

D I O S A / X T P V1.1

導入の手引

輸出する際の注意事項

本製品(ソフトウェア)は、外国為替及び外国貿易法で規制される規制貨物(または役務)に該当することがあります。

その場合、日本国外へ輸出する場合には日本政府の輸出許可が必要です。

なお、輸出許可申請手続きにあたり資料等が必要な場合には、お買い上げの販売店またはお近くの当社営業拠点にご相談下さい。

はしがき

本書は、D I O S A / X T P プログラム製品の導入の手引です。

本書の読者としては、業務アプリケーション開発を担当し、HP-UX、TPBASE、TAM、Oracle、その他関連 PP の使用法を一通り心得ているシステム技術者を想定しています。

2012 年 10 月 初版

2019 年 6 月 9 版

本書の関連説明書としては次のものがあります。

- D I O S A / X T P 利用の手引
- D I O S A / X T P メモリキャッシュ 利用の手引
- D I O S A / X T P データストア 利用の手引
- D I O S A / X T P コマンドリファレンス
- D I O S A / X T P A P I リファレンス
- D I O S A / X T P 環境定義リファレンス
- D I O S A / X T P メッセージリファレンス

備考

- (1) Microsoft、Windows は、米国あるいはその他の国における米国 Microsoft Corporation の商標または登録商標です。
- (2) UNIX は、X/Open カンパニーリミテッドが独占的にライセンスしている米国ならびに他の国における登録商標です。
- (3) HP、HP-UX は、Hewlett-Packard 社の商標または登録商標です。
- (4) Linux は、Linus Torvalds の米国およびその他の国における商標または登録商標です。
- (5) Red Hat は、米国およびその他の国における Red Hat, Inc. の商標または登録商標です。
- (6) Oracle と Java は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。
- (7) This product includes software developed by the Apache Group for use in the Apache HTTP server project (<http://www.apache.org/>).
- (8) その他、記載されている会社名、製品名は、各社の登録商標または商標です。

目次

- 第 1 章 概要 1
 - 1.1 目的と特徴 1
 - 1.1.1 目的 1
 - 1.1.2 特徴 1
 - 1.2 構成 3
 - 1.2.1 位置づけ 3
 - 1.2.2 システム構成 3
 - 1.2.3 機能構成 4
 - 1.3 諸概念 5
- 第 2 章 システムの構築 6
 - 2.1 システム概要 6
 - 2.2 環境設計 7
 - 2.2.1 ノード・ネットワーク構成 7
 - 2.2.2 DIOSA に関する設計 8
 - 2.2.3 TPBASE に関する設計 9
 - 2.2.4 TAM に関する設計 13
 - 2.2.5 Oracle データベースに関する設計 14
 - 2.3 環境定義 17
 - 2.3.1 ノード関連(DIOSAMAP、SYSAMP) 17
 - 2.3.2 通信関連(TPATHENV、MFCENV) 22
 - 2.3.3 DB 関連(DBCTRL) 25
 - 2.3.4 CO 制御関連(COCENV) 28
 - 2.3.5 部品関連(APLIB、MMG、OPSENV、TMCENV) 30
 - 2.3.6 運用関連(CMDSEND) 35
 - 2.3.7 TPBASE の環境定義 37
 - 2.3.8 TAM の環境定義 44
 - 2.3.9 Oracle データベースの環境定義 45
 - 2.4 監視設計 46
 - 2.4.1 プロセス監視 46
 - 2.4.2 ログ管理 47
- 第 3 章 システムの運用 48
 - 3.1 定義生成 48
 - 3.1.1 DIOSA 環境定義 48
 - 3.1.2 TPBASE 環境定義 48
 - 3.1.3 TAM 環境定義 48
 - 3.2 起動・停止 49
 - 3.2.1 ノードごとの起動・停止順番 49
 - 3.2.2 DIOSA の起動・停止フロー 50

3.2.3	DIOSA 起動コマンド一覧	52
3.2.4	DIOSA 停止コマンド一覧	58
3.3	環境変更	60
3.3.1	サブコン固有変更	60
3.4	障害時対応	61
3.4.1	ノード障害	61
3.4.2	DIOSA 障害	63
3.4.3	TAM 障害	64
3.4.4	Oracle データベース障害	65
付録 A	リソース一覧	66
A.1	共有メモリ	66
A.2	ソケットファイル	67
付録 B	プロセス一覧	68
B.1	常駐プロセス一覧	68
B.2	TPP 一覧	71
付録 C	データベース一覧	72
C.1	Oracle データベース表	72
付録 D	諸元一覧	73
D.1	共通	73
D.2	アプリケーション実行制御	73
D.3	通信制御	73
D.4	メモリキャッシュ	73
D.5	データストア	73
付録 E	環境定義例	74
E.1	定義例の構成	74
E.2	DIOSA	76
E.2.1	DBCTRL 節	76
E.2.2	DELAYED 節	77
E.2.3	DIOSAMAP 節	79
E.2.4	IMENV 節	81
E.2.5	IMTABLECONF 節	83
E.2.6	MMG 節	87
E.2.7	OPSENV 節	88
E.2.8	SYSMAP 節	89
E.3	TPBASE	90
E.3.1	autoup.conf	90

E. 3. 2	mcs	90
E. 3. 3	DIOSA AP 定義(diosa_env.ap)	90
E. 3. 4	DIOSA VD 定義(diosa_env.vd)	90
E. 3. 5	DIOSA プロセス定義(diosa_env.ped)	90
E. 3. 6	DIOSA トランザクション定義(diosa_env.trns)	91
E. 3. 7	TCP/IP リスナ定義(TCPIP_TPLSN01.ped)	92
E. 3. 8	TCP/IP リスナ SG 定義 (TCPIP_TPLSN01.sg)	92
E. 3. 9	TCP/IP リスナ VD 定義 (TCPIP_TPLSN01.vd)	93
E. 3. 10	TCP/IP リスナ 端末定義 (TCPIP_TPLSN01.term)	93
E. 4	TAM	94
E. 4. 1	tammng.conf	94
E. 4. 2	table.conf	95
E. 4. 3	timer.conf	111
E. 4. 4	mon.conf	111
E. 5	Oracle データベース	113
E. 5. 1	TABLE 生成文	113

第1章 概要

1.1 目的と特徴

1.1.1 目的

DIOSA/XTP は、大規模アプリケーションシステム構築の汎用基盤を提供するソフトウェアです。

社会インフラの重要性が増す中、大規模アプリケーションシステムにおいても、高信頼、高性能、高運用・高稼働性、そしてアプリケーション開発の高生産性・即応性への要求が益々高まっています。

DIOSA/XTP はこれらの要件を満足すべく以下の機能を実現します。

- メモリDBを利用して高速なデータアクセスを実現しつつ、障害時の高速な復旧による業務継続を可能とします。
- システム運行の自動化・省力化、24 時間運用システムを実現するための機能により、高運用・高稼働性を実現します。

1.1.2 特徴

DIOSA/XTP は大きく分けて以下の特徴を備えています。

- アプリケーションプログラムの開發生産性向上
- 24 時間運転システムの実現
- 拡張性の高い大規模・高信頼性・高可用性システムの構築支援

(1) アプリケーションシステムの開発を容易にします。

- オンライントランザクションプログラムの基本処理構造を規定することにより、アプリケーションプログラムの独立性を高め、標準化された開発が行えます。
- 大規模・分散システムを達成するための所在管理や電文のルーティング、電文送受信の制御作業から解放されます。
- アプリケーション開発のためのトレース、性能解析といったプログラム開発の下流工程を支援する機能群により、アプリケーション開発および本番時の運用作業が軽減されます。
- メモリDBを複数のアプリケーションから同時利用可能にし、高速大量処理を可能とします。
- データを分散してメモリDBに配置し、アクセスのための所在管理とルーティング制御を行い、全データへの透過的なアクセスを可能とします。

(2) 24 時間運転システムの稼働を支援します。

- オンライン業務の稼働中に、動作中プログラムの置換を瞬時に行うことが可能であり、業務の追加・変更、プログラム障害時の緊急対応を容易に行うことができます。
- ノードの所在を意識することなく、任意のノードに対し運用指示や状態照会のコマンドを投入することができます。
- オンライン業務の稼働中に、ノード追加などシステム構成の変更や動作環境の変更を行うことが可能で

あり、柔軟なシステムの保守作業を実現します。

(3) **拡張性の高い大規模・高信頼性・高可用性システムを容易に構築することが可能です。**

- OLTP レベルあるいはノードレベルでの分散形態を多様に構成することにより、アプリケーションプログラムへの影響無く、大規模かつ高信頼なシステムを構築することができます。
- メモリ DB の稼動状態を管理し、障害時はマスタ／スレーブ切り替えを自動的に行うことを可能とします。
- Oracle Real Application Clusters を利用したクラスタ構成に対応しており、アプリケーションプログラムは切り替えを意識せずにアクセスするサーバを変更することが可能です。

1.2 構成

1.2.1 位置づけ

DIOSA/XTP は、UNIX オペレーティングシステム、および OLTP や DB などミドルウェアとアプリケーションプログラムの上に立ち、分散オンライントランザクション処理システムのミッションクリティカル性を向上させるアプリケーション実行環境として位置付けられます。

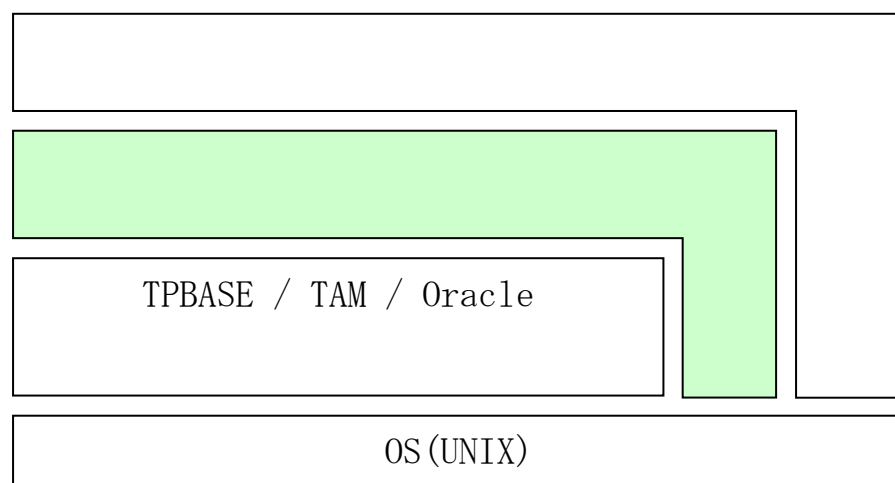
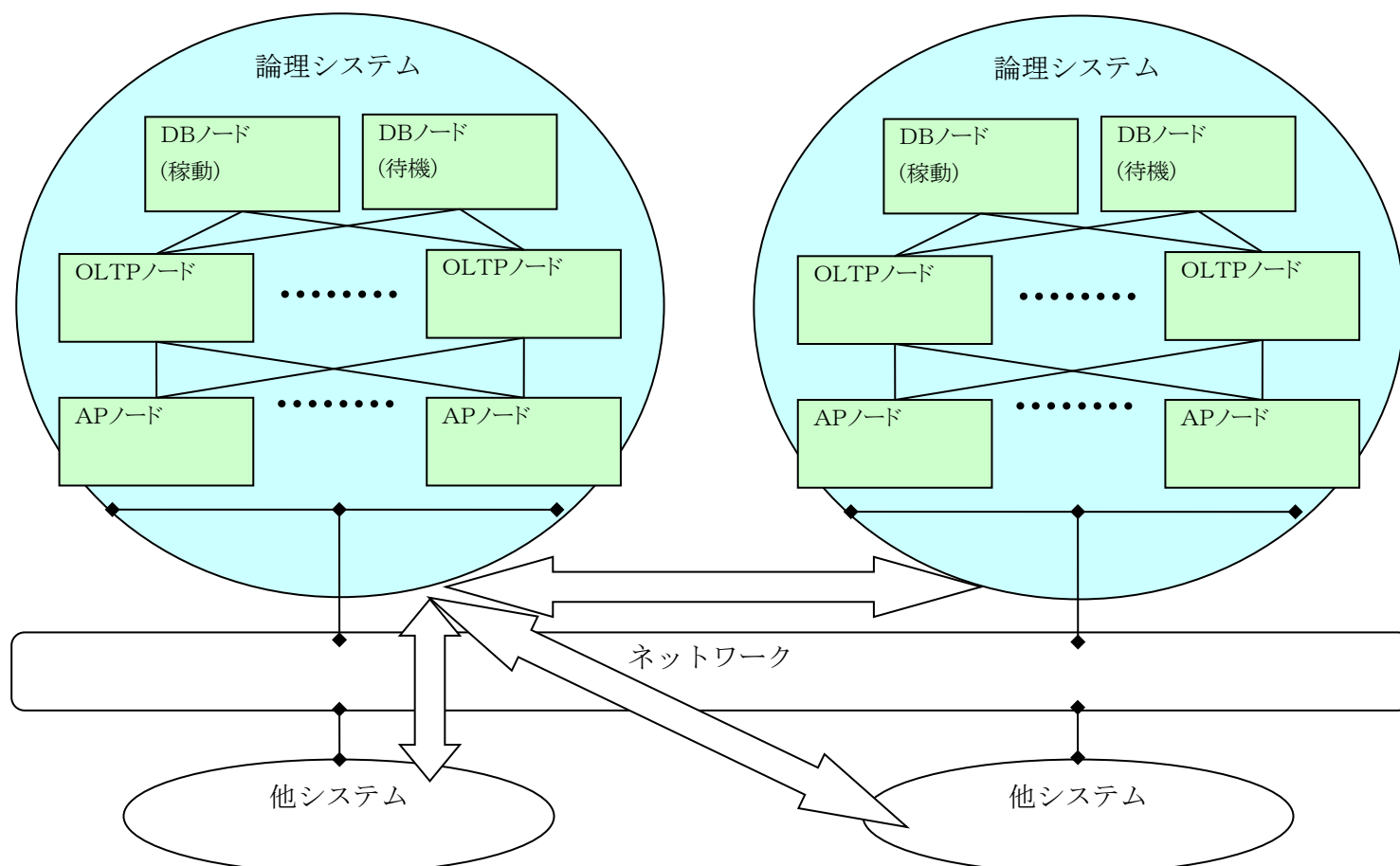


図 1-1 DIOSA/XTP の位置付け

1.2.2 システム構成

DIOSA/XTP の適用可能なシステム構成の例を以下に示します。



1. 2. 3 機能構成

DIOSA/XTP は大きく以下の有償プログラムプロダクト (PP) 群に分けて構成されています。

ライセンス	機能	関連 PP
アプリケーション実行制御	CO 制御機能 バッチアプリケーション制御機能 タイマ制御機能 メモリ管理機能 ロック制御機能 メッセージ出力機能 アプリケーショントレース機能 アプリケーション動的置換機能 経過時間監視機能 稼働統計機能 閉塞管理機能 コマンド配信機能	TPBASE TAM Oracle
通信制御	T パス管理機能 流量制御機能 データベース管理機能	TPBASE Oracle
メモリキャッシュ	インメモリサーバ インメモリサーバ所在管理機能	TAM
データストア	ディレード転送機能	TAM Oracle

1.3 諸概念

DIOSA/XTP を理解するために、予め理解しておくべき概念・用語について説明します。

なお、メモリキャッシュ、データストアに関しては、以下の利用の手引の記述を参照してください。

- ・メモリキャッシュ 利用の手引
- ・データストア 利用の手引

(1) 論理システム

巨大なシステムを構成する独立したシステムの単位であり、運用の単位、通信の単位です。

論理システムは、複数の論理ノード(AP ノード、OLTP ノード、DB ノード)から構成されます。

(2) 論理ノード

論理システムを構成するサーバ上で動作する特定の機能群です。

1つの物理サーバ上で、複数の論理ノードを動作させることが可能です。

ノード種別により動作可能な機能群が変わります。

● DB ノード

Oracle Database が動作するノードです。

主に DB の状態を管理する機能が動作します。

● OLTP ノード

アプリケーションが動作するノードです。

メモリ DB を使った高速処理のアプリケーションを実現できます。

● AP ノード

OLTP ノードと外部の通信を中継するノードです。

このノードで外部／内部のプロトコル変換などを実現します。

(3) Tパス

同一論理システム内の論理ノード間で確立する論理的な通信パスの事を指します。

(4) CO/BO

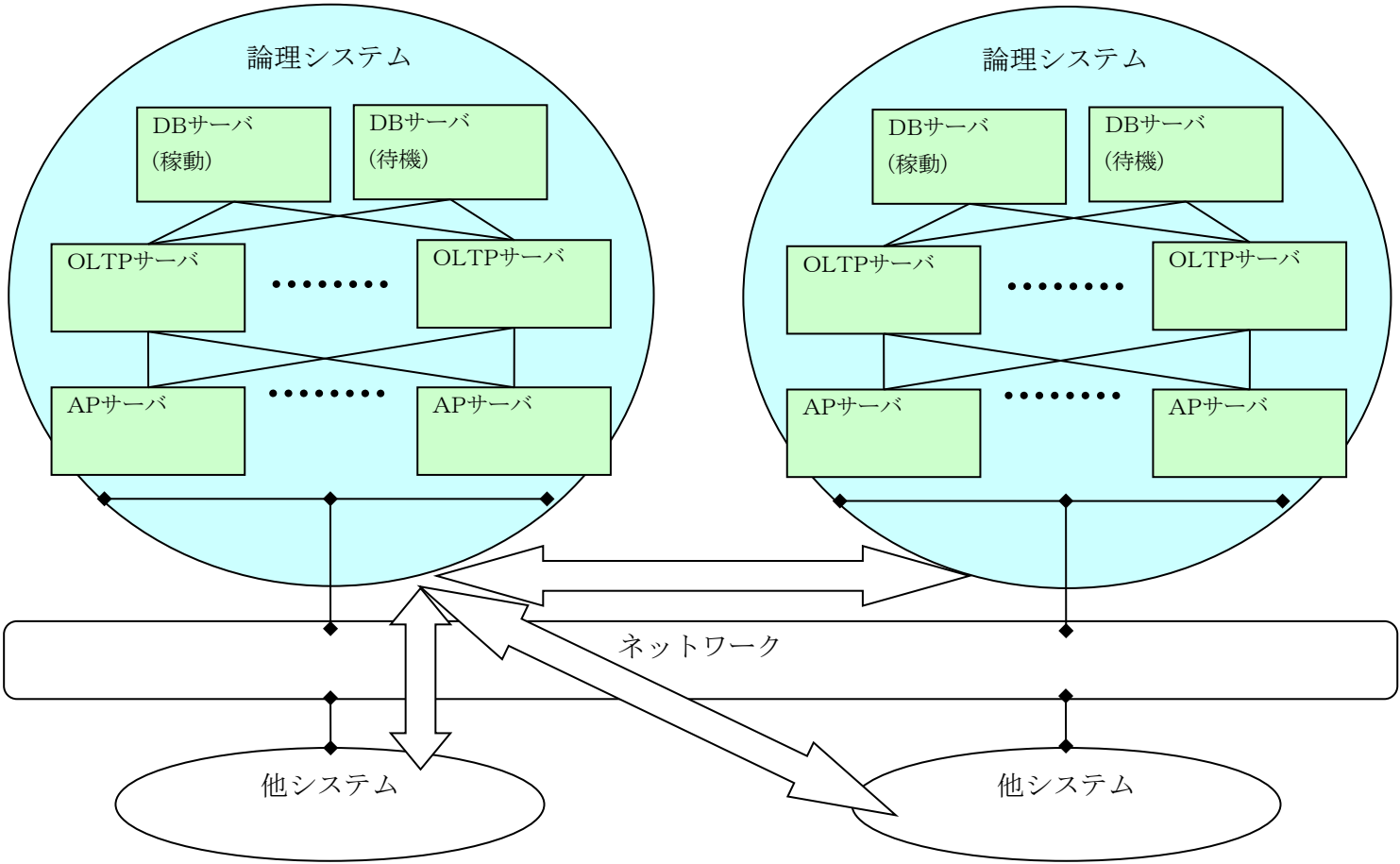
アプリケーションの制御系 (Control Object) と処理系 (Business Object) を分離した AP 構造を示しています。

COは、制御オブジェクトであり、業務要件に従ってBOの組み合わせを制御する業務プログラムです。
BOは、業務オブジェクトであり、DB更新やデータ編集などを行う業務処理部品でCOの制御下で動作します。

第2章 システムの構築

2.1 システム概要

D I O S A / X T P が適用可能なシステムの構成を示します。



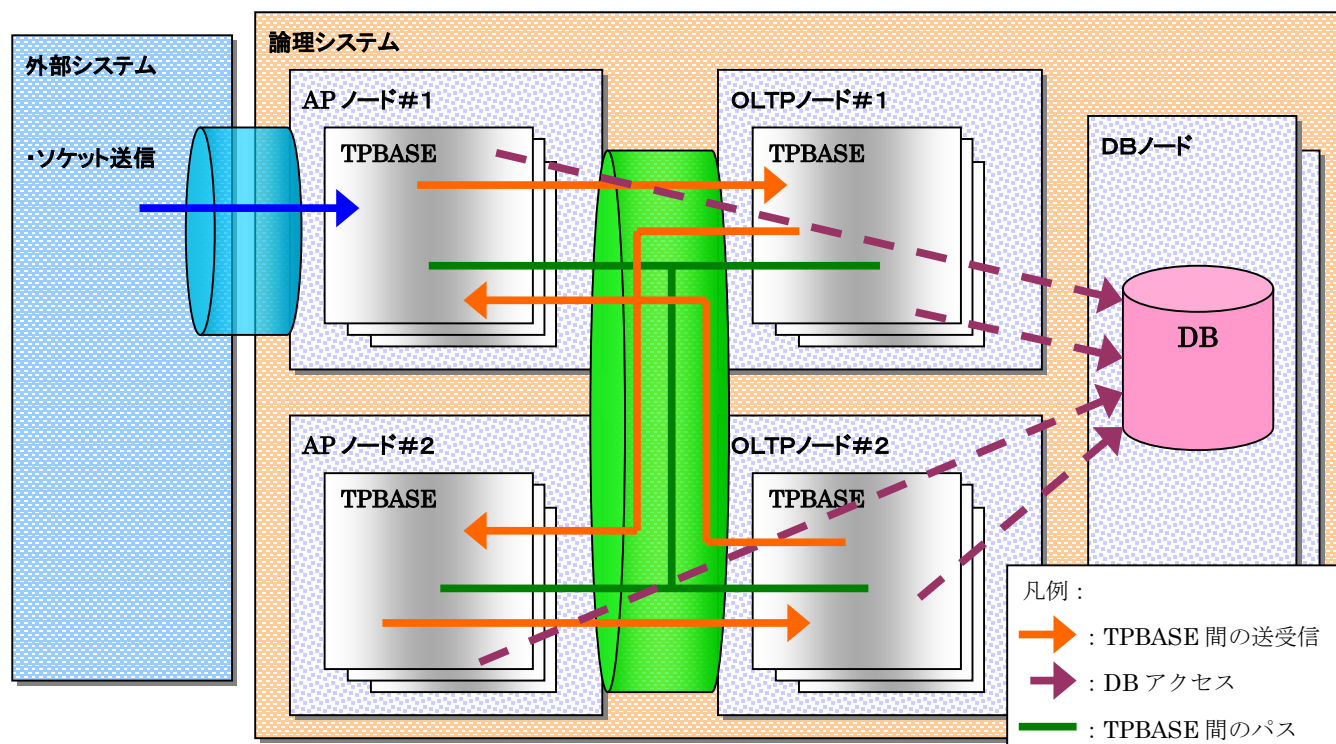
2.2 環境設計

2.2.1 ノード・ネットワーク構成

AP ノードと OLTP ノードの間では TPBASE を用いて通信を行います。外部システムと接続する場合は、外部システムは AP ノードの TPBASE に対してソケット送信をすることで DIOSA/XTP に送信を行うことができます。DIOSA/XTP から外部システムへの送信は機能提供していません。ただし利用者が TPBASE の API を使用することで、外部への応答を送信することはできます。AP ノード間、OLTP ノード間の通信はできません。

DB ノードには TPBASE を配置していないため、他ノードとの TPBASE を利用した通信はできませんが AP, OLTP ノード上からの DB へのアクセスは DIOSA/XTP が処理を隠蔽し、自動的に行います。

TPBASE の設定方法については 2.3.7 TPBASE の環境定義を参照してください。



2.2.2 DIOSA に関する設計

(1) 共通設計

(a) DIOSA/XTP 作業用ディレクトリ

DIOSA/XTP の各機能が共通的に利用するファイルの格納先ディレクトリを、環境変数(DIOSA_TMP)に設定します。

ディレクトリ配下には、以下の情報が格納されます。

{DIOSA_TMP}/{論理ノード名}/lock -----排他制御用ロックファイル

/log -----常駐プロセスからの標準出力、標準エラー出力

/socket -----ソケットファイル

(2) メッセージ出力機能

(a) DIOSA/XTP メッセージログファイル

DIOSA/XTP の運用メッセージや、エラー発生時のメッセージは、メッセージログファイルに出力されます。

ファイルの出力先は環境変数(DIOSA_MSG_LOG_PATH)で指定します。また、ファイルは指定されたファイルサイズごとにスワップして複数のファイルにローテーションしながら出力されますが、ファイルサイズやファイル数を環境変数(DIOSA_MSG_LOG_ROTATE)で指定します。

指定されたディレクトリ配下に、「diosa_{論理ノード名}_msg_log_{連番}」というファイルが作成されます。

(b) ユーザ用メッセージ原型ファイル

利用者が作成した任意のメッセージを DIOSA/XTP メッセージログファイルに出力することができます。出力のためには、メッセージを記述したテキストファイルから、ユーザ用メッセージ原型ファイルを作成する必要があります。

[作成手順]

1. メッセージを記述したテキストファイルを準備する。1 行に 1 メッセージを以下の形式で記述する。

「メッセージ ID¥t メッセージレベル¥t メッセージ種別¥t 英語メッセージ¥t 日本語メッセージ」

※¥t はタブ文字

2. ユーザ用メッセージ原型ファイル名を環境変数(DIOSA_MSG_FORM_FILEPATH)に指定します。
3. 原型メッセージファイル・メンテナンスコマンド(dimsgmtn)を実行します。

> dimsgmtn -i {1 で作成したテキストファイル}

2.2.3 TPBASE に関する設計

TPBASEの基本的な諸概念として、クラス、プロセス、トランザクションの3種類があります。

クラスはプロセスをまとめた属性定義であり、起動プロセス数、再起動回数、プロセスに渡すアーギュメント等を定義します。プロセスはOSで管理されるプロセスと同じで、クラス毎に管理されます。トランザクションは端末から1件の入力メッセージに対する一連の業務処理をいい、その識別子をトランザクションIDと呼びます。トランザクションIDはクラスに紐付けられクラス配下のプロセス群で処理されます。

TPBASEの優先度制御は、クラス内のトランザクションID毎に定義することができます。それ以外の優先度制御はできません。（クラス間の優先度制御などはできません）

クラスはプロセス数を管理しますので、クラスで起動できるプロセス数を分けることで、クラス間の違いを付けることが可能です。発生頻度の高い電文のトランザクションを集めたクラスを定義してプロセス数を多く定義する、また、発生頻度の少ないトランザクションを集めたクラスを定義してプロセス数を少なく定義するなどして、運用に合わせる設計をすることが可能です。

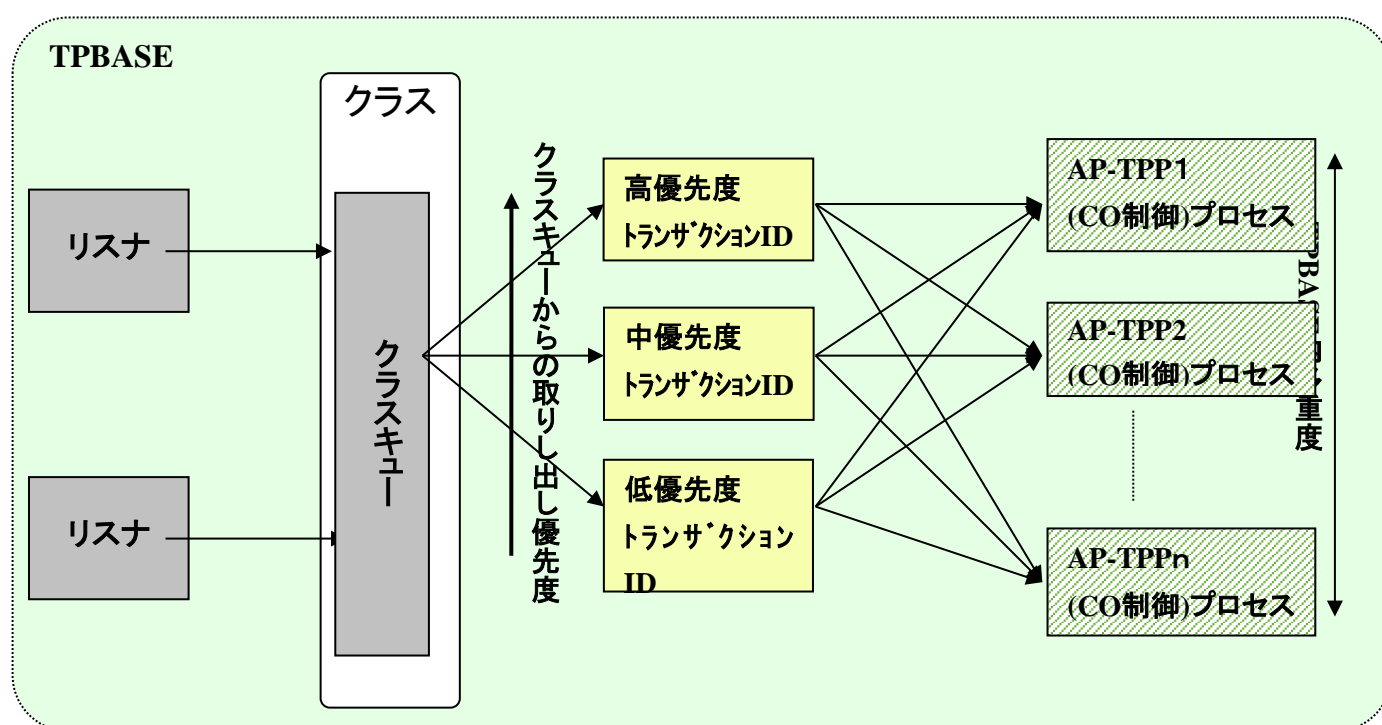
ここでは、TPBASEとDIOSA/XTPの関係から考慮すべきTPBASE環境を述べます。

