | | |
| ORACLE_HANG_CHECK | 監視対象 DB のバージョンが 11g (11.1.0.7 以降) の際、Oracle ハング検知機能を利用した監視を行うかどうかを YES または NO で指定します。省略時は NO が設定されます。 ORACLE_HANG_CHECK については、「3.2.18 Oracle ハング検知機能を利用した監視について(RAC 環境のみ)」を参照してください。 | | |
| CRS_CMD_TIMEOUT | 障害検出時に実行する OCW リトライ判定用スクリプトの実行時間の上限を秒で指定します。指定できる値の範囲は 1~3600 の整数です。省略時は 90 秒が設定されます。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| GET_STATS | 監視モニタ (instmond, lsnrmond, tschkmond) 及び採取モニタ (statsmond) で統計情報の採取を行うかどうかを YES または NO で指定します。省略時は YES が設定されます。 GET_STATS については、「3.2.21 統計情報採取の設定」を参照してください。 | | |

- 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、設定ファイル間で値が重複することがないように、必ず設定してください。
- 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)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については、「第 4 章 ApplicationMonitor の動作」を参照してください。

第 3 章 ApplicationMonitor の設定

Node ステートメントに記述可能なパラメータは以下のとおりです。

| パラメータ名 | 意味 | 1 | 2 |
|-----------------|--|---|---|
| LOG_LEVEL | ログの出力レベルを指定します。指定できる値の範囲は 0~3 の整数値です。省略時は上位のステートメントに指定した値が設定されます。 LOG_LEVEL については、「3.2.8 ログレベルの設定」を参照してください。 | | |
| ORACLE_SID | Oracle のインスタンス識別子を指定します。 このパラメータは省略できません。 | × | × |
| ORACLE_BASE | db ユーザの環境変数 ORACLE_BASE を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| ORACLE_HOME | db ユーザの環境変数 ORACLE_HOME を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| ORA_NLS | db ユーザの環境変数 ORA_NLS10 の値を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| SHLIB_PATH | db ユーザ環境の 64bit 共有ライブラリのパスを指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| NLS_LANG | db ユーザの環境変数 NLS_LANG(言語)を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| TNS_ADMIN | db ユーザの環境変数 TNS_ADMIN を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| MONITOR_USER | 監視 モニタ (instmond, lsnrmond, tschkmond, fdsmond, altmond) 及び採取モニタ (statsmond, ssdump) の実効ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| BOOT_TIMEOUT | 監視 モニタ (instmond, lsnrmond, tschkmond, fdsmond, altmond) 及び採取モニタ (statsmond, ssdump) の起動を確認するまでの待ち時間を秒で指定します。この待ち時間内に起動できない場合システム障害として扱われます。指定できる値の範囲は 1~3600 です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| ORACLE_USER | 監視モニタ (instmond, tschkmond, fdsmond, altmond) 及び採取モニタ (statsmond, ssdump) で監視接続を行うための Oracle ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_USER については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| ORACLE_PASS | 監視モニタ (instmond, tschkmond, fdsmond, altmond) 及び採取モニタ (statsmond, ssdump) で監視接続を行うための Oracle ユーザのパスワードを指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_PASS については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| SYSDBA_CONNECT | ORACLE_USER を指定した際、SYSDBA 権限で接続するかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 SYSDBA_CONNECT については、「3.2.19 パスワード認証環境での監視」を参照してください。 | | |
| MONITOR_CONTROL | モニタ制御デモン (oramond) から監視モニタ (lsnrmond, tschkmond, fdsmond) の制御を行うかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 MONITOR_CONTROL については、「3.2.13 監視モニタ制御の設定」を参照してください。 | | |

3.2 設定ファイル(oramond.conf)の作成と適用

| パラメータ名 | 意味 | 1 | 2 |
|-------------|--|---|---|
| SCRIPT_USER | 障害検出時に実行するスクリプトの実効ユーザ名を指定します。存在しないユーザ名を指定した場合はエラーとなります。省略時は上位のステートメントに指定した値が設定されます。 | | |
| SCRIPT_NAME | 障害検出時に実行するスクリプトのパス名(絶対パス)を指定します。存在しないパス名を指定した場合はエラーとなります。省略時は上位のステートメントに指定した値が設定されます。 必ず実行権が付いていることを確認してください。 | | |
| GET_STATS | 監視モニタ (instmond, lsnrmond, tschkmond) 及び採取モニタ (statsmond)で統計情報の採取を行うかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 GET_STATS については、「3.2.21 統計情報採取の設定」を参照してください。 | | |

- 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)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については、「第4章 ApplicationMonitor の動作」を参照してください。

第 3 章 ApplicationMonitor の設定

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

| パラメータ名 | 意味 | 1 | 2 |
|------------------|---|---|---|
| LOG_LEVEL | ログの出力レベルを指定します。指定できる値の範囲は 0～3 の整数値です。省略時は上位のステートメントに指定した値が設定されます。 LOG_LEVEL については、「3.2.8 ログレベルの設定」を参照してください。 | | |
| MONITOR_USER | インスタンス監視モニタ (instmond) の実効ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| 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はシステム障害として扱われます。指定できる値の範囲は 1～3600 です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| ORACLE_USER | インスタンス監視モニタ (instmond) で監視接続を行うための Oracle ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_USER については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| ORACLE_PASS | インスタンス監視モニタ (instmond) で監視接続を行うための Oracle ユーザのパスワードを指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_PASS については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| SYSDBA_CONNECT | ORACLE_USER を指定した際、SYSDBA 権限で接続するかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 SYSDBA_CONNECT については、「3.2.19 パスワード認証環境での監視」を参照してください。 | | |
| POLL_INTERVAL | インスタンス監視モニタ (instmond) の監視間隔を秒で指定します。指定できる値の範囲は 5～2678400 の整数です。省略時は 90 秒に設定されます。 このパラメータは、全てのノードで同じ値を設定してください。 | | |
| POLL_TIMEOUT | インスタンスのストールと見なすまでの上限時間を秒で指定します。指定できる値の範囲は 1～86400 の整数です。省略時は 120 秒に設定されます。 このパラメータの設定値は、監視開始時、監視対象 DB の環境に合わせて自動的に調整される場合があります。 | | |
| 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、このパラメータに設定された秒数を経過した際、監視を再開します。指定できる値の範囲は 1～3600 の整数です。省略時は 30 秒に設定されます。 | | |
| POLL_DOWN_RETRY | インスタンスのダウンを検出した時に再確認を行う回数を指定します。指定できる値の範囲は 0～10 の整数です。省略時は 0 回に設定されます。 RAC 環境では全ノードで設定値を統一してください。 | | |
| POLL_STALL_RETRY | インスタンスのストールを検出した時に再確認を行う回数を指定します。指定できる値の範囲は 0～10 の整数です。省略時は 0 回に設定されます。 RAC 環境では全ノードで設定値を統一してください。 | | |

3.2 設定ファイル(oramond.conf)の作成と適用

| パラメータ名 | 意味 | 1 | 2 |
|-----------------|--|---|---|
| SERVICE_DOWN | インスタンスの障害時、RESTART_COUNT の回数再起動を待ち合わせても再起動できなかった場合にノード停止を行うかどうかを指定します。以下を指定できます。 YES : ノード停止します。 NO : ノード停止しません。 NO_DOWN : DOWN 障害の場合は停止しません。 NO_STALL : STALL 障害の場合は停止しません。 省略時は YES に設定されます。 SERVICE_DOWN については、「3.2.14 インスタンスノリスナ障害時のグループ停止の設定」を参照してください。 | | |
| 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。指定できる値の範囲は 0 ~ 999 の整数です。省略時は 2 回に設定されます。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| RESTART_TIMEOUT | 障害検知後、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。指定できる値の範囲は 0 ~ 3600 の整数です。省略時は 60 秒に設定されます。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| CRS_CMD_NAME | OCW が管理しているリソースのステータスを取得するためのシェルスクリプトを指定します。省略時は /etc/opt/HA/AM/lbin/crs_control.sh に設定されます。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| 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 YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 GET_STATS については、「3.2.21 統計情報採取の設定」を参照してください。 | | |

- 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)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については、「第 4 章 ApplicationMonitor の動作」を参照してください。

第 3 章 ApplicationMonitor の設定

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

| パラメータ名 | 意味 | 1 | 2 |
|------------------|--|---|---|
| LOG_LEVEL | ログの出力レベルを指定します。指定できる値の範囲は 0~3 の整数値です。省略時は上位のステートメントに指定した値が設定されます。 LOG_LEVEL については、「3.2.8 ログレベルの設定」を参照してください。 | | |
| MONITOR_USER | リスナ監視モニタ (lsnrmond) の実効ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| BOOT_TIMEOUT | リスナ監視モニタ (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はシステム障害として扱われます。指定できる値の範囲は 1~3600 です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| MONITOR_CONTROL | モニタ制御デーモン (oramond) からリスナ監視モニタ (lsnrmond) の制御を行うかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 MONITOR_CONTROL については、「3.2.13 監視モニタ制御の設定」を参照してください。 | | |
| POLL_INTERVAL | リスナの監視間隔を秒で指定します。指定できる値の範囲は 5~2678400 の整数です。省略時は 60 秒に設定されます。 | | |
| POLL_TIMEOUT | リスナの障害と見なすまでの上限時間を秒で指定します。指定できる値の範囲は 1~86400 の整数です。省略時は 90 秒に設定されます。 | | |
| SERVICE_DOWN | リスナの障害時、RESTART_COUNT の回数再起動を待ち合わせても再起動できなかった場合にノード停止を行うかどうかを指定します。以下を指定できます。 YES : ノード停止します。 NO : ノード停止しません。 NO_DOWN : DOWN 障害の場合は停止しません。 NO_STALL : STALL 障害の場合は停止しません。 省略時は NO に設定されます。 SERVICE_DOWN については、「3.2.14 インスタンス/リスナ障害時のグループ停止の設定」を参照してください。 | | |
| RESTART_COUNT | リスナの障害時、リスナの再起動を試みる回数を指定します。指定できる値の範囲は 0~999 の整数です。省略時は 5 回に設定されます。 | | |
| RESTART_INTERVAL | リスナの再起動失敗から次に再起動を行うまでの間隔を秒で指定します。指定できる値の範囲は 1~10 の整数です。省略時は 3 秒に設定されます。 | | |
| NET_SERVICE_NAME | ネットサービス名を指定します。 このパラメータは省略できません。 | × | × |
| TNSPING | Oracle の tns ping コマンドのコマンド名を絶対パスで指定します。省略時は \${ORACLE_HOME}/bin/tnsping に設定されます。 | | |
| LSNRCTL | リスナ再起動コマンドを絶対パスで指定します。省略時は /etc/opt/HA/AM/lbin/lsnr_control.sh に設定されます。 | | |
| RESTART_TIMEOUT | 障害検知後、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。指定できる値の範囲は 0~3600 の整数です。省略時は 10 秒に設定されます。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| TARGET_RESTART | 障害検知時、AM によるリスナの再起動を行うかどうかを YES または NO で指定します。省略時は YES に設定されます。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |

3.2 設定ファイル(oramond.conf)の作成と適用

| パラメータ名 | 意味 | 1 | 2 |
|---------------|--|---|---|
| CRS_CMD_NAME | OCW が管理しているリソースのステータスを取得するためのシェルスクリプトを指定します。省略時は /etc/opt/HA/AM/lbin/crs_control.sh に設定されます。Oracle11gR2 の OCW 環境で有効なパラメータです。 | | |
| ORACLE_BASE | db ユーザまたは grid ユーザの環境変数 ORACLE_BASE を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| ORACLE_HOME | db ユーザまたは grid ユーザの環境変数 ORACLE_HOME を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| ORA_NLS | db ユーザまたは grid ユーザの環境変数 ORA_NLS10 の値を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| SHLIB_PATH | db ユーザまたは grid ユーザ環境の 64bit 共有ライブラリのパスを指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| NLS_LANG | db ユーザまたは grid ユーザの環境変数 NLS_LANG(言語)を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| TNS_ADMIN | db ユーザまたは grid ユーザの環境変数 TNS_ADMIN を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| UP_DOWN_RETRY | リスナ監視時、リスナの状態が不安定で UP-DOWN を繰り返すような障害が発生した際に、リトライを行う総回数を指定します。指定できる値の範囲は 0~10 の整数です。省略時は 5 に設定されます。 UP_DOWN_RETRY については、「3.2.20 安定したリスナ監視の実現」を参照してください。 | | |
| UPTIME_COUNT | リスナ監視時、リスナが安定稼働していると見なすまでの回数を指定します。指定された回数 UP を連続して検知した際、記憶していた再試行回数はクリアされます。指定できる値の範囲は 1~60 の整数です。省略時は 1 に設定されます。 UPTIME_COUNT については、「3.2.20 安定したリスナ監視の実現」を参照してください。 | | |
| GET_STATS | リスナ監視モニタ(Isnrmond)で統計情報の採取を行うかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 GET_STATS については、「3.2.21 統計情報採取の設定」を参照してください。 | | |

- 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)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。

第 3 章 ApplicationMonitor の設定

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

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

```
LISTENER =      この部分がリスナ名になります
(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ネットサービス名は以下を参照してください。

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

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

個々のパラメータの関係については、「第 4 章 ApplicationMonitor の動作」を参照してください。

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

注意： リモートからリスナの管理コマンド(lsnrctl)を実行する場合、パスワードを設定する必要があります。

AMでは、設定ファイルのチェック時、lsnrctlコマンドによりリスナの存在チェックを行っているため、リスナ名にフローティングIPを使用したリスナを指定した場合、設定ファイルのチェックが失敗する可能性があります。フローティングIPを使用したリスナの監視設定のチェック及びapply(動的変更含む)時にコマンド要求が失敗する場合は、以下の設定を行うことによりエラーの発生を回避する、もしくはoraadminコマンドの-Fオプションにより、Oracle関連のチェックを強制的にスキップしてください。

設定方法の詳細については、『Oracle Net Services 管理者ガイド』の「Oracle Net Listenerの構成と管理」及び『Oracle Net Services リファレンス・ガイド』の「Oracle Net Listenerのセキュリティ」を参照してください。

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

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

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

入力時にはパスワードは表示されません。

2. /etc/opt/HA/AM/lbin/lsnr_control.sh において、set password コマンドを使用してパスワードを設定した上で、lsnrctl コマンドを実行するように修正します。lsnr_control.sh の LISTENER_PASSWD に 1. で設定したパスワードを指定します。

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

複数のリスナ監視を行う場合は、/etc/opt/HA/AM/lbin/lsnr_control.sh をコピーし、監視対象とするリスナに合わせパスワードを設定の上、使用してください。

また、監視するリスナ毎、上記でコピー・修正したファイルを LSNRCTL パラメータに設定するようにしてください。

第 3 章 ApplicationMonitor の設定

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

| パラメータ名 | 意味 | 1 | 2 |
|-----------------|--|---|---|
| LOG_LEVEL | ログの出力レベルを指定します。指定できる値の範囲は 0~3 の整数値です。省略時は上位のステートメントに指定した値が設定されます。 LOG_LEVEL については、「3.2.8 ログレベルの設定」を参照してください。 | | |
| MONITOR_USER | 表領域監視モニタ (tschkmond) の実効ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| 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はシステム障害として扱われます。指定できる値の範囲は 1~3600 です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| ORACLE_USER | 表領域監視モニタ (tschkmond) で監視接続を行うための Oracle ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_USER については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| ORACLE_PASS | 表領域監視モニタ (tschkmond) で監視接続を行うための Oracle ユーザのパスワードを指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_PASS については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| SYSDBA_CONNECT | ORACLE_USER を指定した際、SYSDBA 権限で接続するかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 SYSDBA_CONNECT については、「3.2.19 パスワード認証環境での監視」を参照してください。 | | |
| 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 YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 MONITOR_CONTROL については、「3.2.13 監視モニタ制御の設定」を参照してください。 | | |
| POLL_INTERVAL | 監視間隔を秒で指定します。指定できる値の範囲は 5~2678400 の整数です。省略時は 3600 秒に設定されます。 | | |
| POLL_TIMEOUT | 全ての表領域 (TableSpace ステートメント) の検査が完了するまで、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。指定できる値の範囲は 1~86400 の整数です。省略時は 60 秒に設定されます。 | | |
| COMMAND_NAME | 使用可能エクステンションもしくは空き領域率が指定した閾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。存在しないコマンドを指定した場合はエラーとなります。省略時はコマンドの実行を行いません。 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。 | | |
| 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 YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 GET_STATS については、「3.2.21 統計情報採取の設定」を参照してください。 | | |

3.2 設定ファイル(oramond.conf)の作成と適用

- 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)が可能かどうかを示します。可能な場合は、不可能な場合は×です。

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

個々のパラメータの関係については、「第4章 ApplicationMonitor の動作」を参照してください。

第 3 章 ApplicationMonitor の設定

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

| パラメータ名 | 意味 | 1 | 2 |
|-----------------|---|---|---|
| LOG_LEVEL | ログの出力レベルを指定します。指定できる値の範囲は 0～3 の整数値です。省略時は上位のステートメントに指定した値が設定されます。 LOG_LEVEL については、「3.2.8 ログレベルの設定」を参照してください。 | | |
| MONITOR_USER | ディスク領域監視モニタ(fdsmond)の実効ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| 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はシステム障害として扱われます。指定できる値の範囲は 1～3600 です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| ORACLE_USER | ディスク監視モニタ(fdsmond)で監視接続を行うための Oracle ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_USER については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| ORACLE_PASS | ディスク領域監視モニタ(fdsmond)で監視接続を行うための Oracle ユーザのパスワードを指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_PASS については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| SYSDBA_CONNECT | ORACLE_USER を指定した際、SYSDBA 権限で接続するかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 SYSDBA_CONNECT については、「3.2.19 パスワード認証環境での監視」を参照してください。 | | |
| 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 YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 MONITOR_CONTROL については、「3.2.13 監視モニタ制御の設定」を参照してください。 | | |
| POLL_INTERVAL | 監視間隔を秒で指定します。指定できる値の範囲は 5～2678400 の整数です。省略時は 3600 秒に設定されます。 | | |
| POLL_TIMEOUT | 全ての検査が完了するまで、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。指定できる値の範囲は 1～86400 の整数です。省略時は 60 秒に設定されます。 | | |
| COMMAND_NAME | ディスク空き領域率が指定した閾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。存在しないコマンドを指定した場合はエラーとなります。省略時はコマンドの実行を行いません。 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。 | | |
| FREE_SPACE | ディスク空き領域率の閾値を割合で指定します。指定できる値の範囲は 0～100 の整数です。省略時は 10%です。 | | |

3.2 設定ファイル(oramond.conf)の作成と適用

- 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)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については、「第4章 ApplicationMonitor の動作」を参照してください。

第 3 章 ApplicationMonitor の設定

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

| パラメータ名 | 意味 | 1 | 2 |
|-----------------|---|---|---|
| LOG_LEVEL | ログの出力レベルを指定します。指定できる値の範囲は 0～3 の整数値です。省略時は上位のステートメントに指定した値が設定されます。 LOG_LEVEL については、「3.2.8 ログレベルの設定」を参照してください。 | | |
| MONITOR_USER | 統計情報採取モニタ(statsmond)の実効ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | × |
| 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はシステム障害として扱われます。指定できる値の範囲は 1～3600 です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| ORACLE_USER | 統計情報採取モニタ(statsmond)で採取接続を行うための Oracle ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_USER については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| ORACLE_PASS | 統計情報採取モニタ(statsmond)で採取接続を行うための Oracle ユーザのパスワードを指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_PASS については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| SYSDBA_CONNECT | ORACLE_USER を指定した際、SYSDBA 権限で接続するかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 SYSDBA_CONNECT については、「3.2.19 パスワード認証環境での監視」を参照してください。 | | |
| 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 YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 MONITOR_CONTROL については、「3.2.13 監視モニタ制御の設定」を参照してください。 | | |
| POLL_INTERVAL | 採取間隔を秒で指定します。指定できる値の範囲は 5～2678400 の整数です。省略時は 3600 秒に設定されます。 | | |
| POLL_TIMEOUT | 全ての統計情報(Stats ステートメント)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。指定できる値の範囲は 1～86400 の整数です。省略時は 60 秒に設定されます。 | | |
| 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 YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 GET_STATS については、「3.2.21 統計情報採取の設定」を参照してください。 | | |

- 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)が可能かどうかを示します。可能な場合は、不可能な場合は×です。

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

個々のパラメータの関係については、「第4章 ApplicationMonitor の動作」を参照してください。

第 3 章 ApplicationMonitor の設定

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

| パラメータ名 | 意味 | 1 | 2 |
|----------------|---|---|---|
| LOG_LEVEL | ログの出力レベルを指定します。指定できる値の範囲は 0～3 の整数値です。省略時は上位のステートメントに指定した値が設定されます。 LOG_LEVEL については、「3.2.8 ログレベルの設定」を参照してください。 | | |
| MONITOR_USER | システムステートダンプ採取デーモン(ssdump)の実効ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 | | x |
| 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はシステム障害として扱われます。指定できる値の範囲は 1～3600 です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| ORACLE_USER | システムステートダンプ採取デーモン(ssdump)で採取接続を行うための Oracle ユーザ名を指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_USER については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| ORACLE_PASS | システムステートダンプ採取デーモン(ssdump)で採取接続を行うための Oracle ユーザのパスワードを指定します。省略時は上位のステートメントに指定した値が設定されます。 ORACLE_PASS については、「3.2.16 OracleDBA 監査(audit)機能対応の設定」を参照してください。 | | |
| SYSDBA_CONNECT | ORACLE_USER を指定した際、SYSDBA 権限で接続するかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 SYSDBA_CONNECT については、「3.2.19 パスワード認証環境での監視」を参照してください。 | | |
| 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。指定できる値の範囲は 5～2678400 の整数です。省略時は 60 秒に設定されます。 | | |
| 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。指定できる値の範囲は 1～86400 の整数です。省略時は 60 秒に設定されます。 | | |
| GET_DUMP | システムステートを採取するかどうかを YES または NO で指定します。省略時は YES に設定されます。 | | |
| DUMP_INTERVAL | システムステートダンプの採取間隔を秒で指定します。指定できる値の範囲は 1～60 です。省略時は 30 秒に設定されます。 | | |
| DUMP_COUNT | システムステートダンプの採取回数を指定します。指定できる値の範囲は 1～10 の整数です。省略時は 3 回に設定されます。 | | |
| DUMP_LEVEL | 採取するシステムステートダンプのレベルを指定します。指定できる値の範囲は 0～266 です。省略時は 10 に設定されます。 | | |
| DUMP_TIMEOUT | システムステートダンプを採取する時間の上限(秒)を指定します。指定できる値の範囲は 1～3600 の整数です。省略時は 90 秒に設定されます。 このパラメータは全ノードステートメント内の SystemStateDaemon ステートメントにのみ記述できます。 | | |

3.2 設定ファイル(oramond.conf)の作成と適用

- 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)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については、「第4章 ApplicationMonitor の動作」を参照してください。

第 3 章 ApplicationMonitor の設定

Table ステートメントに記述可能なパラメータは以下のとおりです。

| パラメータ名 | 意味 | 1 | 2 |
|----------------|--|---|---|
| TABLE_USER | 表にアクセス可能なユーザ名を指定します。指定できる文字列は半角で最大 30 文字です。ユーザ名に全角文字、スペース、タブ、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。 このパラメータは省略できません。 | × | |
| TABLE_PASSWORD | 表にアクセス可能なユーザ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。 このパラメータは省略できません。 TABLE_PASSWORD については、「3.2.17 Oracle パスワードの扱いについて」を参照してください。 | × | |

- 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)が可能かどうかを示します。可能な場合は、不可能な場合は×です。

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

| パラメータ名 | 意味 | 1 | 2 |
|----------------------|--|---|---|
| USEABLE_EXTENT_BLOCK | 使用可能エクステントの閾値をブロック数で指定します。指定できる値の範囲は 0 ~ 1048576 の整数です。0 を指定した場合は使用可能エクステントサイズの監視を行いません。省略時は 0 です | | |
| FREE_SPACE | 空き領域率の閾値を割合で指定します。指定できる値の範囲は 0 ~ 100 の整数です。省略時は 20%です。 | | |
| GET_STATS | 統計情報の採取を行うかどうかを YES または NO で指定します。省略時は上位のステートメントに指定した値が設定されます。 GET_STATS については、「3.2.21 統計情報採取の設定」を参照してください。 | | |

- 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)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 設定ファイル(oramond.conf)の作成と適用

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

| パラメータ名 | 意味 | 1 | 2 |
|--------------|--|---|---|
| COMMAND_NAME | ディスク空き領域率が指定した閾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。存在しないコマンドを指定した場合はエラーとなります。省略時はコマンドの実行を行いません。 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。 | | |
| FREE_SPACE | ディスク空き領域率の閾値を割合で指定します。指定できる値の範囲は0~100の整数です。省略時は上位のステートメントに指定した値が設定されます。 | | |

- 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)が可能かどうかを示します。可能な場合は、不可能な場合は×です。

Stats ステートメントに記述可能なパラメータは以下のとおりです。

| パラメータ名 | 意味 | 1 | 2 |
|---------------|--|---|---|
| POLL_INTERVAL | 採取間隔を秒で指定します。指定できる範囲は5~2678400の整数です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| 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。指定できる範囲は1~86400の整数です。省略時は上位のステートメントに指定した値が設定されます。 | | |
| GET_STATS | 統計情報の採取を行うかどうかをYESまたはNOで指定します。省略時は上位のステートメントに指定した値が設定されます。 GET_STATS については、「3.2.21 統計情報採取の設定」を参照してください。 | | |

- 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)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。

第3章 ApplicationMonitor の設定

3.2.8. ログレベルの設定

ログの出力レベルは LOG_LEVEL パラメータに指定します。ログの出力レベルは以下のとおりです。また、syslog に出力されるレベルとの関係を示します。

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

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

出力されるメッセージについては、『CLUSTERPRO MC ApplicationMonitor 1.0 for Linux メッセージ一覧』を参照してください。

3.2.9. ノード名の設定

監視対象の Oracle が動作するノード名を NODE_NAME パラメータに指定します。ノード名には IP アドレスを用いることもできます。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 の場合以下のように記述します。

(例)

