

D I O S A / X T P V2.1

コマンドリファレンス

#### 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、業務システムの構築を支援するD I O S A / X T P 製品群のコマンドリファレンスです。

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、OS、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。

2018 年 2 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 利用の手引き
- D I O S A / X T P 導入の手引き
- D I O S A / X T P メモリキャッシュ 利用の手引き
- D I O S A / X T P データストア 利用の手引き
- D I O S A / X T P データ変換・通信オプション 導入の手引き
- D I O S A / X T P データ変換・通信オプション 利用の手引き
- 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	コマンドの構成と記述形式	1
1.1	概要	1
1.2	コマンドの構成	1
1.3	コマンドの説明形式	2
第2章	コマンドリファレンス	4
2.1	アプリケーション実行制御	4
2.1.1	コマンド一覧	4
2.1.2	diaptrctrl (デーモン起動/停止コマンド)	7
2.1.3	diaptrcfedit (直接出力トレース情報ファイル編集コマンド)	8
2.1.4	diaptrcfenv (直接出力動作環境変更コマンド)	11
2.1.5	diaptrcinit (初期化コマンド)	13
2.1.6	diaptrcmedit (メモリ経由出力トレース情報ファイル編集コマンド)	14
2.1.7	diaptrcmenv (メモリ経由出力動作環境変更コマンド)	18
2.1.8	diaptrcmflush (フラッシュコマンド)	20
2.1.9	diaptrcterm (終了コマンド)	21
2.1.10	dibacexec (バッチ AP 制御実行コマンド)	22
2.1.11	dibcmctrl (デーモン起動停止コマンド)	25
2.1.12	dibcmref (閉塞状態参照コマンド)	26
2.1.13	dibcmsync (閉塞状態同期コマンド)	28
2.1.14	dibcmupd (閉塞状態変更コマンド)	29
2.1.15	dicddchg (コマンド配信定義動的変更コマンド)	31
2.1.16	dicddctrl (コマンド配信デーモン起動停止コマンド)	32
2.1.17	dicddhstctl (コマンド配信履歴状態変更コマンド)	33
2.1.18	dicddhstedit (コマンド配信履歴編集コマンド)	34
2.1.19	dicmdsend (コマンド配信コマンド)	36
2.1.20	dicocinit (初期化コマンド)	39
2.1.21	dicocmod (動作変更コマンド)	40
2.1.22	dicocref (稼動状況照会コマンド)	42
2.1.23	dicocrefsg (環境定義照会コマンド)	48
2.1.24	dicocterm (終了コマンド)	54
2.1.25	dicoctxblock (電文保留制御コマンド)	55
2.1.26	dicomcatget (カタログメッセージ標準出力コマンド)	57
2.1.27	didamctrl (死活監視デーモン起動/停止コマンド)	58
2.1.28	didbgerr (障害情報検索・編集コマンド)	59
2.1.29	didbgerrswap (障害情報強制スワップコマンド)	63
2.1.30	didbginit (デバッグトレース機能初期化コマンド)	64
2.1.31	didbglog (ログ情報検索・編集コマンド)	65
2.1.32	didbglogswap (ログ情報強制スワップコマンド)	69

2.1.33	didbgterm (デバッグトレース機能終了コマンド)	70
2.1.34	didlrchg (AP 動的置換)	71
2.1.35	didlrinit (AP 動的置換機能初期化コマンド)	72
2.1.36	didlrreflib (論理ライブラリ定義参照コマンド)	73
2.1.37	didlrreflm (LM 定義参照コマンド)	74
2.1.38	dietgctrl (経過時間監視デーモン起動停止コマンド)	75
2.1.39	dietgref (経過時間監視照会コマンド)	76
2.1.40	diirmadd (SG オブジェクト生成コマンド)	78
2.1.41	diirmdel (SG オブジェクト削除コマンド)	79
2.1.42	diirmref (SG オブジェクト参照コマンド)	80
2.1.43	diirmrep (SG オブジェクト更新コマンド)	83
2.1.44	dimdrctrl (SG 変更通知制御コマンド)	84
2.1.45	dimemchg (SG 変更コマンド)	85
2.1.46	dimemfree (メモリ一括解放コマンド)	86
2.1.47	dimemref (メモリ状態表示コマンド)	87
2.1.48	dimsgdctrl (メッセージ出力デーモン起動停止コマンド)	91
2.1.49	dimsgdisp (メッセージ出力コマンド)	92
2.1.50	dimsglimit (メッセージ出力抑止コマンド)	94
2.1.51	dimsglvmod (メッセージ出力レベル変更コマンド)	96
2.1.52	dimsglvref (メッセージ出力レベル照会コマンド)	97
2.1.53	dimsgmtn (原型メッセージファイルメンテナンスコマンド)	98
2.1.54	diopschg (稼動統計環境変更コマンド)	99
2.1.55	diopsctrl (稼動統計デーモン起動・停止コマンド)	100
2.1.56	diopsedit (稼動統計データ編集コマンド)	101
2.1.57	diopsflush (稼動統計フラッシュコマンド)	106
2.1.58	diopsfput (稼動統計転送コマンド)	107
2.1.59	diopsgather (稼動統計収集コマンド)	108
2.1.60	diopsmrg (稼動統計マージコマンド)	110
2.1.61	distart_com (DIOSA/XTP 起動コマンド)	111
2.1.62	distatref (起動/停止状態管理コマンド)	112
2.1.63	distop (DIOSA/XTP 停止コマンド)	114
2.1.64	ditmcactv (タイマ保留解除コマンド)	115
2.1.65	ditmcdel (タイマ情報削除コマンド)	116
2.1.66	ditmcdmn (タイマデーモン起動コマンド)	117
2.1.67	ditmcdmnstop (タイマデーモン停止コマンド)	118
2.1.68	ditmchold (タイマ保留コマンド)	119
2.1.69	ditmcref (タイマ情報照会コマンド)	120
2.1.70	ditmcset (タイマ登録コマンド)	123
2.1.71	ditmsgset (SG タイマ登録コマンド)	125
2.2	通信制御	126
2.2.1	コマンド一覧	126

2.2.2	digntchgdest (保証電文宛先変更コマンド).....	127
2.2.4	digntdbreset (電文保証 DB 初期化コマンド).....	128
2.2.5	digntdel (保証電文削除コマンド) .....	129
2.2.6	digntinit (電文保証初期化コマンド).....	130
2.2.7	digntmod (電文保証定義変更コマンド) .....	131
2.2.8	digntref (電文保証照会コマンド) .....	132
2.2.9	digntterm (電文保証終了コマンド) .....	138
2.2.10	dilspathctrl (論理システム間パス接続切断コマンド).....	139
2.2.11	dilspathref (論理システム間パス照会コマンド).....	141
2.2.12	dilspathrefstats (論理システム間パス統計情報照会コマンド).....	147
2.2.13	dincmbhchkctrlinit (DB ヘルスチェックデーモン起動コマンド) .....	152
2.2.14	dincmbhchkctrlterm (DB ヘルスチェックデーモン停止コマンド) .....	153
2.2.15	dincmdbinit (DB 監視機能初期化コマンド) .....	154
2.2.16	dincmdbstat (DB 状態照会コマンド) .....	155
2.2.17	dincmdbterm (DB 監視機能終了コマンド) .....	157
2.2.18	dincpathdmnctrl (パス制御機能デーモン起動停止コマンド).....	158
2.2.19	dincpathinit (パス制御機能初期化).....	159
2.2.20	dincpathterm (パス制御機能終了).....	160
2.2.21	dinodepathctrl (論理ノード間パス構成端末制御).....	161
2.2.22	dinodepathref (論理ノード間パス情報照会).....	163
2.2.23	dinodepathrefstats (論理ノード間パス統計情報照会).....	168
2.2.24	diotcref (都度接続管理機能 定義照会コマンド).....	173
2.2.25	diotcrefstats (都度接続管理機能 統計情報照会コマンド).....	176
2.3	メモリキャッシュ .....	179
2.3.1	コマンド一覧.....	179
2.3.2	diimblock (インメモリサーバ閉塞コマンド).....	180
2.3.3	diimcheck (インメモリサーバ所在情報整合性チェックコマンド).....	181
2.3.4	diimchg (メモリキャッシュ環境定義置換コマンド).....	182
2.3.5	diimchghash (ハッシュ値切替コマンド).....	183
2.3.6	diimctrl (インメモリサーバ起動停止コマンド).....	184
2.3.7	diiminit (メモリキャッシュ起動コマンド).....	187
2.3.8	diimmod (メモリキャッシュ動作変更コマンド).....	189
2.3.9	diimnotify (メモリキャッシュ通知コマンド).....	191
2.3.10	diimperfref (メモリキャッシュ性能情報照会コマンド).....	192
2.3.11	diimperfreset (メモリキャッシュ統計情報初期化コマンド).....	200
2.3.12	diimquegc (IMS キュー解放コマンド) .....	201
2.3.13	diimref (メモリキャッシュ情報照会コマンド).....	202
2.3.14	diimtblopenclose (表制御コマンド).....	218
2.3.15	diimterm (メモリキャッシュ停止コマンド).....	220
2.3.16	ditamswap (マスタ切替コマンド).....	221
2.3.17	ditamswitch (マスタ昇格コマンド).....	223

2.4 データストア基盤 .....	224
2.4.1 コマンド一覧.....	224
2.4.2 didltblock(ディレード無効化状態変更) .....	226
2.4.3 didltchg(ディレード定義反映) .....	227
2.4.4 didltcreate(ディレード定義生成) .....	229
2.4.5 didltctrl(デーモン起動停止) .....	232
2.4.6 didltdivchg(ディビジョン切り替え).....	234
2.4.7 didltinit(ディレード起動) .....	235
2.4.8 didltmodreset(ディレード動作変更リセット).....	236
2.4.9 didltpathref(ディレード転送パス状態照会).....	238
2.4.10 didltpooldel(ログデータ削除).....	241
2.4.11 didltpoolmod(プールファイル動作変更).....	242
2.4.12 didltpoolref(プールファイル状況照会).....	243
2.4.13 didltref(ディレード状態照会).....	250
2.4.14 didltrefdb(ディレード定義リスト出力).....	255
2.4.15 didltslmmv(スーパーストリーム所在変更).....	256
2.4.16 didltslmref(スーパーストリーム所在管理情報照会).....	257
2.4.17 didltterm(ディレード停止).....	260
2.4.18 didltupd(ディレード定義変更).....	261
2.4.19 didtlctrl(ログリーダデーモン起動停止).....	264
2.4.20 didtldivend(ログリーダ強制ディビジョン終了).....	265
2.4.21 didtlininit(ログリーダ開始).....	266
2.4.22 didtlmod(ログリーダ動作変更).....	267
2.4.23 didtlref(ログリーダ状況照会).....	268
2.4.24 didtlstart(ログリーダデータ処理開始).....	278
2.4.25 didtlstop(ログリーダデータ処理停止).....	279
2.4.26 didtlterm(ログリーダ終了).....	280
2.4.27 didtrctrl(レシーバデーモン起動停止).....	281
2.4.28 didtrdivend(レシーバ強制ディビジョン終了).....	282
2.4.29 didtrinit(レシーバ開始).....	283
2.4.30 didtrmod(レシーバ動作変更).....	284
2.4.31 didtrref(レシーバ状況照会).....	285
2.4.32 didtrstart(レシーバ転送開始).....	292
2.4.33 didtrstop(レシーバ転送停止).....	293
2.4.34 didtrterm(レシーバ終了).....	294
2.4.35 didtsctrl(センダデーモン起動停止).....	295
2.4.36 didtsdivend(センダ強制ディビジョン終了).....	296
2.4.37 didtsinit(センダ開始).....	297
2.4.38 didtsmod(センダ動作変更).....	298
2.4.39 didtsref(センダ状況照会).....	300
2.4.40 didtsstart(センダ転送開始).....	308

2.4.41	didtsstop(センダ転送停止).....	309
2.4.42	didtsterm(センダ終了).....	310
2.5	データ変換・通信オプション .....	311
2.5.1	コマンド一覧.....	311
2.5.2	dadacchg(環境定義動的変更コマンド) .....	312
2.5.3	dadacinit(起動コマンド) .....	313
2.5.4	dadacref(環境定義情報照会コマンド) .....	314
2.5.5	dadacterm(終了コマンド) .....	318
2.5.6	dadscblock(更新ログ出力抑止状態変更) .....	319
2.5.7	dadscinfochg(更新状況同期) .....	320
2.5.8	dadsclogref(更新ログ照会) .....	322
2.5.9	dadscref(更新ログ出力抑止照会) .....	326
2.5.10	daolmchkstatus(実行状況確認コマンド).....	328
2.5.11	daolminit(オンライン中メンテナンス初期化コマンド).....	329
2.5.12	daolmrefstatus(実行状況照会コマンド).....	330
2.5.13	daolmterm(オンライン中メンテナンス終了コマンド).....	332
2.5.14	daslutamload(TAM ロードコマンド) .....	333
2.5.15	daslutamsave(TAM セーブコマンド) .....	334
2.5.16	daslufltamload(ファイル入力 TAM ロードコマンド).....	335
2.5.17	daslufltamsave(ファイル出力 TAM セーブコマンド).....	337
2.5.18	datrlrefopdata(運用情報照会コマンド).....	338
2.5.19	datrlupdhash(ハッシュ値更新コマンド).....	340
2.5.20	datrlupdopdata(運用情報更新コマンド).....	342
第3章	TPP リファレンス .....	344
3.1	概要 .....	344
3.2	TPP 一覧.....	344
3.3	TPP リファレンス.....	345
3.3.1	dicoctpp(CO 制御 TPP) .....	345
3.3.2	digntdeltp (電文保証 受信情報削除 TPP) .....	346
3.3.3	digntfintpp (電文保証 応答電文受信 TPP) .....	347
3.3.4	digntrsdtp (電文保証 電文再送 TPP) .....	348
3.3.5	dincmctrltp (通信制御 TPP) .....	349
付録A	コマンド一覧.....	350
A.1	コマンド呼び出し可能ノード .....	350
A.1.1	アプリケーション実行制御 .....	350
A.1.2	通信制御 .....	352
A.1.3	メモリキャッシュ .....	353
A.1.4	データストア .....	354
A.1.5	データ変換・通信オプション .....	355

# 第1章 コマンドの構成と記述形式

## 1.1 概要

D I O S A / X T P の各種機能に対する指示を、コマンドとして提供する。

本章では、コマンドの構成と説明形式、および、利用上の一般的な規則について説明する。

## 1.2 コマンドの構成

### (1) コマンドの形式

コマンドは次の基本形式を持つ。

コマンド名 [オプション] . . . [パラメータ] . . .
-----------------------------------

### (2) コマンドの構成項目

#### (a) コマンド名

コマンド名は必ず“di”で始まる。

コマンド名とパラメータとの間は1文字以上の空白で区切る。

#### (b) オプション

オプションは、次の形式である。

オプション形式： -オプション記号 オプション値
--------------------------

“-”とオプション記号との間は空白で区切ってはいけない。

- ・オプション記号とオプション値の間は任意個の空白で区切る。

#### (c) パラメータ

パラメータは、次の形式である。

パラメータ形式： パラメータ値
-----------------

- ・名前、ID、数値、選択子等の即値は、パラメータ値として指定する。
- ・パラメータどうしは、任意個の空白で区切る。

## 1.3 コマンドの説明形式

各コマンドの詳細は「第2章コマンドリファレンス」で説明する。ここでは、各コマンド説明の記述形式と構文表記法について述べる。

### (1) 説明の記述項目

#### 名前

コマンド名、および、処理概要について説明する。

#### 書式

コマンドの構成、すなわち構成するパラメータとその記述順、繰り返し等について構文表記法を用いて説明する。

#### 説明

コマンドが提供する機能、一般的な動作を説明する。

#### オプション

書式で示したパラメータごとに、その意味を概説し、省略や繰り返し等のコマンド内での記述規則およびパラメータ値の記述規則について説明する。

#### カラムの説明

照会系コマンドで表示するカラムとその説明を記述する。

#### 戻り値

コマンドの戻り値を記述する。

#### 注意

コマンドを利用する際に注意すべき点がある場合に、その説明を記述する。

#### 関連

コマンドを利用する際に参照あるいは使用するコマンドがある場合に記述する。

## (2) コマンドの表記法

### (a) 構成要素

コマンド構文は次の構成要素からなる。

#### 構文表記文字

次の文字は構文の記述方法を指示するための文字で、構文の記述時にこれらの文字を指定するわけではない。文字の意味については次項の表記方法で説明する。

—例 [ ] { } | •

#### 空白

構文を読みやすくするために空白を用いることがある。必ずしも空白が指定できることを意味しない。空白が記述できる。

### (b) 表記方法

#### 省略の表現

[ ]でくくられた文字列は省略することができる。

—例 [XYZ]

#### 選択の表現

{ }または[ ]でくくられた複数の指定要素のうち、1つを選んで指定するための表現で、次の2つの書き方がある。[ ]でくくられた選択表現は、何も選ばず全体を省略できる。

- 指定要素を|で区切り1行に列挙する。主にパラメータ値の選択で用いる。  
論理和記号“|”の代わりに行分けで表現する場合もある。
- 指定要素を、行を分けて頭をそろえて列挙する。主にパラメータどうしの選択で用いる。

#### 既定値の表現

[ ]でくくられたなかに選択表現がある場合、下線付きの指定要素は省略時の既定値である。

—例 [LOCK={YES|NO}]

#### 繰り返しの表現

繰り返しは、“...”で表現する。直前の[ ]または{ }部を繰り返し指定することができる。

—例 KEY=ARG1 [, ARG2]...

## 第2章 コマンドリファレンス

### 2.1 アプリケーション実行制御

#### 2.1.1 コマンド一覧

##### (1) CO制御機能

dicocinit	CO 制御の初期化处理をおこなう。
dicocmod	CO 制御の動作環境を変更する。
dicocref	CO 制御 TPP の稼動状況を照会する。
dicocrefsg	環境定義を照会する。
dicocterm	CO 制御の終了処理をおこなう。
dicoctxblock	トランザクション閉塞制御をおこなう。

##### (2) バッチアプリケーション制御機能

dibacexec	バッチ AP 制御機能を通じて CO を実行する
-----------	--------------------------

##### (3) AP 動的置換機能

didlrchg	AP 動的置換機能を利用して呼び出している関数を置換する。
didlrinit	AP 動的置換機能を初期化する。
didlrreflib	APLIB 節 LLIB 項の定義を参照する。
didlrreflm	APLIB 節 LM 項の定義を参照する。

##### (4) タイマ制御機能

ditmcactv	保留しているタイマ情報を解除する。
ditmdel	タイマ情報を削除する。
ditmdmn	タイマデーモンを起動する
ditmdmnstop	タイマデーモンを停止する。
ditmchold	タイマ情報を保留する。
ditmcref	タイマ情報を照会する。
ditmcset	タイマ情報を登録する。
ditmcsgset	SG タイマ情報を登録する。

##### (5) 稼動統計機能

diopschg	稼動統計の環境変更を行う。
diopsctrl	稼動統計デーモンの起動・停止を行う。
diopsedit	稼動統計マージファイルを編集する。
diopsflush	稼動統計情報のファイルフラッシュを行う。
diopsgather	他ノードの稼動統計情報を収集する。

diopsfput	稼動情報を指定ディレクトリにコピーする。
diopsmrg	稼動情報をソート・マージする。

(6)      **メッセージ出力機能**

dimsgdctrl	メッセージ出力デーモンの起動、停止を行う。
dimsgdisp	指定したメッセージをメッセージログファイルへ出力する。
dimsglimit	指定したメッセージ ID の出力抑止の登録・解除、登録されている ID の一覧表示を行う。
dimsglvmod	メッセージの出力基準を変更します。
dimsglvref	出力レベル基準値の照会を行います。
dimsgmtn	原型メッセージファイルを作成、出力します。

(7)      **経過時間監視機能**

dietgctrl	経過時間を監視するデーモンを操作する。
dietgref	経過時間情報を照会する。

(8)      **閉塞管理機能**

dibcmctrl	閉塞管理デーモンを起動または停止する。
dibcmref	閉塞状態を照会する。
dibcmsync	閉塞状態の同期合せをする。
dibcmupd	閉塞状態を変更する。

(9)      **APP トレース**

diapptrcinit	APP トレース機能を開始する。
diapptrcterm	APP トレース機能を終了する。
diapptrcctrl	APP トレースデーモンの起動/停止を行う。
diapptrcfenv	直接出力の動作環境を動的に変更・参照する。
diapptrcfedit	直接出力トレース情報ファイル内の該当するトレース情報をテキスト形式に編集する。
diapptrcmenv	APP トレース機能の動作環境を動的に変更・参照する。
diapptrcmedit	メモリ経由トレース情報ファイル内の該当するトレース情報をテキスト形式に編集する。
diapptrcmflush	APP トレース機能の保存領域のトレース情報をファイルに出力する。

(10)     **マルチスレッド動的変更**

dimdrctrl	SG 変更通知の送信を行う。
-----------	----------------

(11)     **死活監視機能**

didamctrl	死活監視デーモンの起動/停止を行う。
-----------	--------------------

(12)     **メモリ管理機能**

dimemchg	メモリ管理機能の SG を変更する。
dimemfree	共有メモリを解放する。

dimemref          共有メモリの状態およびアボートダンプファイルの内容を編集出力する。

(13)          **起動／停止機能**

distart\_com      DIOSA/XTP を起動する。

distop            DIOSA/XTP を停止する。

distatref        DIOSA/XTP の起動状態を出力する。

(14)          **デバッグトレース機能**

didbginit        デバッグトレース機能を開始する。

didbgterm        デバッグトレース機能を終了する。

didbglog          デバッグトレース機能の共有メモリ上で管理しているログ情報、  
またはログ情報ファイル出力デーモンが出力したログ情報出力ファイルの内容を  
表示/ファイル出力する。

didbgerr          デバッグトレース機能の共有メモリ上で管理している障害情報、  
または障害情報ファイル出力デーモンが出力した障害情報出力ファイルの内容を  
表示/ファイル出力する。

didbglogswap     デバッグトレース機能の共有メモリ内のログ情報をファイルに出力する。

didbgerrswap     デバッグトレース機能の共有メモリ内の障害情報をファイルに出力する。

(15)          **環境定義機能**

diirmadd          SG オブジェクトを新規に生成する。

diirmdel          SG オブジェクトを削除する。

diirmref          SG オブジェクトに関する情報を参照する。

diirmrep          SG オブジェクトを更新する。

(16)          **コマンド配信機能**

dicddchg          コマンド配信の環境定義を動的に置換する。

dicddctrl        コマンド配信デーモンの起動または停止を行う。

dicddhstctl      コマンド配信履歴状態を変更する。

dicddhstedit     コマンド配信履歴ファイルを編集する。

dicmdsend        指定したコマンドを配信先のノードで実行する。

## 2.1.2 diaptrcctrl (デーモン起動/停止コマンド)

### 名前

diaptrcctrl - APP トレースデーモンの起動/停止を行う

### 書式

```
diaptrcctrl [-b]
diaptrcctrl -e
diaptrcctrl -h
```

### 説明

APP トレースデーモンを起動/停止する。事前に **diaptrcinit** コマンドが実行されていない場合、本コマンドは異常終了する。

### オプション

**-b**

APP トレースデーモンを起動する。(省略時の既定値)