(1) 1クラス方式

クラスを1つだけ定義する。

本方式では優先度の高いトランザクションを優先的に処理します。優先度の低いトランザクションは高優先度の処理の合間で動作します。

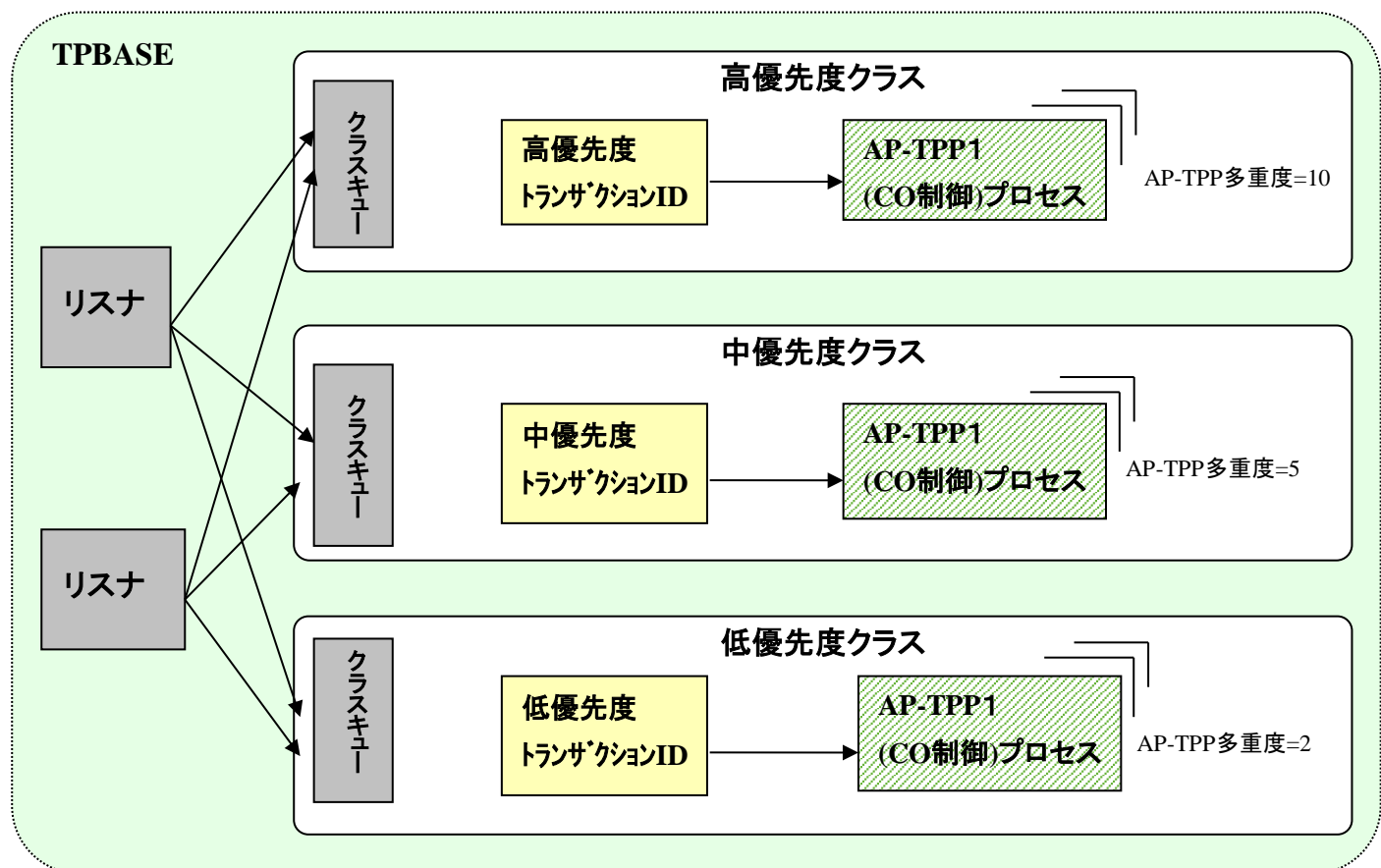
優先度の高いトランザクションが多い状況と、あまり多くない状況で優先度毎の多重度をダイナミックに切り替えます、優先度の高い、または中のトランザクションがない状況では優先度低のトランザクションが全多重度で動作できます、また、中、低優先度がプロセスを占めている割合により、高優先度トランザクションが到着しても即時動作できない場合があります。



(2) nクラス方式

nクラスを定義して、クラス毎に属性を持たせることができます。

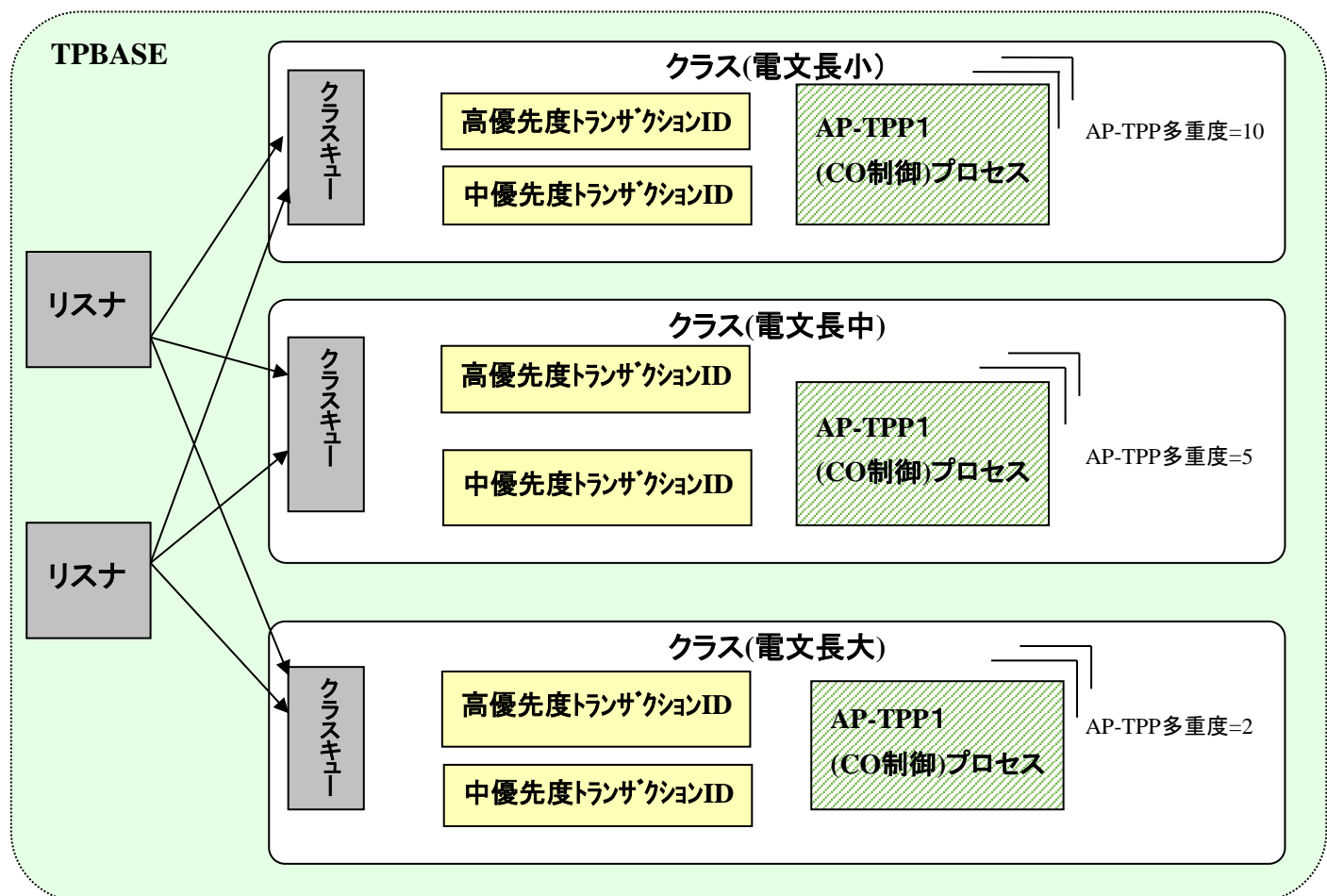
下図では便宜上高、中、低優先度クラスとしています。TPBASE 上クラス間の優先度は制御できません。本方式ではクラスと優先度を対応させ TPP 多重度を変えることによりプロセスを制御します。本方式では高優先度クラス以外の中、低優先度クラスに対応した業務も常に多重度分は動作が可能となります。優先度の低い業務が後回しになる危険性は少なくなります。しかし、高優先度のトランザクションが低優先度のトランザクションと CPU を分け合う等、影響される可能性があります。また、多重度数以上の業務を動作させることができないため、高優先度が少ない状況では低優先度の業務を効率良く動作させることはできません。



(3) nクラス、受信電文長

クラス毎に受信電文長を定義する。D I O S A / X T P ではクラス毎に受信可能な電文長を設定することができます。業務の最大電文長が100MBのように大きい場合、全てのクラス、トランザクションで最大電文長を扱えるようにすると、メモリを圧迫する危険性があります。D I O S A / X T P では電文長によるクラス分けができるようになっています。

最大電文長に近い業務電文の発生率が低い場合等は、クラスのプロセス多重度を絞るように定義します。また、発生率が高い電文長クラスは、プロセス多重度を上げる等を考慮することで、メモリ使用量を抑えることができます。



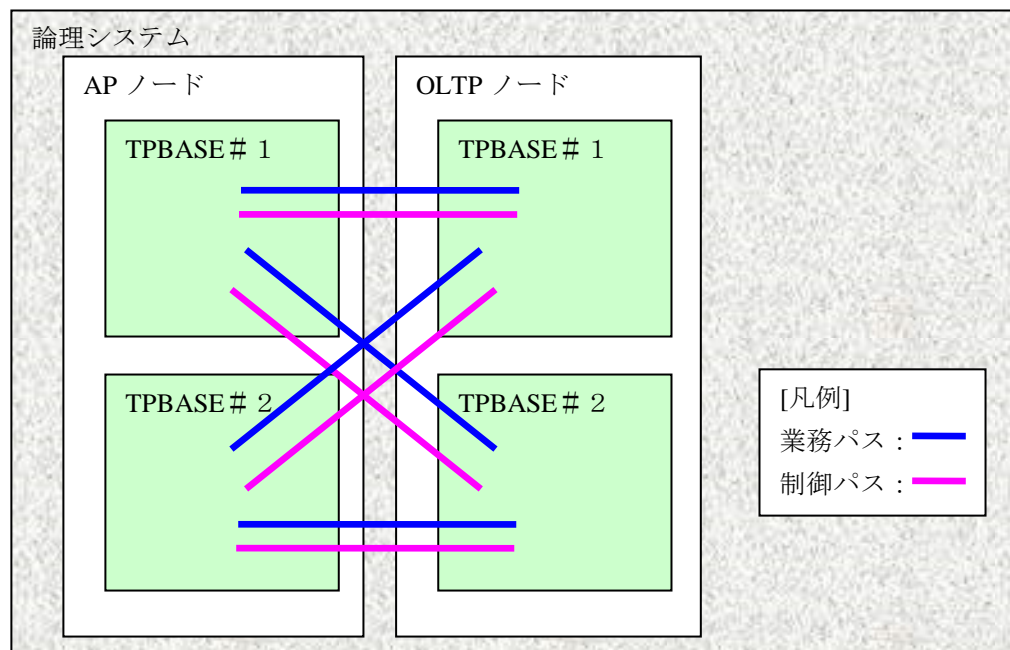
(a) TPBASEの配置

TPBASEはクラス数、プロセス数を増やしても比例してスループットが上がるわけではありません。必要に応じて1ノードにマルチTPBASEの配置を考慮してください。

(4) マルチTPBASE

TPBASEの性能が求める性能に及ばない場合は1ノードにTPBASEを複数配置することができます。

ノード内に複数のTPBASEを配置した場合、APノード、OLTPノードの各TPBASEは全てが互いに接続するように設定してください。



設定方法の詳細は 2.3.2(1) ノード間通信パス用環境定義を参照してください。

(5) **D I O S A / X T P が提供する物件**

D I O S A / X T P を利用するために T P B A S E に以下の設定を行う必要があります。

(a) 制御用 T P P

- C O 制御 T P P (C O 制御機能)
- ヘルスチェック受信 T P P (ノード間パス通信管理機能)
- ヘルスチェック送信 T P P (ノード間パス通信管理機能)
- ヘルスチェック T P P (ノード間パス通信管理機能)
- タイムアウト監視 T P P (ノード間パス通信管理機能)

(b) 制御用ライブラリ

- 通信リスナ出口ルーチン (C O 制御機能)
- 通信リスナ出口ルーチン (ノード間パス通信管理機能)

設定方法の詳細は、2.3.7 TPBASE の環境定義を参照してください。

2.2.4 TAMに関する設計

TAM の環境定義については、DIOSA/XTP メモリキャッシュ利用の手引 第4章 システムの構築を参照してください。

2.2.5 Oracle データベースに関する設計

D I O S A / X T P ではリレーショナル・データベースとして O r a c l e DB を使用しています。アプリケーションプログラムが O r a c l e DB にアクセスしデータを操作するためには、S Q L コンテキストを使用しデータベースへ接続することが必要です。また、O r a c l e DB はデータを格納するデータベース・ファイルを管理するにあたり、データベース・インスタンスを生成します。

D I O S A / X T P は O r a c l e DB とアプリケーションプログラムの間に立ち、アプリケーションプログラムにおける O r a c l e DB に関する管理の負担を軽減します。本節では、D I O S A / X T P が管理するにあたり O r a c l e に関する D I O S A / X T P 固有の概念について説明し、合わせてユーザが設計する内容を述べます。

(1) 接続方式設計

O r a c l e DB に接続するにあたって、D I O S A / X T P では 2 つの方式を選ぶことが出来ます。1 つはデータベース接続出口によるデータベース接続、もう 1 つは環境定義によるデータベース接続です。「表 1 データベース接続方式」に長所と短所を記載しますので、適切な方法を選んでください。

実際の環境定義方法は「第 I 編 2.3.3 DB 関連(DBCTRL)」を参照してください。

表 1 データベース接続方式

方式	詳細	長所	短所
データベース 接続出口	データベース接続を行うための 出口ルーチンを用意し、ルーチン 内で接続を行う方式	ユーザ I D、パスワードを環 境定義に記載することなく、 接続を行うことが出来ます	利用者が接続のためのルー チンを用意する必要があります
環境定義	環境定義にユーザ I D とパスワ ードを設定し、D I O S A / X T P が接続を行う方式	利用者が接続のためのルー チンを用意する必要があり ません	ユーザ I D、パスワードを環 境定義に記載するため、環境 定義を見られる人であれば 誰でも参照することができ ます

(2) リソースグループ設計

D I O S A / X T P はデータベース・インスタンスが2台のO r a c l e R A Cに対応しています。O r a c l e R A C機能は、複数のO r a c l e D Bサーバを経由して同一のD Bの整合性を保ちながらアクセスする機能ですが、無秩序にO r a c l e D Bサーバを選択してD Bにアクセスした場合、各サーバ上のキャッシュを同期させる処理（キャッシュフュージョン）が発生し、全体的なスループットを低下させてしまいます。利用者は「図 2 データベース・インスタンスとリソースグループの関連性」のようにO r a c l e D Bのデータベース・インスタンスとリソースグループの関連性を設計することで、D I O S A / X T P がリソースグループごとに通常利用するD Bノードを1台に限定させることにより、キャッシュフュージョンの発生を抑えることが可能です。

またD I O S A / X T Pでは、一方のデータベース・インスタンスに障害が発生した場合、もう一方のデータベース・インスタンスへと自動的に接続先を切り替えることで、高可用性を実現しています。

なお、O r a c l e R A Cを導入しないシステムで運用を行いたい場合は、データベース・インスタンス1台に全てのリソースグループを関連付けることで対応することが可能です。

実際の環境定義方法は「第 I 編 2.3.3 D B 関連(DBCTRL)」を参照してください。

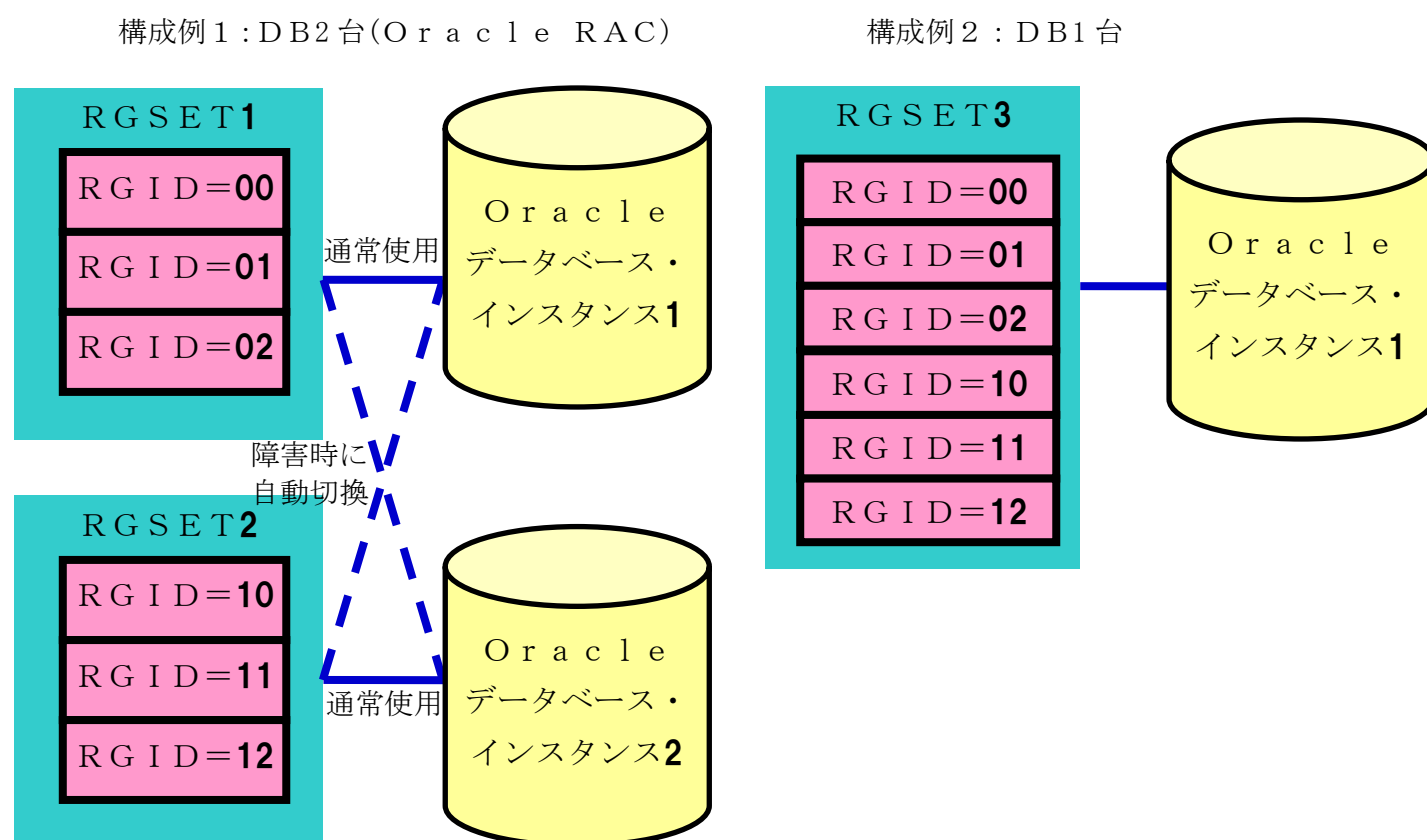


図 2 データベース・インスタンスとリソースグループの関連性

(3) インスタンスグループ設計

D I O S A / X T P では、1 台または2 台のデータベース・インスタンスをセットとして取り扱い、これをインスタンスグループと呼んでいます。インスタンスグループは複数セット用意できるため、「図 3 データベース・インスタンスとインスタンスグループの関連性」のように、同一システム内に稼動系データベース・インスタンスを併設することが可能です。データベース・インスタンスに障害が発生した場合の接続管理は、インスタンスグループ毎に行われます。

利用者はO r a c l e DBに保存するデータ領域の設計を行った上で、データベース・インスタンスとインスタンスグループの設計を行ってください。

なおD I O S A / X T P の機能では、データストアがインスタンスグループに対応しているため、実際の環境定義方法はD I O S A / X T P データストア 利用の手引および「第 I 編 2. 3. 3 DB 関連(DBCTRL)」を参照してください。

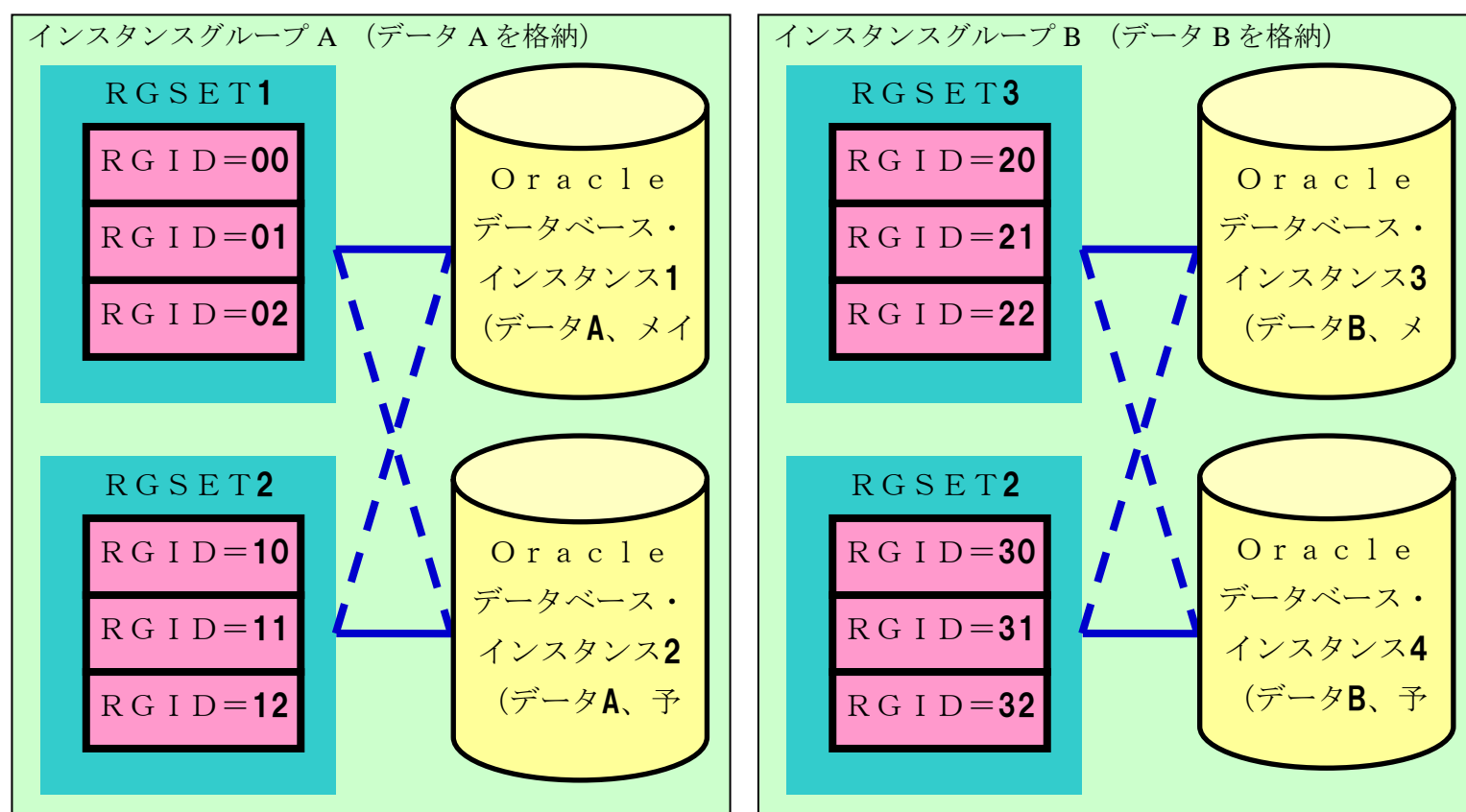


図 3 データベース・インスタンスとインスタンスグループの関連性

(4) ネット・サービス名とインスタンス識別子（S I D）の設計について

D I O S A / X T P ではインスタンス識別子を使ってSMONプロセス監視のためのプロセス名を特定します。環境定義（DBCTRL節のDBNAMEパラメータ）にはO r a c l e DBに接続するためのネット・サービス名（TNSサービス名）を指定します。また、環境定義（DBCTRL節のINSTANCENAMEパラメータ）に該当するインスタンスのインスタンス識別子（S I D）を指定します。ネット・サービス名とインスタンス識別子が同一になるように設計した場合、インスタンス識別子の環境定義への指定は省略可能です。

2.3 環境定義

2.3.1 ノード関連(DIOSAMAP、SYSAMP)

D I O S A M A P 節、S Y S M A P 節では、D I O S A の各機能が認識可能な論理システム内の論理ノードの構成や、論理システムの構成を定義します。

(1) **D I O S A M A P 節**

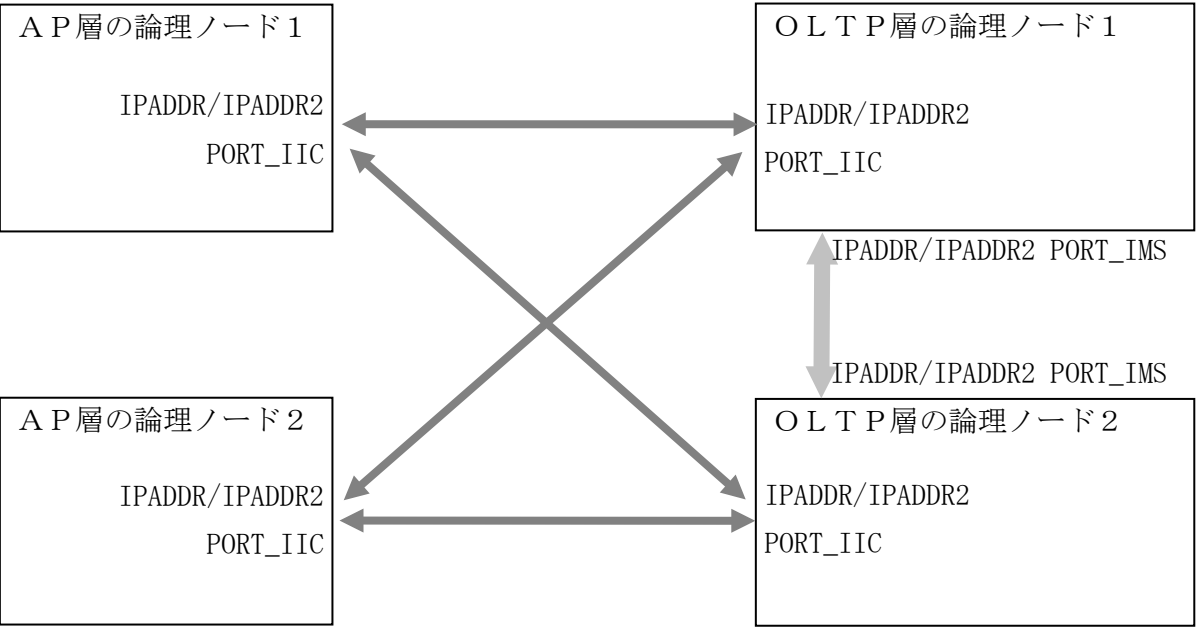
D I O S A M A P 節にはD I O S A の各機能が認識可能な論理システム内の論理ノードの構成を定義します。

D I O S A M A P 節に定義する各種 I P アドレスについて説明します。

(a) I P A D D R、I P A D D R 2

メモリキャッシュ機能が使用し、論理システム内の各論理ノードとの接続を行います。

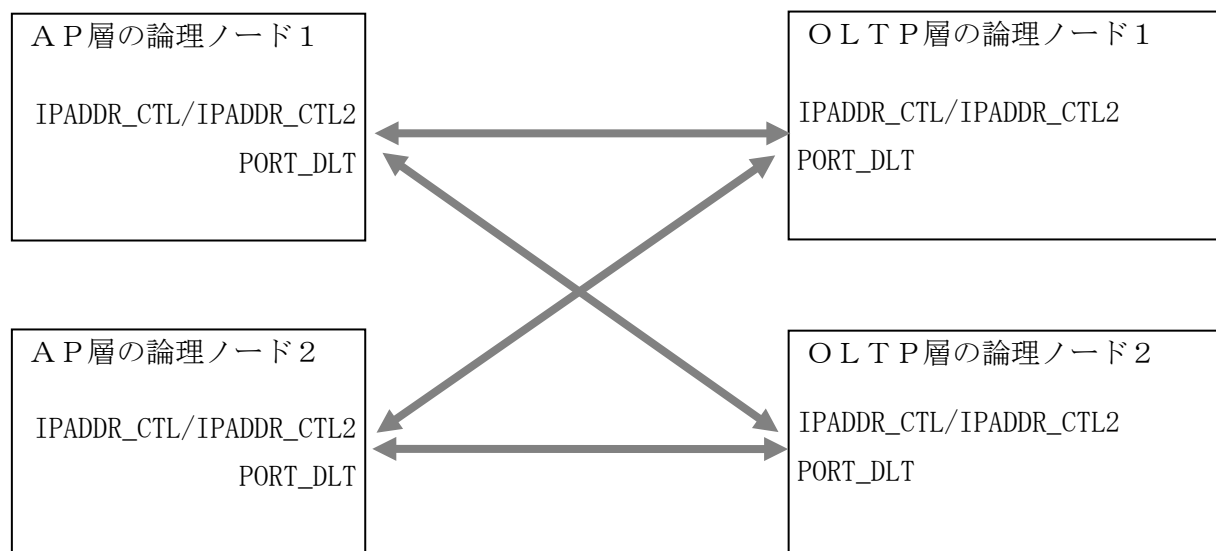
メモリキャッシュ機能の I P A D D R / I P A D D R 2 を使用した接続を以下に示します。