```
NODE_NAME=node1  
NODE_NAME=node2
```

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

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

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

設定ファイルに記述された 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については、「3.2.10 停止ノード選択ポリシーの設定 (RAC 構成時のみ)」を参照してください。

3.2.10. 停止ノード選択ポリシーの設定 (RAC 構成時のみ)

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停止ノードを選択します。本パラメータでは、複数ノードで発生した障害の重要度が同じ場合に、停止するノードの選択条件を指定することができます。

この指定は、HALT_CONTROL_POLICY パラメータで行います。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(oramond)が最初に障害を検出したノードが選択されます。 |
| HALT_CONTROL_POLICY=CONFIGURED | 設定ファイルに記述される NODE_NAME の優先順位に従います。 |

複数ノード障害とは、モニタ制御デーモン(oramond)が1つ目のノードで障害を検出してから、障害ノードの停止処理を開始するまでの間に2つ目のノードで障害を検出した場合を指します。

HALT_CONTROL_POLICY パラメータに CONFIGURED が設定されている場合、障害ノードのうち、NODE_NAME パラメータの上の方に記述されているノードが停止ノードとして選択されます。

3.2.11. 障害ノードの停止動作の設定

障害が発生したノードの停止動作を指定することができます。この指定は、HALT_CONTROL_OPTION パラメータで行います。指定できる値は以下のとおりです。

| | |
|-----------------------------|--|
| HALT_CONTROL_OPTION=ALWAYS | 常に停止処理が行われます。 |
| HALT_CONTROL_OPTION=NEVER | 停止処理を行いません。 |
| HALT_CONTROL_OPTION=OUTLIVE | 他に生存しているノードが存在する場合、ALWAYS として扱われます。他に生存しているノードが存在しない場合、NEVER として扱われます。 |

第3章 ApplicationMonitor の設定

3.2.12. Oracle 構成プロセス停止の設定

ノード停止時に Oracle の構成プロセスに対して kill コマンド (SIGTERM) を実行するかどうかを指定します。この指定は HALT_CONTROL_METHOD パラメータで行います。指定できる値は以下のとおりです。

| | |
|--------------------------|---|
| HALT_CONTROL_METHOD=KILL | Oracle の構成プロセスに対して kill (SIGKILL) を実行します。 |
| HALT_CONTROL_METHOD=NONE | 何もしません。 |

3.2.13. 監視モニタ制御の設定

リスナ監視モニタ (lsnrmond)、表領域監視モニタ (tschkmond)、ディスク領域監視モニタ (fdsmond) 及び統計情報採取モニタ (statsmond) をモニタ制御デーモン (oramond) から制御 (モニタの起動、停止、中断、再開) を行うかどうかを指定します。この指定は MONITOR_CONTROL パラメータで行います。指定できる値は以下のとおりです。

| | |
|---------------------|-----------------|
| MONITOR_CONTROL=YES | 監視モニタの制御を行います。 |
| MONITOR_CONTROL=NO | 監視モニタの制御を行いません。 |

NO に指定した場合は、モニタ制御デーモン (oramond) による制御が行われないため、リスナ監視モニタ、表領域監視モニタ、ディスク領域監視モニタ及び統計情報採取モニタを単独で起動し監視を行うことができます。単独で起動する場合には、「3.6 リスナ監視モニタの単独起動」、「3.7 表領域監視モニタの単独起動」、「3.8 ディスク領域監視モニタの単独起動」または「3.9 統計情報採取モニタの単独起動」を参照してください。

3.2.14. インスタンス / リスナ障害時のグループ停止の設定

インスタンス / リスナの障害時に、RESTART_COUNT の回数再起動を行っても再起動できなかった場合、グループの停止を行うかどうか指定します。この指定は SERVICE_DOWN パラメータで行います。指定できる値は以下のとおりです。

| | |
|-----------------------|----------------------------|
| SERVICE_DOWN=YES | グループの停止を行います。 |
| SERVICE_DOWN=NO | グループの停止を行いません。 |
| SERVICE_DOWN=NO_DOWN | DOWN 障害の場合、グループの停止を行いません。 |
| SERVICE_DOWN=NO_STALL | STALL 障害の場合、グループの停止を行いません。 |

この機能は、リスナ監視モニタ (lsnrmond) がモニタ制御デーモン (oramond) により制御されているか、単独起動しているかに関わらず有効です。また、OCW 監視でも障害検出時の動作について SERVICE_DOWN パラメータにより同様の設定が可能です。

3.2.15. リスナ障害時の障害情報採取の設定

リスナの無応答障害発生時に、障害解析のため、tnsping の core を採取することができます。

tnsping の core を採取するためには、/var/opt/HA/AM/<構成番号>/lsnrmond/<リスナ名>/.tnsping_core を作成します。

複数のリスナ監視を行う場合は、監視対象のリスナ毎に、.tnsping_core ファイルを作成する必要があります。

以下のコマンドを実行することにより、作成してください。

```
# touch /var/opt/HA/AM/<構成番号>/lsnrmond/<リスナ名>/.tnsping_core
```

tnsping の core は、/var/opt/HA/AM/<構成番号>/lsnrmond/<リスナ名> 配下に、core.<pid>の名称で採取されます。

注意： 採取されたcoreは、自動的に削除されることはありません。採取するよう設定した場合、ディスク容量を圧迫することがないように、必要に応じて削除してください。

また、tnspingのpidが以前のものと同重複した場合、ファイルが上書きされる可能性があります。障害発生時は速やかに、/var/opt/HA/AM/<構成番号>/lsnrmond/<リスナ名> 配下のファイルを退避し、障害情報の採取を行ってください。

3.2.16. OracleDBA 監査(audit)機能対応の設定

Oracle の設定で SYS の監査ログ取得が有効化されている場合、SYS ユーザでの接続監視では、監査ログが監視毎に出力され、大量のログファイルが生成されてしまうという現象が発生しますが、ORACLE_USER パラメータ、ORACLE_PASS パラメータの双方を設定することで、SYS ではなく、任意の Oracle ユーザで監視接続を行えるようになります。

ORACLE_USER パラメータ及び ORACLE_PASS パラメータは、oraadmin コマンドの設定ファイルのチェックにより検査されます。正確なチェックを行うため、監視対象の Oracle が動作している状態で、チェックを行うようにしてください。詳細については、「3.2.17 Oracle パスワードの扱いについて」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設定することはできません。

また、接続検査においてOracle未起動以外のエラーが検出された場合、syslogにエラー事由が出力されます。出力イメージについては、別冊の『CLUSTERPRO MC ApplicationMonitor 1.0 for Linux メッセージ一覧』を参照してください。

注意： 任意のOracleユーザで監視接続を行っている場合でも、設定変更（apply）時に監査ログが出力されることがあります。その場合は、OracleのAUDIT_TRAIL初期化パラメータを“none”に設定することで監査ログが出力されなくなります。初期化パラメータを変更できない際は、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。監視表については、「3.1 監視を行うための表を作成する」を参照してください。

3.2.17. Oracle パスワードの扱いについて

Oracle の設定で、パスワードの大/小文字の区別を有効化している場合、AM のパラメータ値に小文字を含める際は、ダブルクォーテーション(“)で囲んでください。Oracle パスワードについての詳細は、『Oracle Database セキュリティ・ガイド』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について（RAC 環境のみ）

Oracle 11g（11.1.0.7）より、RAC 固有プロセスのストール監視機能（本書では Oracle ハング検知機能と呼びます）が追加されました。AM では、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。この指定は、ORACLE_HANG_CHECK パラメータで指定します。利用する場合は YES、利用しない場合は NO を指定してください。

3.2.19. パスワード認証環境での監視

パスワード認証での SYSDBA 権限による監視を行います。この方法で監視を行うためには ORACLE_USER、ORACLE_PASS、SYSDBA_CONNECT パラメータを指定します。SYSDBA 権限で Oracle に接続する場合は SYSDBA_CONNECT パラメータに YES を指定してください。

3.2.20. 安定したリスナ監視の実現

リスナや動作環境などが不安定で、障害リスナを一度再起動して復旧させても、しばらくして障害再発する（UP/DOWN を繰り返す）ようなケースがあります。このような環境で、累積障害検出回数に一定の閾値を設け、障害状態と確定させることができます。この指定は UP_DOWN_RETRY、UPTIME_COUNT により行います。

リスナ障害が発生し、再起動/復旧させても障害検出回数は加算され、障害検出回数が UP_DOWN_RETRY を超えた場合、障害と判断します。

監視結果が UP の状態が UPTIME_COUNT 回続くと、安定稼働状態と見なし、記憶していた障害検出回数をクリアします。

3.2.21. 統計情報採取の設定

インスタンス監視モニタ(instmond)、リスナ監視モニタ(lsnrmond)、表領域監視モニタ(tschkmond)及び統計情報採取モニタ(statsmond)の情報採取を行うかどうかを指定します。この指定は GET_STATS パラメータで行います。指定できる値は以下のとおりです。

| | |
|---------------|----------------|
| GET_STATS=YES | 統計情報の採取を行います。 |
| GET_STATS =NO | 統計情報の採取を行いません。 |

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 |
| リスナ監視モニタ | 監視 TAT 情報 ・監視対象リスナ名 ・監視 TAT |
| 表領域監視モニタ | 表領域情報 ・表領域名 ・空き領域のサイズ(ブロック数) ・総サイズ(ブロック数) ・空き容量率(%) |
| 統計情報採取モニタ | SGA 情報 ・空き容量サイズ(ブロック数) ・総サイズ(ブロック数) ・空き容量率(%) |

また、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。

第3章 ApplicationMonitor の設定

| | |
|---------|--|
| 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
リスナ監視モニタ : リスナ名
表領域監視モニタ : 表領域監視モニタ ID
統計情報採取モニタ : 統計情報採取モニタ ID

注意： 統計情報採取モニタを指定した際、採取する統計情報がない（全ての GET_STATSパラメータをNOに設定している）場合は、エラー事由が出力されます。出力イメージについては、別冊の『CLUSTERPRO MC ApplicationMonitor 1.0 for Linux メッセージ一覧』を参照してください。

採取した情報は、先頭に情報採取日時を付与し、CSV 形式でファイルに出力されます。それぞれの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

フォーマット：情報採取日時, SID 名, 監視 TAT
出力イメージ：2012/10/12 08:23:40, rac1, 1

リスナ監視モニタ

フォーマット：情報採取日時, リスナ名, 監視 TAT
出力イメージ：2012/10/12 08:23:40, LISTENER, 1

表領域監視モニタ

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：2012/10/12 08:23:40, USERS, 152, 1600, 9

統計情報採取モニタ

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：2012/10/12 08:23:40, SGA_SPACE, 345529792.00, 603979776.00, 57.00

3.3. Oracle エラーのフィルタリング設定

AM は、インスタンス監視モニタ(instmond)、リスナ監視モニタ(lsnrmond)の監視時に発生する Oracle エラー毎に障害として扱う（エラー）または障害として扱わない（許可エラー）を選択することができます。

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（テキストファイル）に記述することで設定します。

| | |
|---------------------------|-------------------------------|
| インスタンス監視モニタのフィルタリング設定ファイル | /etc/opt/HA/AM/conf/inst.conf |
| リスナ監視モニタのフィルタリング設定ファイル | /etc/opt/HA/AM/conf/lsnr.conf |

インスタンス監視モニタの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
```

リスナ監視モニタの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。

```
#lsnrmond error check configuration file#

ERROR_CODE_TYPE = ERROR_CODE

-12541   #TNS:no listener.
```

第 3 章 ApplicationMonitor の設定

3.3.1. フィルタリング設定ファイルの記述形式

フィルタリング設定ファイルの記述形式は、インスタンス監視モニタ(instmond)、リスナ監視モニタ(lsnrmond)共に共通です。

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、フィルタリングを行う Oracle エラー番号から構成されます。

- ・コメント

コメントはシャープ記号(#)から始まり、行の最後までとなります。

- ・ ERROR_CODE_TYPE パラメータ

ERROR_CODE_TYPE パラメータは、このパラメータ以降に記述する Oracle エラー番号に対し、障害として扱う (ERROR_CODE) Oracle エラー番号なのか障害として扱わない (ALLOWABLE_CODE) Oracle エラー番号なのかを設定します。記述形式は以下のとおりです。

記述形式：

```
ERROR_CODE_TYPE = [ERROR_CODE|ALLOWABLE_CODE]
```

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、デフォルト設定は全て無効となり、フィルタリング設定ファイルの内容が使用されます。

注意： フィルタリング設定ファイルに不正がある場合、フィルタリング設定ファイルの内容は全て無効となり、デフォルト設定が使用されます。syslog にデフォルト設定を使用する旨のメッセージが出力されます。

注意： ERROR_CODE_TYPE パラメータは、必ず Oracle エラー番号を記述する前の行になければなりません。

- ・ Oracle エラー番号

フィルタリングを行う 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

セッション数を超過しました」の場合、Oracle エラー番号は、-18 と記述します。

3.3.2. フィルタリングのデフォルト設定

インスタンス監視モニタ(instmond)のデフォルト設定は以下のとおりです。インスタンス監視モニタの場合、以下のエラーコードを障害として扱わない(許可エラー)に設定されています。

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

リスナ監視モニタ(lsnrmond)のデフォルト設定は以下のとおりです。リスナ監視モニタの場合、以下のエラーコードを障害として扱う(エラー)に設定されています。

| エラーコード | メッセージ |
|-----------|-----------------|
| TNS-12541 | TNS: リスナがありません。 |

第 3 章 ApplicationMonitor の設定

3.4. 情報採取スクリプト(oradiag)の設定

AM をインストールすると、情報採取スクリプト(oradiag)が/opt/HA/AM/bin 配下にインストールされます。情報採取スクリプトを使用することにより、/var/opt/HA/AM/<構成番号>/oradiag ディレクトリに障害解析情報が採取されます。

障害検出時にモニタ制御デーモン(oramond)が自動で情報採取スクリプトを実行する場合は、設定ファイル(oramond.conf)の SCRIPT_NAME パラメータに情報採取スクリプトの絶対パス名を指定してください。

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

情報採取スクリプトは対象とするモニタ制御デーモンの構成番号 (COMPONENT_ID) を自動的に認識し動作します。

また、自ノード上のインスタンスのステータスに合わせた障害後処理を行う場合は、DOWN_sequence 及び STALL_sequence の各々のシェル関数内に任意の追加処理を記述することで、DOWN に遷移した場合には、DOWN_sequence を、STALL に遷移した場合には、STALL_sequence を実行することが可能になります。