**-e**

APP トレースデーモンを停止する。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 注意

APP トレースデーモンは APP トレース機能初期化コマンド(diaptrcinit)、APP トレース機能終了コマンド(diaptrcterm)に連動して起動/停止するため、通常は本コマンドを実行する必要はない。

何らかの障害が原因で、環境変数 DIOSA\_APPTRCMIDNUM で指定した数の APP トレースデーモンが起動していない場合は、本コマンドで起動することができる。個別での起動停止は行わない。

### 関連

diaptrcinit, diaptrcterm

## 2.1.3 diaptrcfedit (直接出力トレース情報ファイル編集コマンド)

### 名前

diaptrcfedit - トレース情報ファイル(直接出力)内の該当するトレース情報をテキスト形式に編集する

### 書式

```
diaptrcfedit -f FILENAME [-B STARTTIME] [-E ENDTIME] [-v レベル値] [-t データ部のテキスト] [-s  
関数名] [-c コメント部のテキスト] [-o 出力ファイル名] [-d]  
diaptrcfedit -h
```

### 説明

トレース情報ファイル(直接出力)内を指定された条件で検索し、該当するトレース情報をテキスト形式に編集して出力する。指定できる検索条件は、時間帯、レベル、ユーザデータ部の値、関数名、コメント部の値である。なお、複数の条件を設定した場合は、全ての条件にあてはまるトレース情報が編集の対象となる。

### オプション

#### -f

指定したトレース情報ファイル(直接出力)内のトレース情報が検索対象となる。指定可能なトレース情報ファイルは、環境変数 `DIOSA_APPTRCFPATH` か `diaptrcfenv` コマンドの `-P` オプションで指定したパスに出力された、拡張子が `.log` のファイルである。

#### -B

指定した時刻以降のトレース情報を編集対象とする。時刻指定の形式は `hhmm[YYMMDD]` とし、`hh` は時間、`mm` は分、`YY` は年、`MM` は月、`DD` は日で、すべて 2 桁で指定する。`YYMMDD` は省略可能で、省略した場合は本コマンドが実行された日付が採用される。

#### -E

指定した時刻以前のトレース情報を編集対象とする。本オプションの時刻指定の形式は `-B` オプションと同じ。

#### -v

指定した値以下のレベルのトレース情報を編集対象とする。指定可能な値は 1~9 の整数である。

#### -t

指定した値をユーザデータ部に含むトレース情報を編集対象とする。指定する値は `0x` で始まる任意の桁数の 16 進数または `"`(ダブルクォート)で囲まれた任意の長さの ASCII 文字列からなる。指定可能な値の最大長は 64 バイトである。

#### -s

指定した関数内で出力されたトレース情報を編集対象とする。`*`を付加して指定することで、前方一致検索が可能となる。例えば、`"ABC"` という文字列を関数名として指定した場合、`"ABCD"` や `"ABC123"` といった関数内で出力されたトレース情報が編集対象となる。

#### -c

指定した値をコメント部に含むトレース情報を編集対象とする。指定可能な値は最大 49 バイトの ASCII 文字列である(50 バイト目以降は無視される)。指定する値に空白や特殊文字を含ませたい場合は `"`(ダブルクォート)で囲む必要がある。

#### -o

編集結果の出力先となるファイル名を指定する。同じ名前のファイルが既に存在している場合は上書きされる。本オプションが省略された場合、編集結果は標準出力に出力される。

-d

トレース情報毎の情報(時刻、行番号等)を詳細に出力する。本オプションが省略された場合は最小限の情報が出力される。

-h

本コマンドの使用方法を表示する。

項目の説明

以下に-d オプションを指定した場合に、トレース情報編集結果として出力される項目の見出しと意味を挙げる。

RCDTYPE	トレース情報種別("LOG"で固定)
LVL	出力レベル(diosapptrcf 関数の第 5 引数の値)
SRCNAME	ソースファイル名と行番号
DATE	日付(YY-MM-DD 形式)
TIME	時刻(hh:mm:ss.sss 形式)
PROCESS	プロセス名、プロセス ID、スレッド ID
SECTION	関数名
ERRCODE	エラーコード(diosapptrcf 関数の第 4 引数の値)
COMMENT	コメント(diosapptrcf 関数の第 1 引数で指定された文字列)
ADDR	データの格納位置(diosapptrcf 関数の第 2 引数で指定されたデータ)
LENGTH	トレース情報格納時に指定されたデータの長さ(diosapptrcf 関数の第 3 引数の値)

-d オプションを指定しない場合は、TIME、DATE、SRCNAME、COMMENT、ERRCODE、LENGTH の内容がトレース情報毎の情報として出力される。ただし DATE や SRCNAME は、1 つ前のトレース情報と同じ値である場合は出力されない。ERRCODE は 0 の場合は出力されない。

戻り値

0	正常終了
1	異常終了

実行例

トレース情報出力ファイル指定(diapptrcfedit -f)

SYSTEM : HOSTNAME(ssssssssssssssssssss):0xhhhhhhh(XXXXXXX) ①							
[YY-MM-DD] hh:mm:ss.uuu [oooooooooooo]:kkkkk mmmmmmmmmmmmm[(rrrrrrrrr)][ (0xeeeeeeee)]							
②	③	④	⑤	⑥	⑦	⑧	
0xgggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh  cccccccccccccccc							
⑨	⑩		⑪				
:	:		:				
0xgggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh  cccccccccccccccc							
0xgggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh  cccccccccccccccc							

- ①SYSTEM : システム情報を以下の形式で表示する  
“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)”
- ②区間が開始/終了した日付 : 左から順に西暦年の下 2 桁-月-日。直前のログ情報と日付が同じ場合表示しない
- ③区間が開始/終了した時刻 : 左から順に時:分:秒.ミリ秒(3 桁)
- ④ソースファイル名 : 直前に表示したトレース情報と同じ場合表示しない
- ⑤行番号
- ⑥コメント
- ⑦エラーコード : 値が 0 の場合表示しない
- ⑧出力リソースレコード長 : 値が 0 の場合表示しない

- ⑨表示開始アドレス  
⑩バイナリデータの HEX ダンプ  
⑪バイナリデータのキャラクタダンプ

トレース情報出力ファイル詳細指定(diapptrcfedit -f -d)

```
SYSTEM : SSSSS(ssssssssssssssssssssss) : 0xhhhhhhh(XXXXXXXXXX) ①
```

  

```
RCDTYPE: LOG   LVL: N   SRCNAME: oooooooooooooooooo:k kkkk  
②           ③         ④
```

  

```
DATE    : YY-MM-DD      TIME: hh:mm:dd.uuu  
⑤              ⑥
```

  

```
PROCESS: [ppppp:t tttt] qqqqqqqqqq  
⑦
```

  

```
SECTION: xxxxxxxx ERRCODE: rrrrrrrr  
⑧                ⑨
```

  

```
COMMENT: ~~~~~  
⑩
```

  

```
ADDR     : 0xhhhhhhh LENGTH: dddd(eeeeeee)  
⑪                ⑫
```

  

```
0x0000000 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccc  
⑬          ⑭                                ⑮
```

  

```
0x00000010 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccc  
0x00000020 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccc  
⋮          ⋮                                ⋮
```

- |                   |  |
|-------------------|--|
| ①SYSTEM           | ：システム情報を以下の形式で表示する。<br>“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)” |
| ②RCDTYPE          | ：情報の種類 LOG   |
| ③LVL              | ：トレース情報出力のレベル  |
| ④SRCNAME          | ：ソースファイル名を“ソースファイル名:行番号”の形式で表示する                                   |
| ⑤DATE             | ：情報収集を行った日付。左から順に西暦年の下2桁-月-日                                       |
| ⑥TIME             | ：情報収集を行った時刻。左から順に時-分-秒.ミリ秒(3桁)                                     |
| ⑦PROCESS          | ：プロセスの情報を”[プロセス ID:スレッド ID] プロセス名” の形式で表示する                        |
| ⑧SECTION          | ：関数名   |
| ⑨ERRCODE          | ：エラーコード  |
| ⑩COMMENT          | ：コメント  |
| ⑪ADDR             | ：出力リソースの先頭アドレス   |
| ⑫LENGTH           | ：出力リソースのレコード長を”10 進数(16 進数)”の形式で表示する                               |
| ⑬表示オフセット値         |  |
| ⑭バイナリデータの HEX ダンプ |  |
| ⑮バイナリデータのキャラクタダンプ |  |

## 2.1.4 diapptrcfenv（直接出力動作環境変更コマンド）

### 名前

diapptrcfenv - 直接出力の動作環境を動的に変更・参照する

### 書式

```
diapptrcfenv [-v level ] [-f { ON | OFF }] [-u { ON | OFF }] [-P 絶対パス ]
diapptrcfenv [-i]
diapptrcfenv -h
```

### 説明

利用者から出力されるトレース情報ファイル(直接出力)への出力条件を動的に変更・参照する。本コマンドで設定した値は、直後の API 実行から適用される。事前に **diapptrcinit** コマンドが実行されていない場合、本コマンドは異常終了する。

### オプション

**-v**

**level** 以下の出力レベルのトレース情報をファイルに出力する。**level** には 0 から 9 の整数値を指定する。**level** に 0 を指定した場合、トレース情報のファイル出力はされなくなる。

**-f**

トレース情報ファイル(直接出力)への出力時に強制的にフラッシュするかどうかを指定する。

**-u**

ファイルに出力するトレース情報にユーザデータを含めるかどうかを指定する。ユーザデータとは `diosaapptrcf()` 関数の第 2 引数で指定されるデータを指す。

**-P**

トレース情報ファイル(直接出力)の出力先を指定の絶対パスに変更する。

**-i**

現在の設定情報を表示する。(省略時の既定値)

**-h**

本コマンドの使用方法を表示する。

### カラムの説明

**-i** オプション指定時に表示される項目の見出しと意味を挙げる。

SYSTEM	ホスト名、OS 名とバージョン、APP トレース機能で使用する IPC キー
LOGLVL	トレース情報の出力レベル
FLUSHFLG	トレース情報ファイル(直接出力)出力時の強制フラッシュの ON/OFF
OUTUSERDATA	ユーザデータ出力の ON/OFF
LPATH	トレース情報ファイル(直接出力)の出力先絶対パス

### 戻り値

0	正常終了
1	異常終了

### 関連

diapptrcinit

## 実行例

トレース情報ファイル(直接出力)設定値 (diapptrcfenv -i)

```
SYSTEM : SSSSS(ssssssssssssssssssssss) : Oxhhhhhhh(XXXXXXXXXX) ①  
LOGVLV : v OUTUSERDATA: uuu FLUSHFLG: fff  
②          ③          ④  
LOGPATH: PPPPPPPPPPPPPPPPPPP  
⑤
```

- ①SYSTEM : システム情報を以下の形式で表示する。  
“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)”
- ②LOGLVL : ログ情報出力のレベル
- ③OUTUSERDATA : ユーザデータ出力モード  
ユーザデータを出力する : ON  
ユーザデータを出力しない : OFF
- ④FLUSHFLG : 強制書き込みモード  
強制書き込みする : ON  
強制書き込みしない : OFF
- ⑤LOGPATH : トレース情報直接出力ファイル作成先の絶対パス

## 2.1.5 diaptrcinit（初期化コマンド）

### 名前

diaptrcinit - APP トレース機能を開始する

### 書式

diaptrcinit  
diaptrcinit -h

### 説明

現在設定されている環境変数にもとづいてトレース情報を格納する保存領域(共有メモリ)を作成し、APP トレースデーモンプロセスを起動する。以降、ユーザアプリケーションはトレース情報を出力できるようになる。

### オプション

-h  
使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 注意

環境変数 DIOSA\_APPTRCMIDNUM で指定した数(1~3)の APP トレースデーモンプロセスを起動するが、1 つでのデーモン起動時できなかった場合は、コマンドは異常終了する。  
異常となった APP トレースデーモンプロセスはトレース情報出力デーモン起動/停止コマンドで再起動することが可能（個別起動はしない）

### 関連

diaptrctrl, diaptrterm

## 2.1.6 diaptrcmedit (メモリ経由出力トレース情報ファイル編集コマンド)

### 名前

diaptrcmedit - トレース情報ファイル(メモリ経由出力)内の該当するトレース情報をテキスト形式に編集する

### 書式

```
diaptrcmedit -f FILENAME [-B STARTTIME] [-E ENDTIME] [-p プロセス ID | -n プロセス名] [-v レベル値] [-t データ部のテキスト] [-s 関数名] [-c コメント部のテキスト] [-o 出力ファイル名] [-d]
diaptrcmedit -h
```

### 説明

トレース情報ファイル(メモリ経由出力)内を指定された条件で検索し、該当するトレース情報をテキスト形式に編集して出力する。指定できる検索条件は、時間帯、プロセス ID またはプロセス名、レベル、ユーザデータ部の値、関数名、コメント部の値である。なお、複数の条件を設定した場合は、全ての条件にあてはまるトレース情報が編集の対象となる。

### オプション

#### -f

指定したトレース情報ファイル(メモリ経由出力)内のトレース情報が検索対象となる。指定可能なトレース情報ファイルは、環境変数 `DIOSA_APPTRCMPATH` か `diaptrcmenv` コマンドの `-P` オプションで指定したパスに出力された、拡張子が `.trc` のファイルである。

指定されたトレース情報ファイル名は、複数ファイル識別情報として編集対象とする。

#### -B

指定した時刻以降のトレース情報を編集対象とする。時刻指定の形式は `hhmm[YYMMDD]` とし、`hh` は時間、`mm` は分、`YY` は年、`MM` は月、`DD` は日で、すべて 2 桁で指定する。`YYMMDD` は省略可能で、省略した場合は本コマンドが実行された日付が採用される。

#### -E

指定した時刻以前のトレース情報を編集対象とする。本オプションの時刻指定の形式は `-B` オプションと同じ。

#### -p

指定したプロセス ID のプロセスが出力したトレース情報を編集対象とする。本オプションや `-n` オプションを指定しない場合は全プロセスが出力したトレース情報を編集対象とする。

#### -n

指定したプロセス名のプロセスが出力したトレース情報を編集対象とする。本オプションや `-p` オプションを指定しない場合は全プロセスが出力したトレース情報を編集対象とする。

#### -v

指定した値以下のレベルのトレース情報を編集対象とする。指定可能な値は 1~9 の整数である。

#### -t

指定した値をユーザデータ部に含むトレース情報を編集対象とする。指定する値は `0x` で始まる任意の桁数の 16 進数または `"`(ダブルクォート)で囲まれた任意の長さの ASCII 文字列からなる。指定可能な値の最大長は 64 バイトである。

#### -s

指定した関数内で出力されたトレース情報を編集対象とする。\*を付加して指定することで、前方一致検索が可能となる。例えば、“ABC\*”という文字列を関数名として指定した場合、“ABCD”や“ABC123”といった関数内で出力されたトレース情報が編集対象となる。

-c

指定した値をコメント部に含むトレース情報を編集対象とする。指定可能な値は最大 49 バイトの ASCII 文字列である (50 バイト目以降は無視される)。指定する値に空白や特殊文字を含ませたい場合は”(ダブルクォート)で囲む必要がある。

-o

編集結果の出力先となるファイル名を指定する。同じ名前のファイルが既に存在している場合は上書きされる。本オプションが省略された場合、編集結果は標準出力に出力される。

-d

トレース情報毎の情報 (時刻、行番号等) を詳細に出力する。本オプションが省略された場合は最小限の情報が出力される。

-h

本コマンドの使用方法を表示する。

項目の説明

以下に-d オプションを指定した場合に、トレース情報編集結果として出力される項目の見出しと意味を挙げる。

RCDTYPE	トレース情報種別 (“LOG”で固定)
LVL	出力レベル (diosapptrcm 関数の第 5 引数の値)
SRCNAME	ソースファイル名と行番号
DATE	日付 (YY-MM-DD 形式)
TIME	時刻 (hh:mm:ss. sss 形式)
PROCESS	プロセス名、プロセス ID、スレッド ID
SECTION	関数名
ERRCODE	エラーコード (diosapptrcm 関数の第 4 引数の値)
COMMENT	コメント (diosapptrcm 関数の第 1 引数で指定された文字列)
ADDR	データの格納位置 (diosapptrcm 関数の第 2 引数で指定されたデータ)
LENGTH	トレース情報格納時に指定されたデータの長さ (diosapptrcm 関数の第 3 引数の値)

-d オプションを指定しない場合は、PROCESS、TIME、DATE、SRCNAME、COMMENT、ERRCODE、LENGTH の内容がトレース情報毎の情報として出力される。ただし DATE や SRCNAME は、1 つ前のトレース情報と同じ値である場合は出力されない。ERRCODE は 0 の場合は出力されない。

戻り値

0	正常終了
1	異常終了

注意

まだファイルに出力されていない (保存領域に存在している) トレース情報を編集対象にしたい場合は、**diapptrcmflush** コマンドを実行後に本コマンドを実行すること。  
プロセス名は最大 14 文字が有効となる。14 文字より長い名前のプロセスが出力したトレース情報を編集対象とする場合は、プロセス名の先頭 14 文字までを指定すること。

関連

diaptrcmflush

## 実行例

## トレース情報出力ファイル指定(diapptrcmedit -f)

```
FILENAME: fffffff ①  
SYSTEM : HOSTNAME(sssssssssssssssssss)0xhhhhhhh(XXXXXXXXX) ②  
  
PAGE : tttt(ERR INFORMATION) ③  
aaaaaaaaaaaaaaa(ppppp:ss) [tt-tt-tt]uu:uu:uu.uuu  
④                ⑤                ⑥                ⑦  
[ooooooooooooooo]:kkkkk mmmmmmmmmmmmmmmmmmmmmmmmmmm[(rrrrrrrrr)][(eeeeee)]  
⑧                ⑨                ⑩                                ⑪                                ⑫  
Oxgggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh |ccccccccccccc  
⑬                ⑭                                ⑮  
:  
:  
:  
Oxggggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh |ccccccccccccc  
Oxggggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh |ccccccccccccc
```

- |                     |  |
|---------------------|--|
| ①FILENAME           | : -f で指定したトレース情報ファイル名  |
| ②SYSTEM             | : システム情報を以下の形式で表示する<br>“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)” |
| ③PAGE               | : トレース情報保存領域のページ番号   |
| ④プロセス名              |  |
| ⑤プロセス ID とスレッド ID : | (プロセス ID:スレッド ID)  |
| ⑥区間が開始/終了した日付 :     | 左から順に西暦年の下 2 桁-月-日。直前のログ情報と日付が同じ場合表示しない                            |
| ⑦区間が開始/終了した時刻 :     | 左から順に時:分:秒. ミリ秒(3 桁)   |
| ⑧ソースファイル名           | : 直前に表示したトレース情報と同じ場合表示しない  |
| ⑨行番号                |  |
| ⑩コメント               |  |
| ⑪エラーコード             | : 値が 0 の場合表示しない  |
| ⑫出力リソースレコード長        | : 値が 0 の場合表示しない  |
| ⑬表示開始アドレス           |  |
| ⑭バイナリデータの HEX ダンプ   |  |
| ⑮バイナリデータのキャラクタダンプ   |  |

### トレース情報出力ファイル詳細指定(diapptrcmedit -f -d)

[illegible]

- ①FILENAME : -f で指定したトレース情報ファイル名
- ②SYSTEM : システム情報を以下の形式で表示する  
“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)”
- ③PAGE : トレース情報保存領域のページ番号
- ④RCDTYPE : 情報の種類 LOG
- ⑤LVL : トレース情報出力のレベル
- ⑥SRCNAME : ソースファイル名を“ソースファイル名:行番号”の形式で表示する
- ⑦DATE : 情報収集を行った日付。左から順に西暦年の下2桁-月-日
- ⑧TIME : 情報収集を行った時刻。左から順に時-分-秒、ミリ秒(3桁)
- ⑨PROCESS : プロセスの情報を”[プロセス ID:スレッド ID] プロセス名”の形式で表示する
- ⑩SECTION : 関数名
- ⑪ERRCODE : エラーコード
- ⑫COMMENT : コメント
- ⑬ADDR : 出力リソースの先頭アドレス
- ⑭LENGTH : 出力リソースのレコード長を”10進数(16進数)”の形式で表示する
- ⑮表示オフセット値
- ⑯バイナリデータの HEX ダンプ
- ⑰バイナリデータのキャラクタダンプ

## 2.1.7 diaptrcmenv (メモリ経由出力動作環境変更コマンド)

### 名前

diaptrcmenv - メモリ経由出力の動作環境を動的に変更・参照する

### 書式

```
diaptrcmenv [-v level ] [-u { ON | OFF }] [-P 絶対パス ] [-a trcid ]
diaptrcmenv [-i]
diaptrcmenv -h
```

### 説明

利用者から出力されるトレース情報の保存領域への格納条件を動的に変更・参照する。本コマンドで設定した値は、直後の API 実行から適用される。事前に **diaptrcinit** コマンドが実行されていない場合、本コマンドは異常終了する。

### オプション

**-v**

**level** 以下の出力レベルのトレース情報を保存領域に格納する。**level** には 0 から 9 の整数値を指定する。**level** に 0 を指定した場合、トレース情報は保存領域に格納されなくなる。

**-u**

保存領域に格納するトレース情報にユーザデータを含めるどうかを指定する。ユーザデータとは `diosaaptrcm()` 関数の第 2 引数で指定されるデータを指す。

**-P**

トレース情報ファイル(メモリ経由出力)の出力先を指定の絶対パスに変更する。

**-a**

変更対象のトレース識別 ID を指定する。省略時は、全トレース域別 ID を変更対象とする。指定可能な値は、1～環境変数 `DIOSA_APPTRCMIDNUM` による指定値の範囲（最大 3）である。

**-i**

現在の設定情報を表示する。(省略時の既定値)