(b) I P A D D R _ C T L、I P A D D R _ C T L 2

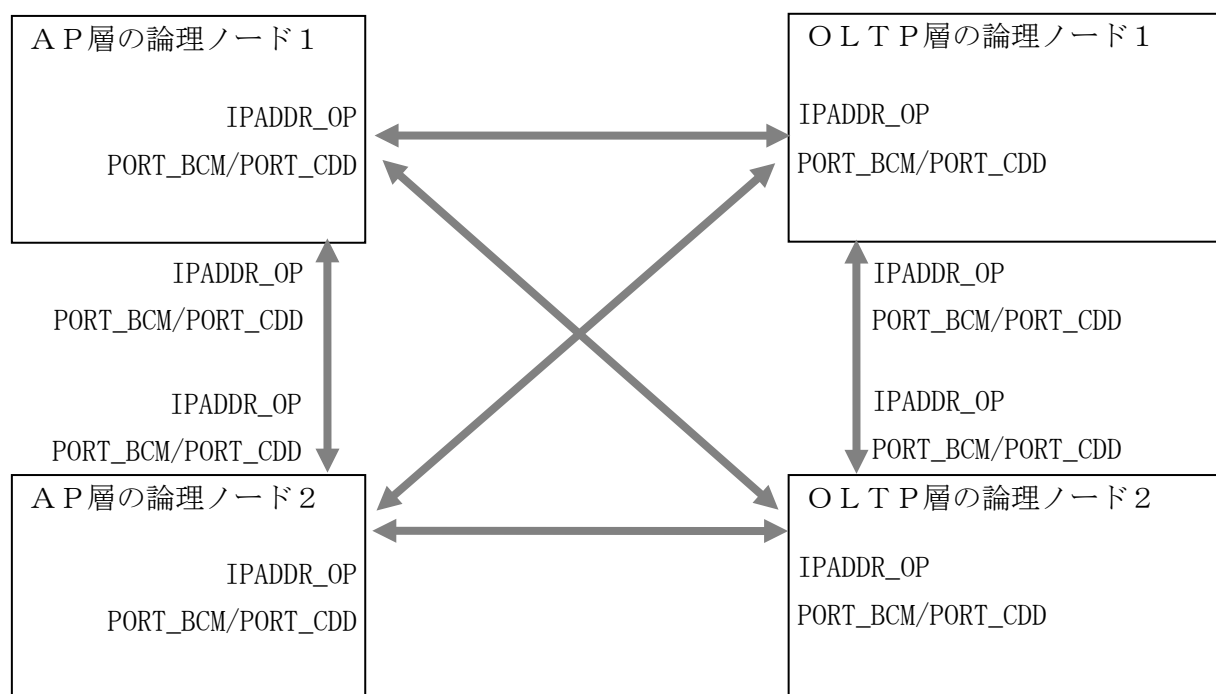
ディレード転送機能が使用し、論理システム内の各論理ノードとの接続を行います。

ディレード転送機能の I P A D D R _ C T L、I P A D D R _ C T L 2 を使用した接続を以下に示します。



(c) IPADDR_OP

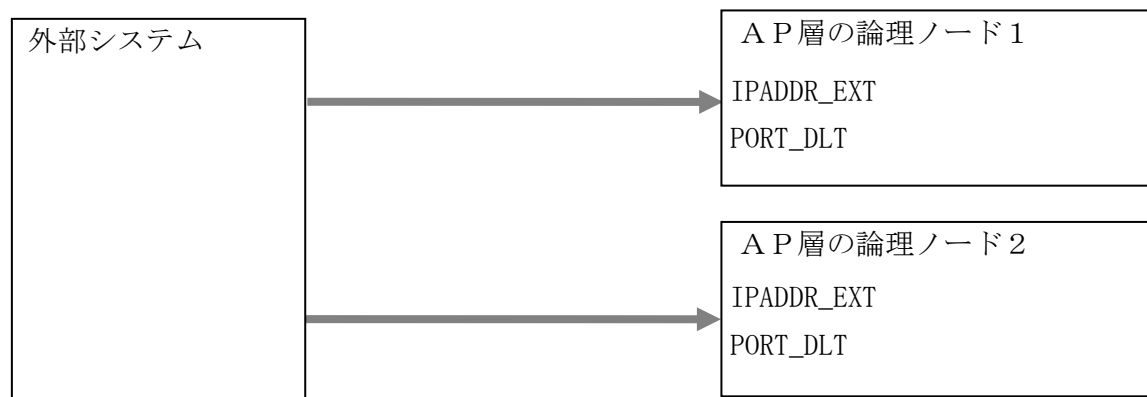
閉塞管理機能及びコマンド配信機能を使用し、論理システム内の各論理ノードとの接続を行います。
閉塞管理機能及びコマンド配信機能の IPADDR_OP を使用した接続を以下に示します。



(d) IPADDR_EXT

ディレード転送機能及びコマンド配信機能を使用し、外部システムとの接続を行います。本 IP アドレスは、外部システムからの要求を受信するために使用します。

使用例として、ディレード転送機能の IPADDR_EXT を使用した接続を以下に示します。



(2) **S Y S M A P 節**

S Y S M A P 節には、論理システムの構成を定義します。

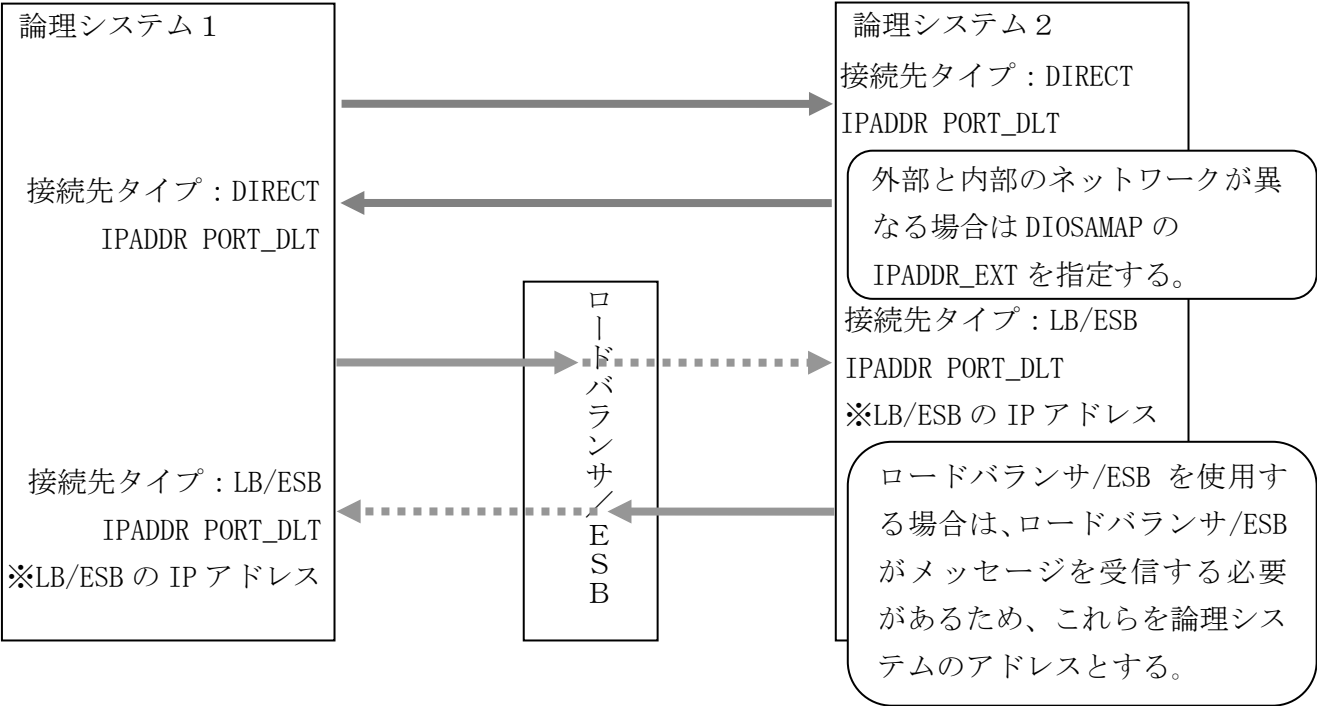
なお、S Y S M A P 節に定義された自論理システムの I P アドレスは、コマンド配信機能及びディレード転送機能が外部システム（相手論理システム）からの要求を受信するために使用します。

また、外部システム（相手論理システム）の I P アドレスについては、コマンド配信機能及びディレード転送機能が外部システム（相手論理システム）に要求を送信するために使用します。

なお、S Y S M A P の I P A D D R に D I O S A M A P の I P A D D R と異なるアドレスを指定する場合（論理システム内と論理システム外のネットワークが異なる場合）は、D I O S A M A P の I P A D D R _ _ E X T に S Y S M A P の I P A D D R を指定する必要があります。

使用例として、ディレード転送機能の I P A D D R を使用した接続を以下に示します。

r

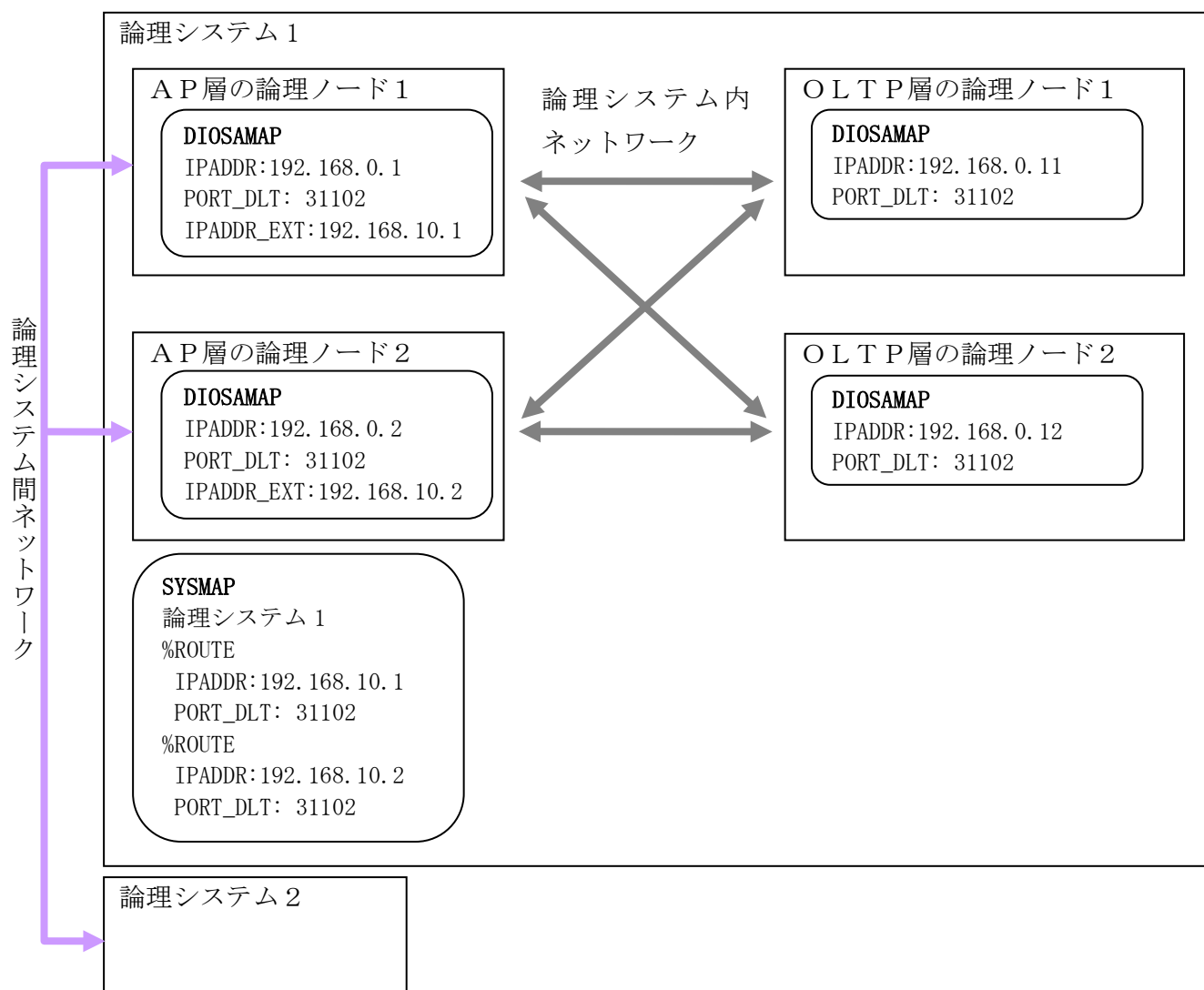


(3) **定義構成例**

(a) 他論理システムとの接続

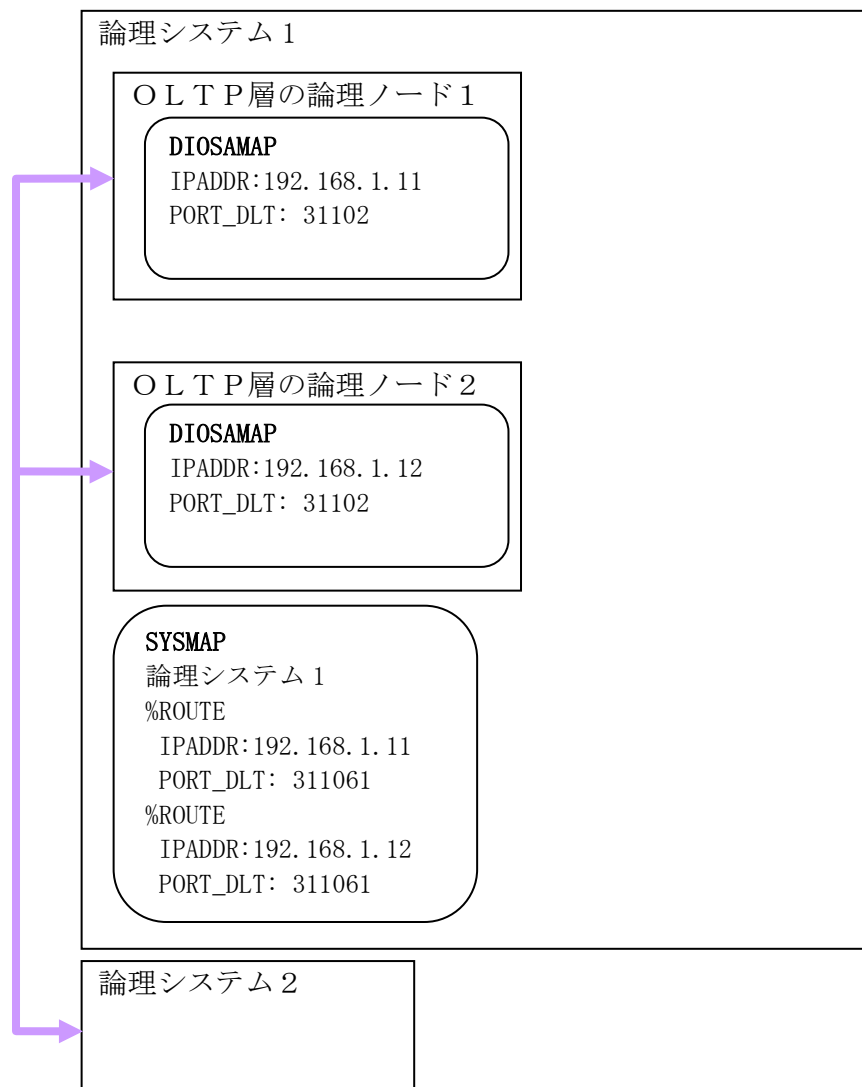
他論理システムと接続するための I P アドレスを S Y S M A P の I P A D D R に定義します。この I P アドレスが論理システム内で使用するネットワークと別になる場合は、D I O S A M A P の I P A D D R _ _ E X T にも同じアドレスを指定する必要があります。

具体的には下図の様に設定します。



(b) AP ノードを使用しない場合

AP ノードを使用しない場合は、SYSMAP の IPADDR に OLT 層の IP アドレスを指定します。また、論理システム内で使用するネットワークと外部ネットワークが同一であれば、SYSMAP の IPADDR と DIOSAMAP の IPADDR は同一の IP アドレスを指定します（IPADDR_EXT は使用しません）。



2.3.2 通信関連(TPATHENV、MFCENV)

ノード間通信パス用環境定義(TPATHENV節)では、DIOSAの各機能が認識可能な論理ノード間のTパス構成を定義します。Tパスとは、論理ノード間の通信を行うためのパスです。Tパスを定義することで、DIOSAの通信が可能となります。また、流量制御機能用環境定義(MFCENV節)では、論理システム内のOLTPノード間の負荷分散およびAPノード間の負荷分散を行うための定義です。

(1) ノード間通信パス用環境定義(TPATHENV節)

ノード間通信パス用環境定義(TPATHENV節)では、以下項目を定義します。

- 基本情報定義(TSEND項)
- Tパス構成定義(LOGSYSTEM項、APNODE項、OLTPNODE項、TPM項、MPATH項、CPATH項)
- パス定義(MPATH項、CPATH項)

(a) 基本情報定義(TSEND項)

ヘルスチェックの処理間隔やタイムアウト時間など、ノード間通信パス管理機能を使用するための基本的な情報を定義します。

(b) Tパス構成定義(LOGSYSTEM項、APNODE項、OLTPNODE項、TPM項)

論理システム内でAPノードのTPBASEがどのOLTPノードのTPBASEと接続するかを定義します。

TPATHENVはAPノードがどのOLTPノードに接続するかという構造で定義します。

APノード、OLTPノードが各2台で相互に接続する場合の定義イメージを下記に示します。

○ノード構成の定義のイメージ

```
APノード#1 ↔ OLTPノード#1
"         ↔ OLTPノード#2

APノード#2 ↔ OLTPノード#1
"         ↔ OLTPノード#2
```

※APノードから見てどのOLTPノードに接続するかを全て定義するため、OLTPノード#1, #2が重複する。

TPBASEについてもノードと同様、APノードのTPBASEがどのOLTPノードのTPモニタに接続するかという構造で定義します。

上記の例で各ノードにTPBASE 2台ずつあり、それらを相互に接続する場合の定義イメージを下記に示します。

○ノード、TPBASE構成の定義のイメージ

```

APノード#1のTPモニタ#1 ↔ OLTPノード#1のTPモニタ#1
    "                ↔ OLTPノード#1のTPモニタ#2
    "                ↔ OLTPノード#2のTPモニタ#1
    "                ↔ OLTPノード#2のTPモニタ#2
APノード#1のTPモニタ#2 ↔ OLTPノード#1のTPモニタ#1
    "                ↔ OLTPノード#1のTPモニタ#2
    "                ↔ OLTPノード#2のTPモニタ#1
    "                ↔ OLTPノード#2のTPモニタ#2
APノード#2のTPモニタ#1 ↔ OLTPノード#1のTPモニタ#1
    "                ↔ OLTPノード#1のTPモニタ#2
    "                ↔ OLTPノード#2のTPモニタ#1
    "                ↔ OLTPノード#2のTPモニタ#2
APノード#2のTPモニタ#2 ↔ OLTPノード#1のTPモニタ#1
    "                ↔ OLTPノード#1のTPモニタ#2
    "                ↔ OLTPノード#2のTPモニタ#1
    "                ↔ OLTPノード#2のTPモニタ#2

```

(c) パス定義 (MPATH項、CPATH項)

通信に使用するTPBASEのRMT名、VD名を定義します。MPATH項には利用者の電文を通信するためのRMT名、VD名を定義します。CPATH項にはDIOSA/XTPが制御情報を通信するためのRMT名、VD名を定義します。CPATH項で定義したパスによってノード間のTパスがオープンしたかどうかを判別します。

TPATHENVで定義するRMT名、VD名は、TPBASEの論理ノード間接続用の端末環境定義(*.term)、論理ノード間接続用のVD環境定義(*.vd)で定義するRMT、VDと同じ名前にする必要があります。

(2) 流量制御機能用環境定義 (MFCENV節)

流量制御機能用環境定義 (MFCENV節) では、以下項目を定義します。

- 基本情報定義 (MFCBASE項)
- 負荷レベル定義 (TRFLEVEL項、THRESHOLD項)
- 負荷監視項目定義 (LNODETRF項)
- 内部論理システム情報定義 (INNERLS項)
- 論理システム間流量制御定義 (MFCGRPTRF項、MFCLS項)
- APノード負荷分散定義 (COMMTRF項、MODETRF項)

流量制御機能はMFCENV節に定義された内容に従って監視対象の負荷状態を変更します。

MFCENV節の設定方法について説明します。

- 流量制御機能を使用するかどうかを設定します。(MFCBASE項)
- 監視対象の負荷状態を変更するための基準である負荷レベルを設定します。負荷レベルは最大15段階に分けて設定することができます(TRFLEVEL項、THRESHOLD項)。
- 監視対象をどの負荷レベルで監視するかを、監視対象ごとに設定します(LNODETRF項)。
- 監視対象がどの負荷レベルに達したら高負荷状態と判断するかの基準となるしきいレベルを監視対象ごとに設定します(INNERLS項、MFCGRPTRF項、MFCLS項、COMMTRF項、MO

DETRF項)。

環境定義の詳細については、環境定義リファレンスを参照してください。

2.3.3 DB 関連(DBCTRL)

(1) DB監視機能用環境定義 (DBCTRL節)

DB監視機能用環境定義 (DBCTRL節) では、Oracle DBのヘルスチェック機能、およびOracle DBへの接続に関する以下の項目を定義します。

- DB監視機能制御用定義 (CONTROL項)
- データベース・インスタンスの定義 (INSTANCE項)
- リソースグループセットの定義 (RGSET項)
- リソースグループIDの定義 (RGIDSET項)
- インスタンスグループの定義 (INSTANCEGRP項) (省略可)

データベースのヘルスチェックは、INSTANCE項に定義されている全てのデータベース・インスタンスに対して実施されます。データベース・インスタンスのヘルスチェックを制御するための情報 (ヘルスチェック間隔、障害と判定するまでのSQL発行リトライ回数等) をCONTROL項で定義します。

データベースのヘルスチェック時、および diosadbconnect 等のDB接続関数を実行した際に、データベースへの接続を行います。データベースへの接続には、データベース接続出口関数を使用する方法と、INSTANCE項に定義されたユーザID/パスワードを使用して接続する方法があります。

データベース接続出口関数を使用する場合は、CONTROL項のCONNECTEXITパラメータを定義します。

データベース接続出口関数を使用しない場合は、CONTROL項のCONNECTEXITパラメータを定義せず、代わりにINSTANCE項のUSERID、PASSWORDパラメータを定義します。

なお、環境変数「DIOSA_NCM_DBEXIT」が定義されている場合は環境変数が優先されるため、環境定義の内容に依らず、環境変数で指定されたデータベース接続出口関数を使用してデータベースへの接続が行われます。

以下、各項の説明を行います。詳細については、環境定義リファレンスを参照してください。

(a) DB監視機能制御用定義 (CONTROL項)

ヘルスチェック間隔などの制御に関する定義を行います。

データベースへの接続にデータベース接続出口関数を使用する場合は、本項にCONNECTEXITパラメータを定義します。

(i) 稼働DBヘルスチェック処理間隔時間 (INTERVAL1)

正常稼働中のデータベース・インスタンスをヘルスチェックする間隔を定義します。

(ii) 障害DBヘルスチェック処理間隔時間 (INTERVAL2)

障害中のデータベース・インスタンスを復旧検出するためのヘルスチェック間隔を定義します。

(iii) SQL発行のリトライ回数 (RETRY)

データベース・インスタンスを障害と判定するまでのSQL発行リトライ回数を定義します。

(iv) デフォルトで使用するリソースグループセット名 (DEFAULTRGSET)

バッチAP制御においてリソースグループセット名が省略された場合、またはリソースグループに依存しない処理の場合に、デフォルトで使用するリソースグループセットを指定します。

(v) データベース接続出口関数名 (CONNECTEXIT)

データベースに接続するためのデータベース接続出口関数名を指定します。

(vi) データベース接続多重度 (MULTI)

データベースに同時に接続できる最大多重度を定義します。

(vii) DBヘルスチェックデーモンが使用するポート番号 (PORTNUM)

DBヘルスチェックデーモンがソケット通信の際に使用するポート番号を定義します。

(b) データベース・インスタンスの定義 (INSTANCE項)

データベース・インスタンスの情報に関する定義を行います。

データベースへの接続にデータベース接続出口関数を使用しない場合は、本項にUSERIDとPASSWORDパラメータを定義します。

(i) ネット・サービス名 (DBNAME)

ネット・サービス名を指定します。

(ii) データベースノード名 (DBLNODENAME)

データベースノード名 (論理ノード名) を指定します。

(iii) ユーザ名 (USERID)

ユーザ名を指定します。

ユーザ名とDB監視機能制御用定義 (CONTROL項) のデータベース接続出口関数名の両方が定義されている場合は、データベース接続出口関数の設定が優先されます。

(iv) パスワード (PASSWORD)

パスワードを指定します。

パスワードとDB監視機能制御用定義 (CONTROL項) のデータベース接続出口関数名の両方が定義されている場合は、データベース接続出口関数の設定が優先されます。

(v) データベース・インスタンスの初期状態 (INITSTATUS)

COLDモード起動時のデータベース・インスタンスの初期状態 (稼動/待機) を定義します。

- ACTIVE : 稼動系データベース・インスタンス
- STANDBY : 待機系データベース・インスタンス

(c) リソースグループセットの定義 (RGSET項)

リソースグループが使用するデータベース・インスタンスを定義します。

本項は、diosadbconnect 等のDB接続関数で指定するリソースグループID/リソースグループセット名と、INSTANCE項で定義されたデータベース・インスタンスとを紐付けるために使用します。

(i) リソースグループセット名 (NAME)

リソースグループセットの名称を定義します。

(ii) 初期接続先データベース・インスタンス (INITINSTANCE)

COLDモード起動時の初期接続先データベース・インスタンスをネット・サービス名 (INSTANCE項のDBNAMEパラメータで指定した名前) で指定します。初期接続先は、稼動系データベース・インスタンス (INSTANCE項のINITSTATUSパラメータがACTIVE) である必要があります。

(d) リソースグループIDの定義 (RGIDSET項)

リソースグループに紐付くリソースグループIDを定義します。本項は、RGSET項内に定義する必要があります。

リソースグループIDの定義方法には、RGIDパラメータを使用して1つずつ定義する方法と、RGIDMIN/RGIDMAXパラメータを使用して範囲指定する方法の二種類があります。

(i) リソースグループID (RGID)

リソースグループセットに含まれるリソースグループIDを定義します。

(ii) 範囲指定するリソースグループIDの下限值 (RGIDMIN)

リソースグループセットに含まれるリソースグループIDを範囲指定する際の下限値を定義します。この定義は範囲指定するリソースグループIDの上限値とセットで指定します。

(iii) 範囲指定するリソースグループIDの上限値 (RGIDMAX)

リソースグループセットに含まれるリソースグループIDを範囲指定する際の上限値を定義します。この定義は範囲指定するリソースグループIDの下限値とセットで指定します。

(e) インスタンスグループの定義 (INSTANCEGRP項)

データベース・インスタンスのペアをインスタンスグループとして定義します。

接続中のデータベース・インスタンスで障害が発生した場合は、同一インスタンスグループ内に定義されている、もう一方のデータベース・インスタンスへと自動的に接続先が切り替わり、動作を続けることが可能です。

本項を定義しない場合は、INSTANCE項の先頭2つ (INSTANCE項が1つしか定義されていない場合は、その1つ) のデータベース・インスタンスがデフォルトインスタンスグループとして取り扱われます。

(i) インスタンスグループ名 (NAME)

インスタンスグループの名前を定義します。

(ii) ネット・サービス名 (DBNAME 1、DBNAME 2)

インスタンスグループに含まれるネット・サービス名 (INSTANCE項のDBNAMEパラメータで指定した名前) を2つまで定義します。

2.3.4 CO 制御関連 (COCENV)

CO制御用環境定義 (COCENV節) では、TPBASEのTPPとして動作するCO制御の実行環境を定義します。

(1) COCENV節

(a) 共通項定義 (COMMON項)

CO制御全体に対する共通的な定義をします。COが異常終了要求を返した際のプロセス停止有無、経過時間超過時の所作、プロセス開始、終了時のメッセージ出力有無等を定義することができます。

(b) 利用者出口項定義 (EXIT項)

利用者出口有無と出口名を定義します。出口名が定義されているものは出口有りと判断します。

CO制御全体に対する共通的な定義となります。出口にはプロセス初期化、プロセス終了、受信電文解析、トランザクション初期化、トランザクション終了、コミット、ロールバック、アボート#1、そしてアボート#2出口があります。

(c) 既定クラス項定義 (DEF__CLASS項)

この定義は実際存在するクラス定義ではなく、以降定義されるクラス (CLASS) 項の既定値となります。CLASS項定義で省略されたパラメータは本項の定義が採用されます。

(d) 既定トランザクション項定義 (DEF__TRANS項)

この定義は実際存在するトランザクション定義ではなく、以降定義されるトランザクション (TRANS) 項の既定値となります。TRANS項定義で省略されたパラメータは本項の定義が採用されます。

(e) クラス項定義 (CLASS項)

TPBASEのクラスに対応します。TPBASE定義のクラス名と合わせます。

クラスで受信できる最大電文長を定義することができます。受信電文長を超える電文は受信が拒否されますので注意してください。

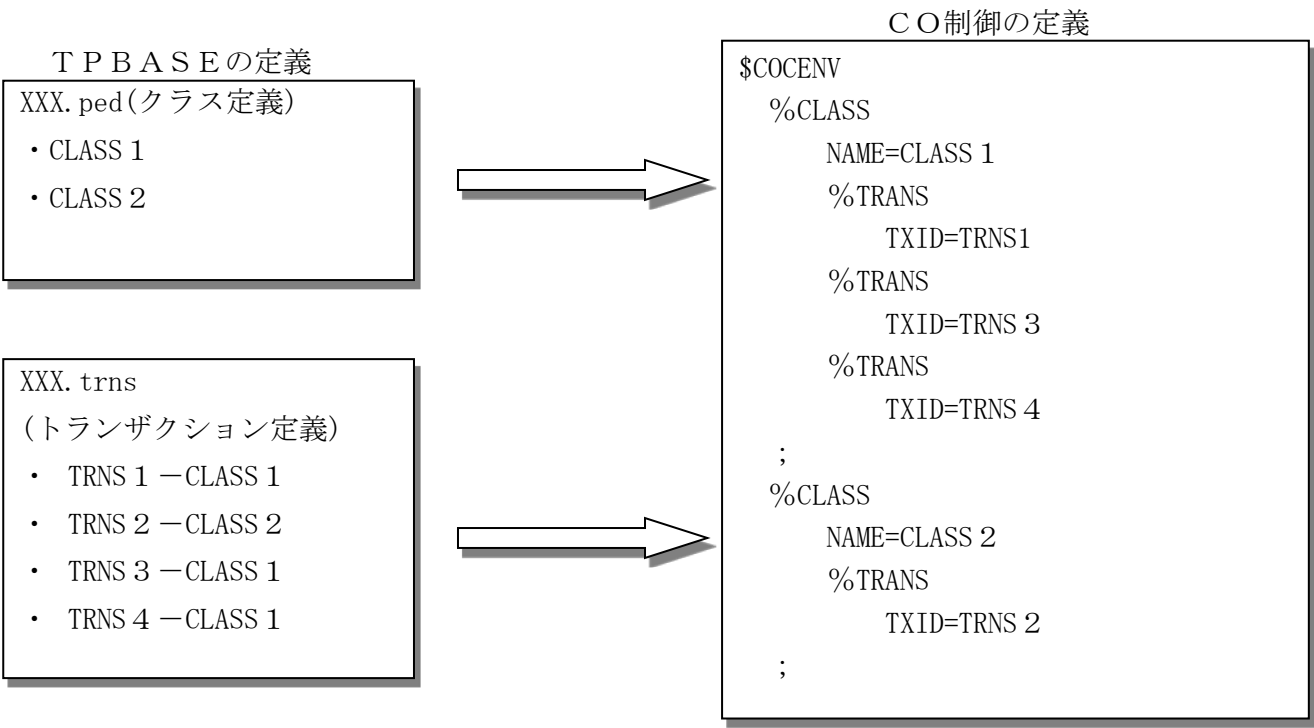
また、電文保留機能を使う場合は、電文保留用トランザクションIDを定義することが可能です。電文保留用トランザクション未定義では、そのクラスから電文保留機能は使うことができません。

(f) トランザクション項定義 (TRANS項)

TPBASEのトランザクションに対応します。ここでは、CO制御機能としての環境定義をおこないます。リトライ回数、経過監視時間、CPU時間、稼動統計採取有無等の定義をおこないます。

(g) TPBASE定義とCOCENV節

TPBASEの定義は、クラス、トランザクションのようにカテゴリで分別して定義しますが、COCENVは関係性を階層化して定義します。



2.3.5 部品関連 (APLIB、MMG、OPSENV、TMCENV)

(1) APLIB (アプリケーション動的置換機能)

アプリケーション動的置換機能を利用するには以下のいずれかをおこなう必要があります。

- ・環境定義 APLIB 節を定義する
- ・環境変数 DIOSA_LIBNAME を設定する

これらの呼び出し手段は 1 つの LM から複数利用可能ですが、関数呼び出し時の処理順序としては、まず環境定義の定義情報を参照して関数呼び出しをおこない、呼び出せなかった場合に環境変数による呼び出しをおこないます。

(a) 環境定義 APLIB 節を定義する

APLIB 節に LM 項、LLIB 項、LIBRARY 項、FUNC 項を定義することで関数呼び出しが可能となります。

FUNC 項を定義している関数のみが呼び出し可能で、LIBRARY 項に定義したライブラリに含まれている関数でも、FUNC 項が定義されていない場合は、呼び出すことはできません。