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 |

SCRIPT_USER に root 以外のユーザ名を指定した場合は、必要な情報を全て採取することができませんので注意してください。

3.5. CLUSTERPRO の設定

CLUSTERPRO に AM を Oracle グループのリソースとして登録することで、障害検出時にグループを停止することができます。設定作業は CLUSTERPRO の Builder により行います。設定方法などの詳細については、「5.5 ApplicationMonitor の Oracle グループへの組み込み」や、CLUSTERPRO のマニュアル『インストール&設定ガイド』などを参照してください。

- ・ AM(oramond)を起動 / 停止するリソースを作成するため、Oracle のグループに EXEC リソースを追加します。設定時の主な注意点を記述します。
 - ・ リソースの起動 / 停止スクリプトに AM の起動 / 停止 (停止処理は oramond に SIGTERM を送ります) 処理を記述します。
 - ・ 起動スクリプトの 同期 / 非同期の選択を”非同期”とします。
 - ・ このリソースが依存するリソースとして Oracle の起動 / 停止をする EXEC リソースを選択します (Oracle の起動完了後に AM を起動させるためです) 。

- ・ AM をプロセス監視するリソースを作成するため、PID モニタリソースを追加します。設定時の主な注意点を記述します。
 - ・ 監視対象リソースとして、AMの起動 / 停止を行うEXECリソースを指定します。
 - ・ 回復対象として Oracle のグループを指定します。
 - ・ 再活性化しきい値に “0” を指定します。

同一ノード上で複数インスタンスの監視を行う場合は、それぞれのグループを作成します。この時、EXEC リソースから起動するモニタ制御デーモン(oramond)は構成番号により識別します。構成番号 (COMPONENT_ID) が 1 の場合、スクリプトには次のように指定します。

```
...
/etc/opt/HA/AM/lbin/oramond -C 1
...
```

モニタ制御デーモン(oramond)の起動オプションについては、「A コマンドリファレンス」の oramond を参照してください。

第 3 章 ApplicationMonitor の設定

RENS 連携する場合、以下の設定が必要になります。

- ・ AM を監視する RENS のプロセスを起動する必要があります。

RENS のモニタプロセス起動スクリプトファイルにプロセス起動処理を追加します。

例 (/opt/mcl/rens/script/monitor_run.sh)

```
${RENS_BIN}/hamon -n haam &
```

- ・ CLUSTERPRO の AM 起動用のリソースの起動スクリプトには、AM 起動処理を記述します。

例 (起動用スクリプト : oramond を監視する場合)

```
/etc/opt/HA/AM/lbin/oramond &
```

PID モニタリソースによる監視は行わない場合、EXEC リソースの起動スクリプトの設定は、“同期”でも構いません。その場合は、起動スクリプトが終了するように記述してください (上記のようにバックグラウンド指定するなど)。

RENS と連携することができるモニタは、oramond の他に lsnrmond (単独起動)、crsmond があります。

- ・ CLUSTERPRO の AM 起動用のリソースの停止スクリプトには、AM 停止処理を記述します。

例 (停止用スクリプト : oramond を監視する場合)

```
kill <oramond の PID>
```

各モニタの停止処処理方法は同じです (モニタプロセスに対して SIGTERM を送信します)。

- ・ RENS と連携する場合には、CLUSTERPRO の外部連携モニタの設定が必要です。設定時の主な注意点を記述します。

- ・ 監視タイミングとして、“活性時”を指定します。
- ・ 対象リソースとして、AM を起動する EXEC リソースを指定します。
- ・ カテゴリとして、“HA/AM”を指定します。

RENS の設定については、マニュアル『サーバ管理基盤セットアップガイド』、『サーバ管理基盤利用の手引き』を参照してください。

3.6. リスナ監視モニタの単独起動

リスナ監視モニタ(lsnrmond)は、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で実行してください。

リスナ監視モニタの起動方法の詳細については、「A コマンドリファレンス」の lsnrmond を参照してください。

リスナ監視モニタを単独起動させ、CLUSTERPRO と連携する場合は、以下のように設定を行うことで、リスナ障害時にグループの停止・切り替えが可能となります。設定方法などの詳細については、CLUSTERPRO のマニュアル『インストール&設定ガイド』などを参照してください。

- ・ lsnrmond を起動 / 停止するリソースを作成するため、リスナのグループに EXEC リソースを追加します。設定時の主な注意点を記述します。
 - ・ リソースの起動 / 停止スクリプトに lsnrmond の起動 / 停止 (停止処理は lsnrmond に SIGTERM を送ります) 処理を記述します。
 - ・ 起動スクリプトの 同期 / 非同期の選択を”非同期”とします。
 - ・ このリソースが依存するリソースとしてリスナの起動 / 停止をする EXEC リソースを選択します (リスナの起動完了後に lsnrmond を起動させるためです) 。

- ・ lsnrmond をプロセス監視するリソースを作成するため、PID モニタリソースを追加します。設定時の主な注意点を記述します。
 - ・ 監視対象リソースとして、lsnrmond の起動 / 停止を行う EXEC リソースを指定します。
 - ・ 回復対象としてリスナのグループを指定します。
 - ・ 再活性化しきい値に “0” を指定します。

第 3 章 ApplicationMonitor の設定

RENS 連携する場合、以下の点に注意が必要です。

- ・ RENS 連携時には、RENS モニタプロセス (hamon) の起動が必要です。手順については、「3.5 CLUSTERPROの設定」と同じです。
- ・ 単独起動の lsnrmond が RENS 機能を利用する場合、プロセスを実行するユーザは、root または RENS のグループに含まれるユーザである必要があります。lsnrmond を root 以外のユーザで実行する場合、または設定ファイルの”MONITOR_USER”パラメータに root 以外のユーザを指定する場合には、該当ユーザを”rens”グループにも所属させてください。

```
# usermod -a -G rens <ユーザ名>
```

RENS の設定については、マニュアル『サーバ管理基盤セットアップガイド』、『サーバ管理基盤利用の手引き』を参照してください。

3.7. 表領域監視モニタの単独起動

表領域監視モニタ(tschkmond)は、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。

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

| | |
|-----------------|------------------------|
| ORACLE_SID | Oracle のインスタンス識別子 |
| ORACLE_BASE | Oracle のベースディレクトリ |
| ORACLE_HOME | Oracle のホームディレクトリ |
| ORA_NLS10 | Oracle の言語ファイル格納ディレクトリ |
| NLS_LANG | Oracle で使用している言語 |
| LD_LIBRARY_PATH | Oracle のライブラリパス |

表領域監視モニタの起動方法の詳細については、「A コマンドリファレンス」の tschkmond を参照してください。

CLUSTERPRO などの監視リソースと連携する場合は、それぞれの監視リソースの設定方法及び登録方法を参照してください。

3.8. 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、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。

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

| | |
|-----------------|------------------------|
| ORACLE_SID | Oracle のインスタンス識別子 |
| ORACLE_BASE | Oracle のベースディレクトリ |
| ORACLE_HOME | Oracle のホームディレクトリ |
| ORA_NLS10 | Oracle の言語ファイル格納ディレクトリ |
| NLS_LANG | Oracle で使用している言語 |
| LD_LIBRARY_PATH | Oracle のライブラリパス |

ディスク領域監視モニタの起動方法の詳細については、「A コマンドリファレンス」の fdsmond を参照してください。

CLUSTERPRO などの監視リソースと連携する場合は、それぞれの監視リソースの設定方法及び登録方法を参照してください。

第3章 ApplicationMonitor の設定

3.9. 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、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。

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

| | |
|-----------------|------------------------|
| ORACLE_SID | Oracle のインスタンス識別子 |
| ORACLE_BASE | Oracle のベースディレクトリ |
| ORACLE_HOME | Oracle のホームディレクトリ |
| ORA_NLS10 | Oracle の言語ファイル格納ディレクトリ |
| NLS_LANG | Oracle で使用している言語 |
| LD_LIBRARY_PATH | Oracle のライブラリパス |

統計情報採取モニタの起動方法の詳細については、「A コマンドリファレンス」の statsmond を参照してください。

CLUSTERPRO などの監視リソースと連携する場合は、それぞれの監視リソースの設定方法及び登録方法を参照してください。

3.10. ProcessSaver との併用

ProcessSaver はプロセスを監視し、プロセス消滅時に自動で再起動を行うことができます。ProcessSaver の詳細については、ProcessSaver のマニュアルを参照してください。

ProcessSaver を併用し、Oracle の監視を行う場合は、以下の構成を推奨します。

- ・ AM によるリスナ監視
- ・ AM によるインスタンスの無応答障害の監視
- ・ ProcessSaver によるリスナ以外のプロセス消滅の監視

ProcessSaver との併用時は、以下の点に注意してください。

- ・ ProcessSaver により Oracle プロセスの再起動を行う場合、プロセス再起動中に AM が障害として検出する可能性があります。Oracle プロセスの再起動処理を行なう場合は、AM 側で障害と認識しないよう、AM の POLL_DOWN_RETRY パラメータの設定を調整してください。

3.11. CRS 監視設定ファイル(crsmond.conf)の作成

AMでOracleのOCWの無応答障害監視を行う場合は、CRS監視設定ファイル(crsmond.conf)を作成する必要があります。CRS監視設定ファイルには監視対象とするOCWデーモンのステータスを取得するコマンド(CRS_STATパラメータ)、CRS監視モニタの動作設定などを定義します。

AMをインストールすると、/etc/opt/HA/AM/conf/sampleディレクトリにCRS監視設定ファイルのサンプルが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。

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

CRS_STATパラメータはOracleの環境に依存しますので、環境を確認の上、必ず設定してください。

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。

CRS監視設定ファイルを作成後、CRS監視モニタの起動を行います。

CRS監視モニタの詳細は、「A コマンドリファレンス」のcrsmondを参照してください。

CLUSTERPRO連携を行う場合は、「3.12 OCWとApplicationMonitorの監視対象設定について」を参照してください。

3.11.1. CRS 監視設定ファイルの記述例

CRS監視設定ファイルは、コメント、パラメータにより構成されたテキストファイルです。全て半角英数文字で記述します。

設定は以下のように記述します(例)。

(例) OCW 監視設定例

```
# crsmond.conf

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

第 3 章 ApplicationMonitor の設定

3.11.2. コメントの記述形式

コメントは、シャープ記号(#)で始まり、行の最後までとなります。

3.11.3. パラメータの記述形式

パラメータは、CRS 監視モニタ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、タブを含める場合は、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。文字列内にダブルクォーテーション(")及び改行コードを使用することはできません。

1つのパラメータにつき1行(**改行コードを含め 1023 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。各種パラメータについては、「3.11.4 パラメータ一覧」を参照してください。

記述形式：

パラメータ名 = パラメータ値

3.11 CRS 監視設定ファイル(crsmond.conf)の作成

3.11.4. パラメーター一覧

CRS 監視設定ファイル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 | デフォルト値 | 1 | 2 |
|----------------|--|------------------------------|---|---|
| | | 値の範囲 | | |
| ORACLE_VERSION | 監視対象の OCW (Oracle Clusterware、Oracle Restart) のバージョンを指定します。バージョンにより監視方法が異なるため必ず指定します。 | なし 11G、11GR2 | × | |
| CRS_STAT | Oracle が提供する crsd/ohasd のステータス表示用コマンドの絶対パス (及び引数) を指定します。 ORACLE_VERSION パラメータの値により指定するコマンドが変わります。 11G : crs_stat 11GR2 : crsctl check has 必須パラメータです。OCW 監視を行う場合、省略することはできません。 | なし 引数上限は 31 まで | × | |
| LOG_LEVEL | syslog に出力するログレベルを指定します。 設定値とログレベルの関係は、「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 | CRS_STAT コマンドの実行間隔を指定します。 | 90 (秒) 10~86400 | | |
| POLL_TIMEOUT | CRS_STAT コマンドの実行タイムアウト値 (実行から無応答であると判断するまでの間隔) を指定します。 | 180 (秒) 10~86400 | | |
| RETRY_COUNT | エラーもしくは無応答である結果が返却された場合、crsd/ohasd を kill するまでの間に CRS_STAT コマンドを再試行する回数を指定します。 起動確認時 (crsd/ohasd を kill した後の起動確認も含む) に、確認回数が RETRY_COUNT をオーバーした場合、SERVICE_DOWN に指定されたアクションに基づいて動作が決定されます。 | 5 (回) 0~24 | | |
| WAIT_INTERVAL | crsd/ohasd を kill した際、init(8)による crsd/ohasd 再起動を待ち合わせる間隔を指定します。 | 180 (秒) 60~86400 | | |
| SERVICE_DOWN | 再試行回数が RETRY_COUNT をオーバーした場合のアクションを指定します。 指定できるアクションは以下の 4 つとなります。 YES : ノード停止します。 NO : ノード停止しません。 NO_DOWN : DOWN 障害の場合は停止しません。 NO_STALL : STALL 障害の場合は停止しません。 省略時は NO に設定されます。 SERVICE_DOWN については、「3.2.14 インスタンス/リスナ障害時のグループ停止の設定」を参照してください。 | NO 指定可能な値については、意味の項を参照のこと | | |
| CUSTOM_USER | CUSTOM_CMD が指定された場合に実行するコマンドの実行ユーザ名を指定します。 存在しないユーザ名を指定した場合はエラーとなります。 | root | | |
| CUSTOM_CMD | 障害検出時に実行するコマンド (パス及び引数) を指定します。 存在しないコマンドを指定した場合はエラーとなります。 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。 | なし 引数上限は 31 まで | | |

第3章 ApplicationMonitor の設定

- 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)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。

3.12. OCW と ApplicationMonitor の監視対象設定について

OCW は、デフォルトで、データベース、インスタンス、ノードアプリケーション等を監視し、監視対象のプロセス停止障害発生時に自動で再起動を行うことができます。OCW の詳細については、Oracle のマニュアル『Oracle Clusterware 管理およびデプロイメント・ガイド』を参照してください。

OCW + CLUSTERPRO + AM で OCW 監視を行う場合は、以下の構成を推奨します。

Oracle11gR1 の RAC 環境：

- ・ AM によるリスナ監視（ただし、Oracle vip を使用している vip 用リスナは除く）
- ・ AM によるインスタンス監視
- ・ OCW によるリソース監視（ただし、AM で監視しているリスナは除く）

上記構成において、インスタンス監視のみ、OCW と AM 両方からの監視となります。OCW がデフォルトの設定の場合、障害検出時にインスタンスの再起動を実施します。この場合、再起動中に AM が障害として検出する可能性があります。そのため、AM 側のパラメータの調整を行うか、もしくは OCW でインスタンスの再起動を抑止する設定を行う（推奨）ことにより、AM が障害と認識しないように設定してください。

設定例を以下に示します。

- ・ AM 側のパラメータを変更する場合

AM の POLL_DOWN_RETRY 及び POLL_STALL_RETRY のパラメータの設定値により、インスタンスの起動から DB の open までの間、障害と認識しないよう調整してください。

- ・ OCW による監視対象の再起動を行わない（OCW で監視対象を再起動しない）場合

監視対象の RESTART_ATTEMPTS パラメータを以下の手順で変更します。

1. **oracle** ユーザで、全てのインスタンスとノードアプリケーションを停止します。

```
% srvctl stop instance -d <dbname> -i <instname>  
% srvctl stop nodeapps -n <hostname>
```

第3章 ApplicationMonitor の設定

2. **root** ユーザで、再起動を行わない設定にする監視対象リソースに関する情報を OCR (Oracle Cluster Registry) からテキストに出力します。

```
# cd $ORA_CRS_HOME/bin
# ./crs_stat -p <CRS リソース名> > $ORA_CRS_HOME/crs/profile/<CRS リソース名>.cap
```

3. 上記で作成された .cap ファイルの RESTART_ATTEMPTS パラメータの値を 0 に修正します。

```
# ./crs_profile -update <CRS リソース名> -o ra=0
```

4. OCR に変更内容を登録します。

```
# ./crs_register <CRS リソース名> -u
```

5. 修正が有効かどうか確認します。

```
# ./crs_stat -p <CRS リソース名> | grep RESTART_ATTEMPTS
```

RESTART_ATTEMPTS が 0 に変更されたことを確認します。

6. **oracle** ユーザに戻り、ノードアプリケーション、インスタンスと他の必要なリソースを起動します。

```
% srvctl start nodeapps -n <hostname>
% srvctl start instance -d <dbname> -i <instname>
```

Oracle11gR2 の環境 :

- ・ AM によるリスナ監視 (ただし、Oracle scan-vip を使用している scan-vip 用リスナは除く)
- ・ AM によるインスタンス監視
- ・ OCW によるリソース監視

Oracle11gR2 では、OCW の状況に合わせて動作するため、対処は必要ありません。

3.12.1. OCW 監視の構築例

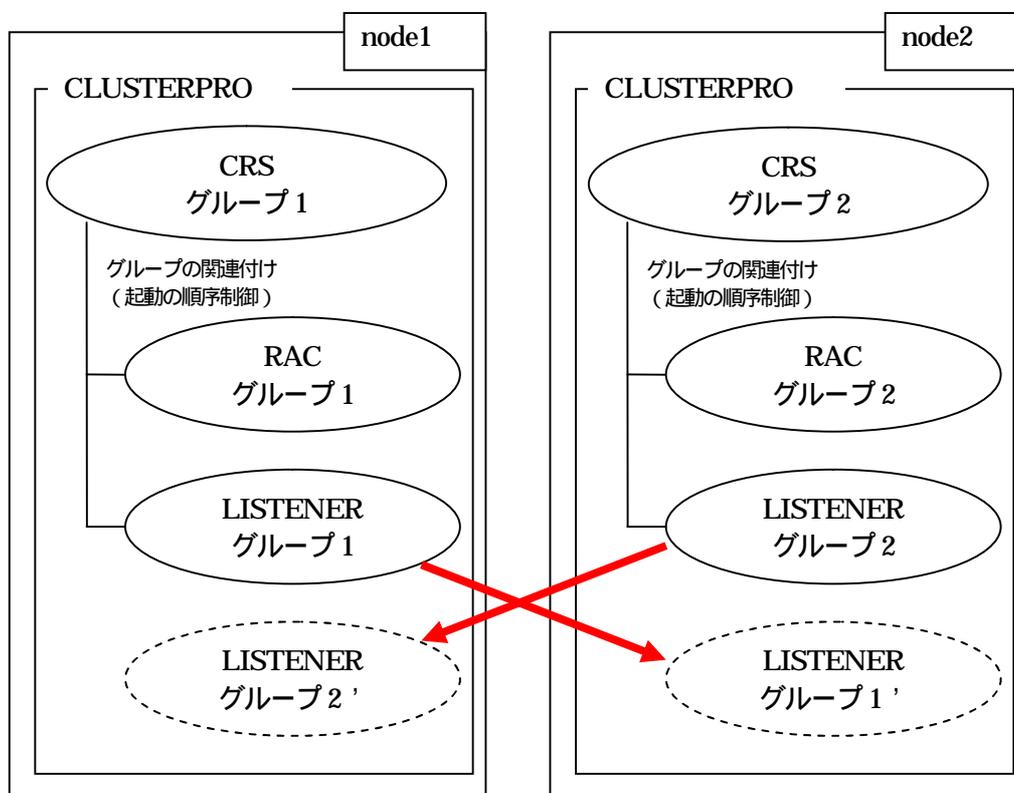


図 3-2 OCW 監視の構成例

AM において OCW 監視を行う場合は、上記構成例のように、RAC グループとは別に CRS グループを作成し、CRS グループのリソースとして、crsmond を登録します。

上記構成例における各グループの動作について、以下に記載します。

第 3 章 ApplicationMonitor の設定

3.12.2. CRS グループ

crsmond をリソースとし、OCW の起動 / 停止及び他のグループの起動停止の順序制御 () を行います。障害発生時にはグループを縮退します。

OCW が OS 起動の延長で自動起動されないように設定をする場合、root ユーザで以下を行います。

Oracle11gR1 の RAC 環境 :

```
# <$ORA_CRS_HOME> /bin/crsctl disable crs
```

Oracle11gR2 の環境 :

```
# <grid ホーム> /bin/crsctl disable has
```

起動順序は、リソースの依存関係があるため、

CRS グループ RAC グループ

停止順序は、

RAC グループ CRS グループ

となるように制御する必要があります。LISTENER グループは任意です。

自動で順序制御する場合は、CRS グループの起動処理に組み込む必要があります。手動またはシェルスクリプトなどを用いて順序制御する場合は、個々のグループ起動コマンドを実行する必要があります。

RAC グループでインスタンスを起動する場合は、OCW 起動の延長でインスタンスが自動起動しないよう、OCW が管理するリソースの AUTO_START パラメータを never に変更する必要があります。

変更例 : (以下の manual を automatic に変更すると、デフォルト値に戻ります)