**-h**

本コマンドの使用方法を表示する。

### カラムの説明

**-i** オプション指定時に表示される項目の見出しと意味を挙げる。

SYSTEM	ホスト名、OS 名とバージョン、APP トレース機能で使用する IPC キー
LOGLVL	トレース情報の出力レベル
OUTUSERDATA	ユーザデータ出力の ON/OFF
LPATH	トレース情報ファイル(メモリ経由出力)の出力先絶対パス

### 戻り値

0	正常終了
1	異常終了

### 注意

**-P** オプションで指定した出力先が有効になるのは、次のトレース情報ファイル(メモリ経由出力)のローテーション時以降となる。

関連

diapptrcinit

## 実行例

トレース情報ファイル(メモリ経由出力)設定値 (diapptrcmenv -i)

SYSTEM : SSSSS(ssssssssssssssssssssssss):0xhhhhhhh(XXXXXXXXXX) ①						
ID	PID	STATUS	TRCLVL	OUTUSERDATA	TRCPATH	
②	③	④	⑤	⑥	⑦	

- |              |   |
|--------------|---|
| ①SYSTEM      | : システム情報を以下の形式で表示する。<br>“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)” |
| ②ID          | : トレース識別 ID   |
| ③PID         | : APP トレースデーモンプロセス ID   |
| ④STATUS      | : APP トレースデーモンプロセスの状態<br>起動している : RUNNING<br>停止している : STOP          |
| ⑤TRCLVL      | : トレース情報出力レベル   |
| ⑥OUTUSERDATA | : ユーザデータ出力モード<br>ユーザデータを出力する : ON<br>ユーザデータを出力しない : OFF             |
| ⑦TRCPATH     | : トレース情報メモリ経由出力ファイル作成先の絶対パス   |

## 2.1.8 diapptrcmflush (フラッシュコマンド)

### 名前

diapptrcmflush - 保存領域のトレース情報をファイルに出力する

### 書式

```
diapptrcmflush [-f] [-a trcid]
```

```
diapptrcmflush -h
```

### 説明

保存領域に格納されているトレース情報を、現在の出力先となっているトレース情報ファイル(メモリ経由出力)に出力する。

事前に **diapptrcinit** コマンドが実行されていない場合や、APP トレースデーモンが不在の場合、本コマンドは異常終了する。

### オプション

**-f**

トレース情報をトレース情報ファイル(メモリ経由出力)へ出力した後、トレース情報ファイル(メモリ経由出力)をローテーションする。

**-a**

ファイル出力対象のトレース識別 ID を指定する。

指定可能な値は、1～環境変数 DIOSA\_APPTRCMIDNUM の指定値の範囲（最大 3）である。

省略時は、全トレース識別 ID をファイル出力対象とする。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

0	正常終了
---	------

1	異常終了
---	------

### 関連

diapptrcinit

## 2.1.9 diaptrcterm (終了コマンド)

### 名前

diaptrcterm - APP トレース機能を終了する

### 書式

diaptrcterm  
diaptrcterm -h

### 説明

APP トレースデーモンが起動中の場合は停止する。この時、保存領域に格納されているトレース情報はトレース情報ファイル(メモリ経由出力)に出力される。その後、APP トレース機能の保存領域を解放する。

### オプション

-h

本コマンドの使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 関連

diaptrcinit, diaptrcctrl

## 2.1.10 dibacexec (バッチ AP 制御実行コマンド)

### 名前

dibacexec - バッチ AP 制御機能を通じて CO を実行する

### 書式

```
dibacexec -f CO_NAME
        [-m MAPID]
        [-s START_EXIT] [-n NORMAL_EXIT]
        [-a ABORT_EXIT] [-w]
        [-e ELAPSED_TIME_LIMIT [-r ELAPSED_TIME_RESET_MAXCOUNT] [-k]]
        [-c CPU_TIME_LIMIT] [-o REPORT_FILE]
        [-p PARAMETER_FILE]
        [-b ROLLBACK_RETRY_MAXCOUNT]
        [-- USER_PARAMETER]

dibacexec -f CO_NAME -d
        [-s START_EXIT] [-n NORMAL_EXIT]
        [-a ABORT_EXIT] [-w]
        [-e ELAPSED_TIME_LIMIT [-r ELAPSED_TIME_RESET_MAXCOUNT] [-k]]
        [-c CPU_TIME_LIMIT] [-o REPORT_FILE]
        [-p PARAMETER_FILE]
        [-b ROLLBACK_RETRY_MAXCOUNT]
        [-- USER_PARAMETER]

dibacexec -h
```

### 説明

**dibacexec** は、バッチアプリケーション制御機能を通じて CO を実行する。

### オプション

#### **-f CO\_NAME**

CO の関数名を指定する。(1～30 文字)

#### **-m MAP\_ID**

DB として TAM を使用し、バッチ AP 制御が自動的に行うコネクト、ディスコネクトが必要な場合に、使用する TAM の MAPID を指定する。MAP\_ID を指定し、COMMT\_EXIT を省略している場合は、コミット、ロールバックの制御もバッチ AP 制御が行う。

#### **-d**

DB として Oracle を使用し、バッチ AP 制御が自動的に行う DB 制御(コネクト、ディスコネクト、コミット、ロールバック)が必要な場合に指定する。

#### **-s START\_EXIT**

初期化出口の関数名を指定する。(1～30 文字)  
省略時は、初期化出口が呼び出されない。

#### **-n NORMAL\_EXIT**

正常終了出口の関数名を指定する。(1～30 文字)  
省略時は、正常終了出口が呼び出されない。

**-a ABORT\_EXIT**

アボート出口の関数名を指定する。(1～30 文字)  
省略時は、アボート出口が呼び出されない。

**-w**

稼動統計情報の採取を行う場合に指定する。  
省略時は、稼動統計情報の採取を行わない。

**-e ELAPSED\_TIME\_LIMIT**

経過時間の制限値を秒単位で指定する。(0～999999)  
制限値を超えた時には、警告メッセージの出力またはプロセスの強制終了が行われる。  
省略時、または 0 を指定した時は経過時間監視を行わない。

**-r ELAPSED\_TIME\_RESET\_MAXCOUNT**

CO／利用者出口からの経過時間リセット要求の最大回数を指定する。(0～999999)  
CO／利用者出口からの経過時間リセット要求回数がこの値を超えた時には、経過時間が超過した時と同様に処置される(警告メッセージの出力またはプロセスの強制終了)。  
ELAPSED\_TIME\_LIMIT が指定していない場合は、指定できない。  
省略時、または 0 を指定した時は無制限となる。

**-k**

経過時間または経過時間リセット回数が超過した時に、プロセスの強制終了とする場合に指定する。  
ELAPSED\_TIME\_LIMIT が指定していない場合は、指定できない。  
省略時は、経過時間が超過した場合には警告メッセージの出力のみ行われる。

**-c CPU\_TIME\_LIMIT**

CPU 時間の制限値を秒単位で指定する。(0～999999)  
**CPUTIME\_LIMIT** 制限値を超えた時には、プロセスの強制終了が行われる。  
省略時、または 0 を指定した時は無制限となる。

**-o REPORT\_FILE**

実行レポート出力先を、絶対パス名または相対パス名で指定する。(1～255 文字)  
REPORT\_FILE には、下記の出力先を指定が可能である。  
既存ファイル指定時は再作成される。  
標準出力／標準エラー出力指定時は利用者メッセージの後に出力される。

記述	出力先
ファイル名(絶対パス名または相対パス名)	指定ファイル
stdout	標準出力
stderr	標準エラー出力

省略時は、実行レポートが出力されない。

## **-p** **PARAMETER\_FILE**

バッチ AP 制御パラメータファイルの名前を、絶対パス名または相対パス名で指定する。(1～255 文字)

PARAMETER\_FILE は dibacexec のコマンドパラメータ記述を減らす目的で利用できる。

パラメータファイルに複数の作業(ジョブ)で共通に使用するパラメータを予め記述しておく、その内容が起動パラメータに追加される。

省略時は、環境変数で指定されているパラメータファイルを使用し、環境変数も省略されていた場合、パラメータファイルは使用されない。

以下にバッチ AP 制御パラメータファイルの記述例を記す。

`-s startexit1 -n normalendexit1 -a abortendexit1`

## **-b** **ROLLBACK\_RETRY\_MAXCOUNT**

ロールバックリトライを行う最大回数を指定する。(0～9999)

最大回数を超えるリトライが発生した時には、異常終了処理が行われる。

省略時は、環境変数で指定されているロールバックリトライ最大回数を使用し、環境変数も省略されていた場合は、ロールバックリトライを行わない。また、0 を指定した時もロールバックリトライを行わない。

## **-** **USER\_PARAMETER**

利用者側で任意に使用するパラメータ群を指定する。(1～256 文字)

必ず、起動パラメータの末尾に記述すること。

diosauca 構造体の\*BatchApInfo 項目に、このパラメータ群のアドレスが設定される。

省略時は、\*BatchApInfo 項目に NULL が設定される。

## **-h**

使用方法を表示する。

## 戻り値

**dibacexec** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
0 以外	異常終了

## 注意

起動パラメータ、バッチ AP 制御パラメータファイル、環境変数で同一パラメータを重複して指定した場合の優先順位は、以下のとおりとする。

起動パラメータ > バッチ AP 制御パラメータファイル > 環境変数  
(優先高 > 優先低)

同一優先順位内でパラメータが重複して指定された場合は、先に指定した値を有効とする。パラメータの重複エラーとはしない。

DB を使わない場合は、-C、-R、-d、-m の指定を省略する。

バッチ AP 制御パラメータファイル中の“-p”パラメータはエラーとする。

## 2.1.11 dibcmctrl (デーモン起動停止コマンド)

### 名前

dibcmctrl - 閉塞管理デーモンを起動または停止する

### 書式

```
dibcmctrl {-b | -e}
```

```
dibcmctrl -h
```

### 説明

閉塞管理デーモンを起動または停止する。

### オプション

**-b**

閉塞管理デーモンを起動する。

**-e**

閉塞管理デーモンを停止する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                      処理が正常終了した。

1                      処理が異常終了した。

## 2.1.12 dibcmref (閉塞状態参照コマンド)

### 名前

dibcmref - 閉塞状態を照会する

### 書式

```
dibcmref {-N | [-l SysName] [-n NodeName] }  
dibcmref {-C | [-c CName] }  
dibcmref -h
```

### 説明

論理システム、ノード、C0 名の閉塞状態を参照する。

### オプション

**-N**

論理システム、ノードの閉塞状態を参照する。

**-l SysName**

閉塞状態を参照する論理システム名を指定する。

**-n NodeName**

閉塞状態を参照するノード名を指定する。

**-C**

閉塞中の C0 名を照会する。

**-c CName**

閉塞状態を確認する C0 名を指定する。

**-h**

ヘルプを表示する。

### カラムの説明

表示の先頭に、"+===== BLOCKAGE DISPLAY =====+"が表示された後、指定した論理システム、ノード、または、C0 名に対する閉塞状態を一覧表示する。

-N, -l, -n 指定時は、下記のカラム順に左から右へ表示される。

LSYSTEM	論理システム名	
NODE	ノード名	
STATUS	ノード閉塞状態	
	ACTIVE	稼働中
	BLOCKAGE	閉塞中

PRE-BLOCKAGE            予閉塞中

-C, -c 指定時は、下記のコラムにて、閉塞中の C0 名一覧が表示される。  
また、-c で指定した C0 名が閉塞中でない場合、C0 名は表示されない。

ALL                      全 C0 閉塞状態

CONAME                  C0 名

戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                        処理が正常終了した。

1                        処理が異常終了した。

注意

本コマンドは、AP ノード、OLTP ノード、DB ノードに投入可能。  
ただし、C0 の閉塞状態の照会は、DB ノードでは使用できない。

実行例

ノード閉塞状態参照(dibcmref -N)

+===== BLOCKAGE DISPLAY =====+ 2015/04/03 10:26:32		
LSYSTEM -----	NODE -----	STATUS
XXXXXXXXX ①	NNNNNN ②	SSSSSSS ③
XXXXXXXXX	NNNNNN	SSSSSSS
:	:	:
:	:	:
:	:	:
+===== END OF DISPLAY =====+		

- ①LSYSTEM       : 論理システム名
- ②NODE           : 論理ノード名  
空白の場合は論理システムの閉塞状態を示す
- ③STATUS        : ノード閉塞状態
- ACTIVE           稼働中
- BLOCKAGE        閉塞中
- PRE-BLOCKAGE    予閉塞中

C0 閉塞状態参照(dibcmref -C)

+===== BLOCKAGE DISPLAY =====+ 2015/04/03 10:30:01	
ALL - CONAME -----	
all ①	
CCCCCC ②	
:	
:	
+===== END OF DISPLAY =====+	

- ①ALL           : 全 C0 閉塞状態
- all       全 C0 閉塞中
- 空白       全 C0 閉塞中でない
- ②CONAME       : 閉塞状態の C0 名
- 閉塞中でない C0 名は表示されない

## 2.1.13 dibcmsync (閉塞状態同期コマンド)

### 名前

dibcmsync - 閉塞状態の同期合せをする

### 書式

```
dibcmsync [-d] {-n | -c}
dibcmsync -h
```

### 説明

他ノードの閉塞状態を問い合わせ、最新の閉塞状態情報で同期合せをする。

### オプション

**-d**

同期合せ後、閉塞状態を表示する。

**-n**

ノード閉塞状態を同期対象とする。

**-c**

C0 の閉塞状態を同期対象とする。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                      処理が正常終了した。

1                      処理が異常終了した。

### 注意

本コマンドは、各ノードの閉塞状態が不一致となっている場合に、閉塞状態の同期を行うものである。  
複数ノードで同時実行すると DIMSG020 のメッセージを出力し失敗する可能性があるため、このような場合は、コマンドの再実行するしくみや、コマンド実行ノードを 1 つにするなど、複数ノードで同時実行されないように考慮すること。

### 関連

dibcmupd,   dibcmref

## 2.1.14 dibcmupd (閉塞状態変更コマンド)

### 名前

dibcmupd - 閉塞状態を変更する

### 書式

```
dibcmupd {-a | -b [-p]} -n NodeName
dibcmupd -a -l SysName -n all
dibcmupd {-a | -b} -c C0Name
dibcmupd {-a | -b} all
dibcmupd -a force
dibcmupd -h
```

### 説明

閉塞状態を変更する。

### オプション

**-a**

閉塞を解除する。

**-b**

閉塞する。

**-p**

予閉塞する。

**-l**

論理システム名を指定する。

**-n**

ノード名を指定する。

all が指定された場合、全ての名前が指定されたことと同義となる。

all 指定は閉塞解除時のみ使用する。

**all**

閉塞/閉塞解除時、C0 制御で管理する情報 (C0 名) の全体閉塞状態を変更する。

閉塞解除時、all 以外で閉塞された個々の項目の閉塞状態は解除されない。

**force**

全 C0 の閉塞を解除する。

閉塞解除でのみ指定可能。

**-c**

C0 名を指定する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                      処理が正常終了した。

1                      処理が異常終了した。

## 注意

論理システム名、ノード名、C0 名として”all”は使用できない。

エラーC0 の閉塞状態を閉塞にすることは可能だが、エラーC0 の呼び出しは閉塞状態に関わらず実行される。

本コマンドは、AP ノード、OLTP ノード、DB ノードに投入可能。

ただし、C0 の閉塞状態変更の場合は、DB ノードでは使用できない。

複数ノードで同時実行すると DIMSG020 のメッセージを出力し失敗する可能性があるため、このような場合は、コマンドの再実行するしくみや、コマンド実行ノードを 1 つにするなど、複数ノードで同時実行されないように考慮すること。

## 関連

dibcmref, dibcmsync

## 2.1.15 dicddchg(コマンド配信定義動的変更コマンド)

### 名前

dicddchg - コマンド配信の環境定義を動的に置換する

### 書式

```
dicddchg
dicddchg -h
```

### 説明

コマンド配信が共有メモリに展開している環境定義情報を、最新の環境定義オブジェクトで動的に置換する。

### オプション

-h

使用方法を表示する。

### 戻り値

dicddchg は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

## 2.1.16 dicddctrl (コマンド配信デーモン起動停止コマンド)

### 名前

dicddctrl - コマンド配信デーモンの起動または停止を行う

### 書式

```
dicddctrl -b
dicddctrl -e [-f force]
dicddctrl -h
```

### 説明

コマンド配信されたコマンドを実行するデーモン(コマンド配信デーモン)の起動または停止を行う。

### オプション

**-b**

コマンド配信デーモンを起動する。

**-e**

コマンド配信デーモンを停止する。

**-f force**

コマンド配信デーモンを強制停止する。

**-h**

使用方法を表示する。

### 戻り値

dicddctrl は、以下のいずれかの終了ステータスを戻して終了する。

- 0     正常終了
- 1     異常終了
- 2     警告終了

## 2.1.17 dicddhstctl (コマンド配信履歴状態変更コマンド)

### 名前

dicddhstctl - コマンド配信履歴状態を変更する

### 書式

```
dicddhstctl -s
dicddhstctl -e
dicddhstctl -x
dicddhstctl -i
dicddhstctl -h
```

### 説明

コマンド配信履歴の採取状態を変更する。

### オプション

**-s**

コマンド配信履歴採取を開始(再開)する。

**-e**

コマンド配信履歴採取を停止する。

**-x**

コマンド配信履歴ファイルのカレントを変更する。(スワップする)

**-i**

コマンド配信履歴ファイルを全て初期化する。

**-h**

使用方法を表示する。

### 戻り値

dicddhstctl は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

## 2.1.18 dicddhstedit(コマンド配信履歴編集コマンド)

### 名前

dicddhstedit - コマンド配信履歴ファイルを編集する

### 書式

```
dicddhstedit FILENAME [-t {ALL|REQ|EXE|STA|ERR}] [-n NODE_NAME...] [-q SEQ_NO] [-T]
                        [-o OUTPUT_FILE] [-s START_TIME] [-e END_TIME]
dicddhstedit FILENAME [-t {ALL|REQ|EXE|STA|ERR}] [-n NODE_NAME...] [-q SEQ_NO] [-T]
                        [-o OUTPUT_FILE] [-b BACK_TIME]
dicddhstedit -h
```

### 説明

コマンド配信履歴情報を利用者が参照および加工しやすい形式に変換し、CSV 形式で出力する。

### オプション

#### FILENAME

編集元のコマンド配信履歴ファイル名を絶対パス名、または相対パス名で指定する。

#### -t

出力対象とする履歴情報を指定する。

ALL 採取されている全情報を出力

REQ 配信要求情報のみを出力

EXE 実行要求情報のみを出力

STA 配信結果情報のみを出力

ERR 配信結果のうちエラー情報のみを出力

省略した場合、ALL が指定されたものとみなす。

#### -n NODE\_NAME

出力対象とする論理ノード名(要求元ノード名)を指定する。

15 文字以内、最大 10 個、カンマ区切りで指定する。

#### -q SEQ\_NO

出力対象とするコマンド配信処理通番を指定する。

省略した場合、全処理通番が出力対象となる。

#### -T

編集結果の 1 件目に項目タイトル(項目名称)を出力する場合に指定する。

#### -o OUTPUT\_FILE

編集結果の出力ファイル名を絶対パス、または相対パスで指定する。

省略時には標準出力となる。

#### -s START\_TIME

出力対象範囲の開始時刻を指定する。

形式: yyyymmddhh24miss (時間は 24 時間形式)

#### -e END\_TIME

出力対象範囲の終了時刻を指定する。

形式: yyyymmddhh24miss (時間は 24 時間形式)

**-b BACK\_TIME**

現在時刻から指定された時間分遡って出力する。

数値：0～3600、単位：分

**-h**

使用方法を表示する。

## 戻り値

dicddhstedit は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

2      警告終了

## 2.1.19 dicmdsend(コマンド配信コマンド)

### 名前

dicmdsend - 指定したコマンドを配信先のノードで実行する

### 書式

```
dicmdsend [-l LSYS_NAME] [-d DST_NAME] [-p [ap|oltp|db]] [-x {all|any}] [-s {yes|no}]
          [-f | -b] [-c RETRY_COUNT] [-i RETRY_INTERVAL]
          [-v API_TIMEOUT] [-u SYS_TIMEOUT] [-w EXEC_TIMEOUT]
          [-o OUTPUT_FILE]
          -t COMMAND_TEXT
dicmdsend -h
```

### 説明

指定したコマンドを要求された各ノードに配信し、配信先のノードで実行する。

### オプション

#### -l LSYS\_NAME

転送先の外部論理システム名を指定する。(文字列: 1~15 バイト、省略可)  
本パラメータが指定された場合、外部論理システムへの配信を行う。

#### -d DST\_NAME

コマンドの配信宛先名を指定する。(文字列: 1~15 バイト、省略可)  
論理ノード名、論理システム名、サーバグループ名での指定が可能である。

#### -p

論理システム指定の場合、コマンド配信対象として論理システム配下のノード属性種別を指定することができる。(省略可)

ap      指定論理システム内の AP ノード属性の全ノードを配信対象とする

oltp    指定論理システム内の OLTP ノード属性の全ノードを配信対象とする

db      指定論理システム内の DB ノード属性の全ノードを配信対象とする

-p を指定し、ap/oltp/db を省略した場合は、指定論理システム内の全ノードを配信対象とする。

#### -x

配信宛先名に論理システム名またはサーバグループ名を指定した場合、配信対象ノードを以下から選択できる。(省略可)

all     指定した配信先のうち、全ノードを対象(既定値)

any     指定した配信先のうち、任意の 1 ノードを対象

#### -s

複数ノードへの配信で配信元も配信対象に含まれるとき、配信元にコマンドを配信するか否かを指定できる。(省略可)

yes     配信元ノードにも配信する(既定値)

no      配信元ノードには配信しない

#### -f

配信先ノードが閉塞中でも強制的にコマンド配信する場合に指定する。(省略可)  
省略した場合は、閉塞中であればコマンド配信しない。

-b

閉塞中のノード情報を配信結果に含めない場合に指定する。(省略可)

省略した場合は、配信結果に閉塞中のノード情報を含める。

-c RETRY\_COUNT

接続リトライ回数を定義する。(数値：0～100、省略可)

0 を指定した場合、リトライしない。

省略した場合、環境変数、環境定義の設定値を参照する。(優先順位：環境変数＞環境定義)

-i RETRY\_INTERVAL

接続リトライ処理の待合せ時間を指定する。(数値：1～3600、単位：秒、省略可)

省略した場合、環境変数、環境定義の設定値を参照する。(優先順位：環境変数＞環境定義)

-v API\_TIMEOUT

コマンド配信応答待ち合わせ時間を指定する。(数値：1～86400、省略可)