DIOSA/XTP で動作する各プロセスを APLIB 節に定義する場合の LM 名、関数名は以下のようになります。
(環境変数 DIOSA_LIBNAME に %F で指定する文字列も、下記の関数名になります。)

プロセスの種類	LM 名	関数名
DIOSA/XTP コマンド	コマンド名	出口関数名 (※1)
DIOSA/XTP デーモン	デーモンプロセス名	出口関数名 (※1)
C0 制御サーバ	TPBASE の TPP 名	出口関数名 C0 名
バッチアプリケーション	バッチ AP 制御実行コマンド名 (dibacexec)	コマンドパラメータで指定する C0 名、出口関数名
ユーザバッチ	プロセス名 (※2)	呼び出し関数名
ユーザデーモン	プロセス名 (※2)	呼び出し関数名

(※1) 出口の呼び出しをおこなわないコマンド、デーモンでは定義不要です。

(※2) シンボリックリンクを作成して LM を実行する場合、リンク先ではなく、作成したシンボリックリンクの名前を LM 名として定義する必要があります。

環境定義の詳細については、リファレンスマニュアルを参照してください。

(b) 環境変数 DIOSA_LIBNAME を設定する

各プロセスの起動時に、環境変数 DIOSA_LIBNAME に指定したライブラリ内の関数は呼び出し可能です。
ライブラリはコロン(:)で区切って 100 ライブラリまで指定することが可能で、複数指定した場合は先頭のライブラリから順にロード、関数検索処理を実行します。

ライブラリ名が関数名を含む場合、%F を記述することで、その部分は関数名に置換されます。

環境変数の詳細については、リファレンスマニュアルを参照してください。

(2) MMG(メモリ管理機能)

メモリ管理機能の利用有無に関わらず、環境定義 MMG 節を定義する必要があります。メモリ管理機能を利用しない場合、MMG 節のみを定義します。

メモリ管理機能を利用する場合、MMG 節に利用する領域種別に応じて初期サイズ、拡張サイズ、最大サイズの設定、及びプログラムアポート時のメモリダンプ出力設定を行います。

プロセスメモリに関しては、環境変数 DIOSA_MEMBUF で初期サイズ、拡張サイズ、最大サイズを設定することができます。環境定義と環境変数の両方を設定した場合、環境変数の値が有効となります。

(a) メモリサイズ設定

メモリ管理で取り扱うメモリは、アプリケーションからのメモリ割り当て要求時に既に確保済みのメモリを切り出して利用します。以下に初期サイズ、拡張サイズ、最大サイズの設定について説明します。

(i) 初期サイズ (WRINIT、RDINIT)

領域種別毎に必要なメモリの初期サイズを設定します。

以下のように領域種別毎にメモリを確保するタイミングが異なります。

- ・更新可共有メモリ／保護属性共有メモリ

DIOSA/XTP 起動時に初期サイズで指定したサイズの共有メモリを確保します。

- ・一括解放対象外プロセス内メモリ／一括解放対象外スレッド内メモリ

初回のアプリケーションからのメモリ割り当て要求時に初期サイズで指定したサイズのプロセスメモリを確保します。

- ・一括解放対象サービス内メモリ

DIOSA/XTP のプロセス初期化時に初期サイズで指定したサイズのプロセスメモリを確保します。

DIOSA/XTP のプロセス初期化処理を呼び出す全プロセス（メモリ管理機能を利用しないプロセス含む）で、初期サイズで設定したプロセスメモリが確保されます。

<注意事項>

初期サイズで設定したサイズのプロセスメモリが全 CO 制御サーバプロセスで確保されるため、初期サイズの設定には考慮が必要となります。例えば、初期サイズを小さく設定し、拡張サイズを大きく設定することで必要なメモリが効率よく確保できます。

(ii) 拡張サイズ (WRINC、RDINC)

領域種別毎に必要なメモリの拡張サイズを定義します。アプリケーションからのメモリ割り当て要求時、セグメント内の空き領域がなくなった場合に動的に拡張サイズのメモリを確保します。

<注意事項>

拡張領域は既存セグメントとは別の領域にメモリを確保する（既存の領域と連続領域とはならない）ため、アプリケーションからの 1 回のメモリ割り当てで要求する最大のメモリサイズを、拡張サイズで設定する必要があります。

(iii) 最大サイズ (WRMAX、RDMAX)

領域種別毎に必要なメモリの最大サイズを定義します。

このサイズを超えるメモリの拡張は行われません。この場合、メモリ割り当て API は DIOSA_ENOBUFS(-7) が返却されます。

(b) プログラムアボート時のメモリダンプ出力設定

以下にプログラムアボート時のメモリダンプ出力設定について説明します。

(i) 利用者用メモリアボートダンプ出力可否 (APWR、APRD、PRC、THR、SVC)

プログラムアボート時の領域種別毎にアボートダンプ出力可否を設定します。

アボートダンプを出力できるプロセスは、CO 制御サーバプロセスとバッチ AP 制御プロセスが対象となります。

(ii) 利用者用メモリアボートダンプ出力先 (DIR)

アボートダンプファイルの出力ディレクトリを設定します。省略した場合、プロセス起動時のカレントディレクトリとなります。

<注意事項>

設定するディレクトリ（省略時はプロセス起動時のカレントディレクトリ）のパーミッションは、メモリ管理の使用有無に関わらず、プロセス実行ユーザの書込み権を付与する必要があります。

(iii) 内部用メモリアボートダンプ出力可否、出力先 (PRC、DIR)

プログラムアボート時のアボートダンプ出力可否とアボートダンプ出力ディレクトリを設定します。

省略した場合、プロセス起動時のカレントディレクトリとなります。

<注意事項>

設定するディレクトリのパーミッションは、利用者用メモリアボートダンプ出力先と同様にプロセス実行ユーザの書込み権を付与する必要があります。

環境変数・環境定義の詳細については、リファレンスマニュアルを参照してください。

(3) OPSENV (稼動統計機能)

稼動統計機能を利用するためには環境定義 OPSENV 節を定義する必要があります。

(a) OPSPARAM 項

稼動統計機能の動作環境として稼動情報ファイル関連の定義を行います。

以下に各設定値について説明します。

(i) F T P ユーザ名、パスワード

稼動統計収集コマンドにて稼動情報ファイルを収集時に使用する F T P のログインユーザ名とパスワードを指定します。省略した場合、収集コマンドでのファイル転送は利用できません。

(ii) 収集ディレクトリパス

稼動統計収集コマンドにて稼動情報ファイルを収集するディレクトリパスを絶対パスで指定します。

(iii) 出力ディレクトリパス

稼動情報ファイルを格納するディレクトリパスを絶対パスで指定します。

(iv) プレフィックス

稼動情報ファイル名のプレフィックス（接頭語）を指定します。

省略時は” O P S ” が設定されます。

(v) 稼動統計ファイルのサイズ

稼動情報ファイルの 1 ファイルのサイズを指定します。

単位はMバイトで、1 ～ 1 0 0 まで指定可能です。省略時は 5 0 が設定されます。

(vi) 稼動統計ファイルの個数

稼動情報ファイルの個数を指定します。

2 ～ 1 0 0 まで指定可能で、省略時は 3 が設定されます。

(vii) 稼動統計ファイルセットの個数

稼動情報ファイルのセット数を指定します。

1 ～ 9 9 まで指定可能で、省略時は 1 が設定されます。

(viii) 強制出力間隔

メモリバッファ中の稼動統計情報をファイルに強制出力する最大経過時間を指定します。

秒単位で 0 ～ 3 6 0 0 まで指定可能です。0 の場合には強制出力は行われません。

省略時は 9 0 が設定されます。

(4) **TMCENV (タイマ制御機能環境定義)**

SG タイマ登録を行うためには環境定義 TMCENV 節を定義する必要があります。

DIOSA 起動時に ditmsgset を実行することで、SG に定義された情報でタイマ登録することができます。

タイマ制御機能の動作自体に影響はありません。

ただし SG に定義するタイマ情報の最大数は、環境変数 DIOSA_TMCTBLMAX と整合性を取る必要があります。

(最大 2048 個、環境変数省略時のデフォルト値 512 個)

環境変数・環境定義の詳細については、リファレンスマニュアルを参照してください。

2.3.6 運用関連 (CMDSEND)

(1) CMDSEND 節 (コマンド配信機能)

CMDSEND 節には、コマンド配信機能の動作環境を定義します。

(a) コマンド配信動作環境定義 (CMDSENDINFO 項)

CMDSENDINFO 項では、コマンド配信機能の動作環境を定義します。

(i) コマンド配信動作情報

コマンド配信時の動作情報として、タイムアウト時間や接続リトライ指定等を定義します。

タイムアウト時間は、APITIMEOUT (コマンド配信 API の応答待ち合わせ時間) と EXECTIMEOUT (ノード間の応答待ち合わせ時間) の 2 つについて定義します。また、配信先ノードへの接続失敗時にリトライする場合、RTRYCNT (接続リトライ回数) と RTRYINTVL (リトライインターバル時間) を定義します。

コマンド配信の結果を API 側で取得したい場合、以下の条件を満たすように各値を指定してください。

- $\text{APITIMEOUT 値} > \text{EXECTIMEOUT 値} + (\text{RTRYCNT 値} \times \text{RTRYINTV 値})$

なお、これらの値についてはコマンド配信を行う際のパラメータとして指定された場合、パラメータで指定された値が優先されます。

(ii) コマンド配信履歴ファイル情報

コマンド配信履歴を採取する場合、HSTTYPE に "NO" 以外の値を指定します。コマンド配信履歴ファイルは、「HSTFLNAME で指定した絶対パスファイル名_論理ノード名. ファイル番号」として生成されます。生成されたコマンド配信履歴ファイルは、HSTFLMAXSIZE で指定されたファイルサイズに達するまで出力され、ファイル番号が HSTFLMAXCNT で指定した値でサイクリックに利用されます。

(b) サーバグループ構成定義 (SRVGRPINFO 項)

SRVGRPINFO 項では、複数の論理ノードの集合をサーバグループとして定義します。

本項で定義したサーバグループは、コマンド配信する際の配信先として指定することが可能となります。

(c) コマンドルーティング定義 (CMDROUTING 項)

CMDROUTING 項では、コマンドに対する配信先情報を定義します。これにより、コマンド配信を行う際の配信先パラメータを省略することができ、配信先を意識することなくコマンド配信を利用することができます。

コマンドルーティングを利用する場合、対象とするコマンド毎に、以下の配信先情報を定義します。省略した場合既定値で配信先を決定します。

- 配信宛先名 (論理ノード名、論理システム名、サーバグループ名)
- 論理ノード属性種別 (全ノード、AP ノード、OLTP ノード、DB ノード)
- 配信対象種別 (全ノード、任意の 1 ノード)
- 配信元対象 (配信元ノードを配信対象に含める、含めない)

なお、上記の配信先情報を定義しているコマンドについてコマンド配信要求する際に、配信先パラメータとしてコマンドルーティングと異なる配信先情報が指定された場合、配信先パラメータで指定された情

報が優先されます。

(d) コマンド実行権限定義 (CMDPERM 項)

ユーザ及びユーザグループ毎に配信可能なコマンドを制限する場合、CMDPERM 項を定義します。

本項では、制限対象とするコマンド群を CMDGRP 項に定義し、ユーザ及びユーザグループ毎に各コマンド群に対する実行権限（実行可・実行不可）を EXPERM 項に定義します。

EXPERM 項に定義されていないコマンドの実行権限については、CDMPERM 項-DFLTPERM の定義に従い、実行可・不可が決定されます。

なお、CMDPERM 項を省略した場合、全てのコマンドが実行可能となります。

2.3.7 TPBASE の環境定義

(1) APノード、OLTPノードで共通の環境定義

ノード間通信パス管理機能、CO制御機能を使用するための定義を行います。

(a) 自動起動定義 (autoup.cnf)

TPBASE 起動時に自動組み込みを行う環境として、下記を定義します。() 内はSG定義ファイルのファイル名であり、*はTPBASEの命名規則に準じた任意の名前です。

- システムプロセスのプロセス環境定義ファイル名 (systpp.ped)
- タイマプロセスのプロセス環境定義ファイル名 (timer.ped)
- VDサーバプロセスのプロセス環境定義ファイル名 (vdserver.ped)
- DIOSA制御用プロセスのプロセス環境定義ファイル名 (*.ped)
- DIOSA制御用プロセスのAP環境定義 (*.ap)
- DIOSA制御用VDのVD環境定義ファイル名 (*.vd)
- 外部接続用リスナプロセスのプロセス環境定義ファイル名 (*.ped)
- 外部接続用の端末定義ファイル名 (DEFAULT.term)
- 論理ノード間接続用リスナプロセスのプロセス環境定義ファイル名 (*.ped)
- 論理ノード間接続用の端末定義ファイル名 (*.term)
- 論理ノード間接続用のVD定義ファイル名 (*.vd)

(b) TPBASE 構成定義 (tpbase.cnf)

TPBASE 構成定義に関する制限事項はありません。

(c) 通信制御機能定義 (mcs)

(i) AUTSTART

NOを指定します。通信パスは、DIOSAが管理します。

(ii) EXCEPTION

0100CEAFを指定します。他の指定では、DIOSAの例外時処理が正しく動作しない可能性があります。

(iii) RESPCONNECT

YESを指定します。通信パスをDIOSAが管理するために必要な定義です。

(iv) RESPTROUBLE

YESを指定します。通信パスをDIOSAが管理するために必要な定義です。

(v) SYSMSGLANG

ENGLISHを指定します。UTF-8環境で動作するために必要な定義です。

(vi) VDRECVBUF

32KBを超える電文の送受信を行う場合に最大電文長を指定します。32KB以下の電文しか取り扱わない場合は指定する必要はありません。

(d) システムプロセスのプロセス環境定義 (systpp.ped)

システムプロセスのプロセス環境定義に関する制限事項はありません。

- (e) タイマプロセスのプロセス環境定義 (timer.ped)
 タイマプロセスのプロセス環境定義に関する制限事項はありません。

- (f) VDサーバプロセスのプロセス環境定義 (vdserver.ped)
- (i) ARG S
 「-T 0」を指定します。

- (g) D I O S A制御用プロセスのプロセス環境定義 (*.ped)
- (i) ヘルスチェック受信T P P
 - ① P R O E P I L I B
 /D I O S Aインストールディレクトリ名/lib/libdxtpncomtprecvprotpb.so を指定します。
 - ② P R O E N T R Y
 d i _ n c m _ P r o l o g H l t h R c v を指定します。
 - ③ E P I E N T R Y
 d i _ n c m _ E p i l o g H l t h R c v を指定します。
 - ④ P A T H
 /D I O S Aインストールディレクトリ名/bin/dincmhlrecv を指定します。

- (ii) ヘルスチェック送信T P P
 - ① P R O E P I L I B
 /D I O S Aインストールディレクトリ名/lib/libdxtpncomtprsendprotpb.so を指定します。
 - ② P R O E N T R Y
 d i _ n c m _ P r o l o g H l t h S e n d を指定します。
 - ③ E P I E N T R Y
 d i _ n c m _ E p i l o g H l t h S e n d を指定します。
 - ④ P A T H
 /D I O S Aインストールディレクトリ名/bin/dincmhlsend を指定します。

- (iii) ヘルスチェックT P P
 - ① P R O E P I L I B
 /D I O S Aインストールディレクトリ名/lib/libdxtpncomtprespprotpb.so を指定します。
 - ② P R O E N T R Y
 d i _ n c m _ P r o l o g H l t h を指定します。
 - ③ E P I E N T R Y
 d i _ n c m _ E p i l o g H l t h を指定します。
 - ④ P A T H
 /D I O S Aインストールディレクトリ名/bin/dincmhlresp を指定します。

(iv) タイムアウト監視TPP

① PROEPILIB

/DIOSAインストールディレクトリ名/lib/libdxtpncomtptoutprotpb.so を指定します。

② PROENTRY

d i _ n c m _ P r o l o g T i m e o u t を指定します。

③ EPIENTRY

d i _ n c m _ E p i l o g T i m e o u t を指定します。

④ PATH

/DIOSAインストールディレクトリ名/bin/dincmhltmout を指定します。

(h) DIOSA制御用プロセスのAP環境定義 (*.ap)

(i) APID

任意の業務名を指定します。

(ii) APSTAT

パラメータ省略、またはENABLE（業務を自動活性する）を指定します。

(i) DIOSA制御用プロセスのトランザクション環境定義 (*.trns)

(i) DIOSA制御用TPPの共通定義

① TRSTAT

パラメータ省略、またはENABLE（自動活性を行う）を指定します。

② TPPLANG

Cを指定します。

③ TRTYPE

IN（入力型トランザクション）を指定します。

④ LINK

INIT（業務立ち上げ時にリンク）を指定します。

⑤ CPUTIME

60を指定することを推奨します。

⑥ ELPTIME

60を指定することを推奨します。

⑦ ABORT

CONTを指定します。

⑧ TIMEOVER

CONTを指定します。

(ii) ヘルスチェック受信TPP

① TRNSID

D17HRVを指定します。

② TPPLIB

/DIOSAインストールディレクトリ名/lib/libdxtpncomtprecvtpb.so を指定します。

③ T P P E N T R Y

d i _ n c m _ H l t h R c v T p p を指定します。

(iii) ヘルスチェック送信 T P P

① T R N S I D

D 1 7 H S D を指定します。

② T P P L I B

/D I O S A インストールディレクトリ名/lib/libdxtpncomtptsendtpb. so を指定します。

③ T P P E N T R Y

d i _ n c m _ H l t h S e n d T p p を指定します。

④ C L A S S

D I O S A 制御用プロセスのプロセス環境定義で指定した C L A S S で指定したクラス名を指定します。

(iv) ヘルスチェック T P P

① T R N S I D

D 1 7 H R P を指定します。

② T P P L I B

/D I O S A インストールディレクトリ名/lib/libdxtpncomtptresptpb. so を指定します。

③ T P P E N T R Y

d i _ n c m _ H l t h T p p を指定します。

④ C L A S S

D I O S A 制御用プロセスのプロセス環境定義で指定した C L A S S で指定したクラス名を指定します。

(v) タイムアウト監視 T P P

① T R N S I D

D 1 7 H T O を指定します。

② T P P L I B

/D I O S A インストールディレクトリ名/lib/libdxtpncomtptouttpb. so を指定します。

③ T P P E N T R Y

d i _ n c m _ T i m e o u t T p p を指定します。

④ C L A S S

D I O S A 制御用プロセスのプロセス環境定義で指定した C L A S S で指定したクラス名を指定します。

(j) D I O S A 制御用プロセスの V D 環境定義 (* . v d)

(i) D I O S A 制御用 V D の共通定義

① V D T Y P E

T R N S (トランザクション型 V D) を指定します。

② A P

D I O S A 制御用プロセス A P 環境定義の A P I D で指定した業務名を指定します。

(ii) ヘルスチェック送信 T P P の要求受信用 V D

① V D

V D I H L S 0 0 1 を指定します。

② T R N S

D 1 7 H S D を指定します。

(iii) タイムアウト監視 T P P の要求受信用 V D

① V D

V D I H L T 0 0 1 を指定します。

② T R N S

D 1 7 H T O を指定します。

(k) 論理ノード間接続用リスナプロセスのプロセス環境定義 (*. ped)

論理ノード間の通信を行うクラスの定義を行います。

ARGS SGFNAME パラメータで、論理ノード間接続用リスナプロセスの S G パラメータファイル名 (*. sg) を絶対パスで指定してください。

(l) 論理ノード間接続用リスナプロセスの S G パラメータ (*. sg)

(i) T P S E X I T _ C A L L

1 (リスナで T P S E X I T を呼び出す) を指定します。

(ii) T P S E X I T _ L I B N A M E

l i b d x t p c o c l s n t p b . s o を指定します。

(iii) A U T O T R M G E N

O F F (端末定義自動生成機能を使用しない) を指定します。

(m) 論理ノード間接続用の V D 環境定義 (*. vd)

論理ノード間通信用の制御パスおよび業務パス用の V D 定義を行います。

(i) V D T Y P E

T E R M (端末型 V D) を指定します。

(ii) D I S C E R R

P U R G E (非接続 V D キューには電文を送信しない、端末切断時には滞留電文を削除する) を指定します。

(n) 論理ノード間接続用の端末環境定義 (*. term)

論理ノード間接続を行うための端末定義を行います。

(i) T E R M I D

接続先の論理端末名と同じ名前を指定します。

(ii) T E R M S T A T

E N A B L E (自動接続する) を指定します。

(iii) `INITAP`

`DIOSA`制御用プロセス`AP`環境定義の`APID`で指定した業務名を指定します。

(iv) `CONNECTMODE`

`CONCURRENT`（業務、`VD`を同時に接続できる）を指定します。

(v) `PROTOCOL`

`TCP/IP`を指定します。

(vi) `SYMSG`

`NO`を指定します。

(vii) `PORTNO_NOCHK`

`NO`（接続相手の端末のポート番号をチェックする）を指定します。

(viii) `TCP_KEEPALIVE`

`YES`を指定し、`TCP/IP`レベルのキープアライブを指定します。

本パラメータを指定しない場合、ノード障害時に障害ノードへ接続された端末の切断を検出できず、端末が接続されたままの状態となる場合があります。この場合、ノード復旧時に端末が再接続されないため、`TPBASE`のコマンドを使用して、手動で切断～接続を行う必要があります。

(o) 環境変数

32KBを超える電文の送受信を行う場合には、環境変数`TP_TX_DATA_SIZE`に最大電文長（Kバイト単位）を指定します。32KB以下の電文しか取り扱わない場合は指定する必要はありません。

(2) **APノードでのみ必要な環境定義**

外部システムから電文を受信するためのリスナの定義が必要となります。

(a) 自動起動定義（`autoup.cnf`）

`TPBASE`起動時に自動組み込みを行う環境として、下記を定義します。（）内は`SG`定義ファイルのファイル名であり、*は`TPBASE`の命名規則に準じた任意の名前です。

`AP`ノード、`OLTP`ノードで共通の環境定義で作成したファイルに追加で記載をしてください。

- 外部接続用リスナプロセスのプロセス環境定義ファイル名（*.ped）
- 外部接続用の端末定義ファイル名（`DEFAULT.term`）

(b) 外部接続用リスナプロセスのプロセス環境定義（*.ped）

論理システム間通信、シミュレータとの通信を行うクラスの定義を行います。

`ARGS SGFNAME` パラメータで、外部接続用リスナプロセスの`SG`パラメータファイル名（*.sg）を絶対パスで指定してください。

(c) 外部接続用リスナプロセスの`SG`パラメータ（*.sg）

(i) `TPSEXIT_CALL`

1（リスナで`TPSEXIT`を呼び出す）を指定します。

(ii) TPSEXIT_LIBNAME

libdxtpncmlsntpb.soを指定します。

(iii) AUTOTRMGEN

ON（端末定義自動生成機能を使用する）を指定します。

(iv) MSGHD_SIZE

26～224の間の任意の値を指定する

(d) 外部接続用の端末環境定義 (DEFAULT.term)

外部論理システムとの接続、端末との接続、端末シミュレータとの接続を行うための端末定義を行います。

(i) LSNQNAME

TCP/IPを指定します。

(ii) CONNECTMODE

EXCLUSIVEを指定します。

(iii) FREEFORMAT

YESを指定します。

(iv) INITAP

DIOSA制御用プロセスAP環境定義のAPIDで指定した業務名を指定します。

(v) PROTOCOL

TCP/IPを指定します。

(3) **OLTPノードでのみ必要な環境定義**

OLTPノードのみ必要な環境定義は特にありません。

2.3.8 TAM の環境定義

TAM の環境定義については、以下のマニュアルを参照してください。

- DIOSA/XTP メモリキャッシュ利用の手引 第4章 システムの構築
- DIOSA/XTP データストア利用の手引 第4章 システムの構築

2.3.9 Oracle データベースの環境定義

D I O S A / X T P を動作させるにあたり、O r a c l e DB の制御表やストアプロシージャの生成が必要です。なお、データストアの O r a c l e DB 環境定義についてはデータストア 利用の手引の記述を参照してください。

(1) Oracle DB 表生成

O r a c l e DB 表を作成するため、サンプルとして提供している SQL ファイル ({ D I O S A / X T P インストールディレクトリ } / sample / sql 参照) を実行してください。

(a) 表の生成

S Q L ファイルを実行して表を生成します。

それぞれの表の生成用 S Q L ファイルは、サンプルの下記ファイルに記述されています。

対象	生成用ファイル
DIOSA_NCM_DBHC_01～32	create_table_ncm.sql

(2) ストアドプロシージャの生成

S Q L ファイルを実行してストアプロシージャを生成します。

```
> cd {DIOSA/XTP インストールディレクトリ}/sql/{環境の文字コード}
> sqlplus Oracle ログイン情報 @COM/create_com.sql
```

2.4 監視設計

2.4.1 プロセス監視

ほとんどの DIOSA/XTP の常駐プロセスは、DIOSA/XTP のプロセス監視機能によって監視されています。同一の常駐プロセスが一定時間内に一定回数異常終了した場合、監視機能の常駐プロセス自体が停止するため、外部のプロセス監視ツールで DIOSA/XTP の常駐プロセスを監視する場合は、監視機能の常駐プロセスとメッセージ出力をおこなうための常駐プロセスのみを監視対象としてください。

デーモンプロセスは起動パラメータとして、「-N 論理ノード名」を指定されているため、同一マシン上で複数の論理ノードを起動している場合でも識別することが可能です。

プロセス名	説明
dimsgd	メッセージ出力をおこなう
didamd	DIOSA/XTP の常駐プロセス監視をおこなう

表 2-2 外部からの監視が必要な常駐プロセス

DIOSA/XTP のプロセス監視機能のパラメータ設定は環境変数でおこないます。（詳細は環境定義リファレンスを参照してください。）

環境変数名	説明
DIOSA_DAM_DOWNCNT	再起動回数
DIOSA_DAM_MONINVL	監視間隔
DIOSA_DAM_RESINVL	再起動監視時間
DIOSA_DAM_RTYCNT	再起動失敗時のリトライ回数
DIOSA_DAM_RTYINVL	再起動失敗時のリトライ間隔

表 2-3 DIOSA/XTP プロセス監視条件設定用環境変数

また、一部の常駐プロセスについては、各機能の独自の監視機能によって監視されているため、上記環境変数とは異なる条件で監視を実施しています。

機能名	監視条件設定箇所
データベース管理機能	環境定義 DBCTRL 節
IM サーバ所在管理機能	環境定義 IMENV 節
ディレード機能	環境定義 DELAYED 節

表 2-4 DIOSA/XTP プロセス監視機能以外による常駐プロセス監視

DIOSA/XTP で動作するすべての常駐プロセスの動作ノード、起動停止方法については、「付録 B プロセス一覧」を参照してください。

2.4.2 ログ管理

以下のファイルは定期的なメンテナンス作業が必要です。

(1) メッセージログファイル

メッセージログファイルは複数のファイルをサイクリックに使用しますが、古いファイルは順次上書きされます。メッセージを履歴として全て残したい場合は、ファイルスワップ時に実行するコマンドを環境変数(DIOSA_MSG_CMD)に指定し、ファイルをコピーする等の運用が必要です。

(2) 常駐プロセスからの標準出力

常駐プロセスから標準出力や標準エラー出力に出力されたメッセージは、
{DIOSA_TMP}/{論理ノード名}/log/{常駐プロセス名}. [std|err]というファイルに格納されます。

本ファイルはファイルローテーション等の制御は実施していないため、定期的にファイルをコピーして初期化する等の運用が必要です。

第3章 システムの運用

3.1 定義生成

3.1.1 DIOSA 環境定義

作成した環境定義は、DIOSA の起動前にオブジェクトの生成を行う必要があります。

生成したオブジェクトファイルの内容にしたがい動作します。

- (a) 環境定義オブジェクトファイル作成

はじめて環境定義オブジェクトを生成するときに以下のコマンドを実行します。

```
diirmadd -E 環境定義ファイル名 ...
```

- (b) 環境定義オブジェクトファイル更新

すでに環境定義オブジェクトファイルを生成しているときに以下のコマンドで更新します。

```
diirmrep -E 環境定義ファイル名 ...
```

3.1.2 TPBASE 環境定義

作成した環境定義は、変換操作は必要ありません。

環境定義の詳細については、TPBASE のマニュアルを参照してください。

3.1.3 TAM 環境定義

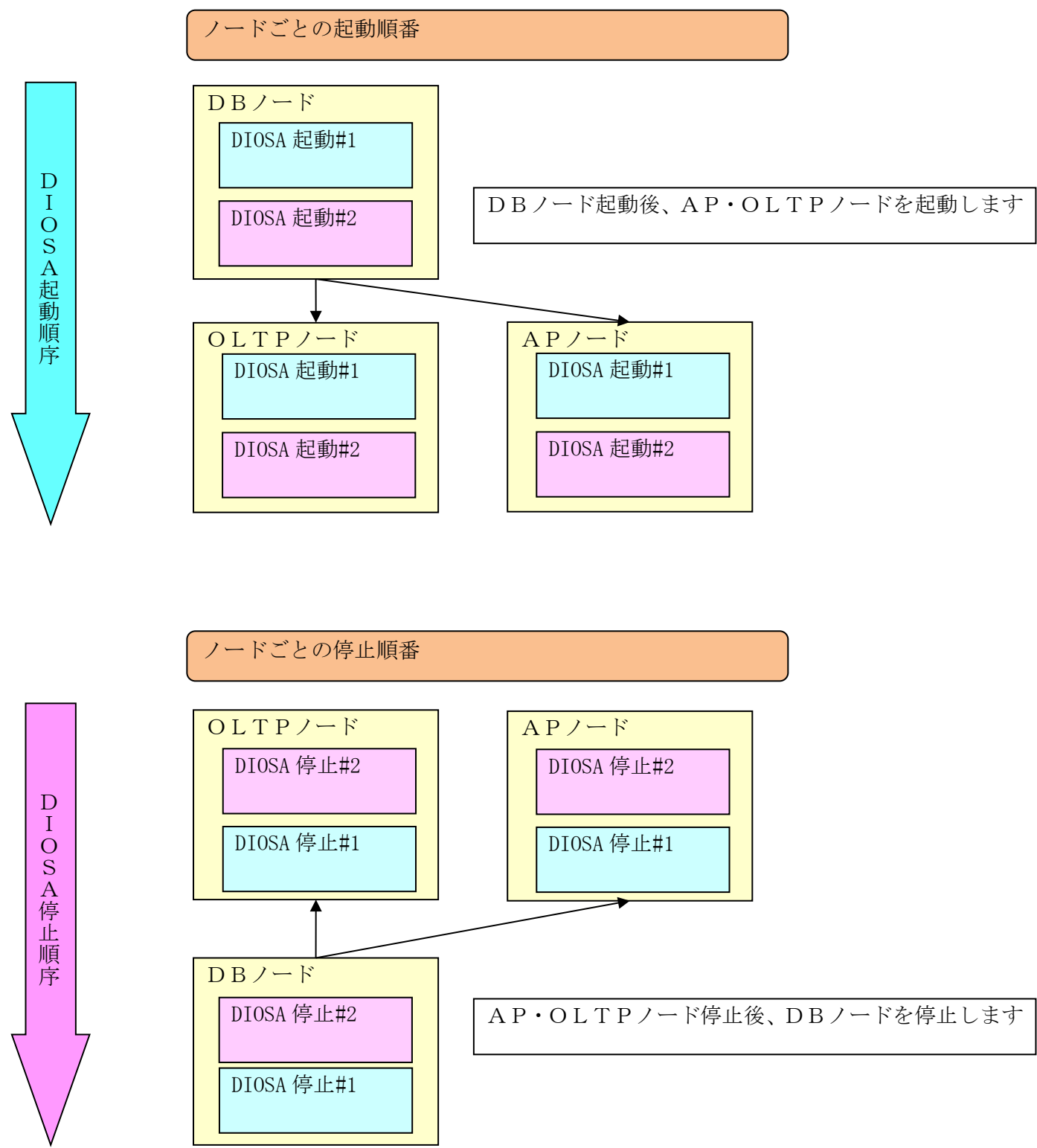
作成した環境定義は、tamcfgmaint コマンドや tamcreate を使って変換や反映する操作が必要となります。