```
% srvctl modify database -d <database_name> -y manual
```

- ・ crsmond を起動 / 停止するリソースを作成するため、CRS のグループに EXEC リソースを追加します。設定時の主な注意点を記述します。
 - ・ リソースの起動 / 停止スクリプトに crsmond の起動 / 停止処理 (停止処理は crsmond に SIGTERM を送ります) を記述します。
 - ・ 起動スクリプトの 同期 / 非同期の選択を”非同期”とします。
 - ・ このリソースが依存するリソースとして OCW の起動 / 停止をする EXEC リソー

スを選択します（OCW の起動完了後に crsmond を起動させるためです）。

- ・ crsmond をプロセス監視するリソースを作成するため、PID モニタリソースを追加します。設定時の主な注意点を記述します。
 - ・ 監視対象リソースとして、crsmond の起動 / 停止を行う EXEC リソースを指定します。
 - ・ 回復対象として CRS グループを指定します。
 - ・ 再活性化しきい値に “0” を指定します。

CRS グループ作成の詳細及び設定方法などの詳細については、「5.5 ApplicationMonitor の Oracle グループへの組み込み」や、CLUSTERPRO のマニュアル『インストール&設定ガイド』などを参照してください。

RENS 連携する場合、以下の点に注意が必要です。

- ・ RENS 連携時には、RENS モニタプロセス（hamon）の起動が必要です。手順については、「3.5 CLUSTERPROの設定」と同じです。

RENS の設定については、マニュアル『サーバ管理基盤セットアップガイド』、『サーバ管理基盤利用の手引き』を参照してください。

3.12.3. RAC グループ

oramond をリソースとし、インスタンスの起動 / 停止（ ）を行います。障害発生時にはグループを縮退します。

OCW がインスタンスを自動で起動する設定となっている場合は、ここでのインスタンス起動/停止は実行しないようにします。

RAC グループへのリソース登録の記述例は、「3.5 CLUSTERPROの設定」を参照してください。

3.12.4. LISTENER グループ

単独起動の lsnrmond をリソースとし、リロケータブル IP を使用するリスナの起動 / 停止を行います。障害発生時にはグループをフェイルオーバーします。

LISTENER グループへのリソース登録の記述例は、「3.6 リスナ監視モニタの単独起動」を参照してください。

第 3 章 ApplicationMonitor の設定

第4章 ApplicationMonitor の動作

4.1. ApplicationMonitor の起動と停止

モニタ制御デーモン(oramond)は、各ノード上で CLUSTERPRO から Oracle グループのリソースとして起動されます。モニタ制御デーモンは起動すると構成情報ファイル(oramond<構成番号>.bin)を読み込み、設定に従い、インスタンス監視モニタ(instmond)、リスナ監視モニタ(lsnrmond)、表領域監視モニタ(tschkmond)、ディスク領域監視モニタ(fdsmond)、リコンフィグ監視モニタ(altmond)、統計情報採取モニタ(statsmond)、システムステータス採取デーモン(ssdump)を起動(1、 2)します。

- (1) AM において、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、インスタンス監視モニタ、リコンフィグ監視モニタ、システムステータス採取デーモン以外の監視モニタは監視を行いません。
- (2) 監視対象がスタンバイ・データベースの場合、各監視モニタは Oracle の起動状態に合わせた監視を行います。

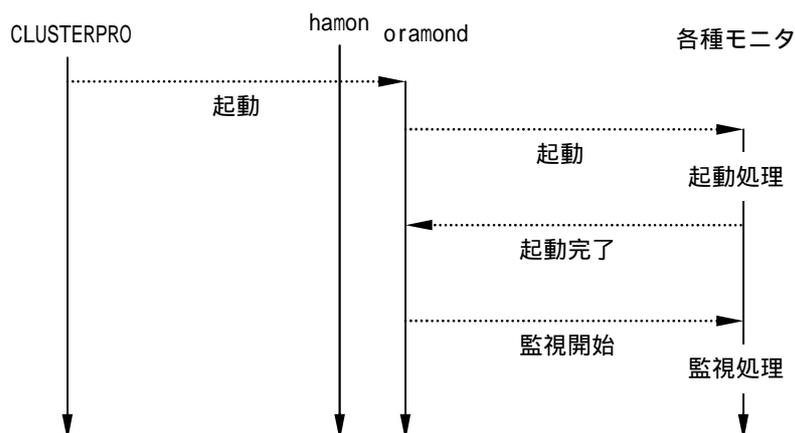


図 4-1 監視モニタ起動時の動作

モニタ制御デーモンは、各監視モニタから起動完了の通知を受け取ると、監視開始の通知を各監視モニタに送信します。これにより各監視モニタ及び各採取モニタは処理を開始します。

モニタ制御デーモンは SIGTERM を受信すると全ての監視モニタを終了させた後、正常に終了します。

注意： AMにおいて、CRS監視モニタは、モニタ制御デーモンにより起動及び停止されることはありません。OCW監視を行う場合は、別途CRS監視モニタを起動する必要があります。

監視モニタの起動に失敗した時は、以下の動作になります。

(例)

BOOT_TIMEOUT = 10

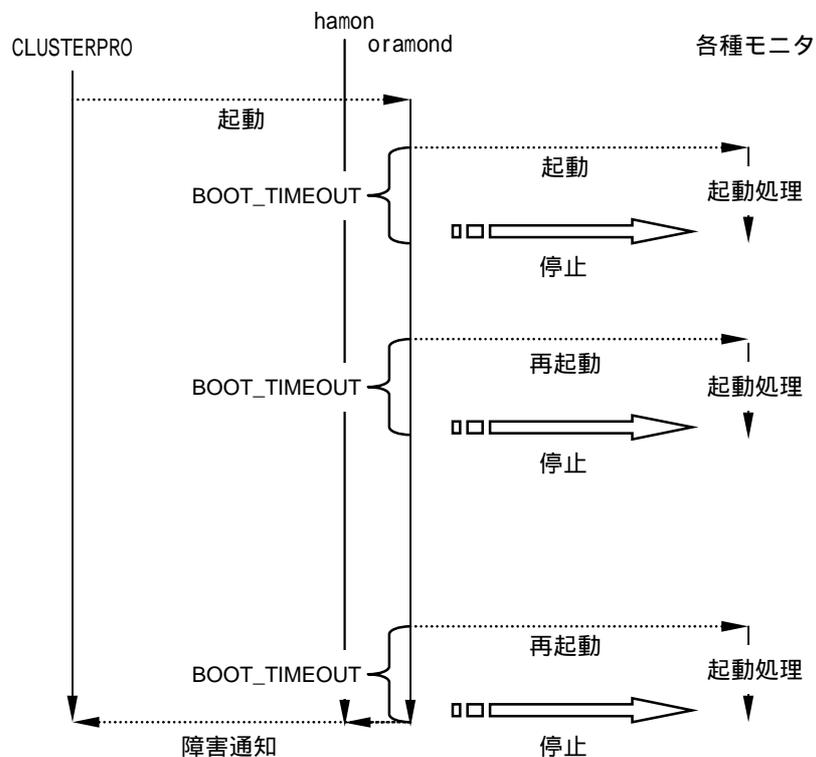


図 4-2 監視モニタ起動失敗時の動作

監視モニタの起動が BOOT_TIMEOUT に設定した時間までに完了しなかった場合、最大 5 回までの再起動を試みます。インスタンス監視モニタとリコンフィグ監視モニタ及び SERVICE_DOWN が YES に設定されているリスナ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。

RENS 連携する場合でも、AM が障害の監視 / 通知をする動作は基本的に変わりません。障害検出時、oramond 停止のタイミングで、CLUSTERPRO (PID 監視)、hamon (障害イベント) にそれぞれ障害検出したことが通知されます。

lsnrmond (単独起動時)、crsmond の CLUSTERPRO、RENS 連携時の動作についても同じです。

4.2. インスタンス監視時の動作

インスタンス監視モニタ(instmond)は起動すると、POLL_INTERVAL に設定した間隔でインスタンスの監視を行います。インスタンスの監視方法は V\$ 表の参照(1)、SYSTEM 表領域上のオブジェクトの更新 / 参照、ユーザ指定表の参照です(図中のインスタンス監視モニタのプロセス名は接尾語を除いたものとなっています)。

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及びデータベースが MOUNT 状態の場合は、インスタンスに対する起動確認のみ行います。

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

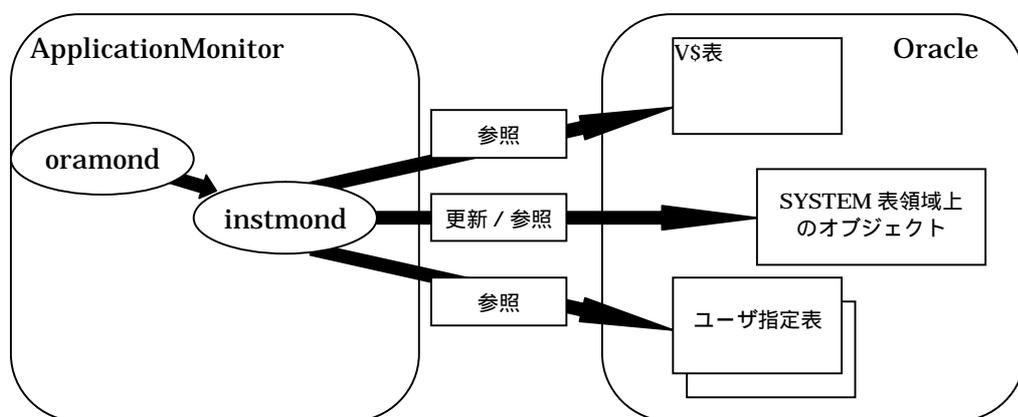


図 4-3 インスタンス監視モニタの概要

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 |
|-----------------|-------|---|
| 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(インスタンス毎に異なるオブジェクトを作成します)。 |
| 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(インスタンス毎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。 |

第4章 ApplicationMonitor の動作

インスタンスの監視動作は、以下のようになります。

(例)

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

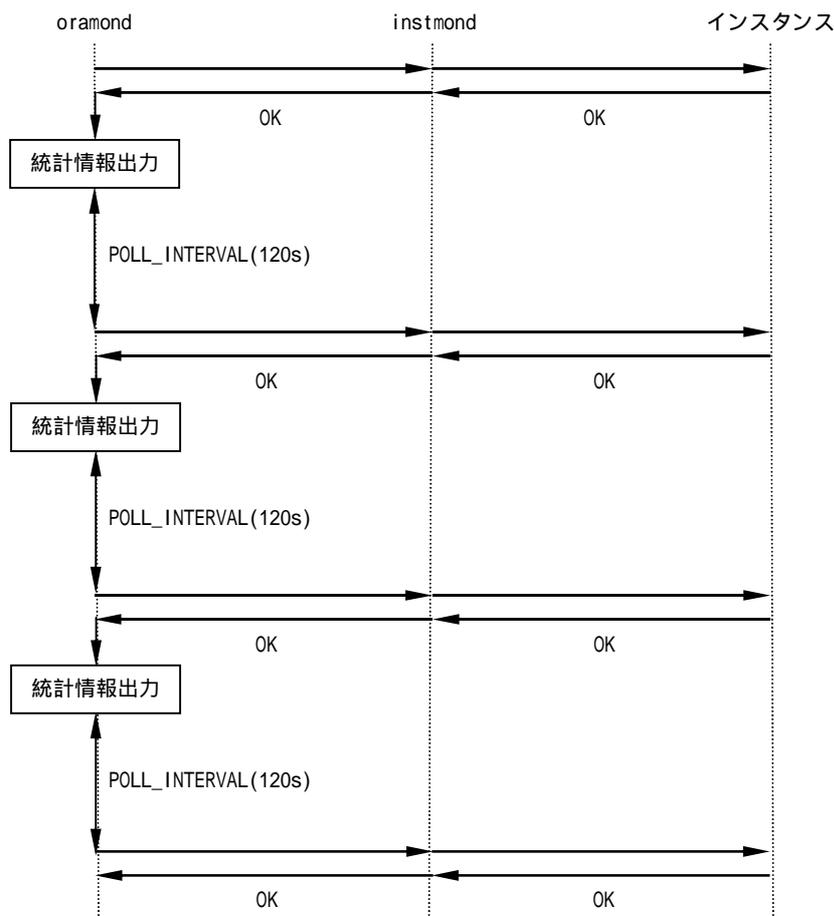


図 4-4 インスタンス監視時の動作

インスタンス監視モニタは **POLL_INTERVAL** 毎にインスタンスへアクセスします。インスタンスへのアクセスが正常の場合、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がエラーになった場合、以下のような動作になります。

(例)

```
InstanceMonitor {
  POLL_INTERVAL = 120
  POLL_DOWN_RETRY = 2
}
```

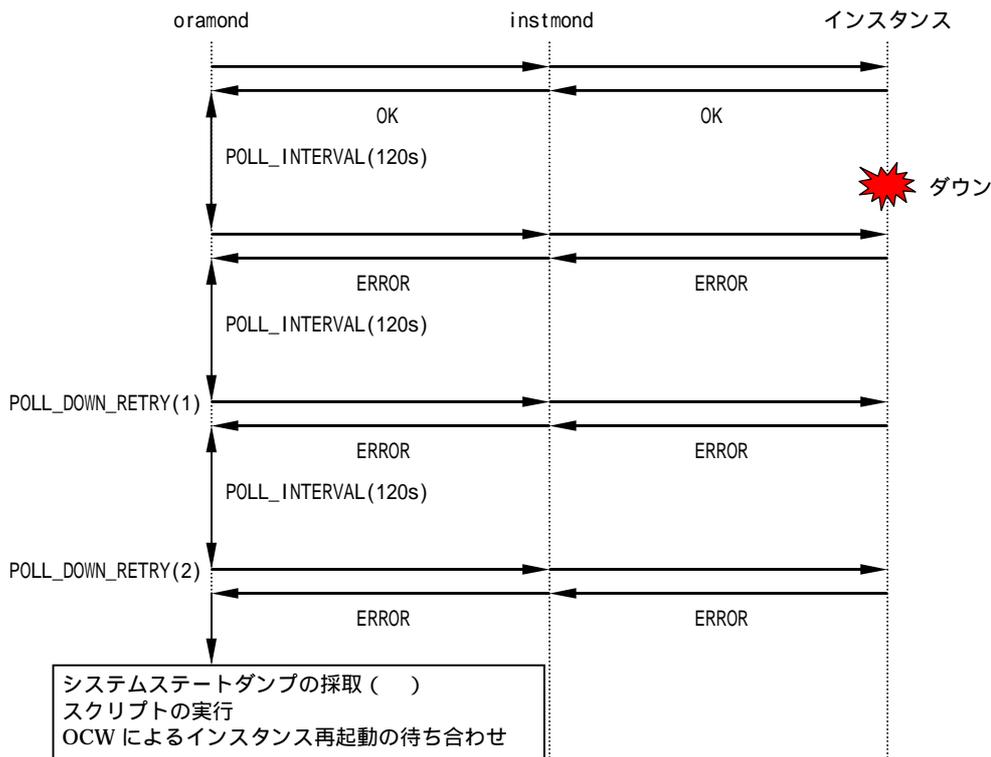


図 4-5 インスタンス監視エラー時の動作

インスタンスへのアクセスがエラーになった場合、**POLL_DOWN_RETRY** に 1 以上の値が設定されていると、**POLL_DOWN_RETRY** の回数だけリトライを行います。リトライすることでインスタンスへのアクセスが成功した場合、障害と見なしません。所定の回数リトライを行ってもインスタンスへのアクセスがエラーになる時は、システムステートダンプの採取 () スクリプトの実行へ処理が移ります。**POLL_DOWN_RETRY** に 0 が設定されている時は、インスタンスへのアクセスがエラーになるとすぐシステムステートダンプの採取 () スクリプトの実行へ処理が移ります。情報採取処理完了後、OCW によるインスタンス再起動を待ち合わせます (Oracle11gR2、OCW 環境のみ)。再起動が成功すれば監視を継続します。

障害ノードであることが決定されても、設定ファイルの **SERVICE_DOWN** パラメータの設定により、ノード停止処理を行うか判断されます。**SERVICE_DOWN** パラメータは、各

第 4 章 ApplicationMonitor の動作

インスタンス障害時にも同様に動作継続するかどうかの判断に使われます。

- () インスタンス監視エラー時のシステムステートダンプ採取は、RAC 構成時のみ非障害ノードにおいて採取されます。

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

(例)

```
InstanceMonitor {
  POLL_INTERVAL = 120
  POLL_TIMEOUT = 90
  POLL_STALL_RETRY = 1
}
```

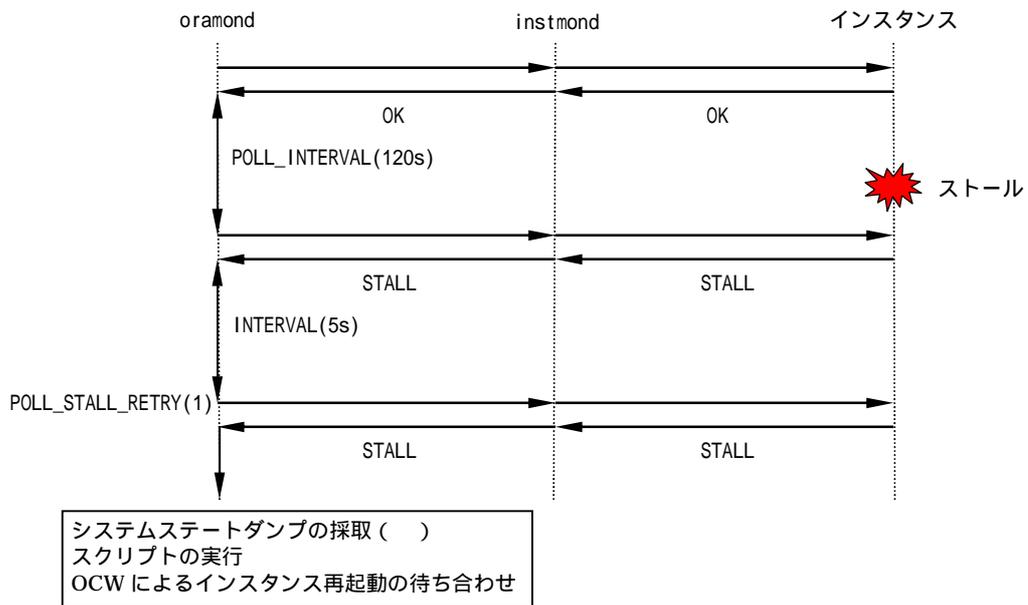


図 4-6 Oracle バックグラウンドプロセスの STALL 検知時の動作

第4章 ApplicationMonitor の動作

インスタンスの監視が POLL_TIMEOUT に指定した時間内に完了しない場合、以下のよ
うな動作になります。

(例)