省略した場合、環境変数、環境定義の設定値を参照する。(優先順位：環境変数＞環境定義)

-u SYS\_TIMEOUT

システム間のコマンド配信結果待ち合わせ時間を指定する。(数値：1～86400、省略可)

省略した場合、既定値(35 秒)で動作する。

-l オプションが指定された場合のみ有効となる。

-w EXEC\_TIMEOUT

コマンド実行応答待ち合わせ時間を指定する。(数値：1～86400、省略可)

省略した場合、環境変数、環境定義の設定値を参照する。(優先順位：環境変数＞環境定義)

-o OUTPUT\_FILE

実行したコマンドの結果出力先を絶対パス名、または相対パス名で指定する。

標準出力は「出力ファイル名\_ノード名.std」、標準エラーは「出力ファイル名\_ノード名.err」で作成される。

省略時は標準出力へ出力する。(文字列：1～192 バイト、省略可)

-t COMMAND\_TEXT

コマンドテキストを指定する。(文字列：1～1023 バイト、省略不可)

-h

使用方法を表示する。

## 戻り値

dicmdsend は、以下のいずれかの終了ステータスを戻して終了する。

0 正常終了

1 異常終了

2 警告終了

## 注意

- -l パラメータには、環境定義 SYSMAP 節に定義された論理システム名を指定すること。
- -b を指定し、配信対象となるノードが全て閉塞中であつた場合、コマンドは異常終了する。
- コマンドに対する配信先情報を環境定義 CMDSEND 節 CMDRT 項に定義することにより、コマンドの配信先パラメータを省略することができる。

以下の表に、省略可能な配信先パラメータと対応する環境定義について示す。

パラメータ	対応する環境定義
-d	CMDRT 項 DSTNAME
-p	CMDRT 項 LSTYPE
-x	CMDRT 項 TARGET
-s	CMDRT 項 SELF

配信宛先名が論理システムで、-p が省略された場合、CMDRT 項に対応するコマンドの登録が無かった場合は、コマンドは異常終了する。

## 2.1.20 dicocinit(初期化コマンド)

### 名前

dicocinit - CO 制御の初期化処理をおこなう。

### 書式

```
dicocinit [ -c | -w ]  
dicocinit -h
```

### 説明

CO 制御の環境定義オブジェクトを読み込み、CO 制御 TPP 起動準備をおこなう。  
また、CO 制御監視デーモンを起動する。

### オプション

**-c**

新規の環境定義オブジェクトで起動する。(コールドモード)

**-w**

前回運転時の環境を引き継ぐ。(ウォームモード)(既定値)

**-h**

使用方法を表示する。

### 戻り値

0	処理が正常終了しました。
1	処理が異常終了しました。
2	処理が警告終了しました。

### 関連

dicocterm, dicocrefsg,

## 2.1.21 dicocmod(動作変更コマンド)

### 名前

dicocmod - CO 制御の動作環境を変更する。

### 書式

```
dicocmod -O {YES|NO} [-m TPBASE_MONITOR_NAME] -t TxID
dicocmod -E [-e LIMIT_ELAPS] [-c LIMIT_CPU] [-r MAX_RESET] [-m TPBASE_MONITOR_NAME] -t TxID
dicocmod -h
```

### 説明

CO 制御の動作環境を変更する。変更された値は CO 制御ウォーム起動に引き継がれる。

### オプション

**-O {YES|NO}**

稼動統計の採取情報を変更する。

YES : 採取する。

NO : 採取しない。

**-E**

経過時間、CPU 時間制限値を変更する。

**-m TPBASE\_MONITOR\_NAME**

動作変更する TPBASE モニタ名を指定する。

本オプションを指定しない場合は、コマンド実行ノードの全ての TPBASE モニタが対象となる。

**-t TxID**

動作変更するトランザクション ID を指定する。

**-e LIMIT\_ELAPS**

経過時間制限値を変更する。(0～999999)

0 を指定した場合、無制限となる。

**-c LIMIT\_CPU**

CPU 時間制限値を変更する。(0～999999)

0 を指定した場合、無制限となる。

**-r MAX\_RESET**

リセット最大回数を変更する。(0～999999)

0 を指定した場合、無制限となる。

**-h**

使用方法を表示する。

### 戻り値

0	処理が正常終了しました。
1	処理が異常終了しました。
2	処理が警告終了しました。

### 注意

-E 指定時は、少なくとも -e、-c、-r の何れか 1 つを指定する必要がある。

関連

dicocinit, dicocrefsg

## 2.1.22 dicocref (稼動状況照会コマンド)

### 名前

dicocref - CO 制御 TPP の稼動状況を照会する。

### 書式

```
dicocref -S [-m TPBASE_MONITOR_NAME] [-s THRESHOLD_VALUE] [-c CLASS_NAME]
dicocref -Q [-m TPBASE_MONITOR_NAME] [-s THRESHOLD_VALUE] [-z] [-c CLASS_NAME] [-t TxID]
dicocref -P [-m TPBASE_MONITOR_NAME] [-z] [-c CLASS_NAME]
dicocref -C [-m TPBASE_MONITOR_NAME] [-z] [-c CLASS_NAME] [-a CO_NAME]
dicocref -h
```

### 説明

クラスキュー、トランザクション、CO の実行状況を照会する。

また、滞留件数を除く情報のリセットをおこなう。

### オプション

#### -S

TPBASE クラスキューの滞留電文数を照会する。

#### -Q

TPBASE クラスキューと TxID の滞留、受信電文数等を照会する。

#### -P

CO 制御 TPP のプロセス単位情報を照会する。

#### -C

CO 単位情報を照会する。

#### -m TPBASE\_MONITOR\_NAME

照会したい TPBASE モニタ名を指定する。(英数字 1～8)

省略した場合は、環境定義(DIOSAMAP 節)で定義された全 TPBASE モニタが照会対象となる。

#### -c CLASS\_NAME

照会したいクラス名を指定する。(英数字 1～32)

省略した場合は、環境定義(COCENV 節)で定義された全クラスが照会対象となる。

#### -t TxID

照会したいトランザクション ID を指定する。(英数字 1～6)

省略した場合は、環境定義(COCENV 節)で定義された全トランザクションが照会対象となる。

#### -a CO\_NAME

照会したい CO 名を指定する。(英数字 1～30)

省略した場合は、全 CO が照会対象となる。

#### -s THRESHOLD\_VALUE

滞留電文数がしきい値を超えているもののみを表示する場合に指定する。(数字 0～9999999)

省略した場合は、滞留電文数に関係なく全て表示される。

#### -z

滞留数以外の情報をリセットする。

情報照会後にリセットされる。

-Q を指定した場合、クラスとトランザクション ID には対応しない。ノード内全 TPBASE モニタが指定された TPBASE モニタの全トランザクション数がリセットされる。

-h

使用方法を表示する。

## 戻り値

0	処理が正常終了しました。
1	処理が異常終了しました。
2	処理が警告終了しました。

## 注意

C0 名を指定して電文数をリセットする場合、リセット対象となる C0 制御 TPP は、リセット要求後、最初のトランザクションでリセットが実行される。従って、トランザクションが実行される前に別 C0 名のリセット要求がおこなわれると後の C0 名が有効となり、前にリセット要求がおこなわれた C0 がリセットされないことがあります。

## 関連

dicocinit

カラムの説明

オプション -S

①TPBASE モニタ名

②クラス名

③滞留電文数

%dicocref -S		
+===== WAITING MESSAGE DISPLAY =====+      YYYY/MM/DD HH:MM:SS		
TPM	CLASS	WAITINGMSG
①	②	③
tpbase01	c2345678901234567890123456789012	10
	class-B	1234567890
tpbase2	c2345678901234567890123456789012	1000
	class-B	0
+===== END OF DISPLAY =====+		

オプション -Q

①TPBASE モニタ名

②クラス名

③トランザクション ID

④トランザクションの状態

START     トランザクション開始から TPP 実行の間  
ABORT     異常終了出口ルーチン実行中  
TPP       TPP 実行中  
END       TPP 終了からトランザクション完了までの間  
IDLE       メッセージ受信待ち  
DUMP       ダンプ出力中  
NOACT     トランザクションが活性状態でない

⑤総トランザクション数

⑥滞留電文数

```
%dicocref -Q
+===== WAITING MESSAGE DISPLAY =====+          YYYY/MM/DD HH:MM:SS
TP-MONITOR: tpbasel ①
CLASS                                TXID    STATUS    TXCOUNT  WAITINGMSG
 ②                                ③        ④        ⑤         ⑥
c2345678901234567890123456789012          5001      100503
                                Tx0001  START      5000      100000
                                Tx2    TPP         1         500
                                Tx0003  IDLE         0          3
class-middle          5001  1234567890
                                Tx0004  START      5000  1000000000
                                Tx5    TPP         1    234567890
                                Tx0006  IDLE         0          0

TP-MONITOR: tpbase2
CLASS                                TXID    STATUS    TXCOUNT  WAITINGMSG
:
+===== END OF DISPLAY =====+
```

オプション -P

- ①TPBASE モニタ名
- ②クラス名
- ③プロセス ID
- ④累積電文数
- ⑤正常終了電文数
- ⑥ピーク時の秒間処理件数
- ⑦ピーク時間

```
%dicocref -P
+===== CO CONTROL TPP INFO DISPLAY =====+   YYYY/MM/DD HH:MM:SS
TP-MONITOR: tpbase01 ①
CLASS: c2345678901234567890123456789012 ②
  PID   INPUT-MSG     DONE   PEAK  PEAK-TIME
  ③      ④           ⑤      ⑥      ⑦
    120 1234567890   123456    10 2011/12/31 12:34:45
123456      1000    1000    10 2011/12/31 12:34:45

TP-MONITOR: tp2
CLASS: class-test
  PID   INPUT-MSG     DONE   PEAK  PEAK-TIME
24943      1000    1000    10 2011/12/31 12:34:45
   56      1000    1000    10 2011/12/31 12:34:45
+===== END OF DISPLAY =====+
```

オプション -C

- ①TPBASE モニタ名
- ②クラス名
- ③プロセス ID
- ④CO 名
- ⑤累積電文数
- ⑥正常終了電文数
- ⑦ピーク時の秒間処理件数
- ⑧ピーク時間

```
%dicocref -C
+===== CO CONTROL TPP INFO DISPLAY =====+
                                                    YYYY/MM/DD HH:MM:SS
TP-MONITOR: tpbase01 ①
CLASS: c2345678901234567890123456789012 ②
  PID(*)/CONAME                INPUT-MSG    DONE    PEAK    PEAK-TIME
  ③          ④                ⑤          ⑥      ⑦      ⑧
* 123456                1234567890  123456      10  2011/12/31 12:34:45
  C03456789012345678901234567890  1234567890  123456
  C0001                        0        0
*   455                2000    2000  -----  ----/--/-- --:--:--
  C03456789012345678901234567890    1000    1000
  C0002                        1000    1000

+===== END OF DISPLAY =====+
```

## 2.1.23 dicocrefsg (環境定義照会コマンド)

### 名前

dicocrefsg - 環境定義を照会する。

### 書式

```
dicocrefsg -B
dicocrefsg -V [-m TPBASE_MONITOR_NAME]
dicocrefsg -C [-k {AP|OLTP}] [-n NODE_NAME] [-m TPBASE_MONITOR_NAME] [-a CO_NAME]
dicocrefsg -X [-k {AP|OLTP}] [-n NODE_NAME] [-m TPBASE_MONITOR_NAME] [-t TxID]
dicocrefsg -h
```

### 説明

CO 制御が共有メモリに展開している環境定義を照会する。

### オプション

#### -B

COCENV 節の COMMON 項、EXIT 項、COGROUP 項の情報を照会する。

#### -V

COCENV 節の TPBASE モニタ情報を照会する。  
照会できる TPBASE モニタは、コマンド実行論理ノード上の TPBASE モニタとなる。

#### -C

CO 指定の宛先情報を照会する。

#### -X

トランザクション ID 指定の宛先情報を照会する。

#### -k {AP|OLTP}

指定されたノードタイプの情報を照会する。  
本オプションを指定しない場合は、全ての論理ノードが対象となる。

#### -n NODE\_NAME

指定された論理ノードの情報を照会する。  
本オプションを指定しない場合は、全ての論理ノードが対象となる。

#### -m TPBASE\_MONITOR\_NAME

指定された TPBASE モニタの情報を照会する。  
-V オプション指定時は、指定された TPBASE モニタを対象とし、本オプションを指定しない場合は、コマンド実行ノードの全ての TPBASE モニタが対象となる。  
-C、-X オプション指定時は、宛先の TPBASE モニタが対象となる。

#### -c CO\_NAME

指定されたクラスの情報を照会する。  
本オプションを指定しない場合は、全ての CO 名が対象となる。

#### -t TxID

指定されたトランザクション ID の情報を照会する。  
本オプションを指定しない場合は、全てのトランザクション ID が対象となる。

-h

使用方法を表示する。

#### 戻り値

0	処理が正常終了しました。
1	処理が異常終了しました。
2	処理が警告終了しました。

#### 関連

dicocinit、dicocmod

カラムの説明

オプション -B

- ①デーモン起動タイムアウト値
- ②デーモン停止タイムアウト値
- ③プロセス監視間隔
- ④アボート時プロセス停止有無
- ⑤経過時間超過時の処置
- ⑥メッセージ DICOC001 の表示有無
- ⑦メッセージ DICOC002 の表示有無
- ⑧メッセージ DICOC003 の表示有無
- ⑨メッセージ DICOC004 の表示有無
- ⑩プロセス初期化出口名
- ⑪プロセス終了出口名
- ⑫受信電文解析出口名
- ⑬トランザクション初期化出口名
- ⑭トランザクション終了出口名
- ⑮アボート#1 出口名
- ⑯アボート#2 出口名
- ⑰CO グループ名
- ⑱CO 名

```
%dicocrefsg -B
+===== SG(COCENV) DISPLAY =====+      YYYY/MM/DD  HH:MM:SS

INITIME   = 120  ①      DICOC001 = NO    ⑥
KILLTIME  = 60   ②      DICOC002 = YES   ⑦
WATCHINVL = 60   ③      DICOC004 = NO    ⑧
ABORTREQ  = CONT ④      DICOC005 = YES   ⑨
ELPOVER   = STOP ⑤

<EXIT>
PRCINIT   = prcinit      ⑩
PRCTERM   = (not defined) ⑪
ANALYZE   = analyze      ⑫
TRANSINIT = transinit    ⑬
TRANSTERM = transterm    ⑭
ABORT1    = abort1       ⑮
ABORT2    = abort2       ⑯

<COGROUP>
COGROUPNAME      CONAME
 ⑰              ⑱
```

g23456789012345	c23456789012345678901234567890
	c23456789012345678901234567890
	c23456789012345678901234567890
	c23456789012345678901234567890
g23456789012345	c23456789012345678901234567890
	c23456789012345678901234567890
	c23456789012345678901234567890
+===== END OF DISPLAY =====+	

オプション -V

- ①TPBASE モニタ名
- ②クラス名
- ③電文最大長
- ④電文保留用トランザクション ID
- ⑤電文保留用 VD 名
- ⑥トランザクション ID
- ⑦VD 名
- ⑧CO 名
- ⑨エラーCO 名
- ⑩リトライ最大回数
- ⑪稼働統計情報の採取有無
- ⑫CPU 時間制限値
- ⑬経過時間制限値
- ⑭CO の稼働情報採取有無
- ⑮経過時間リセットの許可回数
- ⑯連鎖時に経過時間、CPU 時間のリセット有無
- ⑰CO グループ名

```
%dicocrefsg -V
+===== SG(COCENV) TPBASE-MONITOR DISPLAY =====+      YYYY/MM/DD  HH:MM:SS
<TP-MONITOR>
TP-MONITOR : tpbasemon1 ①
<CLASS>
NAME      = c2345678901234567890123456789012 ②
MSGMAX    = 1000000                               ③
RESTTXID  = TXID01                                ④
RESTVD    = VDTXID01                              ⑤

<TRANS>
TXID      = trns01                                ⑥
VD        = v23456789012345678901234             ⑦
CONAME    = c23456789012345678901234567890      ⑧
ERRCO     = c23456789012345678901234567890      ⑨
RETRY     = 10      ⑩
OPS       = YES    ⑪ COINFO    = NO      ⑭
CPUTIME   = 999999⑫ ELPRESET  = 100     ⑮
ELPTIME   = nn     ⑬ COCHAIN   = RESET   ⑯
COGROUP   = g23456789012345 ⑰
COGROUP   = g23456789012345
COGROUP   = g23456789012345
COGROUP   = g23456789012345
+===== END OF DISPLAY =====+
```

オプション -C

①CO 名

②CO グループ名

③論理ノード名

④ノードタイプ

⑤TPBASE モニタ名

⑥トランザクション ID

%dicocrefsg -C					
+===== CO LOCATION INFO DISPLAY =====+			YYYY/MM/DD HH:MM:SS		
CONAME	COGROUP	LNODE	NODETYPE	TP-MONITOR	TXID
①	②	③	④	⑤	⑥
123456789012345678901234567890	123456789012345	123456789012345	AP	12345678	123456
CONAME_001	COG1	LNODENAME1	OLTP	TPM001	TXIDZ
CONAME_001	COG1	LNODENAME2	AP	TPM002	TXIDA
CONAME_002	COG2	LNODENAME1	OLTP	TPM002	TXIDS
+===== END OF DISPLAY =====+					

オプション -X

①トランザクション ID

②論理ノード名

③ノードタイプ

④TPBASE モニタ名

%dicocrefsg -X				
+===== TXID LOCATION INFO DISPLAY =====+				YYYY/MM/DD HH:MM:SS
TXID	LNODE	NODETYPE	TP-MONITOR	
①	②	③	④	
123456	123456789012345	AP	12345678	
TXID1	LNODENAME1	OLTP	TPM001	
TXID1	LNODENAME2	AP	TPM002	
TXID2	LNODENAME1	OLTP	TPM003	
+===== END OF DISPLAY =====+				

## 2.1.24 dicocterm(終了コマンド)

### 名前

dicocterm - CO 制御の終了処理をおこなう。

### 書式

dicocterm [-h]

### 説明

CO 制御の終了処理をおこなう。

また、CO 制御監視デーモンを停止する。

### オプション

-h

使用方法を表示する。

### 戻り値

0	処理が正常終了しました。
1	処理が異常終了しました。
2	処理が警告終了しました。

### 注意

CO 制御用 TPBASE を停止した後に投入する。

### 関連

dicocinit

## 2. 1. 25      dicoctxblock (電文保留制御コマンド)

### 名前

dicoctxblock - 保留用 TxID の閉塞制御をおこなう。

### 書式

```
dicoctxblock [ -a | -b | -r ] [ -m TPBASE_MONITOR_NAME ]  
dicoctxblock -h
```

### 説明

退避用 TxID の閉塞、解除、および閉塞状態の照会をおこなう。

### オプション

**-a**

閉塞を解除する。

**-b**

閉塞する。

**-r**

閉塞状態を照会する。(既定値)

**-m TPBASE\_MONITOR\_NAME**

指定した TPBASE モニタ名配下の閉塞、解除、および閉塞状態の照会する。

省略した場合は、環境定義(COCENV 節)で定義された全クラスが照会対象となる。

**-h**

使用方法を表示する。

### 戻り値

- |   |              |
|---|--------------|
| 0 | 処理が正常終了しました。 |
| 1 | 処理が異常終了しました。 |
| 2 | 処理が警告終了しました。 |

### 関連

dicocinit, dicocterm, dicocrefsg

カラムの説明

以下に、 dicoctxblock -r が表示するカラムの見出しと各カラムの意味

- ①TP-MONITORTP モニタ名
- ②CLASSクラス名
- ③TXIDTxID
- ④STAT  
トランザクション状態  
BLK 閉塞中  
ACT 使用可
- ⑤VDVD 名
- ⑥STAT  
VD の状態  
BLK 閉塞中  
ACT 使用可

1	2	3	4	5	6	7	8
1234567890123456789012345678901234567890123456789012345678901234567890							
% dicoctxblock -r							
+===== TXID STATUS DISPLAY =====+				YYYY/MM/DD HH:MM:SS			
TP-MONITOR: tpbase01 ①							
CLASS		TXID	STAT	VD	STAT		
②		③	④	⑤	⑥		
c234567890123456789012345678901A		tx0001	ACT	vd3456789012345678901234	ACT		
c234567890123456789012345678901B		tx02	BLK	vd0021	ACT		
class-middle-priorityA		tx02	BLK	vd0021	ACT		
class-middle-priorityB		tx22	BLK	vd0021	ACT		
TP-MONITOR: tp2							
CLASS		TXID	STAT	VD	STAT		
c234567890123456789012345678901A		tx0001	BLK	vd3456789012345678901234	ACT		
c234567890123456789012345678901B		tx02	ACT	vd0021	ACT		
class-middle-priorityA		tx02	ACT	vd0021	ACT		
class-middle-priorityB		tx02	ACT	vd0021	ACT		
+===== END OF DISPLAY =====+							

## 2.1.26 dicomcatget(カタログメッセージ標準出力コマンド)

### 名前

dicomcatget - カタログメッセージを標準出力する。

### 書式

dicomcatget subcomid set msg def [str]...