環境定義の詳細については、TAM のマニュアルを参照してください。

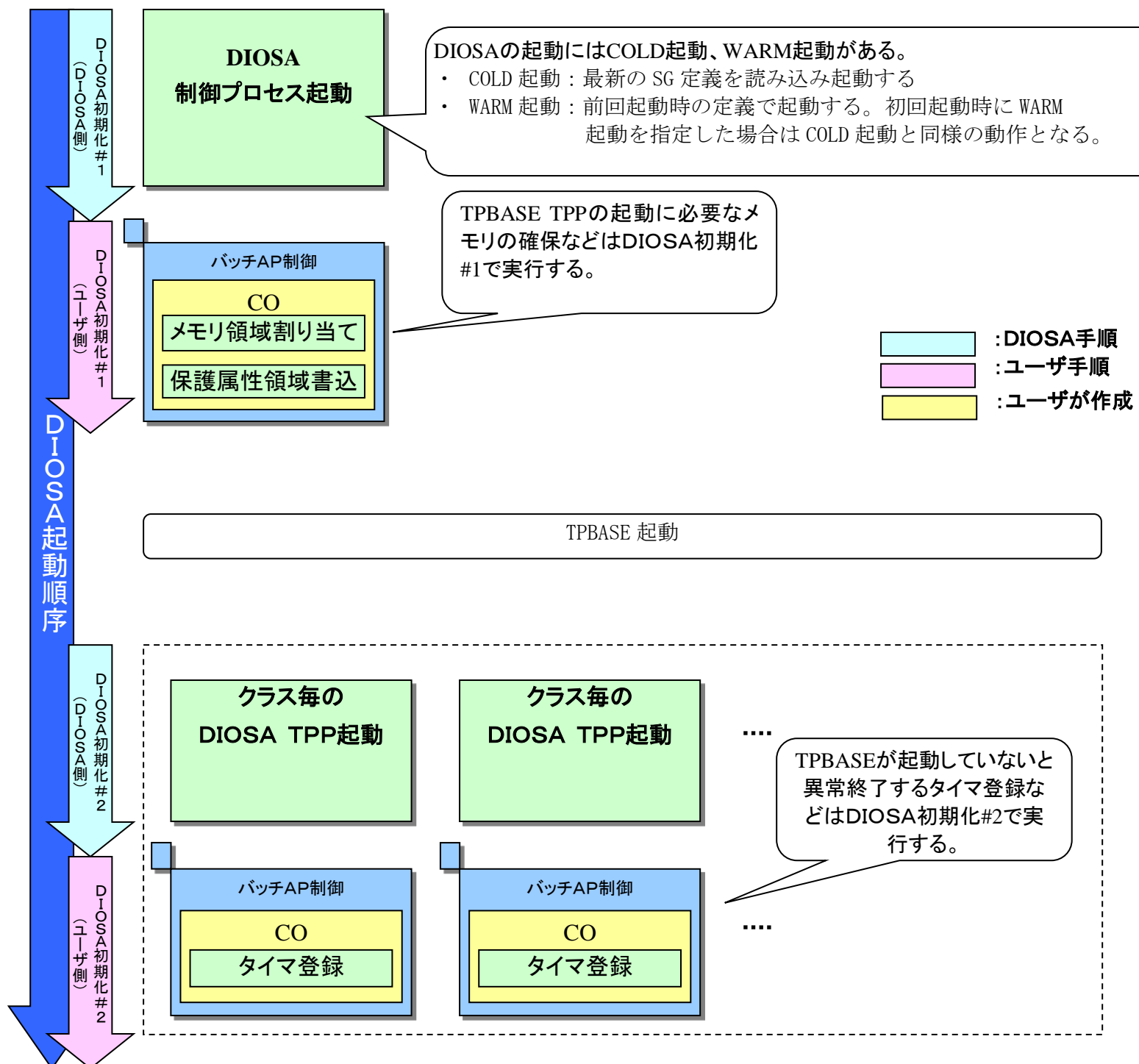
3.2 起動・停止

3.2.1 ノードごとの起動・停止順番

以下の図に DIOSA の起動停止順を記述します。



3.2.2 DIOSA の起動・停止フロー



(1) DIOSA 初期化 # 1

状態：TPBASE 起動前

説明：CO などの TPBASE TPP 稼動前にできる必要な初期化作業を行う。

たとえば、各 CO で共通に使用する共有メモリの確保とデータの設定等に用いる。

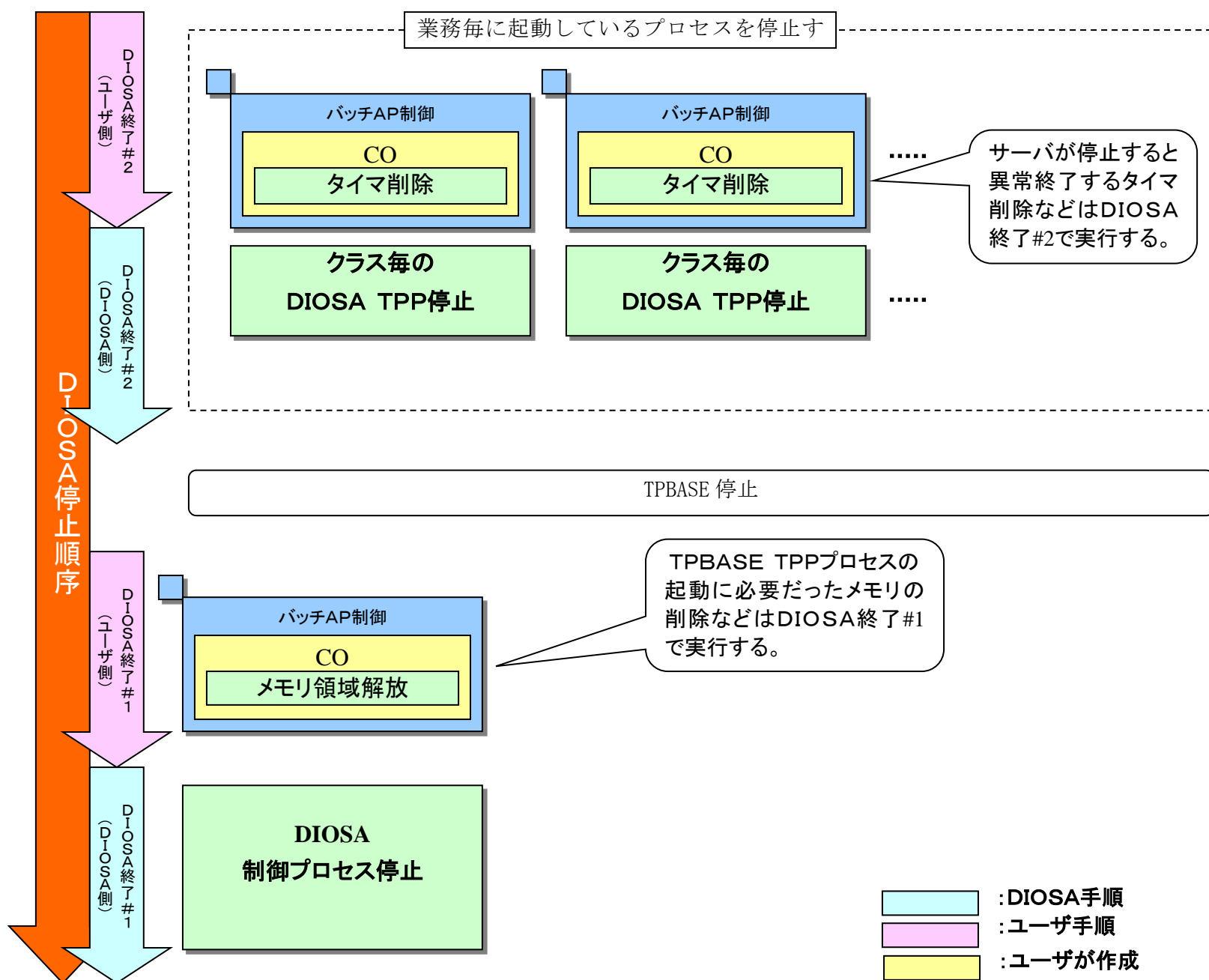
(2) DIOSA 初期化 # 2

状態：TPBASE 起動後

説明：CO などの TPBASE TPP が稼動していないとできない必要な初期化作業、業務毎の初期化作業を行う。

たとえば、CO を呼び出すタイマの登録等に用いる。

なお、TPBASE は起動するまで時間がかかる場合があるため、必要に応じて待ち合わせを行う。



(3) DIOSA終了#2

状態：TPBASE 停止前

説明:CO などの TPBASE TPP が稼動していないとできない必要な後始末作業、業務毎の後始末作業を行う。

たとえば、CO を呼び出すタイマの削除等に用いる。

(4) DIOSA終了#1

状態：TPBASE 停止後

説明：CO などの TPBASE TPP 停止後に必要な後始末作業を行う。

たとえば、各 CO で使用していた共有メモリの解放等に用いる。

3. 2. 3 DIOSA 起動コマンド一覧

以降の表にノード毎のD I O S A起動コマンド一覧を記します。

(1) DBノード

	コマンド名	オプション	実行 有無	概要
D I O S A # 1	didbginit	didbginit	必須	デバッグトレース機能の初期化处理、ログ情報ファイル出力デーモン(デーモン名: didbglogd)及び障害情報ファイル出力デーモン(デーモン名: didbgerrd)を起動します ただし、次の環境変数を設定した場合は各デーモンを起動しません DIOSA_DBGLOGPAGE =1 (または 省略)の場合、ログ情報ファイル出力デーモンは起動しません。 DIOSA_DBGERRPAGE =1 (または 省略)の場合、障害情報ファイル出力デーモンは起動しません。 環境変数の詳細については環境定義リファレンスを参照してください
	dimsgdctrl	dimsgdctrl -b	必須	メッセージ出力デーモンの起動、停止を行います
	distart_com	distart_com [-c -w]	必須	DIOSA を起動するために必要な初期化处理と、各サブコンポーネントを初期化する API を実行します
	didlrinit	didlrinit [-c -w]	必須	アプリケーション動的置換機能の初期化をおこないます
	didbglogswap	didbglogswap -f [-P 絶対パス]	選択	ログ情報保存領域を強制スワップし、ログ情報出力ファイルへ出力後、ユーザ指定によりログ情報出力ファイルに対しても強制スワップを行います。ログ情報ファイル出力デーモンがデーモン死活監視に登録されていない場合は、登録要求を行います 環境変数 DIOSA_DBGLOGPAGE =1 (または 省略)の場合は本コマンドの実行は不要です
	didbgerrswap	didbgerrswap -f [-P 絶対パス]	選択	障害情報保存領域を強制スワップし、障害情報出力ファイルへ出力後、ユーザ指定により障害情報出力ファイルに対しても強制スワップを行います。障害情報ファイル出力デーモンがデーモン死活監視に登録されていない場合は、登録要求を行います 環境変数 DIOSA_DBGERRPAGE =1 (または 省略)の場合は本コマンドの実行は不要です
	dibcmctrl	dibcmctrl -b	必須	閉塞管理デーモンを起動または停止します
	dicddctrl	dicddctrl -b	必須	コマンド配信コマンドを実行するデーモンを起動します
	dincmdbinit	dincmdbinit [-c -w]	必須	DB 管理機能の共有メモリを作成し、環境定義から情報をセットする処理を行います
	dincmdbhchkctrlinit	dincmdbhchkctrlinit	必須	DB のヘルスチェックデーモンの起動を行います

※各コマンドの詳細は、コマンドリファレンスを参照してください。

起動確認は、distatref コマンドと各機能の状態照会コマンドを使って行います。

なお、distatref コマンドで「Running」と表示された機能は、機能の起動コマンドが実行された状態であることを表していますが、各機能が現在正常に動作しているかどうかは、各機能の照会コマンド等で確認する必要があります。

状態確認の実行例

```
# distatref -v
NODE STARTUP STATUS      : Running
LNODE TYPE                : DB
LNODE NAME                 : db1
```

<INFORMATION ON EACH FUNCTION>

Start-End Function	: Running
Dynamic Library Replace Function	: Running
Daemon Alive Monitoring	: Running
Blockade Management Function	: Running
Command Delivery Function	: Running
Data Base Monitoring	: Running
Delayed Transfer	: Running

(2) APノード

	コマンド名	オプション	実行有無	概要
D I O S A # 1	didbginit	didbginit	必須	デバッグトレース機能の初期化处理、ログ情報ファイル出力デーモン(デーモン名: didbglogd)及び障害情報ファイル出力デーモン(デーモン名: didbgerrd)を起動します ただし、次の環境変数を設定した場合は各デーモンを起動しません DIOSA_DBGLOGPAGE =1 (または 省略)の場合、ログ情報ファイル出力デーモンは起動しません。 DIOSA_DBGERRPAGE =1 (または 省略)の場合、障害情報ファイル出力デーモンは起動しません。 環境変数の詳細については環境定義リファレンスを参照してください
	dimsgdctrl	dimsgdctrl -b	必須	メッセージ出力デーモンの起動、停止を行います
	distart_com	distart_com [-c -w]	必須	DIOSA を起動するために必要な初期化处理と、各サブコンポーネントを初期化する API を実行します
	diapptrcinit	diapptrcinit	選択	アプリケーショントレース出力デーモンを起動します
	didlrinit	didlrinit [-c -w]	必須	アプリケーション動的置換機能の初期化をおこないます
	didbglogswap	didbglogswap -f [-P 絶対パス]	選択	ログ情報保存領域を強制スワップし、ログ情報出力ファイルへ出力後、ユーザ指定によりログ情報出力ファイルに対しても強制スワップを行います。ログ情報ファイル出力デーモンがデーモン死活監視に登録されていない場合は、登録要求を行います 環境変数 DIOSA_DBGLOGPAGE =1 (または 省略)の場合は本コマンドの実行は不要です
	didbgerrswap	didbgerrswap -f [-P 絶対パス]	選択	障害情報保存領域を強制スワップし、障害情報出力ファイルへ出力後、ユーザ指定により障害情報出力ファイルに対しても強制スワップを行います。障害情報ファイル出力デーモンがデーモン死活監視に登録されていない場合は、登録要求を行います 環境変数 DIOSA_DBGERRPAGE =1 (または 省略)の場合は本コマンドの実行は不要です
	dibcmctrl	dibcmctrl -b	必須	閉塞管理デーモンを起動または停止します
	dieddctrl	dieddctrl -b	必須	コマンド配信コマンドを実行するデーモンを起動します
	dincmdbinit	dincmdbinit [-c -w]	選択	DB 管理機能の共有メモリを作成し、環境定義から情報をセットする処理を行います ※Oracle DB を使用しない場合は不要です。
	dincmdbhchkctrlinit	dincmdbhchkctrlinit	選択	DB のヘルスチェックデーモンの起動を行います ※Oracle DB を使用しない場合は不要です。
	dincmtpathinit	dincmtpathinit [-c -w]	選択	Tパス管理機能の初期化を行います。
	dimfcinit	dimfcinit [-c -w]	選択	流量制御機能の初期化を行います。
	diopsctrl	diopsctrl -b	選択	CO制御・パッチアプリケーション制御からの稼動情報を受信してファイルへ出力する稼動統計デーモンの起動、停止を行います
	dietgctrl	dietgctrl -b [-t 監視間隔時間]	選択	経過時間を監視するデーモンの起動、停止をおこないます
	dicocinit	dicocinit [-c -w]	選択	CO 制御の初期化处理を行います
# 2	dimfcstart	dimfcstart -b	選択	流量情報管理デーモンの起動または停止を行います
	ditmedmn	ditmedmn [-t インターバル時間]	必須	登録されたタイマを監視するデーモンを起動します
	ditmsgset	ditmsgset	選択	SG で指定されたタイマを登録します

※各コマンドの詳細は、コマンドリファレンスを参照してください。

起動確認は、distatref コマンドと各機能の状態照会コマンドを使って行います。

なお、distatref コマンドで「Running」と表示された機能は、機能の起動コマンドが実行された状態であることを表していますが、各機能が現在正常に動作しているかどうかは、各機能の照会コマンド等で確認する必要があります。

状態確認の実行例

```

# distatref -v
NODE STARTUP STATUS      : Running
LNODE TYPE               : AP
LNODE NAME               : ap1

<INFORMATION ON EACH FUNCTION>

Start-End Function       : Running
Dynamic Library Replace Function : Running
Daemon Alive Monitoring  : Running
Timer Control Function   : Running
Elapsed Time Guard Function : Running
Blockade Management Function : Running
Command Delivery Function : Running
Data Base Monitoring     : Running
T-Path Management       : Running
Message Flow Control     : Running
In-Memory Server Information Control : Running
Operation Statistics Function : Running
Control-Object Controller : Running
Delayed Transfer         : Running
Delayed Transfer / Sender : Running
Delayed Transfer / Receiver : Running
Delayed Transfer / Log Reader : Running

```

(3) O L T P ノード

	コマンド名	オプション	実行有無	概要
D I O S A # 1	didbginit	didbginit	必須	デバッグトレース機能の初期化处理、ログ情報ファイル出力デーモン(デーモン名: didbglogd)及び障害情報ファイル出力デーモン(デーモン名: didbgerrd)を起動します ただし、次の環境変数を設定した場合は各デーモンを起動しません DIOSA_DBGLOGPAGE =1 (または 省略)の場合、ログ情報ファイル出力デーモンは起動しません。 DIOSA_DBGERRPAGE =1 (または 省略)の場合、障害情報ファイル出力デーモンは起動しません。 環境変数の詳細については環境定義リファレンスを参照してください
	dimsgdctrl	dimsgdctrl -b	必須	メッセージ出力デーモンの起動、停止を行います
	distart_com	distart_com [-c -w]	必須	DIOSA を起動するために必要な初期化处理と、各サブコンポーネントを初期化する API を実行します
	diapptrcinit	diapptrcinit	選択	アプリケーショントレース出力デーモンを起動します
	didlrinit	didlrinit [-c -w]	必須	アプリケーション動的置換機能の初期化をおこないます
	didbglogswap	didbglogswap -f [-P 絶対パス]	選択	ログ情報保存領域を強制スワップし、ログ情報出力ファイルへ出力後、ユーザ指定によりログ情報出力ファイルに対しても強制スワップを行います。ログ情報ファイル出力デーモンがデーモン死活監視に登録されていない場合は、登録要求を行います 環境変数 DIOSA_DBGLOGPAGE =1 (または 省略)の場合は本コマンドの実行は不要です
	didbgerrswap	didbgerrswap -f [-P 絶対パス]	選択	障害情報保存領域を強制スワップし、障害情報出力ファイルへ出力後、ユーザ指定により障害情報出力ファイルに対しても強制スワップを行います。障害情報ファイル出力デーモンがデーモン死活監視に登録されていない場合は、登録要求を行います 環境変数 DIOSA_DBGERRPAGE =1 (または 省略)の場合は本コマンドの実行は不要です
	dibcmctrl	dibcmctrl -b	必須	閉塞管理デーモンを起動または停止します
	dieddctrl	dieddctrl -b	必須	コマンド配信コマンドを実行するデーモンを起動します
	dincmdbinit	dincmdbinit [-c -w]	選択	DB 管理機能の共有メモリを作成し、環境定義から情報をセットする処理を行います ※Oracle DB を使用しない場合は不要です。
	dincmdbhchkctrlinit	dincmdbhchkctrlinit	選択	DB のヘルスチェックデーモンの起動を行います。 ※Oracle DB を使用しない場合は不要です。
	dincmtpathinit	dincmtpathinit [-c -w]	選択	Tパス管理機能の初期化を行います。
	dimfcinit	dimfcinit [-c -w]	選択	流量制御機能の初期化を行います。
	diopsctrl	diopsctrl -b	選択	CO制御・バッチアプリケーション制御からの稼動情報を受信してファイルへ出力する稼動統計デーモンの起動、停止を行います
	dietgctrl	dietgctrl -b [-t 監視間隔時間]	選択	経過時間を監視するデーモンの起動、停止をおこないます
	dicocinit	dicocinit [-c -w]	選択	CO 制御の初期化处理を行います
# 2	dimfcstart	dimfcstart -b	選択	流量情報管理デーモンの起動または停止を行います
	ditmedmn	ditmedmn [-t インターバル時間]	必須	登録されたタイマを監視するデーモンを起動します
	ditmsgset	ditmsgset	選択	SG で指定されたタイマを登録します

※各コマンドの詳細は、コマンドリファレンスを参照してください。

起動確認は、distatref コマンドと各機能の状態照会コマンドを使って行います。

なお、distatref コマンドで「Running」と表示された機能は、機能の起動コマンドが実行された状態であることを表していますが、各機能が現在正常に動作しているかどうかは、各機能の照会コマンド等で確認する必要があります。

状態確認の実行例

```

# distatref -v
NODE STARTUP STATUS      : Running
LNODE TYPE                : OLTP
LNODE NAME                : oltp1

<INFORMATION ON EACH FUNCTION>

Start-End Function       : Running
Dynamic Library Replace Function : Running
Daemon Alive Monitoring  : Running
Timer Control Function   : Running
Elapsed Time Guard Function : Running
Blockade Management Function : Running
Command Delivery Function : Running
Data Base Monitoring     : Running
T-Path Management       : Running
Message Flow Control     : Running
In-Memory Server Information Control : Running
Operation Statistics Function : Running
Control-Object Controller : Running
Delayed Transfer         : Running
Delayed Transfer / Sender : Running
Delayed Transfer / Receiver : Running
Delayed Transfer / Log Reader : Running

```

3.2.4 DIOSA 停止コマンド一覧

以降の表にノード毎のD I O S A停止コマンド一覧を記します。

(1) DBノード

	コマンド名	オプション	実行 有無	概要
D I O S A # 1	dincmdbhchkctrlterm	dincmdbhchkctrlterm	必須	DB のヘルスチェックデーモンの停止を行います
	dincmdbterm	dincmdbterm	必須	DB 管理機能の共有メモリを削除する処理を行います
	dicddctrl	dicddctrl -e [-f force]	必須	コマンド配信コマンドを実行するデーモンを起動します
	dibcmctrl	dibcmctrl -e	必須	閉塞管理デーモンを起動または停止します
	distop	distop	必須	DIOSA を停止します
	dimsgdctrl	dimsgdctrl -e	必須	メッセージ出力デーモンの起動、停止を行います
	didbgterm	didbgterm	必須	デバッグトレース機能で使したデバッグ共有メモリを解放します。この時、障害情報ファイル出力デーモンも終了します

※各コマンドの詳細は、コマンドリファレンスを参照してください。

(2) APノード

	コマンド名	オプション	実行 有無	概要
# 2	ditmcdmstop	ditmcdmstop	必須	タイマデーモンを停止します
	dimfstart	dimfstart -e	選択	流量情報管理デーモンの起動または停止を行います
D I O S A # 1	dicocterm	dicocterm	選択	CO 制御の終了処理を行います
	dietgctrl	dietgctrl -e	選択	経過時間を監視するデーモンの起動、停止をおこないます
	diopscctrl	diopscctrl -e	選択	CO制御・パッチアプリケーション制御からの稼動情報を受信してファイルへ出力する稼動統計デーモンの起動、停止を行います
	dimfcterm	dimfcterm	選択	流量制御機能の終了処理を行います。流量制御テーブルの情報クリアおよび解放を行います
	dincmtpathterm	dincmtpathterm	選択	Tパス管理機能の共有メモリを削除する処理を行います
	dincmdbhchkctrlterm	dincmdbhchkctrlterm	選択	DB のヘルスチェックデーモンの停止を行います
	dincmdbterm	dincmdbterm	選択	DB 管理機能の共有メモリを削除する処理を行います
	dicddctrl	dicddctrl -e [-f force]	必須	コマンド配信コマンドを実行するデーモンを起動します
	dibcmctrl	dibcmctrl -e	必須	閉塞管理デーモンを起動または停止します
	diapptrcterm	diapptrcterm	選択	アプリケーショントレース出力デーモンを停止します。
	distop	distop	必須	DIOSA を停止します
	dimsgdctrl	dimsgdctrl -e	必須	メッセージ出力デーモンの起動、停止を行います
	didbgterm	didbgterm	必須	デバッグトレース機能で使したデバッグ共有メモリを解放します。この時、障害情報ファイル出力デーモンも終了します

※各コマンドの詳細は、コマンドリファレンスを参照してください。

(3) O L T P ノード

	コマンド名	オプション	実行 有無	概要
# 2	ditmcdmnstop	ditmcdmnstop	必須	タイマデーモンを停止します
	dimfcstart	dimfcstart -e	選択	流量情報管理デーモンの起動または停止を行います
D I O S A # 1	dicocterm	dicocterm	選択	CO 制御の終了処理を行います
	dietgctrl	dietgctrl -e	選択	経過時間を監視するデーモンの起動、停止をおこないます
	diopscrtl	diopscrtl -e	選択	CO 制御・バッチアプリケーション制御からの稼動情報を受信して ファイルへ出力する稼動統計デーモンの起動、停止を行います
	dimfcterm	dimfcterm	選択	流量制御機能の終了処理を行います。流量制御テーブルの情報クリ アおよび解放を行います
	dincmtpathterm	dincmtpathterm	選択	Tパス管理機能の共有メモリを削除する処理を行います
	dincmdbhchkctrlterm	dincmdbhchkctrlterm	選択	DB のヘルスチェックデーモンの停止を行います
	dincmdbterm	dincmdbterm	選択	DB 管理機能の共有メモリを削除する処理を行います
	dicddctrl	dicddctrl -e [-f force]	必須	コマンド配信コマンドを実行するデーモンを起動します
	dibcmctrl	dibcmctrl -e	必須	閉塞管理デーモンを起動または停止します
	diapptreterm	diapptreterm	選択	アプリケーショントレース出力デーモンを停止します。
	distop	distop	必須	DIOSA を停止します
	dimsgdctrl	dimsgdctrl -e	必須	メッセージ出力デーモンの起動、停止を行います
	didbgterm	didbgterm	必須	デバッグトレース機能でを使用したデバッグ共有メモリを解放しま す。この時、障害情報ファイル出力デーモンも終了します

※各コマンドの詳細は、コマンドリファレンスを参照してください。

3.3 環境変更

3.3.1 サブコン固有変更

(1) アプリケーション動的置換定義変更

プロセスの追加や、リンクするライブラリ/関数の追加、削除等を実施する場合、以下の手順で置換をおこないます。

(a) APLIB 節更新

変更後の定義に合わせて、APLIB 節の内容を修正します。

(b) 環境定義オブジェクトファイル作成

環境定義オブジェクトファイルを更新します。

```
diirmrep -E 環境定義ファイル名
```

(c) アプリケーション動的置換コマンド実行

アプリケーション動的置換コマンドを実行し、変更後の定義を有効にします。

```
didlrchg
```

コマンド実行以降に開始するトランザクション、起動するプロセスについては、全て置換後の定義に従ってアプリケーションの呼び出しをおこなうようになります。

3.4 障害時対応

3.4.1 ノード障害

D I O S A / X T P が動作するノードが異常終了した場合の復旧方法について説明します。

(1) **A P ノード障害からの復旧**

A P ノードが意図せず停止した場合、D I O S A は障害状態を認識して該当ノードへの通信を行わなくなります。また、メモリキャッシュやデータストアなどの製品が動作していた場合も同様に他のノードで処理が継続されます。出力されているメッセージ等を元に障害の原因を特定、原因を取り除いた後、A P ノードを再起動します。再起動すると自動的にノードの起動を認識しますので、特に操作を行う必要はありません。

(2) **O L T P ノード障害からの復旧**

O L T P ノードが意図せず停止した場合、D I O S A は障害状態を認識して該当ノードへの通信を行わなくなります。また、メモリキャッシュやデータストアなどの製品が動作していた場合も同様に他のノードで処理が継続されます。出力されているメッセージ等を元に障害の原因を特定、原因を取り除いた後、O L T P ノードを再起動します。再起動すると自動的にノードの起動を認識します。O L T P ノードが再起動した後に、TAM マスタを復旧したノードに移動する場合は、A P ノードで以下のコマンドを実行して、復旧したノードとの T パスが接続 (STATUS が NORMAL となっている) していることを確認してください。T パスが接続前に T A M マスタを移動した場合、A P ノードからのデータ通知が失敗する場合があります。

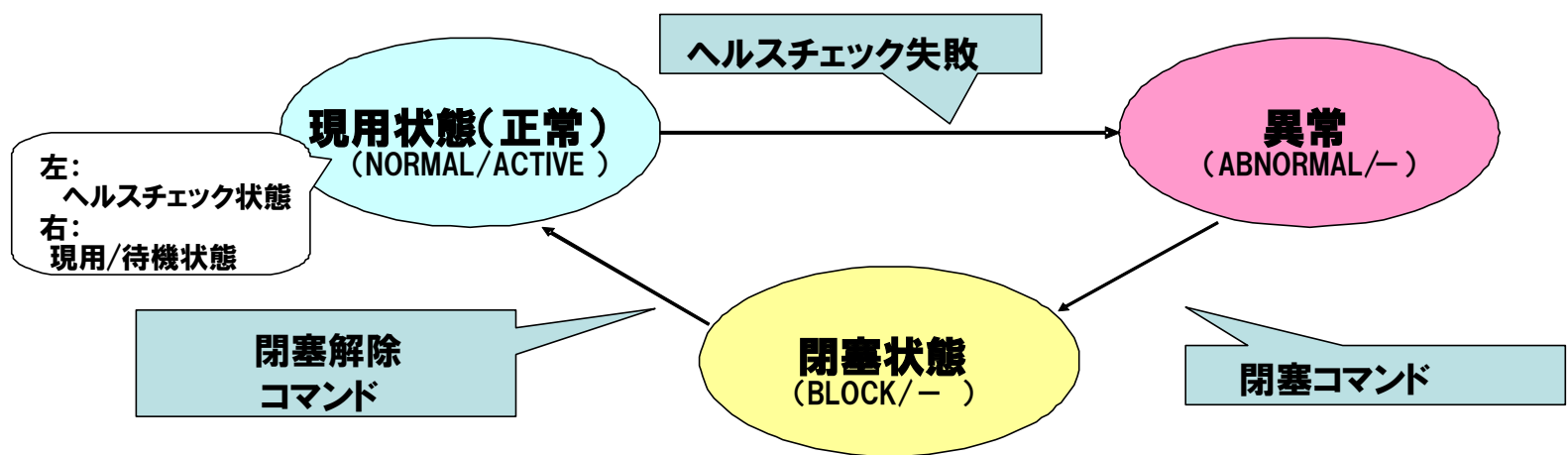
```
# dincmpathref
+===== T-PATH STATUS DISPLAY =====+      2012/06/01 10:54:25
*** T-PATH INFORMATION ***
LOGICAL SYSTEM : --
LSYSNAME       : LSYS1
NODENAME       : ap1

NODE           STATUS      LOADSTATUS   LOADLEVEL
oltp1          OPEN        NORMAL       00
oltp2          OPEN        NORMAL       00
+===== END OF DISPLAY =====+
```