```
InstanceMonitor {  
  POLL_INTERVAL    = 120  
  POLL_TIMEOUT     = 90  
  POLL_STALL_RETRY = 1  
}
```

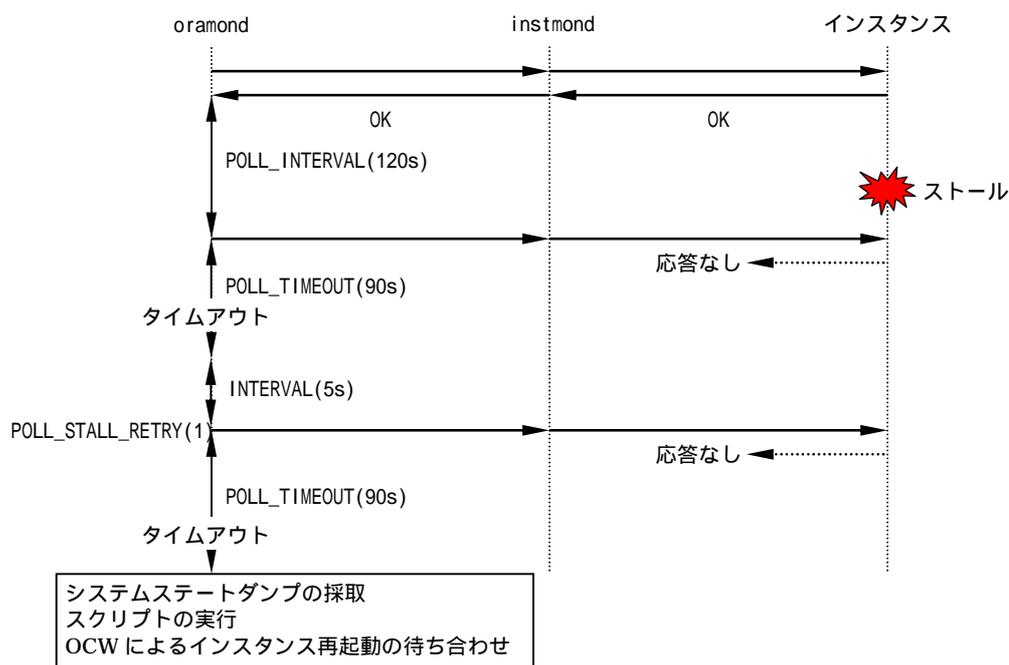


図 4-7 インスタンス監視タイムアウト時の動作

Oracle バックグラウンドプロセスのストールを検知した場合、またはインスタンスへのアクセスが POLL_TIMEOUT に設定した時間までに完了しなかった場合、POLL_STALL_RETRY に 1 以上の値が設定されていると、POLL_STALL_RETRY の回数だけリトライを行います。リトライすることでインスタンスへのアクセスが成功した場合、障害と見なしません。所定の回数リトライを行っても状態が改善しない（インスタンスへのアクセスがタイムアウトする、またはストールを検知する）時は障害となり、システムステートダンプの採取、スクリプトの実行へと処理が移ります。POLL_STALL_RETRY に 0 が設定されている時は、STALL 障害を検知（インスタンスへのアクセスがタイムアウトまたはストールを検知）するとすぐシステムステートダンプの採取、スクリプトの実行へと処理が移ります。情報採取処理完了後、OCW によるインスタンス再起動を待ち合わせます（Oracle11gR2、OCW 環境のみ）。再起動が成功すれば監視を継続します。

障害ノードであることが決定されても、設定ファイルの `SERVICE_DOWN` パラメータの設定により、ノード停止処理を行うか判断されます。`SERVICE_DOWN` パラメータは、各インスタンス障害時にも同様に動作継続するかどうかの判断に使われます。

第4章 ApplicationMonitor の動作

4.3. リスナ監視時の動作

リスナ監視モニタ(lsnrmond)は起動すると、POLL_INTERVAL に設定した間隔毎に、Oracle の tnsping コマンドを用いてリスナの状態を監視します。

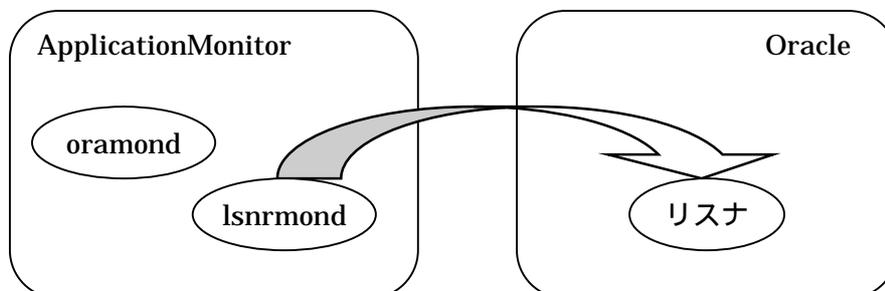


図 4-8 リスナ監視モニタの概要

リスナ監視モニタは POLL_INTERVAL の間隔でリスナを監視します。

(例)

```
ListenerMonitor LISTENER {  
  POLL_INTERVAL      = 60  
  NET_SERVICE_NAME   = ORA11G  
  GET_STATS          = YES  
}
```

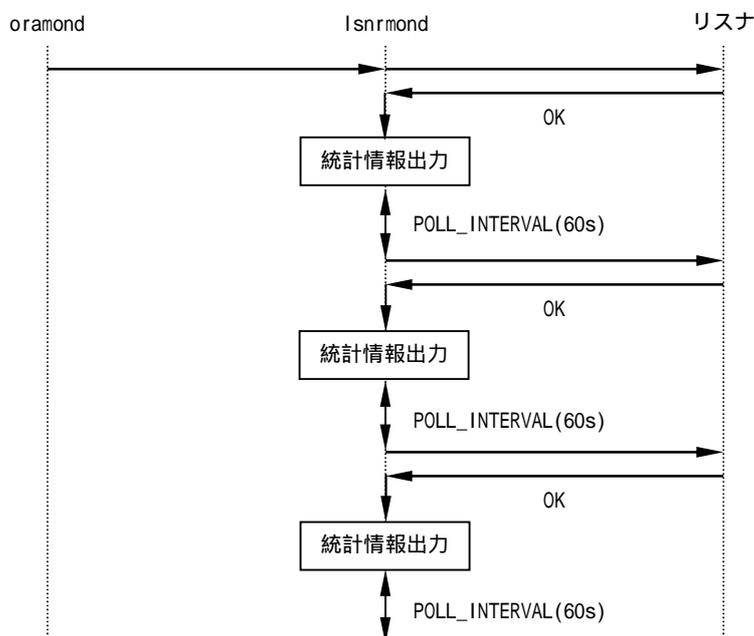


図 4-9 リスナ監視時の動作

リスナの監視がエラーになった場合、以下のような動作になります。

(AM がリスナ再起動を試行する)

(例)

```
ListenerMonitor LISTENER {
  POLL_INTERVAL    = 60
  RESTART_COUNT    = 2
  RESTART_INTERVAL = 3
  NET_SERVICE_NAME = LSNR
  TARGET_RESTART  = YES
}
```

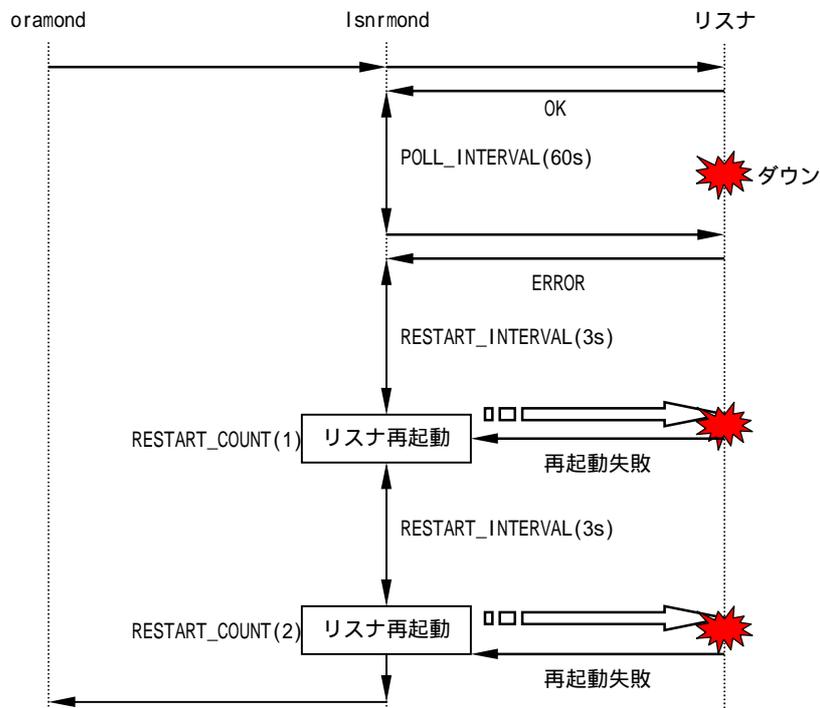


図 4-10 リスナ監視エラー時の動作

リスナ監視モニタは、リスナの障害を検出した場合、`RESTART_INTERVAL` の間隔で、`RESTART_COUNT` の回数だけリスナの再起動を試みます。再起動を行ってもリスナが起動できない場合は、モニタ制御デーモンへ障害を通知します。

障害ノードであることが決定されても、設定ファイルの `SERVICE_DOWN` パラメータの設定により、ノード停止処理を行うか判断されます。`SERVICE_DOWN` パラメータは、各リスナ障害時にも同様に動作継続するかどうかの判断に使われます。

第4章 ApplicationMonitor の動作

(OCW によるリスナ再起動を待ち合わせる)

(例)

```
ListenerMonitor LISTENER {  
  POLL_INTERVAL    = 60  
  NET_SERVICE_NAME = LSNR  
  TARGET_RESTART   = NO  
}
```

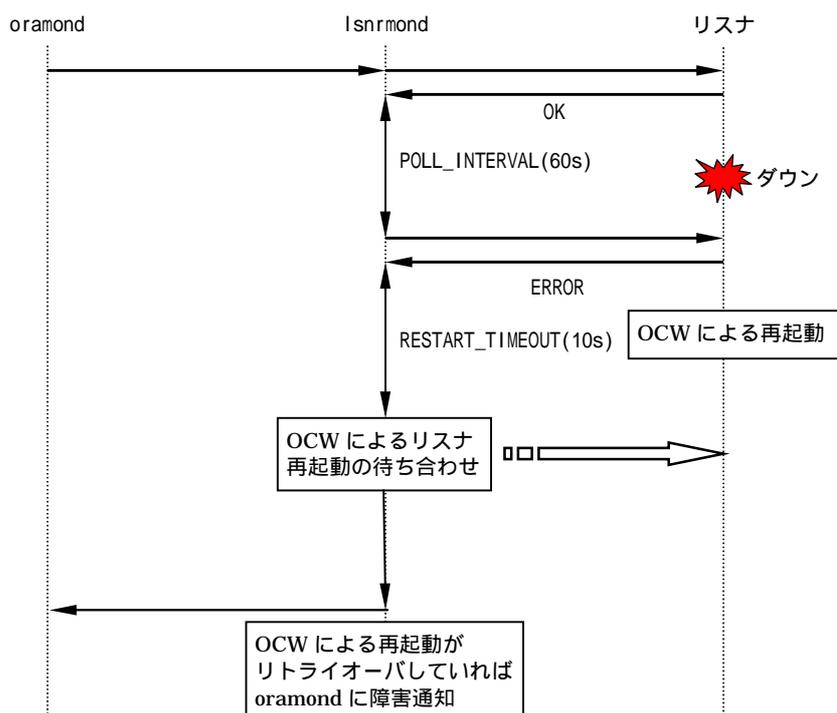


図 4-11 リスナ監視エラー時の動作

Oracle11gR2 OCW 環境でリスナの障害を検出した場合、OCW によるリスナ再起動を待ち合わせます。OCW によるリスナ再起動が成功していれば監視を継続します。再起動を行ってもリスナが起動できない場合は、モニタ制御デーモンへ障害を通知します。

障害ノードであることが決定されても、設定ファイルの **SERVICE_DOWN** パラメータの設定により、ノード停止処理を行うか判断されます。**SERVICE_DOWN** パラメータは、各リスナ障害時にも同様に動作継続するかどうかの判断に使われます。

リスナの監視がストールした場合、以下のような動作になります。

(例)

```
ListenerMonitor LISTENER {
  POLL_INTERVAL    = 60
  POLL_TIMEOUT     = 90
  RESTART_COUNT    = 2
  RESTART_INTERVAL = 3
  NET_SERVICE_NAME = LSNR
}
```

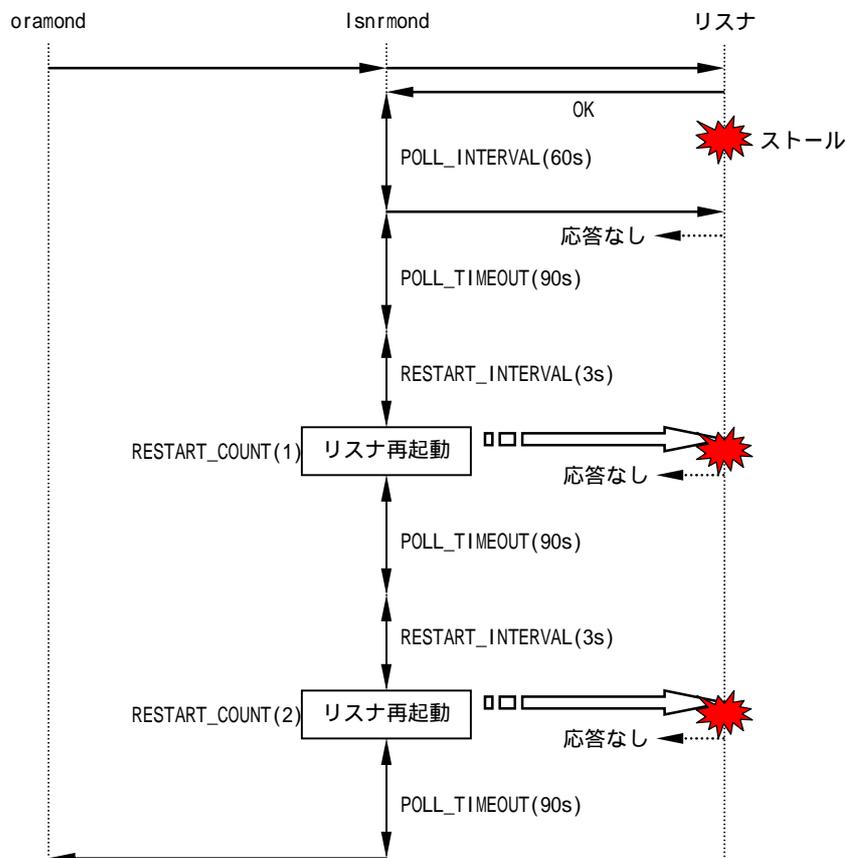


図 4-12 リスナ監視タイムアウト時の動作

リスナの監視が `POLL_TIMEOUT` に設定した時間までに完了しなかった場合、`RESTART_COUNT` に 1 以上の値が設定されていると、`RESTART_COUNT` の回数だけリスナの再起動を行います。所定の回数リトライを行ってもリスナが起動しない場合、モニタ制御デーモンへ障害を通知します。`RESTART_COUNT` に 0 が設定されている場合は、リスナの監視がタイムアウトするとすぐにモニタ制御デーモンへ障害を通知します。

障害ノードであることが決定されても、設定ファイルの `SERVICE_DOWN` パラメータの

第 4 章 ApplicationMonitor の動作

設定により、ノード停止処理を行うか判断されます。SERVICE_DOWN パラメータは、各リスナ障害時にも同様に動作継続するかどうかの判断に使われます。

4.4. 表領域監視時の動作

表領域監視モニタ(tschkmond)は起動すると POLL_INTERVAL に設定した間隔で表領域の監視を行います。表領域の監視方法はデータディクショナリ表の参照です。

(例)

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

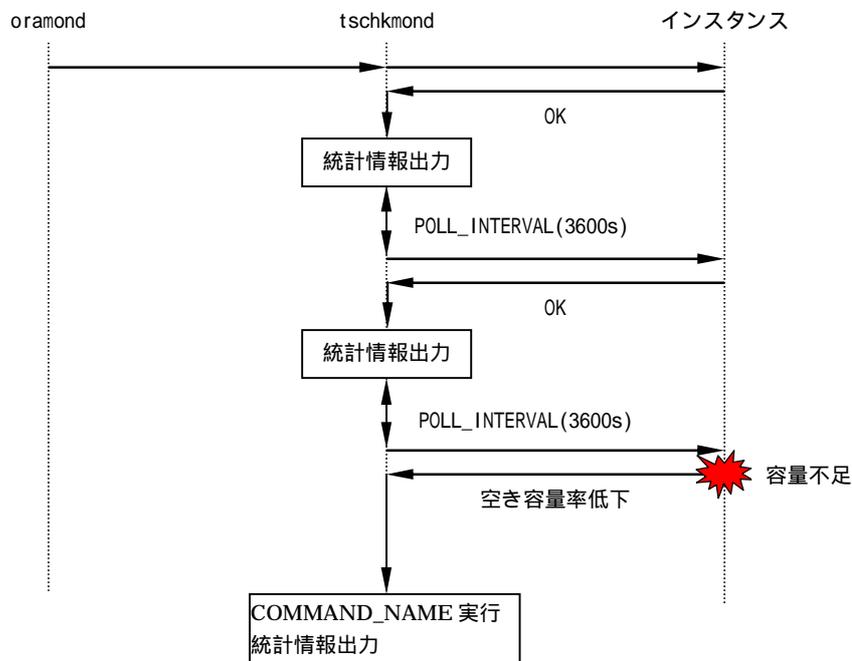


図 4-13 表領域監視時の動作

表領域監視モニタは、POLL_INTERVAL 毎に表領域の最大使用可能エクステントサイズ及び空き領域率をチェックし、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領域率が指定された閾値を下回ったことを検出すると、syslog にエラーメッセージを出力し、COMMAND_NAME に指定したシェルスクリプトを実行します。

なお、データベースが MOUNT 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。

第4章 ApplicationMonitor の動作

4.5. ディスク領域監視時の動作

ディスク領域監視モニタ(fdsmond)は起動すると POLL_INTERVAL に設定した間隔でディスク領域の監視を行います。監視対象は、アーカイブログ出力先、ユーザトレース出力先、バックグラウンドトレース出力先、コアファイル出力先及び監査証跡出力先です。

(例)

```
DiskSpaceMonitor {  
  POLL_INTERVAL = 3600  
  FREE_SPACE    = 10  
}
```

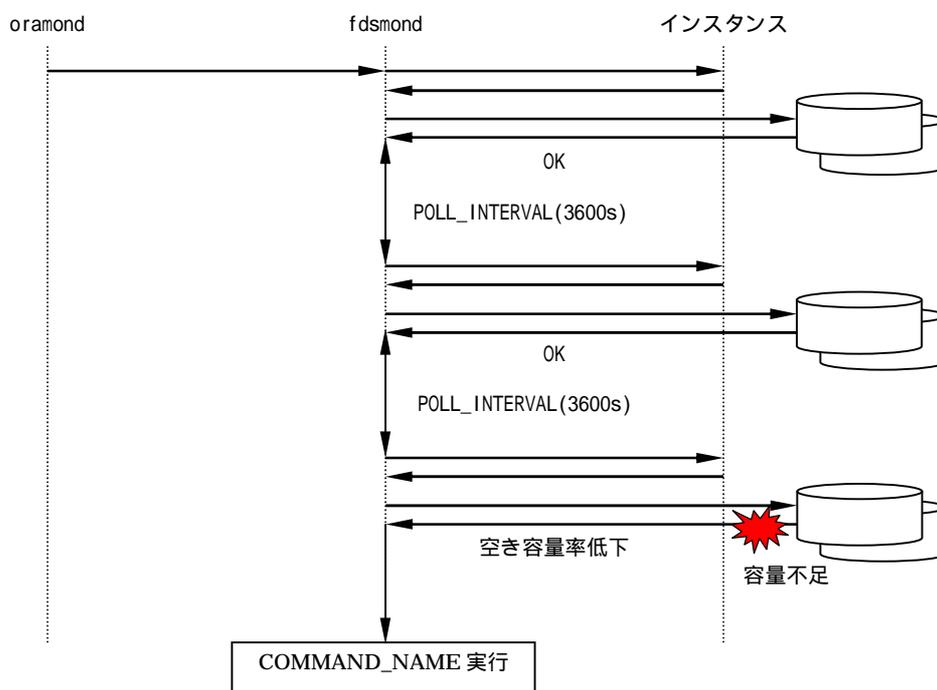


図 4-14 ディスク領域監視時の動作

監視対象のディスク領域の空き領域率が指定された閾値を下回ったことを検出すると、syslog にエラーメッセージを出力し、COMMAND_NAME に指定したシェルスクリプトを実行します。

なお、データベースが MOUNT 状態の際、任意の Oracle ユーザによる監視を行う設定となっている場合(ORACLE_USER / ORACLE_PASS パラメータが設定されている場合) は、インスタンスに対する起動確認のみ行います。

4.6. 統計情報採取時の動作

統計情報採取モニタ(statsmond)は起動すると POLL_INTERVAL に設定した間隔で統計情報の採取を行います。採取対象は、Oracle の SGA 情報です。

(例)