### 説明

指定されたカタログ情報を表示する。

### オプション

パラメータ説明

パラメータ	I	説明
subcomid	I	サブコン識別子(3文字)(必須)
set	I	カタログセット番号(必須)
msg	I	メッセージ番号(必須)
def	I	カタログ情報取得に失敗したときに表示する、デフォルトメッセージ(必須)
str	I	カタログメッセージに渡す置換文字列情報(省略可, 最大5つまで指定可能)

### 戻り値

0                      正常終了

1                      異常終了

### 一般規則

- ・ (DIOSA ルートディレクトリ)/msg/%L/diosa\_%N.cat にカタログ情報が登録されていること。  
→ カatalogファイルオープンに失敗した場合、CD0 メッセージを表示しエラーとする。
- ・ 指定した サブコン識別子、セット番号、メッセージ番号に該当するメッセージ情報が存在しない場合、def で指定したメッセージを表示し、エラーとする

## 2.1.27 didamctrl (死活監視デーモン起動／停止コマンド)

### 名前

didamctrl - 死活監視デーモンの起動／停止を行う

### 書式

```
didamctrl [-b]
didamctrl -e
didamctrl -h
```

### 説明

**didamctrl -b** は、死活監視デーモンを起動する。

**didamctrl -e** は、死活監視デーモンを停止する。

### オプション

**-b**

死活監視デーモンを起動する。(規定値)

**-e**

死活監視デーモンを停止する。

**-h**

使用方法を表示する。

### 戻り値

didamctrl は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

### 注意

didamctrl は、DIOSA/XTP 初期化コマンドが実行されていない場合、異常終了する。

DIOSA/XTP 起動コマンド実行時の環境変数で指定された動作パラメータ値(監視間隔等)を引き継ぐものとする。didamctrl により変更は行なえない。

didamctrl -e は、死活監視デーモンの停止のみとし、デーモン管理テーブルの削除は行なわない。

## 2.1.28 didbgerr (障害情報検索・編集コマンド)

### 名前

didbgerr - デバッグトレース機能の共有メモリ上で管理している障害情報、または障害情報ファイル出力デーモンが出力した障害情報出力ファイルの内容を表示/ファイル出力する。

### 書式

```
didbgerr [-p プロセス ID | -n プロセス名] [-t データ部のテキスト] [-s 関数名] [-c コメント部のテキスト] [-o 出力ファイル名] [-d]  
didbgerr -f 入力ファイル名 [-p プロセス ID | -n プロセス名] [-t データ部のテキスト] [-s 関数名] [-c コメント部のテキスト] [-B 開始時刻] [-E 終了時刻] [-o 出力ファイル名] [-d]  
didbgerr -h
```

### 説明

デバッグトレース機能の共有メモリ内の障害情報および障害情報出力ファイルから時間帯・プロセス ID|プロセス名・区間・データの内容・コメントをキーにして、必要な障害情報を検索し、バイナリ形式からテキスト形式に編集して画面表示やファイル出力を行う。なお、複数のキーを指定した場合、それら全てのキーにあてはまるログ情報が検索・出力される。

### オプション

#### -f

入力ファイルを指定する。指定するファイルは、障害情報ファイル出力デーモンが出力したバイナリ形式の障害情報出力ファイルであること。

#### -p

プロセス ID を指定する。このオプションで指定されたプロセス ID の情報だけを出力する。**-n** オプションと同時に指定できない。

#### -n

プロセス名を指定する。このオプションで指定されたプロセス名の情報だけを出力する。**-p** オプションと同時に指定できない。

#### -t

検索対象のデータ部のテキストを指定する。このオプションで指定したテキストと一致するテキストをデータ部に含む障害情報を編集の対象とする。指定方法は、0x で始まる任意の桁数の 16 進数、または、「`“`」で囲まれた任意の長さの ASCII コード文字列で、指定可能な長さは 64 バイトである。

#### -s

検索対象の関数名を指定する。このオプションで指定した関数名と一致する関数の障害情報を編集の対象とする。関数名には前方一致を意味する記号として\*が使用可能である。例えば、先頭が”ABC”である関数名全ての場合は、ABC\*と指定する。指定可能な長さは 31 バイトである。

#### -c

検索対象のコメント部のテキストを指定する。このオプションで指定したテキストと一致するテキストをコメント部に含む障害情報を編集の対象とする。指定方法は、「`“`」で囲まれた任意の長さの ASCII コード文字列で、指定可能な長さは 49 バイトである。

#### -B

—E

-o 出力ファイル名

-d

$$-h$$

戻り値

0	正常終了
1	異常終了

注意

関連項目

## 実行例

## 障害情報表示:障害情報出力ファイル指定(didbgerr -f)

```
SYSTEM : HOSTNAME(ssssssssssssssssss)0xhhhhh(XXXXXXX) ①  
PAGE   : tttt(ERR INFORMATION) ②  
qqqqqqqqqqqqqqqq(ppppp:ss) [tt-tt-tt ]uu:uu:uu.uuu  
③          ④          ⑤          ⑥  
[ooooooooooooo]:kkkkk mmmmmmmmmmmmmmmmm[rrrrrrrr][ LENGTH:dddd(0xHHHHHHH)]  
⑦          ⑧    ⑨          ⑩          ⑪  
Oxgggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh |ccccccccccccc  
⑫          ⑬          ⑭  
:  
Oxggggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh |ccccccccccccc  
Oxggggggggg hhhh hhhh hhhh hhhh hhhh hhhh hhhh |ccccccccccccc
```

- ①SYSTEM : システム情報を以下の形式で表示する  
“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)”
- ②PAGE : トレース情報保存領域のページ番号
- ③プロセス名
- ④プロセス ID とスレッド ID : (プロセス ID:スレッド ID)
- ⑤区間が開始/終了した日付 : 左から順に西暦年の下 2 桁-月-日。直前のログ情報と日付が同じ場合表示しない
- ⑥区間が開始/終了した時刻 : 左から順に時:分:秒. ミリ秒(3 桁)
- ⑦ソースファイル名 : 直前に表示したトレース情報と同じ場合表示しない
- ⑧行番号
- ⑨コメント
- ⑩エラーコード : 値が 0 の場合表示しない
- ⑪LENGTH : 出力リソースレコード長を“10 進数(16 進数)”で表示する。値が 0 の場合表示しない
- ⑫表示開始アドレス
- ⑬バイナリデータの HEX ダンプ
- ⑭バイナリデータのキャラクタダンプ

障害情報表示：障害情報出力ファイル詳細指定(`didbgerr -f -d`)

```

SYSTEM : HOSTNAME(ssssssssssssssssssssss):0xhhhhhhhh(XXXXXXXXX) ①

PAGE   : ttttt(ERR INFORMATION) ②

RCDTYPE: ERR   LVL: N   SRCNAME: ooooooooooooooooooooo:kkkkk
③             ④             ⑤

DATE   : YY-MM-DD      TIME: hh:mm:dd.uuu
⑥             ⑦

PROCESS: [ppppp:ttttt] qqqqqqqqqqq
⑧

SECTION: xxxxxxxx  ERRCODE: rrrrrrrr
⑨             ⑩

COMMENT: ~~~~~~
⑪

ADDR    : 0xhhhhhhh  LENGTH: ddddd(0xeeeeeeee)
⑫             ⑬

0x00000000 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccccc
⑭             ⑮

0x00000010 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccccc
⑮             ⑯

0x00000020 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccccc
⑯             ⑰

:           :           :

```

- ①SYSTEM : システム情報を以下の形式で表示する。  
“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)”
- ②PAGE : トレース情報保存領域のページ番号
- ③RCDTYPE : 情報の種類 ERR
- ④LVL : トレース情報出力のレベル
- ⑤SRCNAME : ソースファイル名を“ソースファイル名:行番号”の形式で表示する
- ⑥DATE : 情報収集を行った日付。左から順に西暦年の下2桁-月-日
- ⑦TIME : 情報収集を行った時刻。左から順に時-分-秒.ミリ秒(3桁)
- ⑧PROCESS : プロセスの情報を”[プロセス ID:スレッド ID] プロセス名”の形式で表示する
- ⑨SECTION : 関数名
- ⑩ERRCODE : エラーコード
- ⑪COMMENT : コメント
- ⑫ADDR : 出力リソースの先頭アドレス
- ⑬LENGTH : 出力リソースのレコード長を”10進数(16進数)”の形式で表示する
- ⑭表示オフセット値
- ⑮バイナリデータのHEXダンプ
- ⑯バイナリデータのキャラクタダンプ

## 2.1.29 didbgerrswap (障害情報強制スワップコマンド)

### 名前

didbgerrswap - デバッグトレース機能の共有メモリ内の障害情報をファイルに出力する

### 書式

```
didbgerrswap [-m]
didbgerrswap -f [-P 絶対パス]
didbgerrswap -h
```

### 説明

デバッグトレース機能の共有メモリ内に格納されている障害情報を、現在の出力先となっている障害情報出力ファイルに出力する。

事前に **didbginit** コマンドが実行されていない場合や、障害情報ファイル出力デーモンが不在の場合、本コマンドは異常終了する。

### オプション

**-m**

デバッグトレース機能の共有メモリ内の障害情報を障害情報出力ファイルへ出力する。(省略時の既定値)

**-f**

デバッグトレース機能の共有メモリ内の障害情報を障害情報出力ファイルへ出力した後、障害情報出力ファイルをスワップする。

**-P**

障害情報出力ファイルの出力先パスを変更する時に指定する。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 関連

didbginit

## 2.1.30 didbginit (デバッグトレース機能初期化コマンド)

### 名前

didbginit - デバッグトレース機能を開始する

### 書式

```
didbginit
didbginit -h
```

### 説明

現在設定されている環境変数にもとづいてログ情報、障害情報を格納する保存領域(共有メモリ)を作成し、ログ情報ファイル出力デーモンプロセス、障害情報ファイル出力デーモンプロセスを起動する。

### オプション

-h  
使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 関連

didbgterm

## 2.1.31 didbglog(ログ情報検索・編集コマンド)

### 名前

didbglog - デバッグトレース機能の共有メモリ上で管理しているログ情報、またはログ情報ファイル出力デーモンが出力したログ情報出力ファイルの内容を表示/ファイル出力する。

### 書式

```
didbglog [-p プロセス ID | -n プロセス名] [-v レベル値] [-t データ部のテキスト] [-s 関数名] [-c  
コメント部のテキスト] [-o 出力ファイル名] [-d]  
didbglog -f 入力ファイル名 [-p プロセス ID | -n プロセス名] [-v レベル値] [-t データ部のテキ  
スト] [-s 関数名] [-c コメント部のテキスト] [-B 開始時刻] [-E 終了時刻] [-o 出力ファイル名] [-d]  
didbglog -h
```

### 説明

デバッグトレース機能の共有メモリ内のログ情報およびログ情報出力ファイル(デーモン経由)から時間帯・プロセス ID|プロセス名・ログ情報出力レベル・区間・データの内容・コメントをキーにして、必要なデーモン経由ログ情報を検索し、バイナリ形式からテキスト形式に編集して画面表示やファイル出力を行う。なお、複数のキーを指定した場合、それら全てのキーにあてはまるログ情報が検索・出力される。

### オプション

#### -f

入力ファイルを指定する。指定するファイルは、ログ情報ファイル出力デーモンが出力したバイナリ形式のログ情報出力ファイルであること。

#### -p

プロセス ID を指定する。このオプションで指定されたプロセス ID の情報だけを出力する。**-n** オプションと同時に指定できない。

#### -n

プロセス名を指定する。このオプションで指定されたプロセス名の情報だけを出力する。**-p** オプションと同時に指定できない。

#### -v

入力ファイルから検索するログ情報のレベルを指定する。指定可能な値は 1～9。ここで指定したレベルよりも小さいレベル値のログ情報を編集の対象とする。

#### -t

検索対象のデータ部のテキストを指定する。このオプションで指定したテキストと一致するテキストをデータ部に含むログ情報を編集の対象とする。指定方法は、0x で始まる任意の桁数の 16 進数、または、「`“`」で囲まれた任意の長さの ASCII コード文字列で、指定可能な長さは 64 バイトである。

#### -s

検索対象の関数名を指定する。このオプションで指定した関数名と一致する関数のログ情報を編集の対象とする。関数名には前方一致を意味する記号として\*が使用可能である。例えば、先頭が”ABC”である関数名全ての場合は、ABC\*と指定する。指定可能な長さは 31 バイトである。

#### -c

検索対象のコメント部のテキストを指定する。このオプションで指定したテキストと一致するテキストをコメント部に含むログ情報を編集の対象とする。指定方法は、「`“`」で囲まれた任意の長さの ASCII コード文字列で、指定可能な長さは 49 バイトである。

#### -B

—E

-0

-d

$$-h$$

- |                   |   |
|-------------------|---|
| ①SYSTEM           | : システム情報を以下の形式で表示する<br>“ホスト名 (OS のバージョン): トレース共有メモリ IPC キー (論理ノード名称)” |
| ②PAGE             | : トレース情報保存領域のページ番号  |
| ③プロセス名            |   |
| ④プロセス ID とスレッド ID | : (プロセス ID: スレッド ID)  |
| ⑤区間が開始/終了した日付     | : 左から順に西暦年の下 2 桁-月-日。直前のログ情報と日付が同じ場合表示しない                             |
| ⑥区間が開始/終了した時刻     | : 左から順に時:分:秒.ミリ秒(3 桁)   |
| ⑦ソースファイル名         | : 直前に表示したトレース情報と同じ場合表示しない   |
| ⑧行番号              |   |
| ⑨コメント             |   |
| ⑩エラーコード           | : 値が 0 の場合表示しない   |
| ⑪LENGTH           | : 出力リソースレコード長を“10 進数(16 進数)”で表示する。値が 0 の場合表示しない                       |
| ⑫表示開始アドレス         |   |
| ⑬バイナリデータの HEX ダンプ |   |
| ⑭バイナリデータのキャラクタダンプ |   |

ログ情報表示：ログ情報出力ファイル詳細指定 (didbglog -f -d)

```

SYSTEM : HOSTNAME(ssssssssssssssssssssss):0xhhhhhhh(XXXXXXXX) ①
PAGE : ttttt(LOG INFORMATION) ②
RCDTYPE: ERR LVL: N SRCNAME: ooooooooooooooooooooo:kkkkk
③ ④ ⑤
DATE : YY-MM-DD TIME: hh:mm:dd.uuu
⑥ ⑦
PROCESS: [ppppp:ttttt] qqqqqqqqqq
⑧
SECTION: xxxxxxxx ERRCODE: rrrrrrrr
⑨ ⑩
COMMENT: ~~~~~~
⑪
ADDR : 0xhhhhhhh LENGTH: dddd(0xeeeeee)
⑫ ⑬
0x00000000 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccccc
⑭ ⑮ ⑯
0x00000010 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccccc
0x00000020 hhhh hhhh hhhh hhhh hhhh hhhh hhhh |cccccccccccccccc
: :

```

- ①SYSTEM : システム情報を以下の形式で表示する。  
“ホスト名(OS のバージョン):トレース共有メモリ IPC キー(論理ノード名称)”
- ②PAGE : トレース情報保存領域のページ番号
- ③RCDTYPE : 情報の種類 ERR
- ④LVL : トレース情報出力のレベル
- ⑤SRCNAME : ソースファイル名を“ソースファイル名:行番号”の形式で表示する
- ⑥DATE : 情報収集を行った日付。左から順に西暦年の下2桁-月-日
- ⑦TIME : 情報収集を行った時刻。左から順に時-分-秒.ミリ秒(3桁)
- ⑧PROCESS : プロセスの情報を”[プロセス ID:スレッド ID] プロセス名”の形式で表示する
- ⑨SECTION : 関数名
- ⑩ERRCODE : エラーコード
- ⑪COMMENT : コメント
- ⑫ADDR : 出力リソースの先頭アドレス
- ⑬LENGTH : 出力リソースのレコード長を”10進数(16進数)”の形式で表示する
- ⑭表示オフセット値
- ⑮バイナリデータの HEX ダンプ
- ⑯バイナリデータのキャラクタダンプ

## 2. 1. 32 didbglogswap (ログ情報強制スワップコマンド)

### 名前

didbglogswap - デバッグトレース機能の共有メモリ内のログ情報をファイルに出力する

### 書式

```
didbglogswap [-m]
didbglogswap -f [-P 絶対パス]
didbglogswap -h
```

### 説明

デバッグトレース機能の共有メモリ内に格納されているログ情報を、現在の出力先となっているログ情報出力ファイルに出力する。

事前に **didbginit** コマンドが実行されていない場合や、ログ情報ファイル出力デーモンが不在の場合、本コマンドは異常終了する。

### オプション

**-m**

デバッグトレース機能の共有メモリ内のログ情報をログ情報出力ファイルへ出力する。(省略時の既定値)

**-f**

デバッグトレース機能の共有メモリ内のログ情報をログ情報出力ファイルへ出力した後、ログ情報出力ファイルをスワップする。

**-P**

ログ情報出力ファイルの出力先パスを変更する時に指定する。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 関連

didbginit

## 2.1.33 didbgterm (デバッグトレース機能終了コマンド)

### 名前

didbgterm - デバッグトレース機能を終了する

### 書式

```
didbgterm
didbgterm -h
```

### 説明

ログ情報ファイル出力デーモン、障害情報ファイル出力デーモンが起動中の場合は停止する。この時、保存領域に格納されているログ情報や障害情報はログ情報ファイルや障害情報ファイルに出力される。その後、デバッグトレース機能の保存領域を解放する。

### オプション

-h

本コマンドの使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 関連

didbginit

## 2. 1. 34 didlrchg (AP 動的置換)

### 名前

didlrchg - AP 動的置換機能を利用して呼び出している関数を置換する

### 書式

didlrchg [-a|-F|-l {LogicalLibraryName}...] [-E] [-h]

### 説明

**didlrchg** は AP 動的置換機能を利用して関数呼び出しをおこなっているプロセスに対して、ライブラリが置き換わったことを通知する。環境定義 (APLIB 節) を記述して関数呼び出しをおこなっている場合、置換対象を **-a**, **-F**, **-l** のいずれかのオプションで指定する。

環境変数を記述して関数呼び出しをおこなっている場合、**-E** オプションを指定することで、ロード済みのライブラリを再ロードさせることができる。

### オプション

#### -a

環境定義の変更を通知する。(パラメータ省略時の既定値)

LM 項とその配下の論理ライブラリがまったく更新されていない LM についてはライブラリの再ロードはおこなわない。

#### -F

環境定義に定義されたすべてのライブラリ再ロードを通知する。

変更されていない共有ライブラリも再ロードされる。

ライブラリ間に関数名または外部変数の依存関係がある場合に使用する。

#### -l LogicalLibraryName

**-a** オプションの動作に加え、**LogicalLibraryName** で指定した論理ライブラリについては、定義が変更されていなくても再ロード対象とする。論理ライブラリはスペース区切りで複数指定可能である。

コマンド実行時に定義が変更されているものについては、-l の指定に関わらず再ロード対象となる。

#### -E

環境変数 DIOSA\_LIBNAME を使用して関数呼び出しをおこなっているプロセスに対して、DIOSA\_LIBNAME によりロードされた全ライブラリのアンロードを指示する。

指定がない場合、DIOSA\_LIBNAME によりロードされたライブラリのアンロードはおこなわれない。

#### -h

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了
2	引き継ぎファイル作成エラー

### 関連

didlrinit

## 2.1.35 didlrinit (AP 動的置換機能初期化コマンド)

### 名前

didlrinit - AP 動的置換機能を初期化する

### 書式

didlrinit [ -c | -w ] [-h]

### 説明

AP 動的置換機能を初期化し、機能利用可能にする。**-c**、**-w** でスタートモードを指定する。

### オプション

**-c**

APLIB 節の情報をもとに、AP 動的置換機能を初期化する。

**-w**

前回動作時の情報を引き継いで AP 動的置換機能を初期化する。(既定値)

**-h**

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了
2	引き継ぎファイル作成エラー

### 注意

- ウォームスタートが指定されたが、引き継ぎファイルが存在しない場合、コールドスタートする。
- 共有ライブラリ置換コマンド等の実行前に本コマンドが実行されている必要がある。

### 関連

distart\_com, distop

## 2.1.36 didlrreflib (論理ライブラリ定義参照コマンド)

### 名前

didlrreflib - APLIB 節 LLIB 項の定義を参照する

### 書式

didlrreflib [-s] [-l {論理ライブラリ名}...] [-h]

### 説明

APLIB 節 LLIB 項の定義を画面に表示する。

### オプション

-s

関数定義を出力しない。

-l

指定された論理ライブラリの情報のみを出力する。  
指定しない場合、全てのライブラリ情報を表示する。

-h

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了
2	指定対象が存在しない

### 関連

didlrinit, didlrreflm

### 実行例

```
+===== LLIB DEFINITION DISPLAY =====+ 2012/03/27 00:00:00 ①
LLIB: lib_1 ②
LOAD: INIT ③ UNLOAD: NO ④
LIB: /home/user/lib/libtest1.so ⑤
REV: 1 ⑥
  FUNC: function1 ⑦ FREV: 1 ⑧
  FUNC: function1   FREV: 1
      :
LLIB: lib_2
      :
+===== END OF DISPLAY =====+
```

- ① コマンド実行時刻
- ② 論理ライブラリ名
- ③ ロードタイミング
  - INIT : 起動時ロード
  - DEFERRED : 呼び出し時ロード
- ④ アンロードタイミング
  - NO : アンロードしない
  - TRNS : トランザクション終了時アンロード
- ⑤ ライブラリ名
- ⑥ ライブラリリビジョン
- (-s オプションを指定した場合、以下の情報は出力されない)
- ⑦ 関数名
- ⑧ 関数リビジョン

## 2.1.37 didlrreflm (LM 定義参照コマンド)

### 名前

didlrreflm - APLIB 節 LM 項の定義を参照する。

### 書式

didlrreflm [-d] [-l {LM 名}...] [-h]

### 説明

APLIB 節 LM 項の定義を画面に表示する。

### オプション

-d

APLIB 節 DFLTLLIB 項の情報を出力する。

-l

指定された LM の情報のみを出力する。

指定しなかった場合、DFLTLLIB 項、および全ての LM 項の情報が出力される。

-h

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了
2	指定対象が存在しない

### 関連項目

didlrinit, didlrreflib

### 実行例

#### DFLTLLIB 項表示 (-d)

```
+===== LM DEFINITION DISPLAY =====+ 2012/03/27 00:00:00 ①
LM: **DEFAULT** ②
LOADMODE: NO ③
LLIB: llib_common1 ④
LLIB: llib_common2

LM: lm1
LOADMODE: NO
LLIB: llib_common1
LLIB: llib_common2
LLIB: llib_11
LLIB: llib_12

LM: lm2
LOADMODE: NO
LLIB: llib_common1
LLIB: llib_common2
LLIB: llib_21
:
+===== END OF DISPLAY =====+
```

- ① コマンド実行時刻
- ② LM 名
- ③ ロード時のシンボル解決有無
  - YES : シンボル解決あり
  - NO : シンボル解決なし
- ④ 論理ライブラリ名

## 2.1.38 dietgctrl (経過時間監視デーモン起動停止コマンド)

### 名前

`dietgctrl` -経過時間を監視するデーモンを操作する。

### 書式

```
dietgctrl [-b] [-t INTERVAL_TIME]
dietgctrl -e
dietgctrl -h
```

### 説明

`dietgctrl -b` は、経過時間を監視するデーモンを起動する。

`dietgctrl -e` は、経過時間を監視するデーモンを停止する。

### オプション

`-b`

経過時間を監視するデーモンを起動する。(既定値)

`-e`

経過時間を監視するデーモンを停止する。

`-t INTERVAL_TIME`

デーモンが監視する間隔時間を秒単位で設定する。(数字: 1～int の最大値、省略可)