(3) **D B ノード障害からの復旧**

出力されているメッセージ等を元に障害の原因を特定し、原因を取り除いた後、障害 D B ノードを再起動します。D B ノード再起動後、閉塞状態参照コマンド (dibcmref) を実行し、閉塞状態の確認を行います。当該 D B ノードが閉塞状態に移行していた場合は、何れかのノードで閉塞状態変更コマンド (dibcmupd) を実行し、当該 D B ノードの閉塞を解除します。

なお、環境変数 D I O S A _ D B I N T E G R A T E _ A U T O により閉塞を抑止していた場合は、障害 D B ノードが閉塞状態に移行することがないため、障害 D B ノードを再起動後、A P / O L T P ノードからのヘルスチェックが成功した時点で稼働系として組み込まれます。D B ノードの状態管理については下図を参照してください。



現用状態(正常)	通常に DB サーバが利用されている(利用可能)な状態です。
閉塞状態	閉塞状態の場合は、当該 DB サーバのヘルスチェックは実行されません。 閉塞された DB ノードの利用を開始する場合は、閉塞解除コマンドを実行します。
異常	ヘルスチェックが失敗すると、異常状態になりますが、自動的に閉塞コマンドが実行され、閉塞状態に移行します。ただし、閉塞を抑止している場合には、ヘルスチェックが成功すると自動的に、現用状態(正常)に移行します。

図 4 DBノードの状態管理

3.4.2 DIOSA 障害

通常の停止手順で DIOSA/XTP が停止できず強制停止させたい場合、起動中の常駐プロセスの停止、作成した IPC 資源の削除が必要です。

DIOSA/XTP を運用するユーザアカウントが他製品と別になっていて、かつ同一物理サーバ上で複数の論理ノードを起動していない場合は、該当ユーザアカウントで起動したプロセスをすべて停止し、確保した IPC 資源をすべて削除すれば強制停止できます。

上記に該当しないケースでは、以下の手順で強制停止を実施する必要があります。

(1) 常駐プロセスの停止

DIOSA の常駐プロセスは、パラメータに「-N 論理ノード名」を指定して起動されています。ps コマンドで対象プロセスを特定し、kill コマンドで強制停止させてください。

起動するプロセスの一覧については、「付録 B.1 常駐プロセス一覧」を参照してください。

(2) IPC 資源の削除

DIOSA で作成する IPC 資源は、ID の上位 4 桁が以下の定義で指定した値で作成されます。

- ・環境定義 DIOSAMAP 節の%LNODE 項-IPCKEY パラメータで指定した値
- ・環境変数 DIOSA_APPTRCIPC で指定した値
- ・環境変数 DIOSA_DBGIPC で指定した値

ipcs コマンドで対象の IPC 資源を確認し、ipcrm コマンドで削除してください。

また、一部機能では PRIVATE 指定でリソースを確保しています。

以下の方法で削除ください。

- DIOSA 起動時のメッセージ(DIMSG201)の、ID パラメータに出力された数値の ID を持つメッセージキューを削除する。
- 以下の手順を実行する。
 - didbglogswap コマンドを実行する。
 - 環境変数 DIOSA_DBGLOGPATH に指定したディレクトリ配下に作成された「DBG_log_\${DIOSA_DBGIPC}」で始まるファイルに対して、didbglog コマンドを実行する。(本ファイルについては、一定数のファイルをサイクリックに使用するため、確実に情報を残しておくためには、スワップされたタイミングでファイルを退避する運用が必要です。)
 - IPC_CREAT と、IPC_RMID のキーワードで作成/削除された IPC リソースの ID が記録されているので、IPC_CREAT が存在し、IPC_RMID が存在しない(IPC_RMID が存在するが、直後に出力されている errno が 0 以外の場合も含む) IPC 資源に対して、出力されている値を 10 進数に変換した ID で IPC 資源を検索して削除する。

3.4.3 TAM 障害

TAM 障害時の復旧については、DIOSA/XTP メモリキャッシュ利用の手引を参照してください。

3.4.4 Oracle データベース障害

D I O S A / X T P で使用しているデータベース表やO r a c l e D B に障害が発生した場合の復旧方法と処理の再開方法について説明します。

(1) データベース表障害

D I O S A / X T P では、データベース表に様々な制御情報を保有しています。表に障害が発生した場合は、O r a c l e D B の機能を利用してデータリストアおよびロールフォワード／ロールバックを行うことで障害直前の状態に復旧します。表の復旧によりD I O S A / X T P は処理実行可能な状態となるため、障害により停止していた機能を再開することでサービスを再開します。

(2) O r a c l e D B 障害

O r a c l e D B が障害となった場合は、D I O S A / X T P では全DB ノード障害となるため、下記の手順によりデータベースを復旧してください。

1. DB ノードを停止する。
2. O r a c l e D B の障害原因を調査し、取り除く。
3. O r a c l e D B を起動する。
4. DB ノードを起動する。
5. DB ノードが閉塞状態に移行している場合は閉塞を解除する。（閉塞解除手順については、「3.4.1(3) D B ノード障害からの復旧」を参照してください）
6. DB ノード障害により停止したデーモンを再起動する。（詳細は、D I O S A / X T P データストア 利用の手引に記載されている「データベース障害」の項を参照してください）

付録A リソース一覧

A.1 共有メモリ

DIOSA/XTP で使用する共通メモリー一覧です。

キーの上位 4 桁は、環境定義(\$DIOSAMAP)に定義する、論理ノードごとの IPCKEY 定義値です。

キー	サイズ(※)	備考
0x----0d00	備考参照	DIOSA/XTP 起動時に作成 [サイズ見積] $40 + 184 \times \{\text{環境変数 DIOSA_ETGMAXENTRY の値}\}$
0x----1301	1KB	稼動統計機能起動時に作成
0x----1302	1KB	稼動統計機能起動時に作成
0x----1303	1KB	稼動統計機能起動時に作成
0x----1304	1KB	稼動統計機能起動時に作成
0x----2810	1KB	DIOSA/XTP 起動時に作成
0x----282-	備考参照	DIOSA/XTP 起動時に作成 キーの最下位 1 桁は、0 か 1 のいずれかを使用する [サイズ見積] $704 + 32 \times (\text{論理システム数} \times 5 + \text{CMDSEND 節-SRVGRP 項数})$ $+ 296 \times \text{CMDSEND 節-ENV 項数} + 32 \times \text{CMDSEND 節-LNODECMD 項数}$ $+ 32 \times \text{CMDSEND 節-SRVGRP 項数} + 16 \times \text{CMDSEND 節-LNODESRV 項数}$ $+ 88 \times \text{CMDSEND 節-CMDRT 項数} + 8 \times \text{CMDSEND 節-CMDPERM 項数}$ $+ 48 \times \text{CMDSEND 節-CMDGRP 項数} + 64 \times \text{CMDSEND 節-CMDTEXT 項数}$ $+ 56 \times \text{CMDSEND 節-EXPERM 項数} + 32 \times \text{CMDSEND 節-CMD 項数}$
0x----0a00	備考参照	DIOSA/XTP 起動時に作成 [サイズ見積] $192 + (336 + 112 + \{\text{環境変数 DIOSA_TMCLENMAX の値}\})$ $\times ((\{\text{環境変数 DIOSA_TMCTBLMAX の値}\} \times 1.1 \text{ を切り上げ})$ $+ (\{\text{環境変数 DIOSA_TCDITBLMAX の値}\} \times 1.1 \text{ を切り上げ}) + 2)$
0x----0a01	備考参照	タイマ制御機能起動時に作成 [サイズ見積] $16 + 120 \times (\{\text{環境変数 DIOSA_TMCPROC の値}\}$ $+ \{\text{環境変数 DIOSA_TCDIPROC の値}\})$
環境変数 DIOSA_APPTRCIPC の値	備考参照	アプリケーショントレース機能起動時に作成 [サイズ見積] 984 $+ (\{\text{環境変数 DIOSA_APPTRCMPAGE の値}\}$ $\times \{\text{環境変数 DIOSA_APPTRCMSIZE の値}\})$
環境変数 DIOSA_DBGIPC の 値	備考参照	デバッグトレース機能起動時に作成 [サイズ見積] 3296 $+ (\{\text{環境変数 DIOSA_DBGERRPAGE の値}\} \times \{\text{環境変数 DIOSA_DBGERRSIZE の値}\})$ $+ (\{\text{環境変数 DIOSA_DBGLOGPAGE の値}\} \times \{\text{環境変数 DIOSA_DBGLOGSIZE の値}\})$
0x----0900	5.25KB	メモリ管理機能起動時に作成
0x00000000	備考	メモリ管理機能起動時、及びメモリ拡張(アプリケーションからメモリ確保の 要求でメモリ不足)時に作成 [サイズ見積り] 起動時： $(288 + \text{MMG 節-SHM 項-WRINIT の値}) + (288 + \text{MMG 節-SHM 項-RDINIT の値})$ 拡張時： $(288 + \text{MMG 節-SHM 項-WRINC の値}) + (288 + \text{MMG 節-SHM 項-RDINC の値})$ 注意) メモリ種別毎に共有メモリのセグメントが作成される。 メモリ拡張の最大サイズは、メモリ種別毎の最大サイズ(WRMAX, RDMAX)で決ま

		る。
--	--	----

※ 1KB 未満のサイズは 1KB と表記

A.2 ソケットファイル

DIOSA/XTP で使用するソケットファイル一覧です。

ファイルは\${環境変数 DIOSA_TMP 指定ディレクトリ}/{論理ノード名}/socket 配下に作成します。

ファイル名	備考
etg_etg_dmn	経過時間監視機能で使用
ops_xxx	稼動統計機能で使用（xxx は論理システム名）
cdd_cmdsend	コマンド配信機能で使用
cdd_dmctl	コマンド配信機能で使用
cdd_hstctl	コマンド配信機能で使用

付録B プロセス一覧

B.1 常駐プロセス一覧

プロセス名	説明	起動コマンド	停止コマンド	補足
dimsgd	API から要求があったメッセージをファイルへ出力する。 DB、OLTP、AP ノードで動作する。	dimsgdctrl -b	dimsgdctrl -e	
didbglogd	動作ログをファイルへ出力する。 DB、OLTP、AP ノードで動作する。	didbginit	didbgterm	
didbgerrd	エラーログをファイルへ出力する。 DB、OLTP、AP ノードで動作する。	didbginit	didbgterm	
diapptrcd	アプリケーションログをファイルへ出力する。 DB、OLTP、AP ノードで動作する。	diapptrcinit	diapptrcrterm	
didamd	DIOSA の常駐プロセスを監視し、障害時に再起動を行う。 DB、OLTP、AP ノードで動作する。	distart_com	distop	
dimmgd	アプリケーションで使用するメモリを管理する。 DB、OLTP、AP ノードで動作する。	distart_com	distop	
dimdrnotifyd	DIOSA の SG 変更を管理する。 DB、OLTP、AP ノードで動作する。	distart_com	distop	
dibcmdmn	ノードおよび CO の閉塞情報を管理する。 DB、OLTP、AP ノードで動作する。	dibcmctrl -b	dibcmctrl -e	
dicdddmn	要求のあったコマンドを適切なノードへ配信し実行する。 DB、OLTP、AP ノードで動作する。	dicddctrl -b	dicddctrl -e	
dicddhstdmn	コマンド配信履歴を出力する。 DB、OLTP、AP ノードで動作する。	dicddhstctl -s	dicddhstctl -e	
dincmdbhcdmnap	Oracle Database の状態を管理する。 OLTP、AP ノードで動作する。	dincmbhchkctrlinit	dincmbhchkctrlterm	
dincmdbhcdmnmn	Oracle Database の状態を管理する。 OLTP、AP ノードで動作する。	dincmbhchkctrlinit	dincmbhchkctrlterm	
dincmdbhcdmndb	Oracle Database の状態を管理する。 DB ノードで動作する。	dincmbhchkctrlinit	dincmbhchkctrlterm	

プロセス名	説明	起動コマンド	停止コマンド	補足
dincmdbhcdmns	Oracle Database の状態を管理する。 DB ノードで動作する。	dincmdbhchkctrlinit	dincmdbhchkctrlterm	
diicdmn	メモリ DB のマスタ情報の管理と切り替え制御を行う。 OLTP、AP ノードで動作する。	diiminit	diimterm	
diimacssrv	メモリ DB へアクセスする。 OLTP ノードで動作する。	diiminit diimctrl -b	diimterm diimctrl -e	
diimbrgsrv	他ノードのメモリ DB へのアクセスを中継する。 OLTP ノードで動作する。	diiminit diimctrl -b	diimterm diimctrl -e	
diopsctrl	稼動統計情報をファイルへ出力する。 OLTP、AP ノードで動作する。	diopsctrl -b	diopsctrl -e	
dietgdmn	プロセス稼動時間を監視する。 OLTP、AP ノードで動作する。	dietgctrl -b	dietgctrl -e	
dimfcadmdmn	ノードの負荷情報を監視する。 OLTP、AP ノードで動作する。	dimfcstart -b	dimfcstart -e	
ditmcdmn	登録されたタイマを監視し、指定時刻または指定時間後にタイマを実行する。 OLTP、AP ノードで動作する。	ditmcdmn	ditmcdmnstop	
ditmccmdexeccmd	コマンドタイマを実行する。 OLTP、AP ノードで動作する。	-	ditmcdmnstop	タイマ管理機能により自動起動され、タイマ処理が完了すると自動停止する。
ditmccmdexecco	CO タイマを実行する。 OLTP、AP ノードで動作する。	-	ditmcdmnstop	
ditmccmdexectpb	TPBASE タイマを実行する。 OLTP、AP ノードで動作する。	-	ditmcdmnstop	
didltpathcommd	他論理システムへデータを送信する。 AP ノードで定義により動作する。	didltinit	didltterm	
didltpathdispd	他論理システムから受信したデータを適切なスーパーストリームへ転送する。 OLTP、AP ノードで定義により動作する。	didltinit	didltterm	
didltpooldeld	プールファイルの不要データを削除する。 OLTP、AP ノードで定義により動作する。	didltinit	didltterm	

プロセス名	説明	起動コマンド	停止コマンド	補足
didltslmagtd	スーパーストリームが処理するノードを管理する。 OLTP、AP ノードで定義により動作する。	didltinit	didltterm	
didltslmmgrd	スーパーストリームが処理するノードを管理する。 DB ノードで定義により動作する。	didltinit	didltterm	
didtsdmnmngd	ディレードセンダ機能の動作状態を管理する。 OLTP、AP ノードで定義により動作する。	didtsinit	didtsterm	
didtsexecd	ログデータを他論理システムへ送信する。 OLTP、AP ノードで定義により動作する。	didtsinit	didtsterm	
didtrdmnmngd	ディレードレシーバ機能の動作状態を管理する。 OLTP、AP ノードで定義により動作する。	didtrinit	didtrterm	
didtrexecd	他論理システムからログデータを受信する。 OLTP、AP ノードで定義により動作する。	didtrinit	didtrterm	
didtldmnmngd	ディレードログリーダ機能の動作状態を管理する。 OLTP、AP ノードで定義により動作する。	didtlinit	didtlterm	
didtlexecd	ログデータを処理する。 OLTP、AP ノードで定義により動作する。	didtlinit	didtlterm	
dicocdmn	CO 制御 T P P の死活監視をおこなう。 OLTP、AP ノードで動作する	dicocinit	dicocterm	

B.2 TPP 一覧

プロセス名	説明	補足
dicoc TPP	COの起動、停止、実行、エラー処理を制御する。	
dincmhl send	ヘルスチェック電文を監視対象ノードに送信する処理を行う。 ヘルスチェック受信 TPP、タイムアウト監視 TPP と連携してパスの状態を管理する。	
dincmhl rec v	ヘルスチェック対象ノードからのヘルスチェックの応答電文を受信する。 ヘルスチェック送信 TPP、タイムアウト監視 TPP と連携してパスの状態を管理する。	
dincmhl resp	ヘルスチェック電文を受信し、受信した電文に対する応答処理を行う。	
dincmhl t mout	ヘルスチェック対象ノードからの応答未受信のままタイムアウトしているかを監視する。 ヘルスチェック送信 TPP、ヘルスチェック受信 TPP と連携してパスの状態を管理する。	

付録C データベース一覧

C.1 Oracle データベース表

DB ヘルスチェック更新確認用テーブル					
概要	Oracle DB が生存していることを確認する。				
表名	DIOSA_NCM_DBHC_01～32				
データ／制御	制御	レコードサイズ	138 byte	サイズ計算式	32 × レコードサイズ

付録D 諸元一覧

D.1 共通

項目名称	単位	諸元	備考
論理システム数	個数	1～255	
論理ノード数	個数	1～256	1 論理システム内は 256 ノードまで
論理ライブラリ数	個数	0～32767	1 プロセスに対しては最大 5120 個まで

D.2 アプリケーション実行制御

項目名称	単位	諸元	備考
CO 制御電文長	K バイト	3～2097151	制御情報を含む TPBASE の最大電文長以下にすること
CO 制御が動作する TPBASE TPP 数	個数	1～2147483647	
CO 制御が動作する TPBASE クラス数	個数	1～1000	

D.3 通信制御

項目名称	単位	諸元	備考
1 論理ノード内の TPBASE 数	個数	1～16	
DB インスタンス数 (インスタンスグループ数)	個数	0～16	
RAC 構成 DB インスタンス数	個数	2	

D.4 メモリキャッシュ

項目名称	単位	諸元	備考
MAP 数	個数	1～65536	1 レプリケーショングループ内は 256MAP まで
レプリケーショングループ数	個数	1～256	
メインキー	バイト	32	固定長
スレーブ TAM インスタンス数	個数	0～10	
論理表数	個数	1～1023	
セカンダリキー数	個数	論理表ごとに 0～62	プライマリキーは必須

D.5 データストア

項目名称	単位	諸元	備考
ログデータサイズ	バイト	1～2147483640	
スーパーストリーム数	個数	0～1024	MAP あたりのスーパーストリーム数は 100 まで
センダユニット数	個数	0～1	
ログリーダーユニット数	個数	0～16	

付録E 環境定義例

E.1 定義例の構成

以下の構成で環境を構築した場合の定義例を示します。

フロントシステム (LS1)

項目	内容				
論理ノード構成	種別	論理ノード名	IP アドレス		
			業務	運用	
	AP	ap1	192. 168. 0. 1	192. 168. 1. 1	
	AP	ap2	192. 168. 0. 2	192. 168. 1. 2	
	OLTP	oltp1	192. 168. 0. 11	192. 168. 1. 11	
	OLTP	oltp2	192. 168. 0. 12	192. 168. 1. 12	
	DB	db1	192. 168. 0. 21	192. 168. 1. 21	
TAM 構成	インスタンス	MAPID	マスタ	スレーブ	
	INS001	1	oltp1	oltp2	
	INS002	2	oltp2	oltp1	
	INS003	3	oltp1	oltp2	
Oracle DB	インスタンス	user/pass			
	DB01	user1/pass1			
データストア構成	スーパー ストリーム名	プールファイル		ユニット構成	ユーザデータ更新先
		種別	格納先		
	TAM0000000001_1	ライタ	IM	センダ、ログリーダー	DB
	TAM0000000002_1	ライタ	IM	センダ、ログリーダー	DB
	TAM0000000003_1	ライタ	IM	センダ、ログリーダー	DB
	ORA_STREAM1_1	ライタ	DB	センダ	－
	ORA_STREAM2_1	ライタ	DB	センダ	－
使用ポート	機能	ポート番号			
	閉塞管理	31100			
	コマンド配信	31101			
	DB 接続管理	31125			
	メモリキャッシュ	31106-31123 (インメモリサーバ所在管理：31106-31107、インメモリサーバ：31108-31123)			
	データストア	31102-31105			
IPC キー	0x0001****				
TPBASE 関連	論理ノード種別	モニタ名			
	AP	TPMap			
	OLTP	TPMoltp			

データバックアップ用システム (LS2)

項目	内容				
論理ノード構成	種別	論理ノード名	IP アドレス		
			業務	運用	
	AP	ap1	192. 168. 2. 1	192. 168. 3. 1	
	AP	ap2	192. 168. 2. 2	192. 168. 3. 2	
	DB	db1	192. 168. 2. 21	192. 168. 3. 21	
Oracle DB	インスタンス	user/pass			
	DB01	user1/pass1			
データストア構成	スーパー ストリーム名	プールファイル		ユニット構成	ユーザデータ更新先
		種別	格納先		
	TAM0000000001_1	レシーバ	DB	ログリーダー	DB
	TAM0000000002_1	レシーバ	DB	ログリーダー	DB
	TAM0000000003_1	レシーバ	DB	ログリーダー	DB
	ORA_STREAM1_1	レシーバ	DB	ログリーダー	DB
	ORA_STREAM2_1	レシーバ	DB	ログリーダー	DB
使用ポート	機能	ポート番号			
	閉塞管理	31100			
	コマンド配信	31101			
	DB 接続管理	31125			
	データストア	31102-31105			
IPC キー	0x0001****				

E. 2 DIOSA

E. 2. 1 DBCTRL 節

```
$DBCTRL
%CONTROL
    INTERVAL1      = 5000
    INTERVAL2      = 20000
    PORTNUM         = 31125
    RETRY           = 3
;
%INSTANCE
    DBNAME          = DB01
    DBLNODENAME     = db1
    USERID          = user1
    PASSWORD        = pass1
;
%RGSET
    NAME            = RGSET01
    INITINSTANCE    = DB01
    %RGIDSET
        RGID        = 00
;
;
;
```

E. 2. 2 DELAYED 節

フロントシステム用

```
$DELAYED
%COMMON
    DBLAYER      = AP
    FAILOVER     = NO
;
%DSAM
    DELINVL_IM   = 10
    DELINVL_DB   = 60
%DEFAULT
    STACKNUM     = 12
    SWAPSIZE     = 7
    DELTYPE      = AUTO
    DELSTACKCNT  = 2
    COMPRESS     = YES
;
;
%LOGREADER
%DEFAULT
    PRCINIT      = UpdPrcInitExit
    PRCTERM      = UpdPrcTermExit
    TRNINIT      = UpdTrnInitExit
    TRNTERM      = UpdTrnTermExit
;
;
%UNITGROUP
    NAME         = UTGTAM_1
    POOLTYPE      = WRITER
    POOLFILE      = IM
%SNDUNIT
    NAME         = SENDER
    DLSNAME      = LS2
;
%LRDUNIT
    NAME         = UTOLLTOS
    USERDATA     = DB
    RGSETNAME     = RGSET01
    DATAPERLOT   = 5
    MAINAP       = LogUpd0raExit
;
;
%UNITGROUP
    NAME         = UTGORA_1
    POOLTYPE      = WRITER
    POOLFILE      = DB
%SNDUNIT
    NAME         = SENDER
    DLSNAME      = LS2
;
;
%SUPERSTREAM
    NAME         = TAM0000000001_1
    UNITGROUP    = UTGTAM_1
    MAPID        = 1
;
%SUPERSTREAM
    NAME         = TAM0000000002_1
    UNITGROUP    = UTGTAM_1
    MAPID        = 2
;
%SUPERSTREAM
    NAME         = TAM0000000003_1
    UNITGROUP    = UTGTAM_1
    MAPID        = 3
;
%SUPERSTREAM
    NAME         = ORA_STREAM1_1
    UNITGROUP    = UTGORA_1
;
%SUPERSTREAM
    NAME         = ORA_STREAM2_1
    UNITGROUP    = UTGORA_1
;
;
```

データバックアップシステム用

```
$DELAYED
%COMMON
    DBLAYER      = AP
    FAILOVER     = NO
;
%DSAM
    DELINVL_DB  = 60
%DEFAULT
    STACKNUM    = 12
    SWAPSIZE    = 7
    DELTYPE     = AUTO
    DELSTACKCNT = 2
;
;
%LOGREADER
%DEFAULT
    PRCINIT     = UpdPrcInitExit
    PRCTERM     = UpdPrcTermExit
    TRNINIT     = UpdTrnInitExit
    TRNTERM     = UpdTrnTermExit
;
;
%UNITGROUP
    NAME        = UTGORA_1
    POOLTYPE    = RECEIVER
    POOLFILE    = DB
%LRDUNIT
    NAME        = UTOLLDSC
    USERDATA    = DB
    RGSETNAME   = RGSET01
    DATAPERLOT  = 5
    MAINAP      = LogUpdOraExit
;
;
%SUPERSTREAM
    NAME        = TAM0000000001_1
    UNITGROUP   = UTGORA_1
;
%SUPERSTREAM
    NAME        = TAM0000000002_1
    UNITGROUP   = UTGORA_1
;
%SUPERSTREAM
    NAME        = TAM0000000003_1
    UNITGROUP   = UTGORA_1
;
%SUPERSTREAM
    NAME        = ORA_STREAM1_1
    UNITGROUP   = UTGORA_1
;
%SUPERSTREAM
    NAME        = ORA_STREAM2_1
    UNITGROUP   = UTGORA_1
;
;
```

E. 2. 3 DIOSAMAP 節

フロントシステム用

```
$DIOSAMAP
%DEFAULT
    PORT_BCM    = 31100
    PORT_CDD    = 31101
    PORT_DLT    = 31102
    PORT_IIC    = 31106
    PORT_IMS    = 31108
;
%LOGSYSTEM
    NAME = LS1
%LNODE
    NAME      = ap1
    ID        = 100
    INITSTS   = ACT
    TYPE      = AP
    IPADDR    = 192.168.0.1
    IPADDR_OP = 192.168.1.1
    IPCKEY    = 0001
%TPM
    TPMNAME = TMap
;
;
%LNODE
    NAME      = ap2
    ID        = 101
    INITSTS   = ACT
    TYPE      = AP
    IPADDR    = 192.168.0.2
    IPADDR_OP = 192.168.1.2
    IPCKEY    = 0001
%TPM
    TPMNAME = TMap
;
;
%LNODE
    NAME      = oltp1
    ID        = 200
    INITSTS   = ACT
    TYPE      = OLTP
    IPADDR    = 192.168.0.11
    IPADDR_OP = 192.168.1.11
    IPCKEY    = 0001
%TPM
    TPMNAME = TPMoltp
;
;
%LNODE
    NAME      = oltp2
    ID        = 201
    INITSTS   = ACT
    TYPE      = OLTP
    IPADDR    = 192.168.0.12
    IPADDR_OP = 192.168.1.12
    IPCKEY    = 0001
%TPM
    TPMNAME = TPMoltp
;
;
%LNODE
    NAME      = db1
    ID        = 300
    INITSTS   = ACT
    TYPE      = DB
    IPADDR    = 192.168.0.21
    IPADDR_OP = 192.168.1.21
    IPCKEY    = 0001
;
;
;
```

データバックアップシステム用

```
$DIOSAMAP
%DEFAULT
    PORT_BCM    = 31100
    PORT_CDD    = 31101
    PORT_DLT    = 31102
;
%LOGSYSTEM
    NAME = LS2
    %LNODE
        NAME      = ap1
        ID        = 100
        INITSTS    = ACT
        TYPE       = AP
        IPADDR     = 192.168.2.1
        IPADDR_OP  = 192.168.3.1
        IPCKEY     = 0001
    ;
    %LNODE
        NAME      = ap2
        ID        = 101
        INITSTS    = ACT
        TYPE       = AP
        IPADDR     = 192.168.2.2
        IPADDR_OP  = 192.168.3.2
        IPCKEY     = 0001
    ;
    %LNODE
        NAME      = db1
        ID        = 300
        INITSTS    = ACT
        TYPE       = DB
        IPADDR     = 192.168.2.21
        IPADDR_OP  = 192.168.3.21
        IPCKEY     = 0001
    ;
;
;
```

E. 2. 4IMENV 節

※ フロントシステムのみ必要

```
$IMENV
%COMMON
    SWITCH          = AUTO
    CTRLSENDINVL    = 10
    ALLOWCOMMIT     = REPGTHALF
    RESTTXIDBLK     = YES
;
%FAULTDETECT
%NODEFAULT
    HLTHCHKINVL     = 10
    HLTHCHKRETRY    = 3
    TPATHCHK        = YES
    DOWNCNT         = GTHALF
    DOWNTIME        = 0
;
%SERVERFAULT
    CHKINVL         = 40
    DELAYLIMIT      = 834
    MSTRESTART      = 0
    SLVRESTART      = 3
    RESTARTTIME     = 60
;
%BRIDGEFAULT
    CHKINVL         = 300
    RESTART         = 3
    RESTARTTIME     = 60
;
%TAMFAULT
    CHKINVL         = 0
    SWITCHRETRY     = 0
    SWITCHINVL      = 10
;
;
%DEF_MAP
    TXTBLSIZE       = 1020
    IMQUEBUFSIZE    = 2448
    JOURNAL          = NO
    STATS            = YES
    COMMITMODE       = NORMAL
%GROUPCOMMIT
    REQNUM           = 5
    TMOUT            = 0
;
%ACCESSLOG
    OUTPUT           = YES
    BUFSIZE          = 1024
    BUFCYCL          = YES
;
;
%REPGROUP
    ID               = 1
    INSTANCE         = INS001
%MAP
    ID               = 1
    %HASHRANGE
        START = 100
        END   = 199
;
;
%MASTER
    LNODE            = oltp1
    TAMVNODE         = TAMLN001MASTER
;
;
%REPGROUP
    ID               = 2
    INSTANCE         = INS002
%MAP
    ID               = 2
    %HASHRANGE
        START = 200
        END   = 299
;
;
%MASTER
    LNODE            = oltp2
    TAMVNODE         = TAMLN002MASTER
;
;
```

```

;
%REPGROUP
    ID          = 3
    INSTANCE    = INS003
%MAP
    ID          = 3
    %HASHRANGE
        START = 300
        END   = 399
    ;
;
%MASTER
    LNODE       = oltp1
    TAMVNODE    = TAMLN003MASTER
;
;
%BRIDGE
    MULTI       = 2
    CONNECTINVL = 10
    IMQEBUFSIZE = 48959
    STATS       = YES
;
%USERAP
    HASHEXIT    = F0Z999HashExit
    REQTIMEOUT  = 30
    IMQEBUFSIZE = 6068
    MAXNUM      = 30
    USERAREASIZE = 16
    USERAREAXIT = InitOpDataGetExit
;
;