```
StatsMonitor STATSMOND {
  POLL_INTERVAL = 3600
  Stats SGA_SPACE {
    GET_STATS = YES
  }
}
```

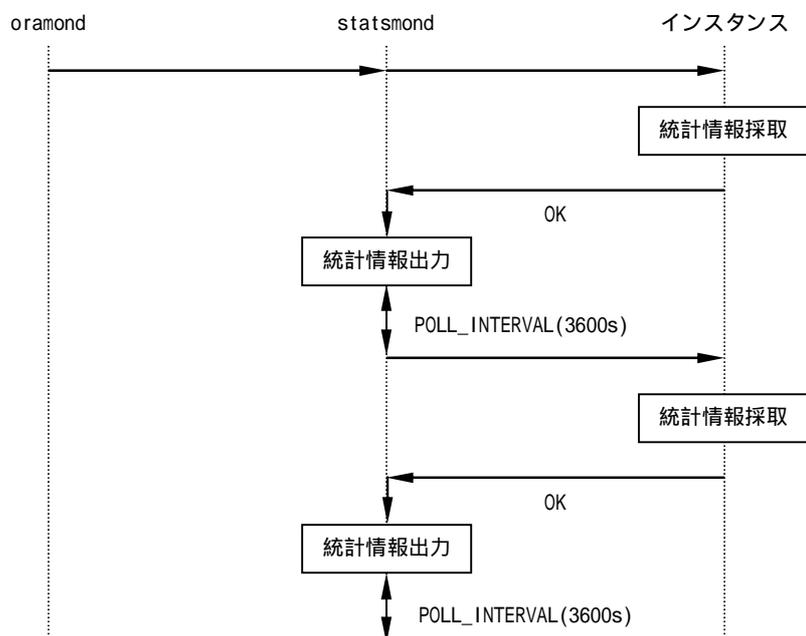


図 4-15 統計情報採取時の動作

統計情報採取モニタは、POLL_INTERVAL 毎に Oracle の SGA 情報を採取し、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すると、syslog にエラーメッセージを出力します。

なお、データベースが MOUNT 状態の際、任意の Oracle ユーザによる監視を行う設定となっている場合(ORACLE_USER / ORACLE_PASS パラメータが設定されている場合)は、インスタンスに対する起動確認のみ行います。

第4章 ApplicationMonitor の動作

4.7. システムステートの採取

モニタ制御デーモン(oramond)は、Oracle で障害を検出した時、GET_DUMP に YES が設定されている全てのシステムステータム採取デーモン(ssdump)にダンプ採取要求を送ります。この要求によりシステムステータム採取デーモンは、システムステータムの採取を行います。また、スクリプトの自動実行が指定されている場合はシステムステータムの採取と同時に実行されます。

(例)

```
SystemStateDaemon {  
  GET_DUMP      =YES  
  DUMP_INTERVAL =30  
  DUMP_COUNT    =3  
  DUMP_TIMEOUT  =90  
}
```

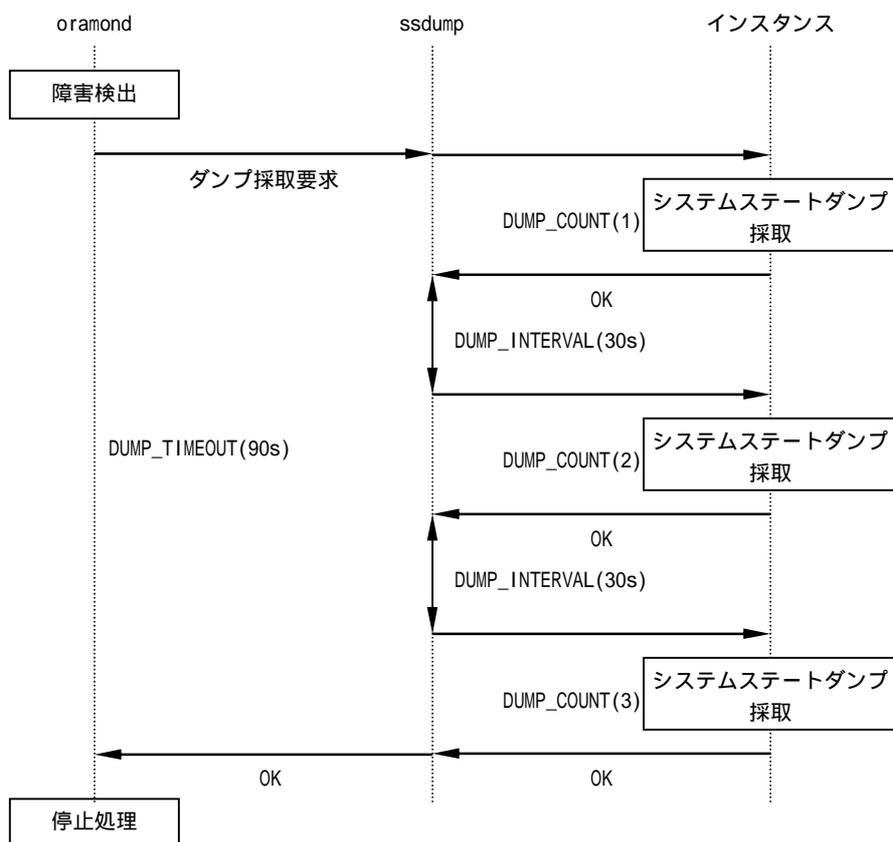


図 4-16 システムステータム採取時の動作

モニタ制御デーモン(oramond)は、ダンプ採取要求後、DUMP_TIMEOUT に指定された時間の間、終了を待ち合わせます。この待ち合わせ時間内にシステムステータムの採取が終了しなかった場合は、採取処理を停止し、ノードの停止処理に移行します。

4.8. スクリプトの実行

モニタ制御デーモン(oramond)は、Oracle で障害を検出した時、SCRIPT_NAME に指定されるスクリプトを全ノードで実行します。スクリプト実行時の動作は以下のようになります。また、システムステータスダンプの採取が指定されている場合はスクリプトの自動実行と同時に実行されます。

(例)

```
SCRIPT_NAME    = /opt/HA/AM/bin/oradiag
SCRIPT_TIMEOUT = 30
```

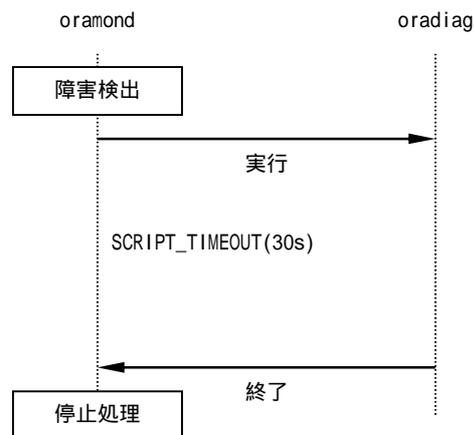


図 4-17 スクリプト自動実行の動作

モニタ制御デーモン(oramond)は、スクリプト実行後、SCRIPT_TIMEOUT に指定された時間の間、終了を待ち合わせます。この待ち合わせ時間内にスクリプトの処理が終了しなかった場合は、ノードの停止処理に移行します。

第 4 章 ApplicationMonitor の動作

4.9. OCW 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 監視モニタは起動すると、CRS 監視設定ファイルの POLL_INTERVAL に設定した間隔毎に、CRS_STAT に設定されたコマンドを用いて OCW デーモンの状態を監視します。

(例)

ORACLE_VERSION = 11GR2

CRS_STAT = “ /oracle/product/11.2.0/crs/bin/crsctl check has ”

POLL_INTERVAL = 90

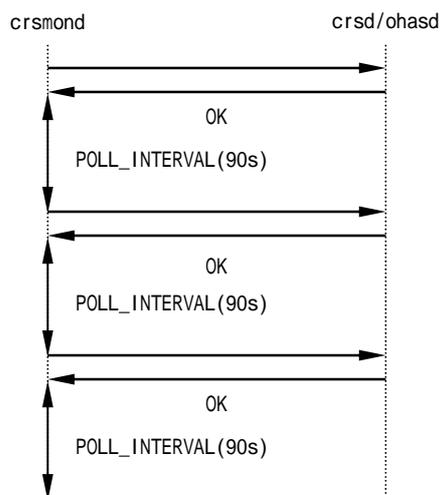


図 4-18 OCW 監視時の動作

OCW デーモンの起動確認時の動作は、以下のようになります。

(例)

```
ORACLE_VERSION = 11GR2
CRS_STAT = " /oracle/product/11.2.0/crs/bin/crsctl check has "
POLL_INTERVAL = 90
RETRY_COUNT = 2
WAIT_INTERVAL = 180
SERVICE_DOWN = NO
```

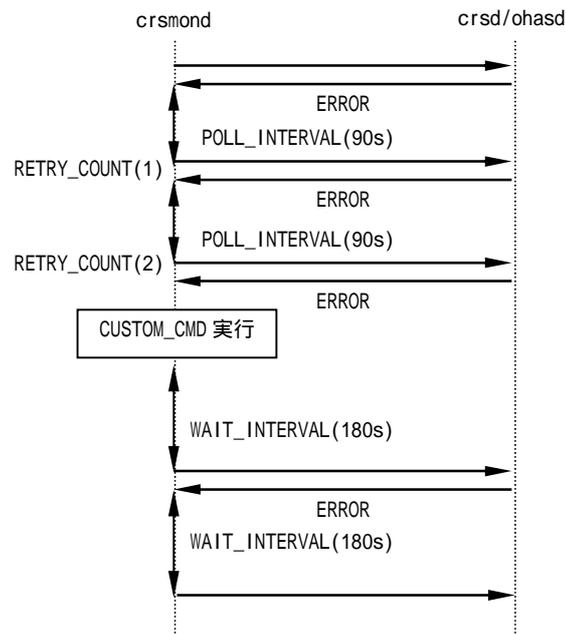


図 4-19 OCW 起動時の動作

CRS 監視モニタが、OCW デーモンの起動確認をする場合、POLL_INTERVAL の間隔で、RETRY_COUNT の回数だけ状態確認を試みます。RETRY_COUNT 回数リトライしても起動を確認できなかった場合、CRS 監視設定ファイルの CUSTOM_CMD が設定されていれば、その処理を実行します。

その後、WAIT_INTERVAL に指定された時間の間、crsd/ohasd 起動待ち合わせを行い、再度確認を行います。この動作は、SERVICE_DOWN が NO の場合、crsd/ohasd の起動が確認されるか、明示的に停止されるまで繰り返します。

第 4 章 ApplicationMonitor の動作

OCW 監視がエラーになった場合、以下のような動作になります。

(例)

```
ORACLE_VERSION = 11GR2
CRS_STAT = " /oracle/product/11.2.0/crs/bin/crsctl check has "
POLL_INTERVAL = 90
POLL_TIMEOUT = 180
RETRY_COUNT = 2
WAIT_INTERVAL = 180
SERVICE_DOWN = NO
```

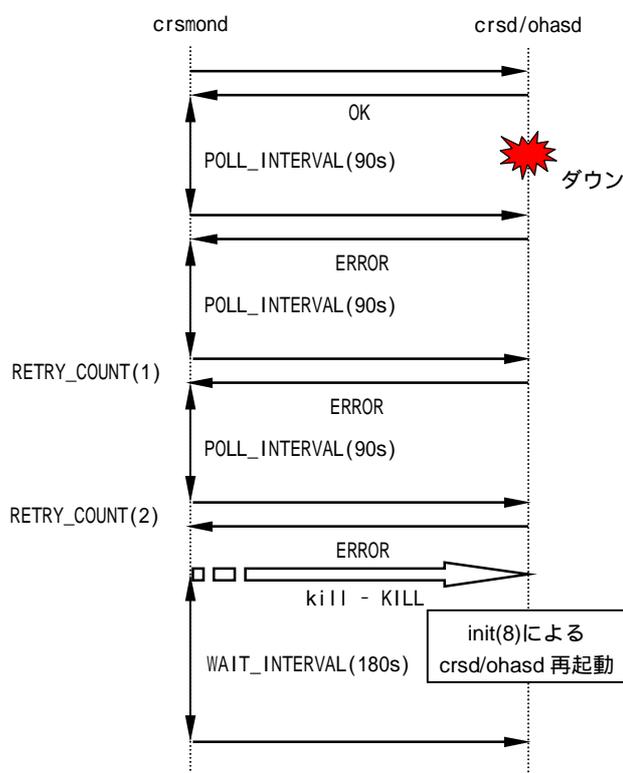


図 4-20 OCW 監視エラー時の動作

CRS 監視モニタは、OCW デーモンの障害を検出した場合、POLL_INTERVAL の間隔で、RETRY_COUNT の回数だけ状態確認を試みます。RETRY_COUNT 回数リトライしても状態が回復しない場合、crsd/ohasd プロセスの強制終了を行います。

CRS 監視設定ファイルの SERVICE_DOWN に NO が設定されている場合は、WAIT_INTERVAL に指定された時間の間、init(8)による crsd/ohasd 再起動の待ち合わせを行い、監視を再開します。監視再開後、crsd/ohasd が起動されていなかった場合の動作は、前述の OCW デーモンの起動確認時の動作に従います。

第 4 章 ApplicationMonitor の動作

OCW の監視が POLL_TIMEOUT に設定した時間までに完了しなかった場合、RETRY_COUNT の回数だけ OCW の状態確認を行います。所定の回数リトライを行っても状態が回復しない場合、crsd/ohasd プロセスを強制終了します。

WAIT_INTERVAL に指定された時間の間、init(8)による crsd/ohasd 再起動の待ち合わせを行い、監視を再開します。crsd/ohasd 再起動の確認ができなかった場合、所定の回数リトライを行います。状態の回復が見込めなかった場合、SERVICE_DOWN に YES が設定されている場合は、CUSTOM_CMD に設定されている処理を実行し、crsmond を停止します。

第5章 ApplicationMonitor の運用

この章では CLUSTERPRO 上に構成されている 2 ノードの Oracle グループに対して、実際に AM の監視機能を追加し、運用を行うための手順を説明します。

5.1. Oracle ハング検知機能の停止 (RAC 環境のみ)

Oracle では、Oracle ハング検知機能 () がデフォルトで有効になっています。

Oracle11gR1 の Oracle ハング検知機能には不具合が報告されているため、機能を停止します。ただし、AM を RAC 構成の環境で監視を行う場合は、Oracle のバージョンに関わらず Oracle ハング検知機能を停止することを推奨します。

Oracle ハング検知機能を停止するには、全てのノードで 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。

() Oracle ハング検知機能とは、Oracle より提供されているストール監視機能です。

・ SPFILE を使用している場合

```
$ sqlplus "/ as sysdba"
SQL> alter system set "_lm_rcvr_hang_kill"=false scope=spfile;
```

・ PFILE を使用している場合

```
PFILE に、以下を追加してください。
_lm_rcvr_hang_kill=false
```

5.2. ApplicationMonitor の設定

既存の Oracle グループの設定を参照し、サンプルファイルを基に AM 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を基に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。

AM で Oracle の監視を行うためには最低限、以下のパラメータを設定する必要があります。

- **NODE_NAME** パラメータ
既存の Oracle グループの設定を参照し、現用、待機系のノード名をそれぞれ指定してください。Node ステートメントではここで指定したノード名のみ使用できます。
- **ORACLE_BASE, ORACLE_HOME, ORA_NLS, SHLIB_PATH** パラメータ
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、適切な値に設定してください。誤った値を指定すると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、適切な値を設定してください。誤った値を指定すると監視が正常に行われません。
- **MONITOR_USER** パラメータ
MONITOR_USER パラメータには、Oracle の OSDBA(dba)グループに所属するユーザ名を指定してください。DBA 権限の無いユーザを指定した場合、監視が正常に行われません。
- **TNS_PING** パラメータ
リスナ監視機能を使用する場合に必要です。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、適切な値に設定してください。誤った値を指定すると監視が正常に行われません。
- **SERVICE_DOWN** パラメータ
インスタンス/リスナ監視で障害を検出した際にグループ切り替えを発生させたい場合は、“yes”に設定してください。“no”に設定した場合は、リスナ障害後のグループ切り替えが行われません。SERVICE_DOWN パラメータについては、「3.2.14 インスタンス/リスナ障害時のグループ停止の設定」も参照してください。

- Node ステートメント
NODE_NAME パラメータに設定したノード名と同じ値を設定してください。
- ORACLE_SID パラメータ
Node ステートメント毎に、各ノードの監視対象インスタンスの SID を正しく指定してください。ORACLE_SID パラメータに誤った値を指定した場合、後述の設定ファイルチェック機能が正しく機能できず、他のパラメータエラーを検出できません。
- ListenerMonitor ステートメント
Node ステートメント毎に存在する 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名を正しく指定してください。
- NET_SERVICE_NAME パラメータ
Node ステートメント毎に存在する ListenerMonitor ステートメント内の NET_SERVICE_NAME パラメータに、監視対象リスナのネットサービス名を正しく指定してください。
- TableSpaceMonitor ステートメント
表領域監視を行わない場合は、TableSpaceMonitor ステートメント全体をコメントアウトするか、設定ファイルから削除してください。表領域監視を行う場合は、「3.2.7 パラメーター一覧」の TableSpaceMonitor ステートメントの説明を参照し、適切な設定を行ってください。
- DiskSpaceMonitor ステートメント
ディスク領域監視を行わない場合は、DiskSpaceMonitor ステートメント全体をコメントアウトするか、設定ファイルから削除してください。ディスク領域監視を行う場合は、「3.2.7 パラメーター一覧」の DiskSpaceMonitor ステートメントの説明を参照し、適切な設定を行ってください。
- StatsMonitor ステートメント
統計情報採取を行わない場合は、StatsMonitor ステートメント全体をコメントアウトするか、設定ファイルから削除してください。統計情報採取を行う場合は、「3.2.7 パラメーター一覧」の StatsMonitor ステートメントの説明を参照し、適切な設定を行ってください。

上記以外の設定パラメータについては、「3.2.7 パラメーター一覧」を参考にして、必要に応じて設定を行ってください。

AM の基本的な設定ファイルが作成できました。

5.3. ApplicationMonitor の設定確認

AM の管理コマンド “oraadmin” を使って、さきほど作成した設定ファイルの内容をテストします。**既存の Oracle グループが動作している状態**で、以下を実行してください。

```
# /opt/HA/AM/bin/oraadmin -c check -f /etc/opt/HA/AM/conf/oramond.conf
Oracle or listener is DOWN, so can't check parameter. Node=dbnode2
Finished to check configuration-file.
#
```

上記の実行結果は、Oracle グループが dbnode1 で動作しており、dbnode2 では動作していない場合のものです。ここで、現在 Oracle グループが動作しているノード上のエラーメッセージが表示されていないことを確認してください。上記以外のエラーメッセージが出力された場合は、「3.2.7 パラメーター一覧」を参照し、パラメータ設定に誤りがないかどうか再度確認してください。

Oracle グループが動作していないノード上でのパラメータはチェックされません。可能であれば、Oracle グループを待機系ノードに移動させた状態で再度 oraadmin コマンドを実行し、待機系ノードのパラメータもチェックされることをおすすめします。

AM を RAC 構成の環境でご使用の際、上記エラーが出力される場合は、全ノードで Oracle グループを動作させた状態で再度、全ノードのパラメータのチェックをされることをおすすめします。

注意： AMにおいて、oraadminによるCRS監視設定ファイルの設定確認は行えません。CRS監視設定ファイルに設定の誤りがある場合、crsmond起動の際、syslogにエラーメッセージが出力されます。
『CLUSTERPRO MC ApplicationMonitor 1.0 for Linux メッセージ一覧』により、エラー内容の詳細を確認の上、誤りを訂正してください。

5.4. ApplicationMonitor の設定配布

設定ファイルをバイナリ化し、設定ノードに配布します。

```
# /opt/HA/AM/bin/oraadmin -c apply -f /etc/opt/HA/AM/conf/oramond.conf
Oracle or listener is DOWN, so can't check parameter. Node=dbnode2
Finished to check configuration-file.
Succeeded in apply configuration file.
#
```

“ Succeeded in apply configuration file. ” が出力されたことを確認してください。これで AM のバイナリ設定ファイルが全ノードに配布されました。

RAC 構成の場合、全ノードで Oracle グループが動作している状態で設定ノードへの配布を行うと、以下のようなメッセージとなります。