省略した場合は、10 秒が設定される。

`-h`

使用方法を表示する。

### 戻り値

`dietgctrl` は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

### 注意

監視間隔時間に設定された時間よりサービスの経過監視時間が短い場合、警告メッセージが出力されない場合がある。

### 関連

`dietgref`

## 2. 1. 39 dietgrep (経過時間監視照会コマンド)

### 名前

dietgrep -経過時間情報を照会する。

### 書式

```
dietgrep [-p PROCESS_ID] [-s SERVICE_NAME] [-u USER_INFORMATION]
dietgrep -h
```

### 説明

現在登録されている経過時間情報を照会する。

### オプション

#### -p PROCESS\_ID

照会する経過時間情報のプロセス ID を指定する。(省略可)

省略した場合は、全てのプロセスの経過時間情報を照会対象とする。

#### -s SERVICE\_NAME

照会する経過時間情報の識別子を指定する。(英数字：1～31 バイト、省略可)

省略した場合は、全ての識別子の経過時間情報を照会対象とする。

#### -u USER\_INFORMATION

照会する経過時間情報のユーザ情報を指定する。(英数字：1～50 バイト、省略可)

省略した場合は、全てのユーザ情報の経過時間情報を照会対象とする。

#### -h

使用方法を表示する。

### カラムの説明

以下に、dietgrep が表示するカラムの見出しと各カラムの意味を説明する。

共通情報は、"+----- COMMON INFORMATION -----+"の表示後、各カラムが以下の順番に従って左から右へプリントされる。

DAEMON PID	起動中の経過時間監視デーモンのプロセス ID
INTERVAL	経過時間監視デーモンの監視間隔時間
MAX-ENTRY	経過時間監視デーモンが管理できる最大登録件数

経過時間情報は、"+----- ENTRY INFORMATION -----+"の表示後、各カラムが以下の順番に従って左から右へプリントされる。

SERVICE-NAME	照会した経過時間情報の識別子(最大 3 1 文字)
PID/TID	照会した経過時間情報のプロセス ID / スレッド ID
STATUS	照会した経過時間情報の状態
WATCH	: 監視中
HOLD	: 監視一時停止
END	: 監視終了

	NO WATCH : 監視対象外
START-TIME	識別子のアプリケーションが開始した時刻
ELAPS	識別子のアプリケーションが開始から現在までの時間
WARN-TIME	識別子のアプリケーションが警告を発する時刻
USER-INFO	diosaetgusinfo() でアプリケーションが登録したユーザ情報文字列

## 戻り値

dietgref は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了

## 関連

dietgctrl

## 2.1.40 diirmadd (SG オブジェクト生成コマンド)

### 名前

diirmadd - SG オブジェクトを新規に生成する。

### 書式

```
diirmadd -E EDL ソースファイル名 ... [-o 実行結果出力ファイル名]  
diirmadd -h
```

### 説明

SG オブジェクトを新規に生成する。

### オプション

#### -E

SG オブジェクト生成時の入力となる EDL ソースファイルを指定する。(1~255 バイト、必須、複数指定可)

絶対/相対パスで指定した場合、そのパスでファイル検索する。

ファイル名で指定した場合、環境変数 "DISOA\_EDLSRCDIR" 配下、コマンド実行時のカレントディレクトリ配下の順にファイル検索する。

EDL ソースファイルを複数指定する場合、各ファイルを半角スペース、またはカンマ区切りで指定する。

#### -o

実行結果を出力するファイルを指定する。(1~255 バイト)

省略した場合、コマンド実行時のカレントディレクトリに "irmlist" という名前のファイルに実行結果を出力する。

#### -h

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 注意

環境変数 DIOSA\_IRMROOT, DIOSA\_LNODENAME の設定が必須である。

### 関連

diirmrep, diirmdel, diirmref

## 2.1.41 diirmdel (SG オブジェクト削除コマンド)

### 名前

diirmdel - SG オブジェクトを削除する。

### 書式

diirmdel [-S セクション名 ...] [-o 実行結果出力ファイル名]

diirmdel -h

### 説明

SG オブジェクトを削除する。

### オプション

-S

削除する SG オブジェクトのセクション名(節名)を指定する。(1~12 バイト、省略/複数指定可)

省略した場合、登録中の全 SG オブジェクトを削除する。

セクションを複数指定する場合、各ファイルを半角スペース、またはカンマ区切りで指定する。

-o

実行結果を出力するファイルを指定する。(1~255 バイト、省略可)

省略した場合、コマンド実行時のカレントディレクトリに"irmlist"という名前のファイルに実行結果を出力する。

-h

使用方法を表示する。

### 戻り値

0        正常終了

1        異常終了

### 注意

環境変数 DIOSA\_IRMROOT の設定が必須である。

### 関連

diirmadd, diirmrep, diirmref

## 2.1.42 diirmref (SG オブジェクト参照コマンド)

### 名前

diirmref - SG オブジェクトに関する情報を参照する。

### 書式

```
diirmref [-s] [-S セクション名 ...] [-o 実行結果出力ファイル名]
diirmref -e -S セクション名 ... [-o 実行結果出力ファイル名]
diirmref -t [-S セクション名 ...] [-o 実行結果出力ファイル名]
diirmref -h
```

### 説明

SG オブジェクトに関する情報を参照する。

### オプション

**-s**

セクションの詳細情報を出力する。

**-e**

SG オブジェクトの入力 EDL ソースイメージを出力する。

このオプションを指定する場合、-S オプションは必須、  
またセクション名は複数指定できない。

**-t**

各システムに対して生成されたセクションの更新履歴を出力する。

**-S**

参照する SG オブジェクトのセクション名(節名)を指定する。(1 つにつき 1~12 バイト)

省略した場合、登録中の全 SG オブジェクトを参照する。

セクションを複数指定する場合、各ファイルを半角スペース、またはカンマ区切りで指定する。

**-o**

実行結果を出力するファイルを指定する。(1~255 バイト)

省略した場合、コマンド実行時のカレントディレクトリに"irmlist"という名前のファイルに実行結果を  
出力する。

**-h**

使用方法を表示する。

### 戻り値

0                      正常終了

1                      異常終了

### 注意事項

環境変数 DIOSA\_IRMROOT の設定が必須である。

### 関連項目

diirmadd, diirmrep, diirmdel

## 実行例

### 環境定義セクション名指定参照(diirmref -S)

```
***** diirmref 15/05/18 17:35:10 *****
DIOSA_IRMROOT=PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP ①
+
LSNAME=HOST          YY/MM/DD hh:mm:ss ②
<secname>    <usrid>  <rev>  <date>  <time>  <mode> ③
SSSSSSSS    UUUUUUU  RRR  YY/MM/DD  hh:mm:ss  jxu
④          ⑤          ⑥  ⑦          ⑧          ⑨
+
***** diirmref STATUS=DONE *****
```

- ①環境変数 DIOSA\_IRMROOT 値
- ②更新日時
- ③セクションヘッダ情報
- ④セクション名
- ⑤ユーザ ID
- ⑥オブジェクトリビジョン
- ⑦更新日付
- ⑧更新時間
- ⑨基盤システム種別

### 環境定義 EDL ソースイメージ参照(diirmref -e -S)

```
/*****
**** < EDL Source Image > --- diirmref ---
****
**** LSNAME=HOST, USER=uuuu ①
**** REV=RRRRR ②, BASESYS=DIOSA, MACHINE=UNIX
**** EDL=ffffffffffffffffffffffffffffffff ③
****/
④
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

- ①ユーザ名
- ②オブジェクトリビジョン
- ③EDL ソースファイル名
- ④EDL ソースイメージ : EDL ソースファイル名の内容を表示する

## 環境定義全セクション更新履歴参照(diirmref -t)

[illegible]

- ①環境変数 DIOSA\_IRMROOT 値
- ②セクション名
- ③セクションヘッダ情報
- ④更新番号
- ⑤更新種別      登録：ADD  
更新：REPLACE
- ⑥ユーザ名
- ⑦オブジェクトリビジョン
- ⑧更新日付
- ⑨更新時間
- ⑩ファイルサイズ
- ⑪レコード数
- ⑫MAX レコード長

## 2.1.43 diirmrep (SG オブジェクト更新コマンド)

### 名前

diirmrep - SG オブジェクトを更新する。

### 書式

diirmrep -E EDL ソースファイル名 ... [-o 実行結果出力ファイル名]

diirmrep -h

### 説明

SG オブジェクトを更新する。

### オプション

-E

SG オブジェクト更新時の入力となる EDL ソースファイルを指定する。(1～255 バイト、複数指定可)

絶対/相対パスで指定した場合、そのパスでファイル検索する。

ファイル名で指定した場合、環境変数 "DISOA\_EDLSRCDIR" 配下、コマンド実行時のカレントディレクトリ配下の順にファイル検索する。

EDL ソースファイルを複数指定する場合、各ファイルを半角スペース、またはカンマ区切りで指定する。

-o

実行結果を出力するファイルを指定する。(1～255 バイト)

省略した場合、コマンド実行時のカレントディレクトリに"irmlist"という名前のファイルに実行結果を出力する。

-h

使用方法を表示する。

### 戻り値

0                      正常終了

1                      異常終了

### 注意

環境変数 DIOSA\_IRMROOT, DIOSA\_LNODENAME の設定が必須である。

### 関連

diirmadd, diirmdel, diirmref

## 2.1.44 dimdrctl (SG 変更通知制御コマンド)

### 名前

dimdrctl - SG 変更通知の送信を行う。

### 書式

```
dimdrctl -n
dimdrctl -a [-d]
dimdrctl -d [-a]
dimdrctl -h
```

### 説明

SG 変更通知の送信を行う。

### オプション

**-n**

SG 変更通知を送信する。

SG 変更通知デーモンの異常終了により、各機能の SG 変更コマンドが失敗した場合に、SG 変更通知デーモンの再起動した後、本コマンドにより通知を再送する。

**-a**

SG 変更先アプリケーションの一覧を照会する。

**-d**

SG 変更通知デーモンの一覧を照会する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。
2	処理が警告終了した。

## 2. 1. 45      **dimemchg (SG 変更コマンド)**

### 名前

dimemchg - メモリ管理機能の SG を変更する

### 書式

dimemchg

### 説明

dimemchg コマンドはメモリ管理機能の SG を変更する。

### オプション

なし

### 戻り値

dimemchg は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

### 注意

共有メモリの拡張サイズの変更、最大サイズの縮小、及び新規追加は動的に変更できない。

## 2.1.46 dimemfree(メモリー一括解放コマンド)

### 名前

dimemfree - 共有メモリを解放する

### 書式

```
dimemfree
dimemfree -h
```

### 説明

メモリ割り当て機能及び再割り当て機能にて割り当てた共有メモリを一括解放する。

### オプション

-h

使用法を表示する。

### 戻り値

dimemfree は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

### 注意

業務処理を実行中の場合でも、本コマンドを実行すると共有メモリが解放される。

## 2.1.47 dimemref(メモリ状態表示コマンド)

### 名前

dimemref - 共有メモリの状態およびアボートダンプファイルの内容を編集出力する

### 書式

```
dimemref [-l | -d] [-i メモリ識別子] [-c メモリ種別] [-e エントリキー]
          [-f アボートダンプファイル名]
dimemref -h
```

### 説明

メモリ管理機能配下の共有メモリの状態およびアボートダンプファイルの内容を編集出力する。プロセスメモリ（“-c” オプションで “pp|tp|sp”）を指定する場合、“-f” オプションと同時に指定する必要がある。アボートダンプファイル名は、フルパスまたはコマンド実行時のカレントディレクトリからの相対パス名で指定する。

### オプション

-l

一覧表示する。（既定値）

-d

ダンプ表示する。

-i

編集出力を行うメモリのメモリ識別子を指定する。（省略可）  
省略した場合は、全てのメモリ識別子を対象とする。

-c

編集出力を行うメモリのメモリ種別を指定する。（省略可）  
省略した場合は、全てのメモリ種別を対象とする。  
メモリ種別に指定できるのは以下のものがある。

aw      更新可共有メモリ

ar      保護属性共有メモリ

pp      一括解放対象外プロセスメモリ（一括解放対象外プロセス内メモリ）

tp      一括解放対象外プロセスメモリ（一括解放対象外スレッド内メモリ）

sp      一括解放対象プロセスメモリ（一括解放対象サービス内メモリ）

-e

抽出するデータのエントリキーを指定する。（省略可）  
省略した場合は、全てのエントリキーを対象とする。

-f

アボートダンプファイルからの出力を行いたい場合に、編集出力する対象のアボートダンプファイル名を指定する。（省略可）  
省略した場合、編集出力する対象がメモリになる。

-h

使用法を表示する。

### 戻り値

dimemref は、以下のいずれかの終了ステータスを戻して終了する。

- 0     正常終了
- 1     異常終了

#### 注意

照会コマンドでは排他を行っていないため、下記 API で更新中に照会を行った場合は、更新途中のデータが表示される場合がある。

※DIOSMALLOC、DIOSREALLOC、DIOSMFREE (IDL2-API)、  
diosamalloc、diosarealloc、diosafree (C-API)

実行例

メモリ状態一覧表示(dimemref -l)

```
+===== MEMORY DUMP DISPLAY =====+ DATE:2012/10/18   TIME:19:00:00
LSYS:XXXXXXXX ①
LNODE:XXXXXXXX ②
CLASS:XXXXXXXXX ③
==USED == ④
ID:XXXXX EKEY:XXXX SIZE:999999999 ADDRESS:0XXXXXXXXX/0XXXXXXXXX
⑤          ⑥          ⑦          ⑧
:           :           :           :
ID:XXXXX EKEY:XXXX SIZE:999999999 ADDRESS:0XXXXXXXXX/0XXXXXXXXX [ This block is crushed]⑨
=== FREE === ⑩
[0] SIZE:1056⑪ [ This block is crushed]⑫
+===== END OF DISPLAY =====+
```

- ①LSYS : 論理システム名
- ②LNODE : 論理ノード名
- ③CLASS : メモリ種別名  
APNOPROT : 更新可共有メモリ  
APPROT : 保護属性共有メモリ  
PRCNOALL : プロセスメモリ (-f 指定のみ出力される)  
THRNOALL : スレッドメモリ (-f 指定のみ出力される)  
SVCALL : サービスメモリ (-f 指定のみ出力される)
- ④==USED== : 使用中ブロックのヘッダ
- ⑤ID : メモリ識別子
- ⑥EKEY : エントリキー名
- ⑦SIZE : 使用中ブロックユーザ要求サイズ
- ⑧ADDRESS : ブロック開始アドレス／終了アドレス
- ⑨This block is crushed : 破壊されているブロックを示す
- ⑩==FREE== : 未使用ブロックのヘッダ
- ⑪SIZE : 未使用ブロックサイズ
- ⑫This block is crushed : 破壊されているブロックを示す

## メモリ状態ダンプ表示(dimemref -d)

```

+===== MEMORY DUMP DISPLAY =====+ DATE:2012/10/18   TIME:19:00:00
LSYS:XXXXXXXX ①
LNODE:XXXXXXXX ②
CLASS:XXXXXXXXX ③
==USED == ④
ID:XXXX EKEY:XXXX SIZE:999999999 ADDRESS:0XXXXXXXX/0XXXXXXXX [ This block is crushed]
⑤          ⑥          ⑦          ⑧          ⑨
ADDRESS   : +0   +2   +4   +6   +8   +a   +b   +e   : 0123456789abcdef
0XXXXXXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXXXXXXXXXXXXXXXXXX
⑩          ⑪          ⑫
:           :           :
0XXXXXXXX :XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXXXXXXXXXXXXXXXXXX
0XXXXXXXX :XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXXXXXXXXXXXXXXXXXX
=== FREE === ⑬
[0] SIZE:1056⑭ [ This block is crushed]⑮
+===== END OF DISPLAY =====+

```

- ①LSYS : 論理システム名
- ②LNODE : 論理ノード名
- ③CLASS : メモリ種別名
  - APNOPROT : 更新可共有メモリ
  - APPROT : 保護属性共有メモリ
  - PRCNOALL : プロセスメモリ (-f 指定時のみ出力される)
  - THRNOALL : スレッドメモリ (-f 指定時のみ出力される)
  - SVCALL : サービスメモリ (-f 指定時のみ出力される)
- ④==USED== : 使用中ブロックのヘッダ
- ⑤ID : メモリ識別子
- ⑥EKEY : エントリキー名
- ⑦SIZE : 使用中ブロックユーザ要求サイズ
- ⑧ADDRESS : ブロック開始アドレス／終了アドレス
- ⑨This block is crushed : 破壊されているブロックを示す
- ⑩ADDRESS : 表示開始アドレス
- ⑪ : メモリの HEX ダンプ
- ⑫ : メモリのキャラクタダンプ
- ⑬==FREE== : 未使用ブロックのヘッダ
- ⑭SIZE : 未使用ブロックサイズ
- ⑮This block is crushed : 破壊されているブロックを示す

## 2.1.48 dimsgdctrl (メッセージ出力デーモン起動停止コマンド)

### 名前

dimsgdctrl - メッセージ出力デーモンの起動、停止を行う。

### 書式

```
dimsgdctrl -b
dimsgdctrl -e
dimsgdctrl -h
```

### 説明

メッセージ出力デーモンの起動、停止を行う。

### オプション

**-b**

メッセージ出力デーモンを起動する。

**-e**

メッセージ出力デーモンを停止する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。
2	処理が警告終了した。

### 注意

メッセージ出力デーモンの起動指定時には必ず、環境変数 "DIOSA\_MSG\_LOG\_PATH" を設定する。設定されていない場合、メッセージ出力デーモンは起動しない。

本コマンドは、AP ノード、OLTP ノード、DB ノードに投入可能です。

### 関連

dimsglvref, dimsglvmod, dimsglimit, dimsgmtn, dimsgconv  
DIOSA\_MSG\_LOG\_PATH

## 2.1.49 dimsgdisp(メッセージ出力コマンド)

### 名前

dimsgdisp - 指定したメッセージをメッセージログファイルへ出力する。

### 書式

```
dimsgdisp [-p ProcName] [-P PID] [-f] MessageID [String ...]
```

```
dimsgdisp -h
```

### 説明

**MessageID** 指定したメッセージをメッセージログファイルへ出力する。  
メッセージ中に埋め込む置換文字列を **String** で指定することができる。

### オプション

**-p ProcName**

メッセージに出力するプロセス名を指定する。

**-P PID**

メッセージに出力するプロセスのプロセス ID を指定する。

**-f**

DIOSA の起動状態に関係なく、強制的にメッセージ出力を行う。

**MessageID**

出力するメッセージのメッセージ ID を指定する。

**String**

指定した置換テキスト数分、文字を置換する。

10 個まで指定可能。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。
2	処理が警告終了した。

### 注意

コマンドパラメータは、オプション、メッセージ ID、置換文字列 の順に指定すること。

オプションは順不同 (**-p -P**、**-P -p** どちらでも) で指定できる。

**-p**、**-P** を同時に指定した場合、指定したプロセス名/ID でメッセージ出力を行う。(このとき、プロセス ID に対応したプロセス名とは異なるプロセス名を **-p** で指定した場合でもエラーにはならない。)

**-p**、**-P** を省略した場合、コマンド自身のプロセス名/ID をメッセージに出力する。

**-f** を省略した場合、DIOSA 環境情報を取得するため、DIOSA のプロセス初期化処理を行う。

**-f** 指定した場合、DIOSA のプロセス初期化処理の代わりに、環境変数からメッセージ出力に必要な DIOSA 環境情報を取得する。(ただし、環境変数から取得できない論理システム名などの情報はメッセージには出力されない。)

本コマンドでメッセージを出力した場合、表示されるプロセス情報中のスレッド ID は固定的に 1 になる。

置換テキスト数が 11 以上指定された場合は異常終了する。

置換テキストが 1024 バイト ('¥0' 含む) を超えた場合、1024 バイト以降は表示されない。

## 関連

dimsgdctrl, dimsglvref, dimsglvmod, dimsglimit, dimsgmtn, dimsgconv

DIOSA\_MSG\_LOG\_PATH

## 2.1.50 dimsglimit (メッセージ出力抑止コマンド)

### 名前

dimsglimit - 指定したメッセージ ID の出力抑止の登録・解除、登録されている ID の一覧表示を行う。

### 書式

```
dimsglimit -a MessageID
dimsglimit -d MessageID
dimsglimit -c
dimsglimit -v
dimsglimit -h
```

### 説明

指定したメッセージ ID の出力抑止の登録・解除、登録されている ID の一覧表示を行う。

### オプション

#### -a MessageID

指定したメッセージ ID のメッセージ出力を抑止する。

**MessageID** の後部分を省略することで前方一致する複数のメッセージを抑止することができる。

#### -d MessageID

指定したメッセージ ID の出力抑止登録を解除する。

#### -c

出力抑止登録されている全てのメッセージ ID を解除する。

#### -v

出力抑止登録されているメッセージ ID を表示する。

#### -h

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。
2	処理が警告終了した。

### 注意

メッセージ ID に指定できる文字は印字可能文字となる。

### 関連

dimsgdctrl, dimsglvref, dimsglvmod, dimsgmtn, dimsgconv

### 実行例

#### 登録(dimsglimit -a DIXXX001)

```
+===== dimsglimit DISPLAY =====+ 2015/04/03 10:10:56
①
DIXXX000
DIXXX001
2 entries. ②
+===== END OF DISPLAY =====+
```

①出力抑止対象として登録されているメッセージ ID を一覧で表示する

②出力抑止対象登録メッセージ ID 数

### 解除(dimsglimit -d DIXXX001)

```
+===== dimsglimit DISPLAY =====+ 2015/04/03 10:12:38
DIXXX000 ①
1 entries. ②
+===== END OF DISPLAY =====+
```

- ①出力抑止対象として登録されているメッセージ ID を一覧で表示する
- ②出力抑止対象登録メッセージ ID 数

### 全解除(dimsglimit -c)

```
+===== dimsglimit DISPLAY =====+ 2015/04/03 10:13:22
0 entries.
+===== END OF DISPLAY =====+
```

### 一覧表示(dimsglimit -v)

```
+===== dimsglimit DISPLAY =====+ 2015/04/03 10:14:41
①
DIXXX001
DIXXX002
DIXXX003
DIYXX002
DIYXX003
5 entries. ②
+===== END OF DISPLAY =====+
```

- ①出力抑止対象として登録されているメッセージ ID を一覧で表示する
- ②出力抑止対象登録メッセージ ID 数

## 2.1.51 dimsglvmod (メッセージ出力レベル変更コマンド)

### 名前

dimsglvmod - メッセージの出力基準を変更する。

### 書式