```


E. 2. 5 IMTABLECONF 節

```
$IMTABLECONF
/*--DIOSA_DSAM-----*/
%LTABLE
    TYPE      = 0
    NAME      = DIOSA_DSAM
    ID        = 228
%RECORDCONF
    TYPE      = FIXED
    %PRIMARYKEY
        NAME  = DIOSA_DSAM_00_IDX
        %KEYDEF OFFSET = 0 LENGTH = 4 ;
    ;
;
%PTABLE NAME = DIOSA_DSAM_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_SPST-----*/
%LTABLE
    TYPE      = 0
    NAME      = DIOSA_DELAYED_SPST
    ID        = 116
%RECORDCONF
    TYPE      = FIXED
    %PRIMARYKEY
        NAME  = DIOSA_DELAYED_SPST_00_IDX
        %KEYDEF OFFSET = 0 LENGTH = 16 ;
    ;
;
%PTABLE NAME = DIOSA_DELAYED_SPST_0000000001 MAPID = 1 ;
%PTABLE NAME = DIOSA_DELAYED_SPST_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_SPST_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_STRM-----*/
%LTABLE
    TYPE      = 0
    NAME      = DIOSA_DELAYED_STRM
    ID        = 115
%RECORDCONF
    TYPE      = FIXED
    %PRIMARYKEY
        NAME  = DIOSA_DELAYED_STRM_00_IDX
        %KEYDEF OFFSET = 0 LENGTH = 16 ;
        %KEYDEF OFFSET = 16 LENGTH = 16 ;
        %KEYDEF OFFSET = 32 LENGTH = 4 ;
    ;
    %SECONDARYKEY
        NAME  = DIOSA_DELAYED_STRM_01_IDX
        %KEYDEF OFFSET = 0 LENGTH = 16 ;
    ;
    %SECONDARYKEY
        NAME  = DIOSA_DELAYED_STRM_04_IDX
        %KEYDEF OFFSET = 0 LENGTH = 16 ;
        %KEYDEF OFFSET = 68 LENGTH = 4 ;
        %KEYDEF OFFSET = 32 LENGTH = 4 ;
    ;
    %SECONDARYKEY
        NAME  = DIOSA_DELAYED_STRM_05_IDX
        %KEYDEF OFFSET = 0 LENGTH = 16;
        %KEYDEF OFFSET = 68 LENGTH = 4;
        %KEYDEF OFFSET = 128 LENGTH = 8;
    ;
    %SECONDARYKEY
        NAME  = DIOSA_DELAYED_STRM_06_IDX
        %KEYDEF OFFSET = 0 LENGTH = 16;
        %KEYDEF OFFSET = 32 LENGTH = 4;
    ;
;
%PTABLE NAME = DIOSA_DELAYED_STRM_0000000001 MAPID = 1 ;
%PTABLE NAME = DIOSA_DELAYED_STRM_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_STRM_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_UNIT-----*/
%LTABLE
    TYPE      = 0
    NAME      = DIOSA_DELAYED_UNIT
    ID        = 119
%RECORDCONF
    TYPE      = FIXED
    %PRIMARYKEY
        NAME  = DIOSA_DELAYED_UNIT_00_IDX
        %KEYDEF OFFSET = 0 LENGTH = 16 ;
```

```

        %KEYDEF OFFSET = 16 LENGTH = 1 ;
        %KEYDEF OFFSET = 17 LENGTH = 16 ;
    ;
    %SECONDARYKEY
        NAME = DIOSA_DELAYED_UNIT_01_IDX
        %KEYDEF OFFSET = 0 LENGTH = 16 ;
    ;
;
%PTABLE NAME = DIOSA_DELAYED_UNIT_0000000001 MAPID = 1 ;
%PTABLE NAME = DIOSA_DELAYED_UNIT_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_UNIT_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_STACK-----*/
%LTABLE
    TYPE          = 0
    NAME           = DIOSA_DELAYED_STACK
    ID             = 117
    %RECORDCONF
        TYPE = FIXED
        %PRIMARYKEY
            NAME = DIOSA_DELAYED_STACK_00_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
            %KEYDEF OFFSET = 16 LENGTH = 4 ;
        ;
        %SECONDARYKEY
            NAME = DIOSA_DELAYED_STACK_01_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
        ;
        %SECONDARYKEY
            NAME = DIOSA_DELAYED_STACK_02_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
            %KEYDEF OFFSET = 20 LENGTH = 4 ;
        ;
    ;
;
%PTABLE NAME = DIOSA_DELAYED_STACK_0000000001 MAPID = 1 ;
%PTABLE NAME = DIOSA_DELAYED_STACK_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_STACK_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_STACKDATANO-----*/
%LTABLE
    TYPE          = 0
    NAME           = DIOSA_DELAYED_STACKDATANO
    ID             = 118
    %RECORDCONF
        TYPE = FIXED
        %PRIMARYKEY
            NAME = DIOSA_DELAYED_STKDATANO_00_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
            %KEYDEF OFFSET = 16 LENGTH = 16 ;
            %KEYDEF OFFSET = 32 LENGTH = 4 ;
            %KEYDEF OFFSET = 36 LENGTH = 4 ;
        ;
        %SECONDARYKEY
            NAME = DIOSA_DELAYED_STKDATANO_01_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
            %KEYDEF OFFSET = 16 LENGTH = 16 ;
            %KEYDEF OFFSET = 36 LENGTH = 4 ;
        ;
        %SECONDARYKEY
            NAME = DIOSA_DELAYED_STKDATANO_02_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
            %KEYDEF OFFSET = 16 LENGTH = 16 ;
            %KEYDEF OFFSET = 32 LENGTH = 4 ;
        ;
        %SECONDARYKEY
            NAME = DIOSA_DELAYED_STKDATANO_03_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
        ;
        %SECONDARYKEY
            NAME = DIOSA_DELAYED_STKDATANO_04_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
            %KEYDEF OFFSET = 16 LENGTH = 16 ;
            %KEYDEF OFFSET = 32 LENGTH = 4 ;
            %KEYDEF OFFSET = 40 LENGTH = 8 ;
        ;
        %SECONDARYKEY
            NAME = DIOSA_DELAYED_STKDATANO_05_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
            %KEYDEF OFFSET = 36 LENGTH = 4 ;
        ;
    ;
;
%PTABLE NAME = DIOSA_DELAYED_STACKDATANO_0000000001 MAPID = 1 ;

```

```

%PTABLE NAME = DIOSA_DELAYED_STACKDATANO_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_STACKDATANO_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_POOL_0001-----*/
%LTABLE
  TYPE      = 0
  NAME      = DIOSA_DELAYED_POOL_0001
  ID        = 103
%RECORDCONF
  TYPE      = VARLEN
  %PRIMARYKEY
    NAME = DIOSA_DELAYED_POOL_0001_00_IDX
    %KEYDEF OFFSET = 0 LENGTH = 16 ;
    %KEYDEF OFFSET = 16 LENGTH = 8 ;
    %KEYDEF OFFSET = 24 LENGTH = 4 ;
  ;
;
%PTABLE NAME = DIOSA_DELAYED_POOL_0001_0000000001 MAPID = 1 ;
%PTABLE NAME = DIOSA_DELAYED_POOL_0001_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_POOL_0001_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_POOL_0002-----*/
%LTABLE
  TYPE      = 0
  NAME      = DIOSA_DELAYED_POOL_0002
  ID        = 104
%RECORDCONF
  TYPE      = VARLEN
  %PRIMARYKEY
    NAME = DIOSA_DELAYED_POOL_0002_00_IDX
    %KEYDEF OFFSET = 0 LENGTH = 16 ;
    %KEYDEF OFFSET = 16 LENGTH = 8 ;
    %KEYDEF OFFSET = 24 LENGTH = 4 ;
  ;
;
%PTABLE NAME = DIOSA_DELAYED_POOL_0002_0000000001 MAPID = 1 ;
%PTABLE NAME = DIOSA_DELAYED_POOL_0002_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_POOL_0002_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_POOL_0003-----*/
%LTABLE
  TYPE      = 0
  NAME      = DIOSA_DELAYED_POOL_0003
  ID        = 105
%RECORDCONF
  TYPE      = VARLEN
  %PRIMARYKEY
    NAME = DIOSA_DELAYED_POOL_0003_00_IDX
    %KEYDEF OFFSET = 0 LENGTH = 16 ;
    %KEYDEF OFFSET = 16 LENGTH = 8 ;
    %KEYDEF OFFSET = 24 LENGTH = 4 ;
  ;
;
%PTABLE NAME = DIOSA_DELAYED_POOL_0003_0000000001 MAPID = 1 ;
%PTABLE NAME = DIOSA_DELAYED_POOL_0003_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_POOL_0003_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_POOL_0004-----*/
%LTABLE
  TYPE      = 0
  NAME      = DIOSA_DELAYED_POOL_0004
  ID        = 106
%RECORDCONF
  TYPE      = VARLEN
  %PRIMARYKEY
    NAME = DIOSA_DELAYED_POOL_0004_00_IDX
    %KEYDEF OFFSET = 0 LENGTH = 16 ;
    %KEYDEF OFFSET = 16 LENGTH = 8 ;
    %KEYDEF OFFSET = 24 LENGTH = 4 ;
  ;
;
%PTABLE NAME = DIOSA_DELAYED_POOL_0004_0000000001 MAPID = 1 ;
%PTABLE NAME = DIOSA_DELAYED_POOL_0004_0000000002 MAPID = 2 ;
%PTABLE NAME = DIOSA_DELAYED_POOL_0004_0000000003 MAPID = 3 ;
;
/*--DIOSA_DELAYED_POOL_0005-----*/
%LTABLE
  TYPE      = 0
  NAME      = DIOSA_DELAYED_POOL_0005
  ID        = 107
%RECORDCONF
  TYPE      = VARLEN

```

```

        %PRIMARYKEY
            NAME = DIOSA_DELAYED_POOL_0005_00_IDX
            %KEYDEF OFFSET = 0 LENGTH = 16 ;
            %KEYDEF OFFSET = 16 LENGTH = 8 ;
            %KEYDEF OFFSET = 24 LENGTH = 4 ;
        ;
    ;
    %PTABLE NAME = DIOSA_DELAYED_POOL_0005_0000000001 MAPID = 1 ;
    %PTABLE NAME = DIOSA_DELAYED_POOL_0005_0000000002 MAPID = 2 ;
    %PTABLE NAME = DIOSA_DELAYED_POOL_0005_0000000003 MAPID = 3 ;
;
/*--DIOSA_SENDER_UNIT-----*/
%LTABLE
    TYPE          = 0
    NAME           = DIOSA_SENDER_UNIT
    ID             = 141
    %RECORDCONF
        TYPE = FIXED
        %PRIMARYKEY
            NAME = DIOSA_SENDER_UNIT_00_IDX
            %KEYDEF
                OFFSET = 0
                LENGTH = 32
        ;
    ;
;
    %PTABLE NAME = DIOSA_SENDER_UNIT_0000000001 MAPID = 1 ;
    %PTABLE NAME = DIOSA_SENDER_UNIT_0000000002 MAPID = 2 ;
    %PTABLE NAME = DIOSA_SENDER_UNIT_0000000003 MAPID = 3 ;
;
/*--DIOSA_SENDER_STRM-----*/
%LTABLE
    TYPE          = 0
    NAME           = DIOSA_SENDER_STRM
    ID             = 142
    %RECORDCONF
        TYPE = FIXED
        %PRIMARYKEY
            NAME = DIOSA_SENDER_STRM_00_IDX
            %KEYDEF
                OFFSET = 0
                LENGTH = 48
        ;
    ;
;
    %PTABLE NAME = DIOSA_SENDER_STRM_0000000001 MAPID = 1 ;
    %PTABLE NAME = DIOSA_SENDER_STRM_0000000002 MAPID = 2 ;
    %PTABLE NAME = DIOSA_SENDER_STRM_0000000003 MAPID = 3 ;
;
;
;

```

E. 2. 6 MMG 節

```
$MMG
%UAP
%SHM
    WRINIT  = 1024
    WRMAX   = 5120
    WRINC   = 2048
;
%PRC
    WRINIT  = 1024
    WRMAX   = 5120
    WRINC   = 2048
;
%THR
    WRINIT  = 2048
    WRMAX   = 6144
    WRINC   = 2048
;
;
%UDIR
    DIR      = /home/diosa/mmg/
;
%DDMP
    PRC      = YES
    DIR      = /home/diosa/mmg/
;
;
```

E. 2. 7 OPSENV 節

```
$OPSENV
%OPSPARAM
    GATHERDIR  = /home/diosa/gather
    OPSDIR     = /home/diosa/ops
    PREFIX     = OPSPFX
    FILESIZE   = 10
    FILENUM    = 5
    SETNUM     = 2
    SWEETIME   = 0
;
;
```

E. 2. 8 SYSMAP 節

```
$SYSMAP
%LOGSYSTEM
  LSNAME      = LS1
  LSID        = 11
  HOSTTYPE    = 01
%ROUTE
  TYPE        = DIRECT
  IPADDR      = 192.168.0.1
  PORT_CDD    = 31101
  PORT_DLT    = 31102
;
%ROUTE
  TYPE        = DIRECT
  IPADDR      = 192.168.0.2
  PORT_CDD    = 31101
  PORT_DLT    = 31102
;
;
%LOGSYSTEM
  LSNAME      = LS2
  LSID        = 12
  HOSTTYPE    = 01
%ROUTE
  TYPE        = DIRECT
  IPADDR      = 192.168.2.1
  PORT_CDD    = 31101
  PORT_DLT    = 31102
;
%ROUTE
  TYPE        = DIRECT
  IPADDR      = 192.168.2.2
  PORT_CDD    = 31101
  PORT_DLT    = 31102
;
;
;
```

E. 3 TPBASE

E. 3. 1 autoup. conf

systpp. ped	; システム TPP
jnlwrt. ped	; ジャーナルライタ
vdserver. ped	; VD サーバー
timer. ped	
TCPIP_TPLSN01. ped	; TCP/IP リスナ定義
TCPIP_TPLSN01. term	; 端末定義
TCPIP_TPLSN01. vd	; VD 定義
diosa_env. ap	; DIOSA AP 定義
diosa_env. vd	; DIOSA VD 定義
diosa_env. ped	; DIOSA CO 制御定義

E. 3. 2 mcs

AUTSTART	NO
EXCEPTION	0100CEAF
RESPCONNECT	YES
RESPTROUBLE	YES
SYMSGLANG	ENGLISH

E. 3. 3 DIOSA AP 定義(diosa_env. ap)

APID	DXTP_AP
APSTAT	ENABL

E. 3. 4 DIOSA VD 定義(diosa_env. vd)

; CO 制御 TPP		
VD	VDTR0101	
VDTYPE	TRNS	
AP	DXTP_AP	
TRNS	TR0101	
;ヘルスチェック送信 TPP		
VD	VDIHLS001	;VD 名
VDTYPE	TRNS	;VD 種別
AP	DXTP_AP	;AP 名
TRNS	D17HSD	;トランザクション ID
;ヘルスチェック監視 TPP		
VD	VDIHLT001	;VD 名
VDTYPE	TRNS	;VD 種別
AP	DXTP_AP	;AP 名
TRNS	D17HTO	;トランザクション ID

E. 3. 5 DIOSA プロセス定義(diosa_env. ped)

;	
;CO 制御 TPP	
;	
CLASS	CLS_XX
PATH	/opt/diosa_xtp/bin/dicoctpp
ARGS	-ChangeDir /home/diosa /tpbase
ARGS	-TP T11oltp1
#ARGS	-0ltpTplibTrace F=/home/diosa/tpbase/log/trace_coc01cls_log.PID,1000000
MAXSTARTTRIES	30
MAXPROCESS	8
MINPROCESS	4
PROEPILIB	/opt/diosa_xtp/lib/libdxtpcocprotpb. so
PROENTRY	di_coc_CocPrologue
EPIENTRY	di_coc_CocEpilogue
PROEPILANG	C
EXCEPTSIG	INT QUIT ILL FPE BUS SEGV XCPU ABRT
#OUTPUTDEV	/home/diosa/tpbase/log/diosa_coc1. out
#ERRORDEV	/home/diosa/tpbase/log/diosa_coc1. err
;	
; ヘルスチェック TPP	


```

;
CLASS      DXTP_NCMHSEND_CLS
PATH       /opt/diosa_xtp/bin/dincmhlsend
ARGS       -ChangeDir /home/diosa/tpbase
MAXSTARTTRIES 3
MAXPROCESS 1
MINPROCESS 1
PROEPILIB  /opt/diosa_xtp/lib/libdxtpncomtptsendprotpb. so
PROENTRY   di_ncm_PrologHlthSend
EPIENTRY   di_ncm_EpillogHlthSend
PROEPILANG C
#OUTPUTDEV /home/diosa/tpbase/log/diosa_ncm_hlsendl. out
#ERRORDEV  /home/diosa/tpbase/log/diosa_ncm_hlsendl. err

;
CLASS      DXTP_NCMHLRECV_CLS
PATH       /opt/diosa_xtp/bin/dincmhltrech
ARGS       -ChangeDir /home/diosa/tpbase
MAXSTARTTRIES 3
MAXPROCESS 1
MINPROCESS 1
PROEPILIB  /opt/diosa_xtp/lib/libdxtpncomtptrechprotpb. so
PROENTRY   di_ncm_PrologHlthRcv
EPIENTRY   di_ncm_EpillogHlthRcv
PROEPILANG C
#OUTPUTDEV /home/diosa/tpbase/log/diosa_ncm_hltrechl. out
#ERRORDEV  /home/diosa/tpbase/log/diosa_ncm_hltrechl. err

;
CLASS      DXTP_NCMHLMOUT_CLS
PATH       /opt/diosa_xtp/bin/dincmhlmout
ARGS       -ChangeDir /NT77/diosa_xtp/envIT11/tpbase
MAXSTARTTRIES 3
MAXPROCESS 1
MINPROCESS 1
PROEPILIB  /opt/diosa_xtp/lib/libdxtpncomtptoutprotpb. so
PROENTRY   di_ncm_PrologTimeout
EPIENTRY   di_ncm_EpillogTimeout
PROEPILANG C
#OUTPUTDEV /home/diosa/tpbase/log/diosa_ncm_hlmoutl. out
#ERRORDEV  /home/diosa/tpbase/log/diosa_ncm_hlmoutl. err

;
CLASS      DXTP_NCMHLRESP_CLS
PATH       /opt/diosa_xtp/bin/dincmhltrech
ARGS       -ChangeDir /NT77/diosa_xtp/envIT11/tpbase
MAXSTARTTRIES 3
MAXPROCESS 1
MINPROCESS 1
PROEPILIB  /opt/diosa_xtp/lib/libdxtpncomtptrespptpb. so
PROENTRY   di_ncm_PrologHlth
EPIENTRY   di_ncm_EpillogHlt
PROEPILANG C
#OUTPUTDEV /home/diosa/tpbase/log/diosa_ncm_hltrechl. out
#ERRORDEV  /home/diosa/tpbase/log/diosa_ncm_hltrechl. err

```

E. 3. 6 DIOSA トランザクション定義(diosa_env. trns)

```

#トランザクション#01
TRNSID      TR0101
TRPRTY      1
TRSTAT      ENABL
TRTYPE      IN
TPPLIB      /opt/diosa_xtp/lib/libdxtpcocotrntpb. so
TPPENTRY    di_coc_CocTrnsEnt
TPPLANG     C
ABORTXITLIB /opt/diosa_xtp/lib/libdxtpcocabttpb. so
ABORTXIT    di_coc_CocExceptMain
ABORTLANG   C
CLASS       CLS_XX
LINK        INIT
CPUTIME     NOLIM
ELPTIME     NOLIM
ABORT       CONT
TIMEOVER    CONT

#ヘルスチェック送信 TPP
TRNSID      D17HSD
TRPRTY      1

```

TRSTAT	ENABL
TRTYPE	IN
TPPLIB	/opt/diosa_xtp/lib/libdxtpncmtpsendtpb. so
TPPENTRY	di_ncm_HlthSendTpp
TPPLANG	C
CLASS	DXTP_NCMHSEND_CLS
LINK	INIT
CPUTIME	60
ELPTIME	60
ABORT	CONT
TIMEOVER	CONT
#ヘルスチェック受信 TPP	
TRNSID	D17HRV
TRPRTY	1
TRSTAT	ENABL
TRTYPE	IN
TPPLIB	/opt/diosa_xtp/lib/libdxtpncmtprecvtpb. so
TPPENTRY	di_ncm_HlthRcvTpp
TPPLANG	C
CLASS	DXTP_NCMHLRECV_CLS
LINK	INIT
CPUTIME	60
ELPTIME	60
ABORT	CONT
TIMEOVER	CONT
#タイムアウト監視 TPP	
TRNSID	D17HT0
TRPRTY	1
TRSTAT	ENABL
TRTYPE	IN
TPPLIB	/opt/diosa_xtp/lib/libdxtpncmtpouttpb. so
TPPENTRY	di_ncm_TimeoutTpp
TPPLANG	C
CLASS	DXTP_NCMHLMOUT_CLS
LINK	INIT
CPUTIME	60
ELPTIME	60
ABORT	CONT
TIMEOVER	CONT
#ヘルスチェック応答 TPP	
TRNSID	D17HRP
TRPRTY	1
TRSTAT	ENABL
TRTYPE	IN
TPPLIB	/opt/diosa_xtp/lib/libdxtpncmtpresptpb. so
TPPENTRY	di_ncm_HlthTpp
TPPLANG	C
CLASS	DXTP_NCMHLRESP_CLS
LINK	INIT
CPUTIME	60
ELPTIME	60
ABORT	CONT
TIMEOVER	CONT

E. 3. 7 TCP/IP リスナ定義 (TCPIP_TPLSN01. ped)

CLASS	CLS_TPLSN01
PATH	/opt/oltp/bin/tcpiplsn
ARGS	-sg_fname /home/diosa/tpbase/catalog/TCPIP_TPLSN01. sg
MAXSTARTTRIES	4
MAXPROCESS	1
MINPROCESS	1
UNITTIME	10
#OUTPUTDEV	/home/diosa/tpbase/log/TCPIP_TPLSN01. out
#ERRORDEV	/home/diosa/tpbase/log/TCPIP_TPLSN01. err
TRIESRESET	3600
TPAPLOOP	OFF

E. 3. 8 TCP/IP リスナ SG 定義 (TCPIP_TPLSN01. sg)

LSNQNAME	TCPIP_TPLSN01	
pluname	31141	
tpsexit_call	1	
tpsexit_libname	libdxtpcoclsntpb. so	
autotrmgen	OFF	#端末自動生成はしない

DATA_INPUT_WATCHT	0	# 端末からのデータ入力監視時間
-------------------	---	------------------

E. 3. 9 TCP/IP リスナ VD 定義 (TCPIP_TPLSN01. vd)

VD	VDC0000
VDTYPE	TERM
VD	VDM00001
VDTYPE	TERM

E. 3. 10 TCP/IP リスナ 端末定義 (TCPIP_TPLSN01. term)

TERMID	RMTC0000
TID	RMTC0000
LSNQNAME	TCPIP_TPLSN01
PROTOCOL	TCPIP
TERMCATEGORY	SJISAP
SYMSGLANG	JAPAN
SYMSG	NO
TERMSTAT	ENABL
INITAP	DXTP_AP
INITVD	VDC0000
HOSTNAME	192. 168. 0. 1
SERVICENAME	31141
INITDATA	192. 168. 0. 11
PORTNO_NOCHK	YES
CONNECTMODE	CONCURRENT
TERMID	RMTM00001
TID	RMTM00001
LSNQNAME	TCPIP_TPLSN01
PROTOCOL	TCPIP
TERMCATEGORY	SJISAP
SYMSGLANG	JAPAN
SYMSG	NO
TERMSTAT	ENABL
INITAP	DXTP_AP
INITVD	VDM00001
HOSTNAME	192. 168. 0. 1
SERVICENAME	31141
INITDATA	192. 168. 0. 11
PORTNO_NOCHK	YES
CONNECTMODE	CONCURRENT

E. 4 TAM

E. 4. 1 tammng. conf

[cluster]	
multi_instance	=yes
cluster_name	=CL_00
instance_group1	=(INS001@VTT1AP01, INS001@VTT1AP02)
instance_group2	=(INS002@VTT1AP02, INS002@VTT1AP01)
instance_group3	=(INS003@VTT1AP01, INS003@VTT1AP02)
[node]	
node_name	=VTT1AP01
ip_memory_backup	=(10. 247. 177. 33)
[node]	
node_name	=VTT1AP02
ip_memory_backup	=(10. 247. 177. 34)
[vnode]	
virtual_node_name	=TAMLN001MASTER
backup_role	=master
[vnode]	
virtual_node_name	=TAMLN002MASTER
backup_role	=master
[vnode]	
virtual_node_name	=TAMLN003MASTER
backup_role	=master
[vnode]	
virtual_node_name	=TAMLN001SLAVE00
backup_role	=slave
[vnode]	
virtual_node_name	=TAMLN002SLAVE00
backup_role	=slave
[vnode]	
virtual_node_name	=TAMLN003SLAVE00
backup_role	=slave
[instance]	
instance_name	=INS001
base_node	=VTT1AP01
i_init_shm_size	=512
i_init_shm_num	=1
i_max_shm_num	=16
i_shm_extend_size	=260
i_port	=31170
i_port_range	=10
i_start_check_port	=31161
i_ccm_thread_num	=10
i_send_thread_num	=2
i_rcv_thread_num	=5
i_sendproc_thread_num	=2
i_rcvproc_thread_num	=2
i_max_proc_request_num	=64
i_send_piece_size	=62816
i_stc_set_priority	=no
i_send_ratio	=3:1
i_send_delay	=0
i_timer_max_queue_size	=4000
i_auto_lock_check	=no
i_max_tpid_num	=1024
i_net_send_buffer_size	=256
i_net_rcv_buffer_size	=256
i_auto_lock_check_syslog	=summary
i_blockage_to_down	=no
i_do_backup	=yes
i_tx_shm_table_size	=50
i_tx_thread_num	=100
[instance]	
instance_name	=INS002
base_node	=VTT1AP02
i_init_shm_size	=512
i_init_shm_num	=1
i_max_shm_num	=16
i_shm_extend_size	=260

```

i_port                =31180
i_port_range          =10
i_start_check_port    =31162
i_ccm_thread_num      =10
i_send_thread_num     =2
i_recv_thread_num     =5
i_sendproc_thread_num =2
i_recvproc_thread_num =2
i_max_proc_request_num =64
i_send_piece_size     =62816
i_stc_set_priority    =no
i_send_ratio          =3:1
i_send_delay          =0
i_timer_max_queue_size =4000
i_auto_lock_check     =no
i_max_tpid_num        =1024
i_net_send_buffer_size =256
i_net_recv_buffer_size =256
i_auto_lock_check_syslog =summary
i_blockage_to_down    =no
i_do_backup           =yes
i_tx_shm_table_size   =50
i_tx_thread_num       =100

[instance]
instance_name         =INS003
base_node             =VTT1AP01
i_init_shm_size       =512
i_init_shm_num        =1
i_max_shm_num         =16
i_shm_extend_size     =260
i_port               =31190
i_port_range          =10
i_start_check_port    =31163
i_ccm_thread_num      =10
i_send_thread_num     =2
i_recv_thread_num     =5
i_sendproc_thread_num =2
i_recvproc_thread_num =2
i_max_proc_request_num =64
i_send_piece_size     =62816
i_stc_set_priority    =no
i_send_ratio          =3:1
i_send_delay          =0
i_timer_max_queue_size =4000
i_auto_lock_check     =no
i_max_tpid_num        =1024
i_net_send_buffer_size =256
i_net_recv_buffer_size =256
i_auto_lock_check_syslog =summary
i_blockage_to_down    =no
i_do_backup           =yes
i_tx_shm_table_size   =50
i_tx_thread_num       =100

```

E. 4. 2 table.conf

```

[common]
update_log_size      = 1
update_log_extend_size = 1
max_update_log_size  = 50
trace_entry          = 0
record_duplicate_check = no
check_record_num      = 0
mutex_spin_num       = 0
dirty_read           = no

# ディレイドスーパーストリーム管理ファイル(DIOSA_DELAYED_SPST)
[table]
tam_table_name       = DIOSA_DELAYED_SPST_0000000001
tam_table_file       = /home/diosa/tam/table/diosa_delayed_spst_0000000001.tam
backup               = yes
share                = yes
sequence             = no
primary_index_duplicate = no
index_reserve_size   = 0
data_reserve_size    = 0
auto_extend          = yes
auto_extend_size     = 1
max_auto_extend_size = 5

```

```

record_size      = 144
checksum         = no
read_only        = no
index_number     = 1
index_name1      = DIOSA_DELAYED_SPST_00_IDX
index_entry_num1 = 10
keys1            = 0:16
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1    = 1000
auto_release_index1 = no
init_record_num  = 1000
txlog            = no

# ディレイード書き込み制御ファイル(DIOSA_DELAYED_STRM)
[table]
tam_table_name    = DIOSA_DELAYED_STRM_0000000001
tam_table_file    = /home/diosa/tam/table/diosa_delayed_strm_0000000001.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5
record_size       = 176
checksum          = no
read_only         = no
index_number      = 5
index_name1       = DIOSA_DELAYED_STRM_00_IDX
index_entry_num1  = 10
keys1             = 0:16, 16:16, 32:4
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1     = 1000
auto_release_index1 = no
index_name2       = DIOSA_DELAYED_STRM_01_IDX
index_entry_num2  = 10
keys2             = 0:16, 0:16, 16:16, 32:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2     = 1000
auto_release_index2 = no
index_name3       = DIOSA_DELAYED_STRM_04_IDX
index_entry_num3  = 10
keys3             = 0:16, 68:4, 32:4, 0:16, 16:16, 32:4
auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3     = 1000
auto_release_index3 = no
index_name4       = DIOSA_DELAYED_STRM_05_IDX
index_entry_num4  = 3
keys4             = 0:16, 68:4, 128:8, 0:16, 16:16, 32:4
auto_extend_key_num4 = 20
max_auto_extend_count4 = 10
init_key_num4     = 10
auto_release_index4 = no
index_name5       = DIOSA_DELAYED_STRM_06_IDX
index_entry_num5  = 3
keys5             = 0:16, 32:4, 0:16, 16:16, 32:4
auto_extend_key_num5 = 20
max_auto_extend_count5 = 10
init_key_num5     = 10
auto_release_index5 = no
init_record_num   = 1000
txlog             = no

# ディレイードユニット管理ファイル(DIOSA_DELAYED_UNIT)
[table]
tam_table_name    = DIOSA_DELAYED_UNIT_0000000001
tam_table_file    = /home/diosa/tam/table/diosa_delayed_unit_0000000001.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5