```
# /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.
#
```

注意： AMにおいて、oraadminによるCRS監視設定ファイルの全ノード配布は行えません。各ノードで作成してください。

以上で AM の設定は終了です。

第 5 章 ApplicationMonitor の運用

5.5. ApplicationMonitor の Oracle グループへの組み込み

設定を終えた AM を、既存の Oracle グループに組み込みます。CLUSTERPRO の Builder により、AM を起動する EXEC リソース、AM の監視をする PID モニタリソースを作成してください。

変更が必要なパラメータを以下に示します。それ以外のパラメータについては、デフォルト値または個々のシステムに合わせて設定してください。

・ AM を起動 / 停止する EXEC リソース

グループに対する”リソースの追加”を選択し設定します。

| パラメータ | 設定値 |
|--|------------------------------|
| タイプ | execute resource |
| 開始スクリプト "調整" "パラメータ"を選択して設定します。 終了スクリプトは同期のままです。 | 非同期 |
| 依存するリソース "既定の依存関係に従う"のチェックを外してから設定します。 | Oracle 起動用の EXEC リソース を選択 |
| Start script | "編集"で AM の起動処理を設定 |
| Stop script | "編集"で AM の停止処理を設定 1 |

1 停止処理は、oramond に SIGTERM を送る処理を記述します。同様に各モニタを単独起動する場合の停止処理も、各モニタに SIGTERM を送る処理を記述します。

・ AM を監視する PID モニタリソース

“モニタリソースの追加”を選択し設定します。

| パラメータ | 設定値 |
|----------|--------------------------------|
| タイプ | pid monitor |
| 監視タイミング | 活性時 (デフォルト) |
| 対象リソース | "参照" から AM 起動用の EXEC リソース選択 |
| 回復対象 | "参照" から Oracle、AM を起動するグループを選択 |
| 再活性化しきい値 | 0 |

ノードダウンを伴うフェイルオーバを設定する場合、以下についても設定します。

| パラメータ | 設定値 |
|-------------|-----------------------------------|
| フェイルオーバしきい値 | 0 |
| 最終動作 | “クラスタデーモン停止と OS 再起動”などのノード停止動作を選択 |

5.5 ApplicationMonitor の Oracle グループへの組み込み

RENS 連携する場合には、追加の設定が必要です。

RENS の設定については、「3.5 CLUSTERPROの設定」、マニュアル『サーバ管理基盤セットアップガイド』、『サーバ管理基盤利用の手引き』を参照してください。

Builder により作成した設定情報をクラスタに反映（アップロード）します。アップロードするには、事前にクラスタを一時停止する必要があります。

WebManager からクラスタの停止（“サービス” “クラスタ停止”）を実施します。

Builder から設定情報をアップロード（“ファイル” “情報ファイルのアップロード”）します。

CLUSTERPRO の各パラメータの詳細については、CLUSTERPRO のマニュアル『インストール&設定ガイド』を参照してください。

第 5 章 ApplicationMonitor の運用

5.6. Oracle グループの起動

AM の監視設定が組み込まれた Oracle グループを起動します。クラスタ起動の延長で Oracle グループも起動します。クラスタの起動は WebManager より行ってください。

WebManager からクラスタの起動 (“サービス” “クラスタ開始”) を実施します。

AM が起動したことを、Oracle グループが起動したノード上の syslog メッセージで確認してください。

```
...
oramond[6705]: 0:ApplicationMonitor 1.0
oramond[6705]: 0:Loading configuration file /etc/opt/HA/AM/conf/oramond0.
bin.
oramond[6705]: 0:Logging level is 2.
...
oramond[6705]: 0:Monitoring activity on local node is started.
oramond[6705]: 0:instmond[6729] INSTANCE1 started.
oramond[6705]: 0:ssdump[6731] started.
oramond[6705]: 0:lsnrmond[6732] LISTENER1 started.
oramond[6705]: 0:lsnrmond[6732] LISTENER1 status is UP.
oramond[6705]: 0:instmond[6729] INSTANCE1 status is UP.
...
```

Oracle グループの起動処理の延長で AM が起動し、監視対象に指定した全てのインスタンス名、リスナ名について “status is UP” と表示されていれば、監視は正常に行われています。

上記以外のエラーメッセージが出力される、または Oracle グループ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。

5.7. 動作中の ApplicationMonitor のステータス確認

oraadmin コマンドを用いて、AM の状態を確認することができます。以下のコマンドを実行してください。

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による制御(control)が NO であるものが、単独起動モニタになります。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: dbnode1 RESUME
  monitors  name          m-status  t-status  interval  tat/limit  control
-----
  INSTANCE  n/a           ACTIVE    UP        90        4/120     YES
  LISTENER  LISTENER1     ACTIVE    UP        60        5/90     YES
  SSDUMP    n/a           ACTIVE    n/a       n/a       n/a       YES
NODE: dbnode2 UNKNOWN
  monitors  name          m-status  t-status  interval  tat/limit  control
-----
  INSTANCE  n/a           UNKNOWN   UNKNOWN   n/a       n/a       YES
  LISTENER  LISTENER1     UNKNOWN   UNKNOWN   n/a       n/a       YES
  SSDUMP    n/a           UNKNOWN   n/a       n/a       n/a       YES
#
```

グループが動作しているノード上では、監視対象のインスタンス、リスナの状態(t-status)が UP であることが確認できます。現在グループが動作していないノードは“ UNKNOWN ”と表示されます。

RAC 構成時は、設定ノード上のグループが全て動作している場合、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: dbnode1 RESUME
  monitors  name          m-status  t-status  interval  tat/limit  control
-----
  INSTANCE  n/a           ACTIVE    UP        90        1/120     YES
  LISTENER  LISTENER1     ACTIVE    UP        60        0/90     YES
  LISTENER  LISTENER2     ACTIVE    UP        60        0/90     NO
  SSDUMP    n/a           ACTIVE    n/a       n/a       n/a       YES
NODE: dbnode2 RESUME
  monitors  name          m-status  t-status  interval  tat/limit  control
-----
  INSTANCE  n/a           ACTIVE    UP        90        1/120     YES
  TABLESPACE MONITORID     ACTIVE    n/a       3600     n/a       YES
  SSDUMP    n/a           ACTIVE    n/a       n/a       n/a       YES
#
```

第 5 章 ApplicationMonitor の運用

RENS と連携する場合表示、項目が増えます。それぞれ以下の意味を持ちます。

resource-name : RENS リソース名

RENS 用リソース名は、以下の規則で自動的に作成されます。

| モニタ | リソース名 |
|-------------|------------------|
| インスタンス監視モニタ | Instance_<SID 名> |
| リスナ監視モニタ | Listener_<リスナ名> |
| CRS 監視モニタ | CRS |

r-status : RENS に通知済みのリソース値

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: dbnode1 RESUME
  monitors  name      m-status  t-status  interval  tat/limit  control  resource-name  r-status
-----
INSTANCE   n/a      ACTIVE    UP        90        4/120     YES     Instance_sid1  UP
LISTENER   LISTENER1 ACTIVE    UP        60        5/90     YES     Listener_LISTENER1  UP
SSDUMP     n/a      ACTIVE    n/a      n/a       n/a       YES
#
```

RENS がインストールされていない環境では、ヘッダ (resource-name、r-status) は表示されません。

5.8. 監視間隔の調整

AM 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認識しますので、お使いの環境に合わせて POLL_TIMEOUT パラメータを調整する必要があります。

AM の監視を有効にした状態で、Oracle に対し通常の業務を想定した負荷を与えてください。

```
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: dbnode1 RESUME
  monitors  name          m-status  t-status  interval  tat/limit  control
-----
  INSTANCE  n/a            ACTIVE    UP        90        67/120     YES
  LISTENER  LISTENER1     ACTIVE    UP        60        35/90      YES
  SSDUMP    n/a            ACTIVE    n/a       n/a       n/a        YES
NODE: dbnode2 UNKNOWN
  monitors  name          m-status  t-status  interval  tat/limit  control
-----
  INSTANCE  n/a            UNKNOWN   UNKNOWN   n/a       n/a        YES
  LISTENER  LISTENER1     UNKNOWN   UNKNOWN   n/a       n/a        YES
  SSDUMP    n/a            UNKNOWN   n/a       n/a       n/a        YES
#
```

“oraadmin -c show” を実行すると、現在までに Oracle のアクセスが完了するまでにかかった時間の最大値が各モニタの “tat” フィールドに表示されます。表示される値を参考に、ピーク負荷時にストールと誤認識しない範囲内で POLL_TIMEOUT パラメータを調整することができます。POLL_TIMEOUT パラメータを小さくすると、ストールを検出するまでにかかる時間は短くなりますが、ストール誤認識の危険が高くなりますので、システムで許容されるダウンタイム内で可能な限り大きい値を指定することをおすすめします。反対に、現在の設定で既にストールと認識してしまうようであれば、POLL_TIMEOUT にはさらに大きな値を指定してください。

5.9. ApplicationMonitor 動作中の設定パラメータ変更

POLL_TIMEOUT を変更する場合など、Oracle グループを動作させた状態のまま、AM に新しい設定を反映させることができます。AM の設定ファイルを編集後、以下のコマンドを実行してください。

```
# /opt/HA/AM/bin/oraadmin -c apply -f /etc/opt/HA/AM/conf/oramond.conf
Oracle or listener is DOWN, so can't check parameter. Node=dbnode2
Finished to check configuration-file.
Succeeded in apply configuration file.
#
```

AM の動作中に oraadmin の apply リクエストを発行した場合、新しいバイナリ設定ファイルが全ノードに配布されると同時に、動作中の AM に対しても新しい設定が即座に反映されます。

注意： 単独起動しているモニタには反映されません。単独起動しているモニタがある場合は、上記コマンドを実行した後、各モニタに対し、個別にリクエストを発行する必要があります。

```
# /opt/HA/AM/bin/oraadmin -c apply -m lsnr -s LISTENER1 -n dbnode1
Success: lsnrmond (node = dbnode1, listener_name = LISTENER1) is accepted (request = apply).
#
```

動作中に設定変更が可能なパラメータについては、「3.2.7 パラメータ一覧」を参照してください。

注意： リスナ監視モニタ数、表領域監視モニタ数及び統計情報採取モニタ数を動的に変更することはできません。変更する場合は、AMを停止し、deleteコマンドで設定を削除した上で、適用を行ってください。

なお、設定ファイルのチェック/適用時に監視対象がフローティングIPにより構成されており、かつそのアドレスが無効化されている場合、チェックが失敗します。該当アドレスを有効な状態にして設定ファイルのチェック/適用を行うようにしてください。

CLUSTERPRO連携を行っている場合で、上記二つに該当する場合は、以下手順に従い、設定を更新してください。

1. リソースと連携しているグループを停止し、AMを停止する。
2. グループからリソース(AM)を外す。
3. フローティングIPを有効化するため、リソースを外したグループを起動する。
4. AMの設定をチェック/適用する。
5. グループを停止し、リソース(AM)を登録する。
6. グループを起動して設定が反映されているか確認する。

注意： OracleがMOUNT状態時、AMの動作中にcheck/applyリクエストを発行した場合、Oracle関連パラメータのチェックに失敗することがあります。その場合は、AMを停止した状態でリクエストを再発行するか、-Fオプションを付与してリクエストを再発行してください。

-Fオプションを付与した場合、Oracle関連パラメータのチェックを行いませんので、設定値に問題がないことを十分ご確認ください。

5.10. Oracle グループのメンテナンス

Oracle グループを動作させたままの状態、データベースのバックアップや表領域のメンテナンス等の作業を実施したいことがあるかも知れません。そのような場合に AM の監視処理を一時的に停止し、メンテナンスに伴う過大な負荷や長時間の表ロック等を障害と誤認識してグループが停止することを避けることができます。

AM の監視処理を一時的に停止するには、oraadmin コマンドの suspend リクエストを使用します。

なお、時刻補正を伴う作業を行う場合は、suspend 要求を行う際、-M オプションを付与し、oramond 内部で動作する自身の無応答監視を一時停止するようにしてください。

```
# /opt/HA/AM/bin/oraadmin -c suspend
Connect to monitor-control(master).
Succeeded in suspend.

# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: dbnode1 SUSPEND
```

| monitors | name | m-status | t-status | interval | tat/limit | control |
|----------|-----------|----------|----------|----------|-----------|---------|
| INSTANCE | n/a | UNKNOWN | UNKNOWN | n/a | n/a | YES |
| LISTENER | LISTENER1 | UNKNOWN | UNKNOWN | n/a | n/a | YES |
| LISTENER | LISTENER2 | ACTIVE | UP | 60 | 12/90 | NO |
| SSDUMP | n/a | UNKNOWN | n/a | n/a | n/a | YES |

```
NODE: dbnode2 UNKNOWN
```

| monitors | name | m-status | t-status | interval | tat/limit | control |
|----------|-----------|----------|----------|----------|-----------|---------|
| INSTANCE | n/a | UNKNOWN | UNKNOWN | n/a | n/a | YES |
| LISTENER | LISTENER1 | 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 |

```
#
```

注意： 上記コマンドでは、単独起動モニタを停止することはできません。
 単独起動しているモニタがある場合は、各モニタに対し個別に停止指示を行ってください。

```
# /opt/HA/AM/bin/oraadmin -c suspend -m lsnr -s LISTENER2 -n dbnode1
Success: lsnrmnd (node = dbnode1, listener_name = LISTENER2) is accepted (request = suspend).
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: dbnode1 SUSPEND
```

| monitors | name | m-status | t-status | interval | tat/limit | control |
|----------|-----------|----------------|----------|----------|-----------|---------|
| INSTANCE | n/a | UNKNOWN | UNKNOWN | n/a | n/a | YES |
| LISTENER | LISTENER1 | UNKNOWN | UNKNOWN | n/a | n/a | YES |
| LISTENER | LISTENER2 | SUSPEND | UP | 60 | 12/90 | NO |
| SSDUMP | n/a | UNKNOWN | UNKNOWN | n/a | n/a | YES |

```
NODE: dbnode2 UNKNOWN
```

| monitors | name | m-status | t-status | interval | tat/limit | control |
|----------|-----------|----------|----------|----------|-----------|---------|
| INSTANCE | n/a | UNKNOWN | UNKNOWN | n/a | n/a | YES |
| LISTENER | LISTENER1 | 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 |

```
#
```

oraadmin の show リクエストを実行し、単独起動モニタも含めて、AM の状態が “ SUSPEND ” に変わったことが確認できたら、必要な作業を実施してください。SUSPEND 中は AM から Oracle に対する全てのアクセスが抑止されます。

第 5 章 ApplicationMonitor の運用

作業終了後、SUSPEND 中の Oracle 監視を再開させるためには、oraadmin の resume リクエストを使用します。

```
# /opt/HA/AM/bin/oraadmin -c resume
Connect to monitor-control(master).
Succeeded in resume.

# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: dbnode1 RESUME
  monitors  name          m-status  t-status  interval  tat/limit  control
-----
INSTANCE   n/a           ACTIVE    UP        90        67/120     YES
LISTENER   LISTENER1     ACTIVE    UP        60        35/90      YES
LISTENER   LISTENER2     SUSPEND   UP        60        12/90      NO
SSDUMP     n/a           ACTIVE    n/a       n/a       n/a        YES
NODE: dbnode2 UNKNOWN
  monitors  name          m-status  t-status  interval  tat/limit  control
-----
INSTANCE   n/a           UNKNOWN   UNKNOWN   n/a       n/a        YES
LISTENER   LISTENER1     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
#
```

注意： 上記コマンドでは、単独起動モニタによる監視を再開することはできません。単独起動しているモニタがある場合は、各モニタに対し個別に再開指示を行ってください。

```

# /opt/HA/AM/bin/oraadmin -c resume -m lsnr -s LISTENER2 -n dbnode1
Success: lsnrmond (node = dbnode1, listener_name = LISTENER2) is accepted (request = resume).
# /opt/HA/AM/bin/oraadmin -c show
Connect to monitor-control(master).
NODE: dbnode1 RESUME
  monitors  name                m-status  t-status  interval  tat/limit  control
-----
INSTANCE   n/a                  ACTIVE    UP        90        67/120     YES
LISTENER   LISTENER1           ACTIVE    UP        60        35/90      YES
LISTENER   LISTENER2           ACTIVE    UP        60        12/90      NO
SSDUMP     n/a                  ACTIVE    n/a       n/a       n/a        YES
NODE: dbnode2 UNKNOWN
  monitors  name                m-status  t-status  interval  tat/limit  control
-----
INSTANCE   n/a                  UNKNOWN   UNKNOWN   n/a       n/a        YES
LISTENER   LISTENER1           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
#

```

注意： AMのモニタ(instmond, tschkmond, fdsmond, statsmond, ssdump)は、起動時にOracleの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 DBに対するアクセスは停止しますが、モニタプロセス自体は動作している状態のため、Oracleパッチ適用の際は、AMの停止も必要になります。

第 5 章 ApplicationMonitor の運用

5.11. ApplicationMonitor による障害の検出

AM の監視機能を組み込んだ Oracle グループが異常終了した場合は、グループの停止が AM の判断によるものかどうか確認を行う必要があります。停止ノードの syslog を参照し、AM のエラーメッセージを確認してください。

```
...
instmond[4332]: [0] sql_error:Oracle error occurred.
instmond[4332]: [0] sql_error: ORA-01034: ORACLE not available
instmond[4332]: [0] sql_error: ORA-27101: shared memory realm does not
exist
instmond[4332]: [0] sql_error: Linux-x86_64 Error: 2: No such file or
directory
oramond[4327]: 0:instmond[4332] INSTANCE1 status is DOWN.
oramond[4327]: 0:Local node has been shifted to disaster mode.
...
oramond[4327]: 0:Issuing AM cluster reconstruction.
oramond[4327]: 0:oramond on this node is terminated.

clusterpro: <type: rm><event: 9> Detected an error in monitoring pidw_am.
(1 : Process does not exist. (pid=13145))
clusterpro: <type: apisv><event: 38> There was a request to failover
group(failover_oracle) from the clprm process(IP=<IPaddress>).
...
```

グループ名 : failover_oracle、AM 監視リソース名 : pidw_am の場合
13145 は AM 起動スクリプトの PID

syslog に上記のような出力が見られた場合は、AM が Oracle の異常を検出してグループが停止しています。共に出力される Oracle のエラーコード等を参考に、障害の原因を判断してください。上記の例では、該当ノード上の Oracle インスタンス、“INSTANCE1” が停止したことにより、AM がインスタンス状態を DOWN と判断し、CLUSTERPRO によるグループ切り替えが発生しています。

SCRIPT_NAME パラメータに製品添付の情報採取スクリプトのパスを指定した場合は、/var/opt/HA/AM/0/oradiag/ディレクトリに障害解析に有益な情報が採取されています。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。

A. コマンドリファレンス

oraadmin

名 前 oraadmin - 管理コマンド

形 式 /opt/HA/AM/bin/oraadmin

動作している全てのモニタに対する要求です。

```
-c show [-t|-d|-S] [-a] [-C <component_id>] [-p <port_number>]
```

oramond によって管理されているモニタのみへの要求です。

```
-c suspend [-M] [-C <component_id >] [-n <node_name>] [-n <node_name>] ...
-c resume [-C <component_id >] [-n <node_name>] [-n <node_name>] ...
-c apply -f <configure_file_name> [-p <port_number>] [-F] [-E]
```

単独起動しているモニタのみへの要求です。

```
-c suspend -m (crs|lsnr|tschk|fds|stats)
(-s <リスナ名>|-i <表領域監視モニタ ID>|-i <統計情報採取モニタ ID>)
-C <component_id> -n <node_name> [-p <port_number>]
-c resume -m (crs|lsnr|tschk|fds|stats)
(-s <リスナ名>|-i <表領域監視モニタ ID>|-i <統計情報採取モニタ ID>)
-C <component_id> -n <node_name> [-p <port_number>]
-c apply -m (crs|lsnr|tschk|fds|stats)
(-s <リスナ名>|-i <表領域監視モニタ ID>|-i <統計情報採取モニタ ID>)
-C <component_id> -n <node_name> [-p <port_number>]
```

構成情報管理サーバ (oraconfd) に対する要求です (監視モニタに対して直接的な要求は行いません)。

```
-c check -f <configure_file_name> [-p <port_number>] [-F] [-E]
-c delete [-p <port_number>] [-C <component_id >]
-c view [-p <port_number>] [-C <component_id >]
```

機能説明 oraadmin は AM の運用及び管理を行うためのコマンドです。<configure_file_name>は任意の設定ファイル名、<port_number>は oraconfd のポート番号、<リスナ名>は監視対象リスナ名、<表領域監視モニタ ID>は表領域監視モニタの ID、<統計情報採取モニタ ID>は統計情報採取モニタの ID、<component_id>は構成番号 (COMPONENT_ID) を指定します。各リクエストの動作は以下のとおりです。

| | |
|---------|--|
| show | 単独起動しているリスナ監視モニタ、表領域監視モニタ、ディスク領域監視モニタ及び統計情報採取モニタを含むモニタ情報を表示します。また、-C の指定に関わらず、各ノードで動作している CRS 監視モニタのステータスも表示します。 |
| show -t | 監視対象の表領域の情報を表示します。 |
| show -d | 監視対象のディスク領域の情報を種別単位に表示します。 |
| show -S | 採取対象の統計情報を表示します。 |
| show -a | モニタ情報の詳細を表示します。 |