```
dimsglvmod [-l Msg_Level] [-t Omit_Time] [-c Omit_Count]  
dimsglvmod -h
```

### 説明

出力レベル基準値(**Msg\_Level**)、同一メッセージ出力抑止時間(**Omit\_Time**)、同時抑止可能メッセージ ID 数(**Omit\_Count**)の変更を行う。

### オプション

**-l Msg\_Level**

メッセージの出力レベル基準値を指定する。(1～5)

レベル基準値で指定した数値以下のレベルを持つメッセージが出力される。

**-t Omit\_Time**

同一メッセージの出力を抑止する秒数を設定する。(0～3600)

**-c Omit\_Count**

抑止中 ID を保持する配列の要素数を設定する。(0～32)

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

- |   |            |
|---|------------|
| 0 | 処理が正常終了した。 |
| 1 | 処理が異常終了した。 |
| 2 | 処理が警告終了した。 |

### 注意

抑止時間、チェック数ともに 0 以外に設定したとき、抑止時間中に同一メッセージの出力があった場合、2 つ目以降のメッセージの出力は抑止される。

同時に抑止できるメッセージの種類は、チェック数で指定した数までとなる。

### 関連

dimsgdctrl, dimsglvref, dimsglimit, dimsgmtn, dimsgconv  
DIOSA\_MSG\_LEVEL, DIOSA\_MSG\_OMIT\_INF

## 2.1.52 dimsglvref (メッセージ出力レベル照会コマンド)

### 名前

dimsglvref - 出力レベル基準値の照会を行う。

### 書式

```
dimsglvref
dimsglvref -h
```

### 説明

出力レベル基準値(MSG\_LEVEL)、同一メッセージ出力抑止時間(OMIT\_TIME)、同時抑止可能メッセージ ID 数(OMIT\_COUNT)の照会を行う。

### オプション

-h  
ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。
2	処理が警告終了した。

### 注意

出力レベル基準値以下のレベルを持つメッセージが出力される。  
抑止時間、チェック数ともに 0 以外るとき、抑止時間中に同一メッセージの出力があった場合、2 つ目以降のメッセージの出力は抑止される。  
同時に抑止できるメッセージの種類は、チェック数で指定した数までとなる。

### 関連

dimsgdctrl, dimsglvmod, dimsglimit, dimsgmtn, dimsgconv  
DIOSA\_MSG\_LEVEL, DIOSA\_MSG\_OMIT\_INF

### 実行例

#### 照会(オプションなし)

```
+===== dimsglvref DISPLAY =====+ 2015/04/03 10:18:38
Output level standard value = 5          ①
The same message control time = 0       ②
Number of checks that control the same message = 0  ③
+===== END OF DISPLAY =====+
```

- ①出力レベル基準値
- ②同一メッセージ抑止時間(秒)
- ③同一メッセージ抑止チェック数

## 2.1.53 dimsgmtn (原型メッセージファイルメンテナンスコマンド)

### 名前

dimsgmtn - 原型メッセージファイルを作成、出力する。

### 書式

```
dimsgmtn -i InFile
dimsgmtn -o OutFile
dimsgmtn -h
```

### 説明

原型メッセージファイルを更新する場合、**-o** で現状の原型メッセージをファイルに取得後、そのファイルを編集し、**-i** でファイルを取り込む。

メッセージ一覧ファイルは、以下形式で作成する。

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

メッセージ ID は 8 桁以下の印字可能文字で、ファイル内でユニークになるよう指定する。なお、“DI”又は“DA”で始まるメッセージ ID については DIOSA で予約されているため使用することはできない。

メッセージレベルは 1~5 を指定する。この値が出力レベル基準値以下の場合にメッセージ出力される。

メッセージ種別は E(異常), W(警告), I(通知)のいずれかを指定する。

英語メッセージ本文には、ASCII 文字のみを使用しメッセージ本文を指定する。置換文字列を埋め込む場所には@(半角)を使用する。メッセージ本文は 500 バイト以内で記述する必要があるが、@(半角)と%(半角)は 2 バイトとして計算する。

日本語文字列には、SJIS のメッセージ本文を指定する。置換文字列を埋め込む場所には@(全角)を使用する。メッセージ本文は 500 バイト以内で記述する必要があるが、%(半角)は 2 バイトとして計算する。

### オプション

**-i InFile**

入力メッセージ一覧ファイルから原型メッセージファイルを作成する。

**-o OutFile**

原型メッセージファイルの内容を出力メッセージ一覧ファイルに出力する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。
2	処理が警告終了した。

### 関連

dimsgdctrl, dimsglvref, dimsglvmod, dimsglimit, dimsgconv  
DIOSA\_MSG\_FORM\_FILEPATH

## 2. 1. 54      diopschg (稼動統計環境変更コマンド)

### 名前

diopschg - 稼動統計の環境変更を行う

### 書式

diopschg  
diopschg -h

### 説明

稼動統計デーモンへの変更通知を行い、稼動統計機能が使用する環境定義情報を読み込み、共有メモリ上へ展開する。

### オプション

-h

使用方法を表示する。

### 戻り値

diopschg は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

## 2.1.55 diopsctrl (稼動統計デーモン起動・停止コマンド)

### 名前

diopsctrl - 稼動統計デーモンの起動・停止を行う。

### 書式

```
diopsctrl -b
diopsctrl -e
diopsctrl -h
```

### 説明

C O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、停止を行う。

### オプション

-b

稼動統計デーモンを起動する。

-e

稼動統計デーモンを停止する。

-h

使用方法を表示する。

### 戻り値

diopsctrl は、以下のいずれかの終了ステータスを戻して終了する。

- 0 正常終了
- 1 異常終了
- 2 警告終了

### 注意

環境定義の OPSDIR に指定されたディレクトリパスが存在しない場合、稼動統計デーモンは起動しない。

### 関連

diopsflush、diopschg

## 2. 1. 56 diopsedit (稼動統計データ編集コマンド)

### 名前

diopsedit - 稼動統計マージファイルを編集する

### 書式

```
diopsedit OPS_FILENAME [-A TARGET] [-R TYPE] [-k CO_NAME...] [-N NODE_TYPE] [-n NODE_NAME...]
        [-i NODE_ID] [-p PROCESS_ID] [-P PROCESS_SEQUENCE_NUMBER]
        [-t TRANSACTION_ID...] [-m TPBASE_MONITER_NAME...]
        [-s START_TIME] [-e END_TIME] [-d] [-T] [-o OUTPUT_FILE]
diopsedit -h
```

### 説明

稼動統計マージ機能の出力ファイルを、利用者が参照および加工しやすい形式に変換し、C S V形式で出力する。

### オプション

#### OPS\_FILENAME

編集元の稼動統計ファイル名を指定する。

絶対パスあるいは環境定義にて指定された収集ディレクトリ (GATHERDIR) からの相対パスを含めることも可能。

パスが未指定の場合には環境定義の収集ディレクトリ (GATHERDIR) 配下と見なされる。

#### -A TARGET

出力対象とする種別(オンラインあるいはバッチ)を指定する。省略した場合は、両方が出力される。

O        オンラインのみを出力対象とする。

B        バッチのみを出力対象とする。

#### -R TYPE

出力対象とする稼動統計情報種別を指定する。省略した場合は、両方が出力される。

co       CO の稼動統計情報のみを出力対象とする。

tx       トランザクションの稼動統計情報のみを出力対象とする。

#### -k CO\_NAME

出力対象とする CO 関数名を指定する。

30 文字以内、最大 10 個、カンマ区切りで指定する。

#### -N NODE\_TYPE

出力対象とする論理ノード種別を指定する。省略した場合は、両方が出力される。

oltp     OLTP ノードの稼動統計情報のみを出力対象とする。

ap       AP ノードの稼動統計情報のみを出力対象とする。

#### -n NODE\_NAME

出力対象とする論理ノード名を指定する。

15 文字以内、最大 10 個、カンマ区切りで指定する。

#### -i NODE\_ID

出力対象とする論理ノード ID を指定する。

#### -p PROCESS\_ID

出力対象とするプロセス ID を指定する。

**-P PROCESS\_SEQUENCE\_NUMBER**

出力対象とするプロセス内通番を指定する。

**-t TRANSACTION\_ID**

出力対象とするトランザクション ID を指定する。  
6 文字以内、最大 10 個、カンマ区切りで指定する。

**-m TPBASE\_MONITER\_NAME**

出力対象とする TPBASE モニタ名を指定する。  
8 文字以内、最大 10 個、カンマ区切りで指定する。

**-s START\_TIME**

出力対象範囲の開始時刻を指定する。  
形式：yyyymmddhh24miss （時間は 24 時間形式）

**-e END\_TIME**

出力対象範囲の終了時刻を指定する。  
形式：yyyymmddhh24miss （時間は 24 時間形式）

**-d**

処理開始日付を出力する場合に指定する。  
省略時には処理開始時刻のみが出力される。

**-T**

編集結果の 1 件目に項目タイトル(項目名称)を出力する場合に指定する。

**-o OUTPUT\_FILE**

編集結果の出力ファイル名を指定する。  
絶対パスあるいは環境定義にて指定された収集ディレクトリ (GATHERDIR) からの相対パスを含めることも可能。  
パスが未指定の場合には環境定義の収集ディレクトリ (GATHERDIR) 配下と見なされる。  
省略時には標準出力となる。

**-h**

使用方法を表示する。

**戻り値**

diopsedit は、以下のいずれかの終了ステータスを戻して終了する。

- 0 正常終了
- 1 異常終了
- 2 警告終了

**編集結果**

編集された稼動統計情報は、各項目毎にカンマ区切りで出力される。  
編集出力される情報について、以下の表に示す。

[C0 の稼動統計情報]

No.	項目名	出力形式
1	種別	C0
2	オンライン／バッチ種別	0：C0 制御サーバ

		B：バッチ AP 制御
3	論理ノード名	文字列、最大 15 文字
4	論理ノード ID	10 進数値
5	プロセス ID	10 進数値
6	プロセス内通番 (トランザクション識別番号※1)	10 進数値
7	TPBASE トランザクション ID	文字列、最大 6 文字
8	TPBASE モニタ名	文字列、最大 8 文字
9	CO 関数名	文字列、最大 30 文字
10	電文処理開始時刻	YYYY/MM/DD HH:MI:SS. sss (YYYY/MM/DD はオプションで表示切替)
11	処理開始時刻	YYYY/MM/DD HH:MI:SS. sss (YYYY/MM/DD はオプションで表示切替)
12	処理時間	10 進数値、マイクロ秒単位(※2)
13	CPU 使用時間(SYSTEM)	10 進数値、マイクロ秒単位
14	CPU 使用時間(USER)	10 進数値、マイクロ秒単位
15	diosauca の状態コード	10 進数値
16	diosauca の利用者コード	10 進数値
17	ユーザ情報	バイナリ形式：100 文字で 16 進数表示 文字列形式：文字列、最大 50 文字
	...	以降出力なし

[トランザクション単位の稼動統計情報]

No.	項目名	出力形式
1	種別	UC：TX 情報(ユーザ明示コミット) DC：TX 情報(暗黙的コミット) UR：TX 情報(ユーザ明示ロールバック) DR：TX 情報(暗黙的ロールバック)
2	オンライン／バッチ種別	O：CO 制御サーバ B：バッチ AP 制御
3	論理ノード名	文字列、最大 15 文字
4	論理ノード ID	10 進数値
5	プロセス ID	10 進数値
6	プロセス内通番 (トランザクション識別番号※1)	10 進数値
7	TPBASE トランザクション ID	文字列、最大 6 文字
8	TPBASE モニタ名	文字列、最大 8 文字
9	CO 関数名	文字列、最大 30 文字
10	電文処理開始時刻	YYYY/MM/DD HH:MI:SS. sss (YYYY/MM/DD はオプションで表示切替)
11	処理開始時刻	YYYY/MM/DD HH:MI:SS. sss (YYYY/MM/DD はオプションで表示切替)
12	処理時間	10 進数値、マイクロ秒単位
13	CPU 使用時間(SYSTEM)	出力なし

14	CPU 使用時間 (USER)	出力なし
15	diosauca の状態コード	出力なし
16	diosauca の利用者コード	出力なし
17	ユーザ情報	出力なし
18	サービス通番	10 進数値
19	更新 MAPID	10 進数値
20	レコード読込 回数	10 進数値
21	レコード読込 平均処理時間	10 進数値、マイクロ秒単位
22	レコード読込 ブリッジ回数	10 進数値
23	レコード追加 回数	10 進数値
24	レコード追加 平均処理時間	10 進数値、マイクロ秒単位
25	レコード追加 ブリッジ回数	10 進数値
26	レコード更新 回数	10 進数値
27	レコード更新 平均処理時間	10 進数値、マイクロ秒単位
28	レコード更新 ブリッジ回数	10 進数値
29	レコード削除 回数	10 進数値
30	レコード削除 平均処理時間	10 進数値、マイクロ秒単位
31	レコード削除 ブリッジ回数	10 進数値
32	全レコード削除 回数	10 進数値
33	全レコード削除 平均処理時間	10 進数値、マイクロ秒単位
34	全レコード削除 ブリッジ回数	10 進数値
35	コミット/ロールバック 処理時間	10 進数値、マイクロ秒単位
36	コミット/ロールバック ブリッジ回数	10 進数値
37	ロック待ち 有無	WAIT : 回数が 1 以上 出力なし : 回数が 0
38	ロック待ち 回数	10 進数値
39	デッドロック 有無	DEADLOCK : 回数が 1 以上 出力なし : 回数が 0
40	デッドロック 回数	10 進数値
41	ロック情報 相手ノード ID	10 進数値
42	ロック情報 相手プロセス ID	10 進数値
43	ロック情報 相手スレッド ID	10 進数値
44	ロック情報 表 ID	10 進数値
45	ロック情報 レコードキー	64 文字で 16 進数表示
46	グループコミット SG 定義値	10 進数値
47	グループコミット数	10 進数値
48	グループコミット順序	10 進数値
49	グループコミット発生原因	出力なし : 通常コミット CNT : 個数 TIME : 時間 QUEUE : キュー

※1: 同一プロセス内でトランザクションの境界を識別するための情報。

diosacommmit (API) 実行後、トランザクション正常終了後、diosarollback (API) 実行後、トランザクション異常終了後にカウントアップされる。

※2: 各項目の処理時間において、1 マイクロ秒以下は切り捨てとする。

## 2.1.57 diopsflush(稼動統計フラッシュコマンド)

### 名前

diopsflush - 稼動統計情報のファイルフラッシュを行う

### 書式

```
diopsflush
diopsflush -h
```

### 説明

メモリテーブル中に残っている稼動情報をファイルへフラッシュさせるコマンドである。

### オプション

**-h**  
使用方法を表示する。

### 戻り値

diopsflush は、以下のいずれかの終了ステータスを戻して終了する。

- 0     正常終了
- 1     異常終了
- 2     警告終了

### 注意

環境定義の OPSDIR に指定されたディレクトリパスが存在しない場合、稼動統計デーモンは起動しない。

### 関連

diopsctrl

## 2. 1. 58 diopsfput (稼動統計転送コマンド)

### 名前

diopsfput - 稼動情報を指定ディレクトリにコピーする

### 書式

```
diopsfput [-d GATHER_DIRECTORY] [-m def[ault]]
diopsfput [-d GATHER_DIRECTORY] -m reg[ather]
diopsfput [-d GATHER_DIRECTORY] -m all
diopsfput [-d GATHER_DIRECTORY] -m cur[rent]
diopsfput -h
```

### 説明

自ノードの稼動情報ファイルを収集ディレクトリにコピーする。

### オプション

#### **-d GATHER\_DIRECTORY**

収集する稼動情報ファイルを格納するディレクトリを絶対パスあるいは環境定義にて指定されたディレクトリ (GATHERDIR) からの相対パスで指定する。  
省略時は環境定義にて指定されたディレクトリが使用される。

#### **-m def[ault]**

収集は、ファイル出力が完了し、未収集のファイルを対象とする。(既定値)

#### **-m reg[ather]**

収集は、デフォルトの収集対象に加え、収集済みのファイルを対象とする。

#### **-m all**

収集は、出力中のファイルも含め、未使用・マージ済みを除いた全てを対象とする。

#### **-m cur[rent]**

収集は、現在出力中のファイルを対象とする。

#### **-h**

使用方法を表示する。

### 戻り値

diopsfput は、以下のいずれかの終了ステータスを戻して終了する。

- 0     正常終了
- 1     異常終了
- 2     警告終了

### 注意

-d で指定するディレクトリ、または環境定義の GATHERDIR に指定するディレクトリは、環境定義の OPSDIR と同じディレクトリは指定できない。

## 2. 1. 59 diopsgather (稼動統計収集コマンド)

### 名前

diopsgather - 他ノードの稼動統計情報を収集する

### 書式

```
diopsgather [-n NODE_NAME] [-d GATHER_DIRECTORY] [-m def[ault]]
diopsgather [-n NODE_NAME] [-d GATHER_DIRECTORY] -m reg[ather]
diopsgather [-n NODE_NAME] [-d GATHER_DIRECTORY] -m all
diopsgather [-n NODE_NAME] [-d GATHER_DIRECTORY] -m cur[rent]
diopsgather -h
```

### 説明

各ノードで出力された稼動情報ファイルを収集するためのコマンドであり、論理システム内の各 OLTP ノードの稼動情報ファイルを環境定義で指定されたディレクトリあるいはコマンドオプションで指定したディレクトリに収集する。

### オプション

#### **-n NODE\_NAME**

収集する対象のノード名を指定する。

省略時は論理システム内の OLTP ノード全てが収集対象となる。

#### **-d GATHER\_DIRECTORY**

収集する稼動情報ファイルを格納するディレクトリを絶対パスあるいは環境定義にて指定されたディレクトリ (GATHERDIR) からの相対パスで指定する。

省略時は環境定義にて指定されたディレクトリが使用される。

#### **-m def[ault]**

収集は、ファイル出力が完了し、未収集のファイルのみを対象とする。(既定値)

#### **-m reg[ather]**

収集は、デフォルトの収集対象に加え、収集済みのファイルを対象とする。

#### **-m all**

収集は、出力中のファイルも含め、未使用・マージ済みを除いた全てを対象とする。

#### **-m cur[rent]**

収集は、現在出力中のファイルを対象とする。

#### **-h**

使用方法を表示する。

### 戻り値

diopsgather は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了
- 2      警告終了

### 注意

-d で指定するディレクトリ、または環境定義の GATHERDIR に指定するディレクトリは、環境定義の OPSDIR と同じディレクトリは指定できない。

環境定義の FTPUSER を省略した場合、diopsgather は使用できない。

## 2. 1. 60 diopsmrg (稼動統計マージコマンド)

### 名前

diopsmrg - 稼動情報をソート・マージする

### 書式

```
diopsmrg [-m out[put]] [-d GATHER_DIRECTORY] [-o OUTPUT_FILE]
diopsmrg -m app[end] [-d GATHER_DIRECTORY] [-o OUTPUT_FILE]
diopsmrg -h
```

### 説明

稼動統計収集コマンドにより 1 つのノード・ディレクトリに集められた稼動情報ファイルを時系列にソート・マージし、指定されたファイルへ出力する。

### オプション

**-m out[put]**

出力ファイルが存在した場合、上書きされる。(既定値)

**-m app[end]**

既存の出力ファイルへ追加される。

**-d GATHER\_DIRECTORY**

収集した稼動情報ファイルが格納されているディレクトリを絶対パスあるいは環境定義にて指定されたディレクトリ (GATHERDIR) からの相対パスで指定する。

省略時は環境定義にて指定されたディレクトリが使用される。

**-o OUTPUT\_FILE**

マージ結果を出力するファイル名を指定する。

絶対パスあるいは環境定義にて指定されたディレクトリ (GATHERDIR) からの相対パスを含めることも可能。パスが未指定の場合には環境定義の収集ディレクトリ (GATHERDIR) となる。

省略時は標準出力となる。

**-h**

使用方法を表示する。

### 戻り値

diopsmrg は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

2      警告終了

### 関連

diopsgather

## 2.1.61 distart\_com (DIOSA/XTP 起動コマンド)

### 名前

distart\_com - DIOSA/XTP を起動する。

### 書式

```
distart_com [-c|_w]  
distart_com -h
```

### 説明

DIOSA/XTP を起動するために必要な初期化处理と、各サブコンを初期化する API を実行する。  
オプションを省略した場合はウォームモードで起動する。

### オプション

-c

コールドモードで起動を行う。

-w

ウォームモードで起動を行う。

-h

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                      処理が正常終了した。

1                      処理が異常終了した。

### 注意

DIOSA/XTP 起動コマンドの起動指定時には環境変数 "DIOSA\_LNODENAME", "DIOSA\_IRMROOT" を設定すること。  
設定されていない場合、DIOSA/XTP 起動コマンドは異常終了する。

DIOSA/XTP インストールルートディレクトリ、作業用ディレクトリを変更したい場合には、環境変数 "DIOSA\_ROOT"、"DIOSA\_TMP" をそれぞれ設定する。

DIOSA/XTP 初回起動時にはコールドモードで起動すること。

### 関連

distop

## 2.1.62 distatref (起動／停止状態管理コマンド)

### 名前

distatref - DIOSA/XTP の起動状態を出力する。

### 書式

distatref [-v]  
distatref -h

### 説明

現在の DIOSA/XTP の起動状態を出力する。  
本コマンドで「Running」と表示される機能は、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るが、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る。

### オプション

-v  
DIOSA/XTP の起動状態を詳細表示する。  
省略した場合はサマリ表示する。

-h  
ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                      処理が正常終了した。  
1                      処理が異常終了した。

### 関連

distart\_com, distop

### 実行例

#### サマリ表示(オプションなし)

```
DISTE010 I LSBSJDG10.LNOL01. distatref Initialize state reference command started.  
NODE STARTUP STATUS : Running ①  
DISTE012 I LSBSJDG10.LNOL01. distatref Initialize state reference command terminated normally.
```

①NODE STARTUP STATUS : 論理ノード起動状況  
Running : 起動中  
Failed : 障害中  
Stop : 停止中

#### 詳細表示(distatref -v)

```
DISTE010 I LSBSJDG10.LNOL01. distatref Initialize state reference command started.  
NODE STARTUP STATUS : Running ①  
LNODE TYPE : OLTP ②  
LNODE NAME : YYYYYYYY ③  
  