```

```
record_size      = 96
checksum         = no
read_only        = no
index_number     = 2
index_name1      = DIOSA_DELAYED_UNIT_00_IDX
index_entry_num1 = 10
keys1            = 0:16, 16:1, 17:16
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1    = 1000
auto_release_index1 = no
index_name2      = DIOSA_DELAYED_UNIT_01_IDX
index_entry_num2 = 10
keys2            = 0:16, 0:16, 16:1, 17:16
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2    = 1000
auto_release_index2 = no
init_record_num  = 1000
txlog            = no

# ディレードスタック管理ファイル(DIOSA_DELAYED_STACK)
[table]
tam_table_name    = DIOSA_DELAYED_STACK_0000000001
tam_table_file    = /home/diosa/tam/table/diosa_delayed_stack_0000000001.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5
record_size       = 48
checksum          = no
read_only         = no
index_number      = 3
index_name1       = DIOSA_DELAYED_STACK_00_IDX
index_entry_num1  = 10
keys1             = 0:16, 16:4
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1     = 1000
auto_release_index1 = no
index_name2       = DIOSA_DELAYED_STACK_01_IDX
index_entry_num2  = 10
keys2             = 0:16, 0:16, 16:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2     = 1000
auto_release_index2 = no
index_name3       = DIOSA_DELAYED_STACK_02_IDX
index_entry_num3  = 10
keys3             = 0:16, 20:4, 0:16, 16:4
auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3     = 1000
auto_release_index3 = no
init_record_num   = 1000
txlog             = no

# ディレードスタック・通番管理ファイル(DIOSA_DELAYED_STACKDATANO)
[table]
tam_table_name    = DIOSA_DELAYED_STACKDATANO_0000000001
tam_table_file    = /home/diosa/tam/table/diosa_delayed_stackdatano_0000000001.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5
record_size       = 88
checksum          = no
read_only         = no
index_number      = 6
index_name1       = DIOSA_DELAYED_STKDATANO_00_IDX
index_entry_num1  = 10
keys1             = 0:16, 16:16, 32:4, 36:4
```

```

auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1 = 1000
auto_release_index1 = no
index_name2 = DIOSA_DELAYED_STKDATANO_01_IDX
index_entry_num2 = 10
keys2 = 0:16, 16:16, 36:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2 = 1000
auto_release_index2 = no
index_name3 = DIOSA_DELAYED_STKDATANO_02_IDX
index_entry_num3 = 10
keys3 = 0:16, 16:16, 32:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3 = 1000
auto_release_index3 = no
index_name4 = DIOSA_DELAYED_STKDATANO_03_IDX
index_entry_num4 = 10
keys4 = 0:16, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num4 = 20
max_auto_extend_count4 = 10
init_key_num4 = 1000
auto_release_index4 = no
index_name5 = DIOSA_DELAYED_STKDATANO_04_IDX
index_entry_num5 = 10
keys5 = 0:16, 16:16, 32:4, 40:8, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num5 = 20
max_auto_extend_count5 = 10
init_key_num5 = 1000
auto_release_index5 = no
index_name6 = DIOSA_DELAYED_STKDATANO_05_IDX
index_entry_num6 = 10
keys6 = 0:16, 36:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num6 = 20
max_auto_extend_count6 = 10
init_key_num6 = 1000
auto_release_index6 = no
init_record_num = 1000
txlog = no

# ディレイドプールファイル 01 (DIOSA_DELAYED_POOL_0001)
[table]
tam_table_name = DIOSA_DELAYED_POOL_0001_0000000001
tam_table_file = /home/diosa/tam/table/diosa_delayed_pool_0001_0000000001.tam
backup = yes
share = yes
sequence = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend = yes
auto_extend_size = 2
max_auto_extend_size = 2000
record_size = 4152
checksum = no
read_only = no
index_number = 1
index_name1 = DIOSA_DELAYED_POOL_0001_00_IDX
index_entry_num1 = 20
keys1 = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1 = 1
auto_release_index1 = no
init_record_num = 1
txlog = no

# ディレイドプールファイル 02 (DIOSA_DELAYED_POOL_0002)
[table]
tam_table_name = DIOSA_DELAYED_POOL_0002_0000000001
tam_table_file = /home/diosa/tam/table/diosa_delayed_pool_0002_0000000001.tam
backup = yes
share = yes
sequence = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend = yes
auto_extend_size = 2
max_auto_extend_size = 2000

```



```
record_size      = 4152
checksum         = no
read_only        = no
index_number     = 1
index_name1      = DIOSA_DELAYED_POOL_0002_00_IDX
index_entry_num1 = 20
keys1            = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1    = 1
auto_release_index1 = no
init_record_num  = 1
txlog            = no

# ディレイドプールファイル 03 (DIOSA_DELAYED_POOL_0003)
[table]
tam_table_name    = DIOSA_DELAYED_POOL_0003_0000000001
tam_table_file    = /home/diosa/tam/table/diosa_delayed_pool_0003_0000000001.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 2
max_auto_extend_size = 2000
record_size       = 4152
checksum          = no
read_only         = no
index_number      = 1
index_name1       = DIOSA_DELAYED_POOL_0003_00_IDX
index_entry_num1  = 20
keys1             = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1     = 1
auto_release_index1 = no
init_record_num   = 1
txlog             = no

# ディレイドプールファイル 04 (DIOSA_DELAYED_POOL_0004)
[table]
tam_table_name    = DIOSA_DELAYED_POOL_0004_0000000001
tam_table_file    = /home/diosa/tam/table/diosa_delayed_pool_0004_0000000001.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 2
max_auto_extend_size = 2000
record_size       = 4152
checksum          = no
read_only         = no
index_number      = 1
index_name1       = DIOSA_DELAYED_POOL_0004_00_IDX
index_entry_num1  = 20
keys1             = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1     = 1
auto_release_index1 = no
init_record_num   = 1
txlog             = no

# ディレイドプールファイル 05 (DIOSA_DELAYED_POOL_0005)
[table]
tam_table_name    = DIOSA_DELAYED_POOL_0005_0000000001
tam_table_file    = /home/diosa/tam/table/diosa_delayed_pool_0005_0000000001.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 2
max_auto_extend_size = 2000
record_size       = 4152
```

```

checksum          = no
read_only         = no
index_number      = 1
index_name1       = DIOSA_DELAYED_POOL_0005_00_IDX
index_entry_num1  = 20
keys1             = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1     = 1
auto_release_index1 = no
init_record_num   = 1
txlog             = no

# センダ ユニット管理ファイル(DIOSA_SENDER_UNIT)
[table]
tam_table_name     = DIOSA_SENDER_UNIT_0000000001
tam_table_file     = /home/diosa/tam/table/diosa_sender_unit_0000000001.tam
backup             = yes
share              = yes
sequence           = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size  = 0
auto_extend        = yes
auto_extend_size   = 1
max_auto_extend_size = 5
record_size        = 248
checksum           = no
read_only          = no
dirty_read         = no
index_number       = 1
index_name1        = DIOSA_SENDER_UNIT_00_IDX
index_entry_num1   = 10
keys1              = 0:32
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1      = 10
auto_release_index1 = no
init_record_num    = 10
txlog              = no

# センダ ストリーム管理ファイル(DIOSA_SENDER_STRM)
[table]
tam_table_name     = DIOSA_SENDER_STRM_0000000001
tam_table_file     = /home/diosa/tam/table/diosa_sender_strm_0000000001.tam
backup             = yes
share              = yes
sequence           = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size  = 0
auto_extend        = yes
auto_extend_size   = 1
max_auto_extend_size = 5
record_size        = 168
checksum           = no
read_only          = no
dirty_read         = no
index_number       = 1
index_name1        = DIOSA_SENDER_STRM_00_IDX
index_entry_num1   = 10
keys1              = 0:48
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1      = 10
auto_release_index1 = no
init_record_num    = 10
txlog              = no

# デイレードスーパーストリーム管理ファイル(DIOSA_DELAYED_SPST)
[table]
tam_table_name     = DIOSA_DELAYED_SPST_0000000002
tam_table_file     = /home/diosa/tam/table/diosa_delayed_spst_0000000002.tam
backup             = yes
share              = yes
sequence           = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size  = 0
auto_extend        = yes
auto_extend_size   = 1
max_auto_extend_size = 5

```

```

record_size      = 144
checksum         = no
read_only        = no
index_number     = 1
index_name1      = DIOSA_DELAYED_SPST_00_IDX
index_entry_num1 = 10
keys1            = 0:16
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1    = 1000
auto_release_index1 = no
init_record_num  = 1000
txlog            = no

# ディレイード書き込み制御ファイル(DIOSA_DELAYED_STRM)
[table]
tam_table_name    = DIOSA_DELAYED_STRM_0000000002
tam_table_file    = /home/diosa/tam/table/diosa_delayed_strm_0000000002.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5
record_size       = 176
checksum          = no
read_only         = no
index_number      = 5
index_name1       = DIOSA_DELAYED_STRM_00_IDX
index_entry_num1  = 10
keys1             = 0:16, 16:16, 32:4
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1     = 1000
auto_release_index1 = no
index_name2       = DIOSA_DELAYED_STRM_01_IDX
index_entry_num2  = 10
keys2             = 0:16, 0:16, 16:16, 32:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2     = 1000
auto_release_index2 = no
index_name3       = DIOSA_DELAYED_STRM_04_IDX
index_entry_num3  = 10
keys3             = 0:16, 68:4, 32:4, 0:16, 16:16, 32:4
auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3     = 1000
auto_release_index3 = no
index_name4       = DIOSA_DELAYED_STRM_05_IDX
index_entry_num4  = 3
keys4             = 0:16, 68:4, 128:8, 0:16, 16:16, 32:4
auto_extend_key_num4 = 20
max_auto_extend_count4 = 10
init_key_num4     = 10
auto_release_index4 = no
index_name5       = DIOSA_DELAYED_STRM_06_IDX
index_entry_num5  = 3
keys5             = 0:16, 32:4, 0:16, 16:16, 32:4
auto_extend_key_num5 = 20
max_auto_extend_count5 = 10
init_key_num5     = 10
auto_release_index5 = no
init_record_num   = 1000
txlog             = no

# ディレイードユニット管理ファイル(DIOSA_DELAYED_UNIT)
[table]
tam_table_name    = DIOSA_DELAYED_UNIT_0000000002
tam_table_file    = /home/diosa/tam/table/diosa_delayed_unit_0000000002.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5

```

```
record_size      = 96
checksum         = no
read_only        = no
index_number     = 2
index_name1      = DIOSA_DELAYED_UNIT_00_IDX
index_entry_num1 = 10
keys1            = 0:16, 16:1, 17:16
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1    = 1000
auto_release_index1 = no
index_name2      = DIOSA_DELAYED_UNIT_01_IDX
index_entry_num2 = 10
keys2            = 0:16, 0:16, 16:1, 17:16
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2    = 1000
auto_release_index2 = no
init_record_num  = 1000
txlog            = no

# ディレイドスタック管理ファイル(DIOSA_DELAYED_STACK)
[table]
tam_table_name    = DIOSA_DELAYED_STACK_0000000002
tam_table_file    = /home/diosa/tam/table/diosa_delayed_stack_0000000002.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5
record_size       = 48
checksum          = no
read_only         = no
index_number      = 3
index_name1       = DIOSA_DELAYED_STACK_00_IDX
index_entry_num1  = 10
keys1             = 0:16, 16:4
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1     = 1000
auto_release_index1 = no
index_name2       = DIOSA_DELAYED_STACK_01_IDX
index_entry_num2  = 10
keys2             = 0:16, 0:16, 16:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2     = 1000
auto_release_index2 = no
index_name3       = DIOSA_DELAYED_STACK_02_IDX
index_entry_num3  = 10
keys3             = 0:16, 20:4, 0:16, 16:4
auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3     = 1000
auto_release_index3 = no
init_record_num   = 1000
txlog             = no

# ディレイドスタック・通番管理ファイル(DIOSA_DELAYED_STACKDATANO)
[table]
tam_table_name    = DIOSA_DELAYED_STACKDATANO_0000000002
tam_table_file    = /home/diosa/tam/table/diosa_delayed_stackdatano_0000000002.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5
record_size       = 88
checksum          = no
read_only         = no
index_number      = 6
index_name1       = DIOSA_DELAYED_STKDATANO_00_IDX
index_entry_num1  = 10
keys1             = 0:16, 16:16, 32:4, 36:4
```

```

auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1 = 1000
auto_release_index1 = no
index_name2 = DIOSA_DELAYED_STKDATANO_01_IDX
index_entry_num2 = 10
keys2 = 0:16, 16:16, 36:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2 = 1000
auto_release_index2 = no
index_name3 = DIOSA_DELAYED_STKDATANO_02_IDX
index_entry_num3 = 10
keys3 = 0:16, 16:16, 32:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3 = 1000
auto_release_index3 = no
index_name4 = DIOSA_DELAYED_STKDATANO_03_IDX
index_entry_num4 = 10
keys4 = 0:16, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num4 = 20
max_auto_extend_count4 = 10
init_key_num4 = 1000
auto_release_index4 = no
index_name5 = DIOSA_DELAYED_STKDATANO_04_IDX
index_entry_num5 = 10
keys5 = 0:16, 16:16, 32:4, 40:8, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num5 = 20
max_auto_extend_count5 = 10
init_key_num5 = 1000
auto_release_index5 = no
index_name6 = DIOSA_DELAYED_STKDATANO_05_IDX
index_entry_num6 = 10
keys6 = 0:16, 36:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num6 = 20
max_auto_extend_count6 = 10
init_key_num6 = 1000
auto_release_index6 = no
init_record_num = 1000
txlog = no

# ディレイドプールファイル 01 (DIOSA_DELAYED_POOL_0001)
[table]
tam_table_name = DIOSA_DELAYED_POOL_0001_0000000002
tam_table_file = /home/diosa/tam/table/diosa_delayed_pool_0001_0000000002.tam
backup = yes
share = yes
sequence = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend = yes
auto_extend_size = 2
max_auto_extend_size = 2000
record_size = 4152
checksum = no
read_only = no
index_number = 1
index_name1 = DIOSA_DELAYED_POOL_0001_00_IDX
index_entry_num1 = 20
keys1 = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1 = 1
auto_release_index1 = no
init_record_num = 1
txlog = no

# ディレイドプールファイル 02 (DIOSA_DELAYED_POOL_0002)
[table]
tam_table_name = DIOSA_DELAYED_POOL_0002_0000000002
tam_table_file = /home/diosa/tam/table/diosa_delayed_pool_0002_0000000002.tam
backup = yes
share = yes
sequence = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend = yes
auto_extend_size = 2
max_auto_extend_size = 2000

```

```

record_size      = 4152
checksum         = no
read_only        = no
index_number     = 1
index_name1      = DIOSA_DELAYED_POOL_0002_00_IDX
index_entry_num1 = 20
keys1            = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1    = 1
auto_release_index1 = no
init_record_num  = 1
txlog            = no

# ディレイドプールファイル 03 (DIOSA_DELAYED_POOL_0003)
[table]
tam_table_name      = DIOSA_DELAYED_POOL_0003_0000000002
tam_table_file      = /home/diosa/tam/table/diosa_delayed_pool_0003_0000000002.tam
backup              = yes
share               = yes
sequence            = no
primary_index_duplicate = no
index_reserve_size  = 0
data_reserve_size   = 0
auto_extend         = yes
auto_extend_size     = 2
max_auto_extend_size = 2000
record_size         = 4152
checksum            = no
read_only           = no
index_number        = 1
index_name1         = DIOSA_DELAYED_POOL_0003_00_IDX
index_entry_num1    = 20
keys1               = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1       = 1
auto_release_index1 = no
init_record_num     = 1
txlog               = no

# ディレイドプールファイル 04 (DIOSA_DELAYED_POOL_0004)
[table]
tam_table_name      = DIOSA_DELAYED_POOL_0004_0000000002
tam_table_file      = /home/diosa/tam/table/diosa_delayed_pool_0004_0000000002.tam
backup              = yes
share               = yes
sequence            = no
primary_index_duplicate = no
index_reserve_size  = 0
data_reserve_size   = 0
auto_extend         = yes
auto_extend_size     = 2
max_auto_extend_size = 2000
record_size         = 4152
checksum            = no
read_only           = no
index_number        = 1
index_name1         = DIOSA_DELAYED_POOL_0004_00_IDX
index_entry_num1    = 20
keys1               = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1       = 1
auto_release_index1 = no
init_record_num     = 1
txlog               = no

# ディレイドプールファイル 05 (DIOSA_DELAYED_POOL_0005)
[table]
tam_table_name      = DIOSA_DELAYED_POOL_0005_0000000002
tam_table_file      = /home/diosa/tam/table/diosa_delayed_pool_0005_0000000002.tam
backup              = yes
share               = yes
sequence            = no
primary_index_duplicate = no
index_reserve_size  = 0
data_reserve_size   = 0
auto_extend         = yes
auto_extend_size     = 2
max_auto_extend_size = 2000
record_size         = 4152

```

```

checksum          = no
read_only         = no
index_number      = 1
index_name1       = DIOSA_DELAYED_POOL_0005_00_IDX
index_entry_num1  = 20
keys1             = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1     = 1
auto_release_index1 = no
init_record_num   = 1
txlog             = no

# センダ ユニット管理ファイル(DIOSA_SENDER_UNIT)
[table]
tam_table_name     = DIOSA_SENDER_UNIT_0000000002
tam_table_file     = /home/diosa/tam/table/diosa_sender_unit_0000000002.tam
backup             = yes
share              = yes
sequence           = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size  = 0
auto_extend        = yes
auto_extend_size   = 1
max_auto_extend_size = 5
record_size        = 248
checksum           = no
read_only          = no
dirty_read         = no
index_number       = 1
index_name1        = DIOSA_SENDER_UNIT_00_IDX
index_entry_num1   = 10
keys1              = 0:32
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1      = 10
auto_release_index1 = no
init_record_num    = 10
txlog              = no

# センダ ストリーム管理ファイル(DIOSA_SENDER_STRM)
[table]
tam_table_name     = DIOSA_SENDER_STRM_0000000002
tam_table_file     = /home/diosa/tam/table/diosa_sender_strm_0000000002.tam
backup             = yes
share              = yes
sequence           = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size  = 0
auto_extend        = yes
auto_extend_size   = 1
max_auto_extend_size = 5
record_size        = 168
checksum           = no
read_only          = no
dirty_read         = no
index_number       = 1
index_name1        = DIOSA_SENDER_STRM_00_IDX
index_entry_num1   = 10
keys1              = 0:48
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1      = 10
auto_release_index1 = no
init_record_num    = 10
txlog              = no

# DSAM システム制御ファイル(DIOSA_DSAM)
[table]
tam_table_name     = DIOSA_DSAM_0000000003
tam_table_file     = /home/diosa/tam/table/diosa_dsam_0000000003.tam
backup             = yes
share              = yes
sequence           = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size  = 0
auto_extend        = yes
auto_extend_size   = 1
max_auto_extend_size = 5

```

```
record_size      = 96
checksum         = no
read_only       = no
index_number     = 1
index_name1      = DIOSA_DSAM_00_IDX
index_entry_num1 = 10
keys1           = 0:4
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1   = 1000
auto_release_index1 = no
init_record_num = 1000
txlog           = no

# ディレイドスーパーストリーム管理ファイル(DIOSA_DELAYED_SPST)
[table]
tam_table_name    = DIOSA_DELAYED_SPST_0000000003
tam_table_file    = /home/diosa/tam/table/diosa_delayed_spst_0000000003.tam
backup           = yes
share            = yes
sequence         = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend      = yes
auto_extend_size = 1
max_auto_extend_size = 5
record_size      = 144
checksum         = no
read_only       = no
index_number     = 1
index_name1      = DIOSA_DELAYED_SPST_00_IDX
index_entry_num1 = 10
keys1           = 0:16
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1   = 1000
auto_release_index1 = no
init_record_num = 1000
txlog           = no

# ディレイド書き込み制御ファイル(DIOSA_DELAYED_STRM)
[table]
tam_table_name    = DIOSA_DELAYED_STRM_0000000003
tam_table_file    = /home/diosa/tam/table/diosa_delayed_strm_0000000003.tam
backup           = yes
share            = yes
sequence         = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend      = yes
auto_extend_size = 1
max_auto_extend_size = 5
record_size      = 176
checksum         = no
read_only       = no
index_number     = 5
index_name1      = DIOSA_DELAYED_STRM_00_IDX
index_entry_num1 = 10
keys1           = 0:16, 16:16, 32:4
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1   = 1000
auto_release_index1 = no
index_name2      = DIOSA_DELAYED_STRM_01_IDX
index_entry_num2 = 10
keys2           = 0:16, 0:16, 16:16, 32:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2   = 1000
auto_release_index2 = no
index_name3      = DIOSA_DELAYED_STRM_04_IDX
index_entry_num3 = 10
keys3           = 0:16, 68:4, 32:4, 0:16, 16:16, 32:4
auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3   = 1000
auto_release_index3 = no
index_name4      = DIOSA_DELAYED_STRM_05_IDX
index_entry_num4 = 3
keys4           = 0:16, 68:4, 128:8, 0:16, 16:16, 32:4
```



```
auto_extend_key_num4 = 20
max_auto_extend_count4 = 10
init_key_num4 = 10
auto_release_index4 = no
index_name5 = DIOSA_DELAYED_STRM_06_IDX
index_entry_num5 = 3
keys5 = 0:16, 32:4, 0:16, 16:16, 32:4
auto_extend_key_num5 = 20
max_auto_extend_count5 = 10
init_key_num5 = 10
auto_release_index5 = no
init_record_num = 1000
txlog = no

# ディレードユニット管理ファイル(DIOSA_DELAYED_UNIT)
[table]
tam_table_name = DIOSA_DELAYED_UNIT_0000000003
tam_table_file = /home/diosa/tam/table/diosa_delayed_unit_0000000003.tam
backup = yes
share = yes
sequence = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend = yes
auto_extend_size = 1
max_auto_extend_size = 5
record_size = 96
checksum = no
read_only = no
index_number = 2
index_name1 = DIOSA_DELAYED_UNIT_00_IDX
index_entry_num1 = 10
keys1 = 0:16, 16:1, 17:16
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1 = 1000
auto_release_index1 = no
index_name2 = DIOSA_DELAYED_UNIT_01_IDX
index_entry_num2 = 10
keys2 = 0:16, 0:16, 16:1, 17:16
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2 = 1000
auto_release_index2 = no
init_record_num = 1000
txlog = no

# ディレードスタック管理ファイル(DIOSA_DELAYED_STACK)
[table]
tam_table_name = DIOSA_DELAYED_STACK_0000000003
tam_table_file = /home/diosa/tam/table/diosa_delayed_stack_0000000003.tam
backup = yes
share = yes
sequence = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend = yes
auto_extend_size = 1
max_auto_extend_size = 5
record_size = 48
checksum = no
read_only = no
index_number = 3
index_name1 = DIOSA_DELAYED_STACK_00_IDX
index_entry_num1 = 10
keys1 = 0:16, 16:4
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1 = 1000
auto_release_index1 = no
index_name2 = DIOSA_DELAYED_STACK_01_IDX
index_entry_num2 = 10
keys2 = 0:16, 0:16, 16:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2 = 1000
auto_release_index2 = no
index_name3 = DIOSA_DELAYED_STACK_02_IDX
index_entry_num3 = 10
keys3 = 0:16, 20:4, 0:16, 16:4
```

```

auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3 = 1000
auto_release_index3 = no
init_record_num = 1000
txlog = no

# デイレーdstack・通番管理ファイル(DIOSA_DELAYED_STACKDATANO)
[table]
tam_table_name = DIOSA_DELAYED_STACKDATANO_0000000003
tam_table_file = /home/diosa/tam/table/diosa_delayed_stackdatano_0000000003.tam
backup = yes
share = yes
sequence = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend = yes
auto_extend_size = 1
max_auto_extend_size = 5
record_size = 88
checksum = no
read_only = no
index_number = 6
index_name1 = DIOSA_DELAYED_STKDATANO_00_IDX
index_entry_num1 = 10
keys1 = 0:16, 16:16, 32:4, 36:4
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1 = 1000
auto_release_index1 = no
index_name2 = DIOSA_DELAYED_STKDATANO_01_IDX
index_entry_num2 = 10
keys2 = 0:16, 16:16, 36:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num2 = 20
max_auto_extend_count2 = 10
init_key_num2 = 1000
auto_release_index2 = no
index_name3 = DIOSA_DELAYED_STKDATANO_02_IDX
index_entry_num3 = 10
keys3 = 0:16, 16:16, 32:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num3 = 20
max_auto_extend_count3 = 10
init_key_num3 = 1000
auto_release_index3 = no
index_name4 = DIOSA_DELAYED_STKDATANO_03_IDX
index_entry_num4 = 10
keys4 = 0:16, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num4 = 20
max_auto_extend_count4 = 10
init_key_num4 = 1000
auto_release_index4 = no
index_name5 = DIOSA_DELAYED_STKDATANO_04_IDX
index_entry_num5 = 10
keys5 = 0:16, 16:16, 32:4, 40:8, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num5 = 20
max_auto_extend_count5 = 10
init_key_num5 = 1000
auto_release_index5 = no
index_name6 = DIOSA_DELAYED_STKDATANO_05_IDX
index_entry_num6 = 10
keys6 = 0:16, 36:4, 0:16, 16:16, 32:4, 36:4
auto_extend_key_num6 = 20
max_auto_extend_count6 = 10
init_key_num6 = 1000
auto_release_index6 = no
init_record_num = 1000
txlog = no

# デイレーdpoolファイル 01(DIOSA_DELAYED_POOL_0001)
[table]
tam_table_name = DIOSA_DELAYED_POOL_0001_0000000003
tam_table_file = /home/diosa/tam/table/diosa_delayed_pool_0001_0000000003.tam
backup = yes
share = yes
sequence = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend = yes
auto_extend_size = 2
max_auto_extend_size = 2000

```

```

record_size      = 4152
checksum         = no
read_only        = no
index_number     = 1
index_name1      = DIOSA_DELAYED_POOL_0001_00_IDX
index_entry_num1 = 20
keys1            = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1    = 1
auto_release_index1 = no
init_record_num  = 1
txlog            = no

# デイレードプールファイル 02 (DIOSA_DELAYED_POOL_0002)
[table]
tam_table_name      = DIOSA_DELAYED_POOL_0002_0000000003
tam_table_file      = /home/diosa/tam/table/diosa_delayed_pool_0002_0000000003.tam
backup              = yes
share               = yes
sequence            = no
primary_index_duplicate = no
index_reserve_size  = 0
data_reserve_size   = 0
auto_extend         = yes
auto_extend_size    = 2
max_auto_extend_size = 2000
record_size         = 4152
checksum            = no
read_only           = no
index_number        = 1
index_name1         = DIOSA_DELAYED_POOL_0002_00_IDX
index_entry_num1    = 20
keys1               = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1       = 1
auto_release_index1 = no
init_record_num     = 1
txlog               = no

# デイレードプールファイル 03 (DIOSA_DELAYED_POOL_0003)
[table]
tam_table_name      = DIOSA_DELAYED_POOL_0003_0000000003
tam_table_file      = /home/diosa/tam/table/diosa_delayed_pool_0003_0000000003.tam
backup              = yes
share               = yes
sequence            = no
primary_index_duplicate = no
index_reserve_size  = 0
data_reserve_size   = 0
auto_extend         = yes
auto_extend_size    = 2
max_auto_extend_size = 2000
record_size         = 4152
checksum            = no
read_only           = no
index_number        = 1
index_name1         = DIOSA_DELAYED_POOL_0003_00_IDX
index_entry_num1    = 20
keys1               = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1       = 1
auto_release_index1 = no
init_record_num     = 1
txlog               = no

# デイレードプールファイル 04 (DIOSA_DELAYED_POOL_0004)
[table]
tam_table_name      = DIOSA_DELAYED_POOL_0004_0000000003
tam_table_file      = /home/diosa/tam/table/diosa_delayed_pool_0004_0000000003.tam
backup              = yes
share               = yes
sequence            = no
primary_index_duplicate = no
index_reserve_size  = 0
data_reserve_size   = 0
auto_extend         = yes
auto_extend_size    = 2
max_auto_extend_size = 2000
record_size         = 4152

```

```
checksum          = no
read_only         = no
index_number      = 1
index_name1       = DIOSA_DELAYED_POOL_0004_00_IDX
index_entry_num1  = 20
keys1             = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1     = 1
auto_release_index1 = no
init_record_num   = 1
txlog             = no

# ディレードプールファイル 05 (DIOSA_DELAYED_POOL_0005)
[table]
tam_table_name     = DIOSA_DELAYED_POOL_0005_0000000003
tam_table_file     = /home/diosa/tam/table/diosa_delayed_pool_0005_0000000003.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 2
max_auto_extend_size = 2000
record_size       = 4152
checksum          = no
read_only         = no
index_number      = 1
index_name1       = DIOSA_DELAYED_POOL_0005_00_IDX
index_entry_num1  = 20
keys1             = 24:28, 6:2
auto_extend_key_num1 = 500
max_auto_extend_count1 = 0
init_key_num1     = 1
auto_release_index1 = no
init_record_num   = 1
txlog             = no

# センダ ユニット管理ファイル (DIOSA_SENDER_UNIT)
[table]
tam_table_name     = DIOSA_SENDER_UNIT_0000000003
tam_table_file     = /home/diosa/tam/table/diosa_sender_unit_0000000003.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5
record_size       = 248
checksum          = no
read_only         = no
dirty_read        = no
index_number      = 1
index_name1       = DIOSA_SENDER_UNIT_00_IDX
index_entry_num1  = 10
keys1             = 0:32
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1     = 10
auto_release_index1 = no
init_record_num   = 10
txlog             = no

# センダ ストリーム管理ファイル (DIOSA_SENDER_STRM)
[table]
tam_table_name     = DIOSA_SENDER_STRM_0000000003
tam_table_file     = /home/diosa/tam/table/diosa_sender_strm_0000000003.tam
backup            = yes
share             = yes
sequence          = no
primary_index_duplicate = no
index_reserve_size = 0
data_reserve_size = 0
auto_extend       = yes
auto_extend_size  = 1
max_auto_extend_size = 5
record_size       = 168
```

```
checksum          = no
read_only         = no
dirty_read        = no
index_number      = 1
index_name1       = DIOSA_SENDER_STRM_00_IDX
index_entry_num1  = 10
keys1             = 0:48
auto_extend_key_num1 = 20
max_auto_extend_count1 = 10
init_key_num1     = 10
auto_release_index1 = no
init_record_num   = 10
txlog             = no
```

E. 4.3 timer.conf

```
[timer]
commit1_response_timer      = 250
commit2_response_timer      = 250
control_data_response_timer = 25
stc_response_timer2         = 60
txstart_response_timer      = 0
net_reconnect_interval_timer = 180
net_wait_response_timer     = 30
net_wait_data_timer         = 60
net_max_send_retry_num      = 7
net_rcv_count_timer         = 22
net_reconnect_interval_timer2 = 10
link_healthcheck_timer      = 10
link_healthcheck_retry_count = 10
net_ack_delay_timer         = 10
start_stop_watch_timer      = 30
lock_check_timer            = 30
memalloc_retry_timer        = 10
memalloc_retry_num          = 0
tamsave_tx_watch_time       = 0
tamsave_expired_action      = alarm
txlog_write_sync_response_timer = 30
txlog_command_response_timer = 20
```

E. 4.4 mon.conf

```
[log]
log_file_path      = /home/diosa/tam/log
log_file_size      = 5
log_file_swap_message = no

[trace]
trace_file_path      = /home/diosa/tam/log
trace_file_size      = 10
trace_file_swap_message = no

[activity]
activity_file_path      = /home/diosa/tam/log
activity_file_size      = 100
activity_api_access     = yes
activity_api_tat        = yes
activity_api_error      = yes
activity_table_space    = no
activity_table_fragment = no
activity_table_access   = yes
activity_table_tat      = yes
activity_table_error    = yes
activity_backup_size     = yes
activity_backup_throughput = yes
activity_backup_tat      = yes
activity_backup_error    = yes
activity_command_access  = no
activity_tx_num          = 1024
activity_command_tat     = no
activity_command_error   = yes
activity_transaction_access = no
activity_transaction_tat = no
activity_transaction_error = yes
activity_interval_timer  = 1
activity_file_swap_message = no
activity_txlog_access    = yes
activity_txlog_tat       = yes
```

activity_txlog_error	=yes
activity_txlog_write_io	=yes

E.5 Oracle データベース

E.5.1 TABLE 生成文

データベース管理機能

```
CREATE TABLE DIOSA_NCM_DBHC_01
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_02
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_03
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_04
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_05
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_06
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_07
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_08
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_09
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_10
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_11
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_12
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_13
```

```

(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_14
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_15
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_16
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_17
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_18
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_19
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_20
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_21
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_22
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_23
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_24
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_25
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_26
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

```



```
CREATE TABLE DIOSA_NCM_DBHC_27
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_28
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_29
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_30
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_31
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);

CREATE TABLE DIOSA_NCM_DBHC_32
(
    INSNAME    VARCHAR2(137),
    INSCND     NUMBER(1)
);
```

D I O S A / X T P V1.1

導入の手引

2019 年 6 月 9 版

日本電気株式会社

東京都港区芝五丁目 7 番 1 号

TEL (03) 3454-1111 (大代表)

©NEC Corporation 2011, 2015, 2019

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。