次のコマンドオプションは oramond によって管理されているモニタにのみ有効となります。そのため、単独起動しているリスナ監視モニタ、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及び CRS 監視モニタは含まれません。

| | |
|------------|---|
| suspend | 全ノードまたは指定したノードのモニタを一時中断 (サスペンド) 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中断 (サスペンド) すると同時に oramond 自身の監視スレッドを一時停止します。 |

A コマンドリファレンス

resume 全ノードまたは指定したノードのモニタを再開（レジューム）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。

apply 指定された設定ファイルの構成情報を全ノードに反映させます。
-F オプションが指定された場合、ノード毎の Oracle 関連のチェックをスキップし、設定を配布します。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、グループをどうしても停止できない場合に指定することにより、リスナ監視モニタ表領域監視モニタ及び統計情報採取モニタの追加削除を行った設定ファイルの動的配布を可能とします。しかし、単独起動しているモニタが動作中の場合、単独起動モニタが停止します。**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。**

次のコマンドオプションは単独起動しているモニタのみ有効となります。oramond で管理されているモニタへの要求は行えません。

単独起動しているモニタへの要求では、構成番号及びノード名の指定が必須となります。また、各オプションの-m に続け、モニタの種別を crs（CRS 監視モニタ）、lsnr（リスナ監視モニタ）、tschk（表領域監視モニタ）、fds（ディスク領域監視モニタ）、stats（統計情報採取モニタ）から選択し、指定する必要があります。
モニタ種別に lsnr が指定された場合、-s <リスナ名>を必ず指定する必要があります。
モニタ種別に tschk が指定された場合、-i <表領域監視モニタ ID>を必ず指定する必要があります。
モニタ種別に stats が指定された場合、-i <統計情報採取モニタ ID>を必ず指定する必要があります。

suspend -m 指定された単独起動モニタを一時中断（サスペンド）します。

resume -m 指定された単独起動モニタを再開（レジューム）します。

apply -m 指定された単独起動モニタに設定の再読み込みを要求します。

次のコマンドオプションは構成情報管理サーバ（oraconfd）に対する要求となります。

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、ノード毎の Oracle 関連のチェックをスキップします。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、リスナ監視モニタ、表領域監視モニタ及び統計情報採取モニタの追加削除を行った設定ファイルの動的チェックが可能になります。

delete 構成情報ファイル(oramond<構成番号>.bin)を削除します。

view 現在の構成情報を表示します。

使用例 コマンド

oraadmin -c show
を実行した場合、出力は次のようになります。

```
Connect to monitor-control(master).
NODE: dbnode1 RESUME
  monitors  name          m-status  t-status  interval  tat/limit  control
  -----
  INSTANCE  n/a           ACTIVE    UP        90        67/120     YES
  LISTENER  LISTENER1     ACTIVE    UP        60        35/90      YES
  SSDUMP    n/a           ACTIVE    n/a      n/a       n/a        YES
NODE: dbnode2 UNKNOWN
  monitors  name          m-status  t-status  interval  tat/limit  control
  -----
  INSTANCE  n/a           UNKNOWN   UNKNOWN   n/a       n/a        YES
  LISTENER  LISTENER1     UNKNOWN   UNKNOWN   n/a       n/a        YES
  SSDUMP    n/a           UNKNOWN   n/a      n/a       n/a        YES
```

ノード単位に状態が出力されます。

上記の例では、dbnode1 及び dbnode2 がノード名になります。

ノード名の後に、各ノード上の oramond の状態が出力されます。

| | |
|--------------------------------|-----------------------------|
| UNKNOWN | 状態不明。停止中に出力されます。 |
| SUSPEND | 一時停止中 |
| SUSPEND (Meta-Control Stopped) | 一時停止中 (oramond 自己監視スレッド停止中) |
| RESUME | 動作中 |

oramond の状態に続き、各ノード上のモニタの状態が出力されます。

列名とその意味を以下に示します。

monitors 監視モニタ及び採取モニタ

| | |
|------------|-------------------|
| INSTANCE | インスタンス監視モニタ |
| LISTENER | リスナ監視モニタ |
| TABLESPACE | 表領域監視モニタ |
| DISKSPACE | ディスク領域監視モニタ |
| 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 |
| CRS | CRS 監視モニタ |

name リスナ監視モニタ、表領域監視モニタ及び統計情報採取モニタの場合、監視対象のリスナ名、表領域名及び採取対象の統計情報名が出力されます。それ以外は、" n/a " が出力されます。

m-status 監視モニタ及び採取モニタの状態

| | |
|---------|------------------|
| UNKNOWN | 状態不明。停止中に出力されます。 |
| ACTIVE | 動作中 |
| SUSPEND | 単独起動モニタの一時停止 |

t-status 監視対象及び採取対象の状態

| | |
|-----------|---|
| UNKNOWN | 状態不明。停止中に出力されます。 |
| 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 | エラー発生 |
| n/a | モニタが tschkmond、fdsmoond、statsmond 及び ssdump の場合、出力されます。 |

interval 各監視モニタの監視間隔と採取モニタの採取間隔が出力されます。停止中は、" n/a " が出力されます。

tat/limit 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 / POLL_TIMEOUT 値が出力されます。停止中は、" n/a " が出力されます。

control oramond による制御

| | |
|-----|--------------------|
| YES | oramond の制御下にあります。 |
| NO | 単独起動モニタです。 |

RENS と連携する場合、以下の 2 項目が追加されます。

Connect to monitor-control(master).

NODE: dbnode1 RESUME

| monitors name | m-status | t-status | interval | tat/limit | control | resource-name | r-status |
|--------------------|----------|----------|----------|-----------|---------|--------------------|----------|
| INSTANCE n/a | ACTIVE | UP | 90 | 4/120 | YES | Instance_sid1 | UP |
| LISTENER LISTENER1 | ACTIVE | UP | 60 | 5/90 | YES | Listener_LISTENER1 | UP |
| SSDUMP n/a | ACTIVE | n/a | n/a | n/a | YES | | |

resource-name RENS リソース名

| | |
|---------------------|------------------|
| Instance_ < SID 名 > | インスタンス監視モニタの監視情報 |
| Listener_ < リスナ名 > | リスナ監視モニタの監視情報 |
| CRS | CRS 監視モニタの監視情報 |

A コマンドリファレンス

| | |
|--------------|-----------------|
| r-status | RENS に通知したリソース値 |
| UP | 正常 |
| DOWN | 停止 |
| STALL | 無応答 |
| UNKNOWN | 状態不明 |
| SUSPEND | 監視一時停止中 |
| MONITOR_HALT | 監視停止 |

ファイル /opt/HA/AM/bin/oraadmin

注 意 oraadmin 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。oramond 起動中、Oracle 障害発生中及び apply 直後は、しばらくの間 oraadmin を実行することができません。oraadmin は root ユーザのみ実行できます。oramond 動作中の delete 要求はできません。

oraconfd

名前 oraconfd - 構成情報管理サーバ

形式 /etc/opt/HA/AM/lbin/oraconfd [-p *ポート番号*]

機能説明 oraconfd は oramond の構成情報ファイルを管理するためのデーモンです。
通常、/etc/rc.d/init.d/haam_oraconfd によりシステム立ち上げ時に自動起動されます。

-p ポート番号を指定します。デフォルトは、25310 です。

使用例 コマンド
 oraconfd

ファイル /etc/opt/HA/AM/lbin/oraconfd

注意 oraconfd は、各ノード上で1つ起動している必要があります。
複数インスタンスの監視設定を行う際、1つ起動していれば、対応は可能です。
複数起動はできません。
oraconfd は、root ユーザでのみ起動できます。

A コマンドリファレンス

oramond

名 前 oramond - モニタ制御デーモン

形 式 /etc/opt/HA/AM/lbin/oramond [-C 構成番号]

機能説明 oramond は各モニタ(instmond, lsrmnd, tschkmond, fdsmond, altmond, statsmond, ssdump)を制御するためのデーモンです。通常、CLUSTERPRO にグループのリソースとして登録を行い、クラスタ起動時に Oracle グループのリソースとして起動します。

-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

ファイル /etc/opt/HA/AM/lbin/oramond

注 意 oramond は、root ユーザでのみ起動できます。

lsnrmond

名前 Isnrmond - リスナ監視モニタ

形式 /etc/opt/HA/AM/lbin/lsnrmond -s リスナ名 [-C 構成番号]

機能説明 Isnrmond は Oracle のリスナの障害を監視するためのデーモンです。
障害検出時に停止することで oramond または CLUSTERPRO へ障害を通知します。

-s 設定ファイル(oramond.conf)に記述される ListenerMonitor ステートメントのリスナ名を指定します。

使用例 コマンド
lsnrmond -s LISTENER

ファイル /etc/opt/HA/AM/lbin/lsnrmond

注意 root または OSDBA(dba)グループに所属するユーザで起動する必要があります。単体で起動する場合は、設定ファイルの MONITOR_CONTROL パラメータに no が指定されている必要があります。

A コマンドリファレンス

tschkmond

- 名前 前 tschkmond_11G, tschkmond_11GR2 - 表領域監視モニタ
- 形式 式 /etc/opt/HA/AM/lbin/tschkmond_11G -i 表領域監視モニタ ID [-C 構成番号]
 /etc/opt/HA/AM/lbin/tschkmond_11GR2 -i 表領域監視モニタ ID [-C 構成番号]
- 機能説明 tschkmond は Oracle の表領域の空き容量を監視するためのデーモンです。
 容量不足時は、syslog に警告メッセージを表示します。
- i 設定ファイル(oramond.conf)に記述される TableSpaceMonitor ステートメントの表領域監視モニタ ID を指定します。
- 使用例 コマンド
 tschkmond_11G -i TSCHK
 tschkmond_11G -i TSCHK -C 1
- ファイル /etc/opt/HA/AM/lbin/tschkmond_11G
 /etc/opt/HA/AM/lbin/tschkmond_11GR2
- 注意 OSDBA(dba)グループに所属するユーザで起動する必要があります。
 単体で起動する場合は、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 のベースディレクトリ |
| ORACLE_HOME | Oracle のホームディレクトリ |
| ORA_NLS10 | Oracle の言語ファイル格納ディレクトリ |
| NLS_LANG | Oracle で使用している言語 |
| LD_LIBRARY_PATH | Oracle のライブラリパス |
- 単体で起動する場合は、設定ファイルの MONITOR_CONTROL パラメータに no が指定されている必要があります。

fdsmond

- 名前 fdsmond_11G, fdsmond_11GR2 - ディスク領域監視モニタ
- 形式 /etc/opt/HA/AM/lbin/fdsmond_11G [-C 構成番号]
/etc/opt/HA/AM/lbin/fdsmond_11GR2 [-C 構成番号]
- 機能説明 fdsmond は Oracle が使用するローカルディスク領域の空き容量を監視するためのデーモンです。
容量不足時は、syslog に警告メッセージを表示します。
- 使用例 コマンド
fdsmond_11G
fdsmond_11G -C 1
- ファイル /etc/opt/HA/AM/lbin/fdsmond_11G
/etc/opt/HA/AM/lbin/fdsmond_11GR2
- 注意 OSDBA(dba)グループに所属するユーザで起動する必要があります。
単体で起動する場合は、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 のベースディレクトリ |
| ORACLE_HOME | Oracle のホームディレクトリ |
| ORA_NLS10 | Oracle の言語ファイル格納ディレクトリ |
| NLS_LANG | Oracle で使用している言語 |
| LD_LIBRARY_PATH | Oracle のライブラリパス |
- 単体で起動する場合は、設定ファイルの MONITOR_CONTROL パラメータに no が指定されている必要があります。

A コマンドリファレンス

statsmond

- 名前 statsmond_11G, statsmond_11GR2 - 統計情報採取モニタ
- 形式 /etc/opt/HA/AM/lbin/statsmond_11G -i 統計情報採取モニタ ID [-C 構成番号]
/etc/opt/HA/AM/lbin/statsmond_11GR2 -i 統計情報採取モニタ ID [-C 構成番号]
- 機能説明 statsmond は Oracle の SGA 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。
- i 設定ファイル(oramond.conf)に記述される 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。
- 使用例 コマンド
statsmond_11G -i STATS
statsmond_11G -i STATS -C 1
- ファイル /etc/opt/HA/AM/lbin/statsmond_11G
/etc/opt/HA/AM/lbin/statsmond_11GR2
- 注意 OSDBA(dba)グループに所属するユーザで起動する必要があります。
単体で起動する場合は、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 のベースディレクトリ |
| ORACLE_HOME | Oracle のホームディレクトリ |
| ORA_NLS10 | Oracle の言語ファイル格納ディレクトリ |
| NLS_LANG | Oracle で使用している言語 |
| LD_LIBRARY_PATH | Oracle のライブラリパス |
- 単体で起動する場合は、設定ファイルの MONITOR_CONTROL パラメータに no が指定されている必要があります。

crsmond

名 前 crsmond - CRS 監視モニタ

形 式 /etc/opt/HA/AM/lbin/crsmond

機 能 説 明 crsmond は OCW の無応答障害を監視するためのプロセスです。
障害検出時に OCW デーモンプロセスを強制停止することで、init(8)による OCW デーモンプロセスの再起動を促します。また、パラメータ SERVICE_DOWN の設定値により、復旧不可時の動作の設定が行え、CLUSTERPRO のリソースとして連携を行うことが可能となっています。

使 用 例 コマンド
crsmond

フ ァ イ ル /etc/opt/HA/AM/lbin/crsmond プログラム本体
/etc/opt/HA/AM/conf/crsmond.conf CRS 監視設定ファイル

注 意 起動するためには、最低限、CRS 監視設定ファイルに CRS_STAT パラメータを必ず設定する必要があります。
crsmond は root ユーザのみ実行できます。また、同一ノードで複数の crsmond を起動することはできません。

A コマンドリファレンス

索引

- .tnsping_core67
- A**
- altmond13
- ApplicationMonitor の運用115
- ApplicationMonitor の概要1
- ApplicationMonitor の設定19
- ApplicationMonitor の動作91
- ApplicationMonitor の特長3
- C**
- CLUSTERPRO75, 78, 89, 121
- crsmond14, 143
- crsmond.conf.....1, 2, 17, 81
- CRS グループ.....88
- CRS 監視110
- CRS 監視モニタ14, 143
- CRS 監視設定ファイル1, 2, 17, 81
- D**
- Directory ステートメント.....38
- Directory ステートメントのパラメータ ..63
- DiskSpaceMonitor ステートメント34
- DiskSpaceMonitor ステートメントのパラメータ56
- F**
- fdsmnd13
- fdsmnd_11G141
- fdsmnd_11GR2.....141
- I**
- InstanceMonitor ステートメント31
- InstanceMonitor ステートメントのパラメータ48
- instmond12
- L**
- ListenerMonitor ステートメント32
- ListenerMonitor ステートメントのパラメータ50
- LISTENER グループ89
- lsnrmond12, 139
- N**
- Node ステートメント30
- Node ステートメントのパラメータ.....46
- O**
- OCW5, 81, 85
- oraadmin16, 133
- Oracle ハング検知機能.....45, 68, 115
- Oracle 構成プロセス停止43, 66
- oraconfd.....16, 137
- oramond11, 138
- oramond.conf.....17, 24
- oramond<構成番号>.bin17
- P**
- ProcessSaver80
- R**
- RAC グループ89
- RAC 構成 ... 2, 11, 43, 65, 96, 118, 119, 123

| | | |
|---------------------------------------|-------------------|----------------------------|
| S | インスタンス障害の監視 | 4 |
| SERVICE_DOWN | | 66 |
| SGA 情報..... | | 15 |
| ssdump..... | | 15 |
| statsmond | | 15 |
| statsmond_11G..... | | 142 |
| statsmond_11GR2 | | 142 |
| StatsMonitor ステートメント | | 35 |
| StatsMonitor ステートメントのパラメータ | | 58 |
| Stats ステートメント..... | | 39 |
| Stats ステートメントのパラメータ | | 63 |
| SystemStateDaemon ステートメント | | 36 |
| SystemStateDaemon ステートメントのパラメータ | | 60 |
| T | | |
| TableSpaceMonitor ステートメント | | 33 |
| TableSpaceMonitor ステートメントのパラメータ | | 54 |
| TableSpace ステートメント | | 37 |
| TableSpace ステートメントのパラメータ | | 62 |
| Table ステートメント | | 37 |
| Table ステートメントのパラメータ..... | | 62 |
| tschkmond | | 13 |
| tschkmond_11G..... | | 140 |
| tschkmond_11GR2..... | | 140 |
| II | | |
| 一括設定 | | 5 |
| 一括変更 | | 5 |
| インスタンス / リスナ障害時のグループ停止 | | 49, 50, 66, 83, 116 |
| インスタンス監視..... | | 93 |
| インスタンス監視表 | | 20 |
| インスタンス監視モニタ | | 12 |
| か | | |
| 各モニタ状態 | | 6 |
| 各モニタ設定 | | 6 |
| 監視・採取一時中断..... | | 6 |
| 監視・採取再開..... | | 6 |
| 監視 TAT | | 6 |
| 監視一時中断 | | 16 |
| 監視応答時間 | | 6 |
| 監視再開..... | | 16 |
| 監視状態..... | | 16 |
| 監視プロセス | | 11 |
| 監視モニタ制御.... | | 44, 46, 50, 54, 56, 58, 66 |
| 管理インタフェース..... | | 16 |
| 管理コマンド | | 16, 133 |
| け | | |
| 現用待機構成 | | 1, 11 |
| こ | | |
| 構成情報管理サーバ..... | | 16, 137 |
| 構成情報ファイル | | 17 |
| コメント..... | | 29, 82 |
| さ | | |
| 採取一時中断 | | 16 |
| 採取再開..... | | 16 |
| 採取状態..... | | 16 |
| 採取プロセス | | 15 |
| 最大監視応答時間 | | 6 |
| し | | |
| システム・グローバル領域..... | | 6 |
| システムステート | | 5, 108 |
| システムステートダンプ採取デーモン.... | | 15 |
| 障害ノードの停止動作..... | | 43, 65 |
| 情報採取スクリプト..... | | 74 |

| | | | |
|------------------------|---------------|----------------------|---|
| ず | | は | |
| スクリプト..... | 5 | パラメータ | 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 43, 82, 83, 117, 118, 126 |
| スクリプトの実行..... | 109 | ひ | |
| ステートメント..... | 29 | 表領域監視..... | 105 |
| ステートメントの階層..... | 40 | 表領域監視モニタ..... | 13, 140 |
| ぜ | | 表領域監視モニタ ID..... | 33 |
| 設定ファイル..... | 17, 24 | ふ | |
| 全ノードステートメント..... | 30 | フィルタリング設定..... | 7, 71 |
| 全ノードステートメントのパラメータ..... | 43 | フィルタリング設定ファイル..... | 72 |
| ぞ | | フィルタリングのデフォルト設定..... | 73 |
| ソフトウェア構成..... | 9 | 複数インスタンス監視..... | 6 |
| た | | も | |
| 単独起動..... | 7, 77, 79, 80 | モニタ制御デーモン..... | 11, 138 |
| て | | り | |
| 停止ノード選択ポリシー..... | 43, 65 | リコンフィグ監視モニタ..... | 13 |
| ディスク領域監視..... | 106 | リスナ監視..... | 100 |
| ディスク領域監視モニタ..... | 13, 141 | リスナ監視モニタ..... | 12, 139 |
| データベース容量不足の監視..... | 4 | リスナ障害時の障害情報採取..... | 67 |
| と | | リスナ障害の監視..... | 4 |
| 統計情報採取モニタ..... | 15, 142 | リスナ名..... | 32, 52, 139 |
| 統計情報採取モニタ ID..... | 35 | ろ | |
| 統計情報の採取..... | 6 | ログレベル | 43, 46, 48, 50, 54, 56, 58, 60, 64, 83 |
| ね | | ローカルディスク容量不足の監視..... | 5 |
| ネットサービス名..... | 52 | | |
| の | | | |
| ノード名..... | 43, 64 | | |

CLUSTERPRO MC
ApplicationMonitor 1.0 for Linux
利用の手引き

2012年 10月 初 版
2013年 4月 第2版

日本電気株式会社
東京都港区芝五丁目7番地1号
TEL (03) 3454-1111 (代表)

Ⓟ

© NEC Corporation 2012-2013

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。