<INFORMATION ON EACH FUNCTION> ④  
  
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  
DataBase Monitoring : Running  
Inter-Node Communication Path Management: Running  
Message Flow Control : Running
```

Message Guarantee	: 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
Online Maintenance	: Running
Database Access Control	: Running
DISTE012 I LSBSJDG10.LNOL01. distatref Initialize state reference command terminated normally.	

- ①NODE STARTUP STATUS : 論理ノード起動状況
  - Running : 起動中
  - Failed : 障害中
  - Stop : 停止中
- ②LNODE TYPE : 論理ノード種別
  - DB : DB ノード
  - AP : AP ノード
  - OLTP : OLTP ノード
- ③LNODE NAME : 論理ノード名称
- ④INFORMATION ON EACH FUNCTION : 機能別起動情報
  - Running : 起動中
  - Failed : 障害中
  - Stop : 停止中

## 2.1.63 distop (DIOSA/XTP 停止コマンド)

### 名前

distop - DIOSA/XTP を停止する。

### 書式

distop [-f]

distop -h

### 説明

DIOSA/XTP を停止する。

### オプション

-f

オプションが指定された場合、DIOSA/XTP の強制停止を行います。

DIOSA/XTP プロセスの強制停止、IPC 資源の削除、ソケットファイルの削除が行われます。

-h

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                    処理が正常終了した。

1                    処理が異常終了した。

### 注意

distart\_com コマンドにより DIOSA/XTP を起動したユーザと同一ユーザで実行すること。

強制停止オプションは DIOSA/XTP が正常に停止できなかった場合に指定してください。

強制停止オプション指定時は、以下のプロセスの強制停止、IPC 資源、ソケットファイルの削除が行われます。

- 強制停止対象プロセス

「DIOSA/XTP V2.1 導入の手引き」の「付録 B.1 常駐プロセス一覧」に記載されているプロセスの強制停止を行います。

- 削除対象 IPC 資源

IPCKEY の先頭 2 バイトに環境変数 (DIOSA\_IPCKEY) で指定した値が使用されている IPC 資源の削除を行います。

- 削除対象ソケットファイル

以下のディレクトリ配下のファイルを全て削除します。

`${DIOSA_TMP}/${DIOSA_LNODENAME}/socket`

### 関連

distart\_com

## 2.1.64 ditmcactv (タイマ保留解除コマンド)

### 名前

ditmcactv - 保留しているタイマ情報を解除する。

### 書式

```
ditmcactv -i TIMER_ID
```

```
ditmcactv -h
```

### 説明

指定されたメモリ型タイマ情報を保留解除する。

### オプション

**-i TIMER\_ID**

保留解除するタイマ ID を指定する。(省略不可、31 文字以内)

タイマ ID にワイルドカードを含める場合は、引用符で囲って指定すること。

**-h**

使用方法を表示する。

### 戻り値

ditmcactv は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

### 注意

保留解除するタイマ ID は ditmcref (タイマ情報照会コマンド) で検索する。

指定したタイマ ID に一致するタイマ情報が無い場合、ditmcactv は異常終了する。

指定したタイマが、既に保留解除済みの場合、ditmcactv は正常終了する。

### 関連

ditmchold, ditmcdmn, ditmcdmnstop, ditmcdel, ditmcref, ditmcset, ditmcsgset

## 2.1.65 ditmcdel (タイマ情報削除コマンド)

### 名前

ditmcdel - タイマ情報を削除する。

### 書式

```
ditmcdel -i TIMER_ID [-u {user|diosa}]
ditmcdel -h
```

### 説明

指定されたメモリ型タイマ情報を削除する。

### オプション

#### -i TIMER\_ID

削除するタイマ ID を指定する。(省略不可、31 文字以内)  
タイマ ID にワイルドカードを含める場合は、引用符で囲って指定すること。

#### -u

削除するタイマのユーザ種別を指定する。

user	ユーザのタイマを示す(既定値)
diosa	DIOSA 内部のタイマを示す

#### -h

使用方法を表示する。

### 戻り値

ditmcdel は、以下のいずれかの終了ステータスを戻して終了する。

- 0     正常終了
- 1     異常終了

### 注意

削除するタイマ ID は ditmcref (タイマ情報照会コマンド) で検索する。  
指定したタイマ ID に一致するタイマ情報が無い場合、ditmcdel は異常終了する。

### 関連

ditmcdmn, ditmcdmnstop, ditmcref, ditmcset, ditmcsgset, ditmchold, ditmcactv

## 2. 1. 66     ditmcdmn(タイマデーモン起動コマンド)

### 名前

ditmcdmn - タイマデーモンを起動する

### 書式

```
ditmcdmn [-t INTERVAL_TIME]
ditmcdmn -h
```

### 説明

登録されたタイマを監視するデーモンを起動する。

### オプション

**-t INTERVAL\_TIME**

デーモンが処理を行う間隔を秒単位で設定する。(数字：1～3600、省略可)

省略した場合は、前回動作時のインターバル時間で起動する。(初期起動時：60 秒)

タイマデーモンが起動済みの場合、インターバル時間を変更する。

**-h**

使用方法を表示する。

### 戻り値

ditmcdmn は、以下のいずれかの終了ステータスを戻して終了する。

0     正常終了

1     異常終了

### 注意

タイマデーモンを起動する前に DIOSA/XTP が起動していなくてはならない。

同じ DIOSA/XTP 上でデーモンを複数起動することはできない。

タイマデーモン起動中に実行した場合、正常終了する。

### 関連

ditmcdmnstop, ditmceref, ditmcdel, ditmcset, ditmcsgset, ditmchold, ditmcactv

## 2.1.67 ditmcdmnstop (タイマデーモン停止コマンド)

### 名前

ditmcdmnstop - タイマデーモンを停止する。

### 書式

```
ditmcdmnstop
ditmcdmnstop -h
```

### 説明

ノードでタイマを制御(実行)するデーモンを停止する。

### オプション

-h

使用方法を表示する。

### 戻り値

ditmcdmnstop は、以下のいずれかの終了ステータスを戻して終了する。

0 正常終了

1 異常終了

### 注意

DIOSA/XTP を停止させる前に、タイマデーモンを停止させなくてはならない。  
タイマデーモン停止済みの状態で実行した場合、正常終了する。

### 関連

ditmcdmn, ditmcref, ditmcdel, ditmeset, ditmesgset, ditmchold, ditmcactv

## 2.1.68 ditmchold(タイマ保留コマンド)

### 名前

ditmchold - タイマ情報を保留する。

### 書式

```
ditmchold -i TIMER_ID
ditmchold -h
```

### 説明

指定されたメモリ型タイマ情報を保留する。

### オプション

**-i TIMER\_ID**

保留するタイマ ID を指定する。(省略不可、31 文字以内)  
タイマ ID にワイルドカードを含める場合は、引用符で囲って指定すること。

**-h**

使用方法を表示する。

### 戻り値

ditmchold は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

### 注意

保留するタイマ ID は ditmcref(タイマ情報照会コマンド)で検索する。  
指定したタイマ ID に一致するタイマ情報が無い場合、ditmchold は異常終了する。  
指定したタイマが、既に保留済みの場合、ditmchold は正常終了する。

### 関連

ditmcactv, ditmcdmn, ditmcdmnstop, ditmcdel, ditmcref, ditmcset, ditmcsgset

## 2.1.69 ditmcref(タイマ情報照会コマンド)

### 名前

ditmcref - タイマ情報を照会する。

### 書式

```
ditmcref [-i TIMER_ID] [-u {user|diosa}] [-v]  
ditmcref -h
```

### 説明

現在登録されているタイマ情報を照会する

### オプション

#### -i TIMER\_ID

照会するタイマ ID を指定する。(省略不可、31 文字以内)  
タイマ ID にワイルドカードを含める場合は、引用符で囲って指定すること。

#### -u

照会するタイマのユーザ種別を指定する。  
user               ユーザのタイマを示す(既定値)  
diosa               DIOSA 内部のタイマを示す

#### -v

詳細情報をプリントする。(省略可)

#### -h

使用方法を表示する。

### 戻り値

ditmcref は、以下のいずれかの終了ステータスを戻して終了する。

- 0     正常終了
- 1     異常終了
- 2     警告終了

### 注意

照会対象となるタイマ情報が存在しない場合、ditmcref は警告終了する。

### 関連

ditmcdmn, ditmcdmnstop, ditmcdel, ditmcset, ditmsgset, ditmchold, ditmcactv

実行例

タイマ情報照会(ditmcref)

+===== TIMER CONTROL DISPLAY =====+			2016/01/01 12:00:00		
PROCESS ID	: 9999	①			
INTERVAL TIME	: 60(sec)	②			
DAEMON NEXT EXEC TIME	: 2016/01/01 12:00:30	③			
TIMER COUNT	: 5	④			
TIMER ID	TYPE	TIMER VALUE	BREAK COUNT LEFT	SERVICE	
⑤	⑥	⑦	⑧	⑨	
id_001	INTERVAL	----/--/-- 00:00:30	***	C0	
H id_002	INTERVAL	----/--/-- 00:01:00	5	C0	
id_003	INTERVAL	----/--/-- 01:00:00	10	C0	
id_004	CLOCK	----/--/-- 15:00:00	20	C0	
id_005	CLOCK	2016/01/31 23:59:59	1	COMMAND	
+===== END OF DISPLAY =====+					

カラムの説明

①PROCESS ID	タイマデーモンのプロセス ID
②INTERVAL TIME	タイマデーモンのインターバル時間(秒)
③DAEMON NEXT EXEC TIME	タイマデーモンが次に処理をおこなう時刻
④TIMER COUNT	照会対象となるタイマ情報の件数
⑤TIMER ID	タイマ ID タイマが保留状態の場合、タイマ ID の左側に「H」が表示される
⑥TYPE	時刻形式 INTERVAL インターバル指定または即時指定 CLOCK 時刻指定
⑦TIMER VALUE	タイマ値 TIME TYPE が INTERVAL の場合はインターバル時間 TIME TYPE が CLOCK の場合はタイマ実行時刻
⑧BREAK COUNT LEFT	タイマが実行される残り回数 通知回数に DIOSA_NOLIM(無限回)を指定した場合、「***」が表示される
⑨SERVICE	サービス種別 C0 C0 タイマ COMMAND コマンドタイマ TPBASE TPBASE タイマ

タイマ詳細情報照会(ditmcref -v)

+===== TIMER CONTROL DISPLAY =====+				2016/01/01 12:00:00		
PROCESS ID		:	9999	①		
INTERVAL TIME		:	60(sec)	②		
DAEMON NEXT EXEC TIME		:	2016/01/01 12:00:30	③		
TIMER COUNT		:	5	④		
TIMER ID			TYPE	TIMER VALUE	NEXT BREAK TIME	TOTAL (LEFT) SERVICE
⑤			⑥	⑦	⑧	⑨ ⑩ ⑪
id_001			INTERVAL	----/--/-- 00:00:30	2016/01/01 12:00:30	*** ( ***) C0
H id_002			INTERVAL	----/--/-- 00:01:00	2016/01/01 12:01:00	10 ( 5) C0
id_003			INTERVAL	----/--/-- 01:00:00	2016/01/01 13:00:00	20 ( 10) C0
id_004			CLOCK	----/--/-- 15:00:00	2016/01/01 15:00:00	30 ( 20) C0
id_005			CLOCK	2016/01/31 23:59:59	2016/01/31 23:59:59	1 ( 1) COMMAND
+===== END OF DISPLAY =====+						

カラムの説明

①PROCESS ID	タイマデーモンのプロセス ID
②INTERVAL TIME	タイマデーモンのインターバル時間(秒)
③DAEMON NEXT EXEC TIME	タイマデーモンが次に処理をおこなう時刻
④TIMER COUNT	照会対象となるタイマ情報の件数
⑤TIMER ID	タイマ ID タイマが保留状態の場合、タイマ ID の左側に「H」が表示される。
⑥TYPE	時刻形式 INTERVAL インターバル指定または即時指定 CLOCK 時刻指定
⑦TIMER VALUE	タイマ値 TYPE が INTERVAL の場合はインターバル時間 TYPE が CLOCK の場合はタイマ実行時刻
⑧NEXT BREAK TIME	次回実行時刻
⑨TOTAL	タイマが実行する通算回数  (更新登録されたタイマ情報の場合、実行済回数と実行予定回数の合計となる)  通知回数に DIOSA_NOLIM(無限回)を指定した場合、「***」が表示される
⑩LEFT	タイマが実行される残り回数  通知回数に DIOSA_NOLIM(無限回)を指定した場合、「***」が表示される
⑪SERVICE	サービス種別  C0 C0 タイマ COMMAND コマンドタイマ TPBASE TPBASE タイマ

## 2.1.70 ditmcset(タイマ登録コマンド)

### 名前

ditmcset - タイマ情報を登録する。

### 書式

```
ditmcset -T TIMER_TYPE [-d TIMERVERAL_DATE] -t TIMERVERAL_TIME [-c COUNT] -i TIMER_ID [-s co]
          -C CONAME -x TRANSACTION_ID -M Message [-k]
ditmcset -T TIMER_TYPE [-d TIMERVERAL_DATE] -t TIMERVERAL_TIME [-c COUNT] -i TIMER_ID -s cmd
          -M Message [-k]
ditmcset -h
```

### 説明

指定されたタイマを登録する。

### オプション

#### -T TIMER\_TYPE

タイマの時刻形式を指定する。(省略不可)

inter	インターバル指定
imme	即時指定
clock	時刻指定

#### -d TIMERVERAL\_DATE

タイマ通知年月日を指定する。(数字 8 桁: 19900101~20891231)

-T clock の場合のみ指定可能であり、任意の日時にタイマ通知を行う。

-T clock 指定時に省略した場合、-t で指定した時刻にタイマ通知を毎日行う。

#### -t TIMERVERAL\_TIME

-T inter、-T imme の場合は、タイマの通知時間間隔を指定する。

-T clock の場合は、タイマ通知時刻を指定する。

(数字 6 桁: 時分秒)

#### -c COUNT

通知回数を指定する。(0~999)

0 を指定した場合、通知回数は無限回となる。省略時は 1 が設定される。

-T clock と -d が指定された場合は、指定不可である。

#### -i TIMER\_ID

登録するタイマ ID を 16 文字以内の英数字で指定する。(省略不可)

#### -s SERVICE

タイマの通知先を指定する。

co	C0 タイマを登録する(既定値)
cmd	コマンドタイマを登録する

#### -C CONAME

宛先の C0 名を 30 文字以内で指定する。

[-s co] (C0 タイマ)の時は省略不可。[-s cmd] (コマンドタイマ)の時は指定不可。

#### **-x TRANSACTION\_ID**

トランザクション ID を 6 文字以内で指定する。

[**-s co**] (C0 タイマ)の時は省略不可。[**-s cmd**] (コマンドタイマ)の時は指定不可。

#### **-M Message**

C0 タイマの場合、C0 に通知するメッセージを指定する。

コマンドタイマの場合、コマンドイメージを指定する。

どちらの場合も、環境変数 DIOSA\_TMCLNMAX に設定したバイト数以内で指定する。(省略不可)

#### **-k**

タイマ ID の重複チェックを行うか否かを指定する。(省略可)

パラメータが指定された時のみタイマ ID の重複チェックを行なう。

重複チェックを行なった時はタイマ情報を上書きして警告終了する。

重複チェックを行なわなかった時はタイマ情報を上書きして正常終了する。

#### **-h**

使用方法を表示する。

### 戻り値

ditmcset は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了
- 2      警告終了

### 注意

タイマの時刻形式で clock(時刻指定)を選択した場合、現在時刻より過去日を設定することはできない。

タイマ制御の仕様上、実行対象のコマンドが “126, 127” で exit すると、コマンド実行エラーと区別ができない。

本コマンドで登録した C0 タイマによって C0 に通知されるメッセージは、**-M** で指定された NULL 終端を含まない文字列となる。

### 関連

ditmcdmn, ditmcdmnstop, ditmcdel, ditmceref, ditmcsgset, ditmchold, ditmcactv

## 2.1.71 ditmsgset (SG タイマ登録コマンド)

### 名前

ditmsgset - SG タイマ情報を登録する。

### 書式

```
ditmsgset  
ditmsgset -h
```

### 説明

環境定義で指定されたタイマを登録する。

### オプション

**-h**  
使用方法を表示する。

### 戻り値

ditmsgset は、以下のいずれかの終了ステータスを戻して終了する。

- 0     正常終了
- 1     異常終了
- 2     警告終了

### 関連

ditcdmn, ditcdmnstop, ditcdel, ditmcref, ditmcset, ditmchold, ditmcactv

## 2.2 通信制御

### 2.2.1 コマンド一覧

#### (1) DB 監視機能

dincmdbhchkctrlinit	DB ヘルスチェックデーモンを起動する。
dincmdbhchkctrlterm	DB のヘルスチェックデーモンを停止する。
dincmdbinit	DB 監視機能を起動する。
dincmdbstat	DB の状態を照会する。
dincmdbterm	DB 監視機能を終了する。

#### (2) パス管理機能

dincmpathinit	パス管理機能を起動する。
dincmpathterm	パス管理機能を終了する。
dincmpathdmnctrl	パス管理機能のデーモンを起動/停止する。
dinodepathctrl	論理ノード間パスを接続/切断する
dinodepathref	論理ノード間パスの状態を照会する。
dinodepathrefstats	論理ノード間パスの状態の統計情報を照会する。
dilspathctrl	論理システム間パス(常時接続)を接続/切断する。
dilspathref	論理システム間パス(常時接続)の状態を照会する。
dilspathrefstats	論理システム間パス(常時接続)の統計情報を照会する。
diotcref	論理システム間パス(都度接続)の定義情報を照会する。
diotcrefstats	論理システム間パス(都度接続)の統計情報を照会する。

#### (3) 電文保証

digntdbreset	電文保証機能のマスタ TAM、OracleDB を初期化する
digntinit	電文保証機能を起動する
digntterm	電文保証機能を停止する
digntmod	電文保証機能の環境定義を変更する
digntdel	電文保証対象の送信電文情報を削除する
digntchgdest	登録済み保証電文の宛先を変更する
digntref	電文保証機能の各種情報を照会する

## 2.2.2 digntchgdest (保証電文宛先変更コマンド)

### 名前

digntchgdest - 登録済み保証電文の送信先アクセスポイントを変更する。

### 書式

```
digntchgdest -b BEFORENAME -a AFTERNAME -D { im | db } [ -C MSGPERCOMMIT ]
digntchgdest -h
```

### 説明

登録済み保証電文の送信先アクセスポイントを変更する。

### オプション

#### **-b BEFORENAME**

変更前のアクセスポイント名を指定する。(必須)  
SG 未定義のアクセスポイント名も指定が可能である。

#### **-a AFTERNAME**

変更後のアクセスポイント名を指定する。(必須)  
SG 定義されたアクセスポイント名のみ指定が可能である。

#### **-D { im | db }**

アクセス先の DB を指定する。(必須)  
im : メモリキャッシュ(対象とするのはコマンド実行ノードにマスタがある MAPID 群)  
db : OracleDB

#### **-C MSGPERCOMMIT**

1 コミットで更新対象とする電文の最大数を指定する。(省略可)  
MSGPERCOMMIT には 1～32767 が指定可能である。  
複数電文が更新対象となったときに、MSGPERCOMMIT のレコード数でトランザクションを分けられる。  
本オプションが未指定の場合、10 が指定されたものとして動作する。

#### **-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した
1	処理が異常終了した

### 注意

- OLTP ノードでのみ実行可能。
- メモリキャッシュを指定する場合、全ての OLTP ノードでコマンドを実行する必要がある。
- 常時接続のアクセスポイントから、都度接続のアクセスポイントへの変更は認めない。  
また、都度接続のアクセスポイントから、常時接続のアクセスポイントへの変更も認めない。

### 関連

digntinit, digntterm

## 2.2.4 digntdbreset (電文保証 DB 初期化コマンド)

### 名前

digntdbreset - 電文保証機能の DB (TAM/OracleDB) を初期化する。

### 書式

```
digntdbreset -D { im | db }  
digntdbreset -h
```

### 説明

電文保証機能の DB (メモリキャッシュ/OracleDB) を初期化する。

### オプション

**-D { im | db }**

アクセス先の DB を指定する。(必須)

im : メモリキャッシュ (対象とするのはコマンド実行ノードにマスタがある MAPID 群)

db : OracleDB

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0 処理が正常終了した

1 処理が異常終了した

### 注意

- OLTP 層のノードで実行する。TAM の場合は実行ノードのマスタを対象とするので全ノードで、OracleDB の場合はいずれかのノードで実行する。
- 送信中の電文や受信した電文の情報を全て削除するため、電文の再送や二重処理検査が行われなくなること  
に留意する。

### 関連

digntinit

## 2.2.5 digntdel (保証電文削除コマンド)

### 名前

digntdel - 処理が未完了の保証電文を削除する

### 書式

```
digntdel { -A | -a ACCESSPOINT | -k MSGKEY } -D { im | db } [ -C MSGPERCOMMIT ]
digntdel -h
```

### 説明

電文保証対象の送信メッセージを削除する。通常は処理完了や、再送処理のリトライオーバーを契機に自動的に削除されるが、処理が完了していない保証電文を強制的に削除したい場合に使用する。

### オプション

#### -A

全ての保証電文を削除する。

#### -a ACCESSPOINT

宛先アクセスポイントを指定する。指定されたアクセスポイント宛の保証電文が全て削除される。

#### -k MSGKEY

保証電文の識別子を指定する。指定された識別子の保証電文が削除される。

MSGKEY は 16 進数表記で 1～32 バイトの領域を表現する (先頭に 0x は付けない)。指定が 32 バイトに満たない場合、先頭から指定されたサイズ分のデータが一致する全ての保証電文が削除対象となる。

MSGKEY に指定する文字数が奇数の場合はパラメータエラーとなる。

#### -D { im | db }

アクセス先の DB を指定する。(必須)

im : メモリキャッシュ (対象とするのはコマンド実行ノードにマスタがある MAPID 群)

db : OracleDB

#### -C MSGPERCOMMIT

1 コミットで削除対象とする電文の最大数を指定する。(省略可)

MSGPERCOMMIT には 1～32767 が指定可能である。

複数電文が削除対象となったときに、MSGPERCOMMIT のレコード数でトランザクションを分けられる。

本オプションが未指定の場合、10 が指定されたものとして動作する。

#### -h

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0 処理が正常終了した

1 処理が異常終了した

### 注意

- OLTP ノードでのみ実行可能。
- メモリキャッシュを指定する場合、全ての OLTP ノードでコマンドを実行する必要がある。

### 関連

digntdbreset

## 2.2.6 digntinit(電文保証初期化コマンド)

### 名前

digntinit - 電文保証機能を起動する

### 書式

digntinit [-w | -c]

digntinit -h

### 説明

電文保証機能を起動する。

### オプション

**-w**

ウォーム起動する。前回運転時の環境を引き継いで起動する。  
パラメータが省略された場合は本オプションが指定されたものとする。

**-c**

コールド起動する。  
環境定義を新規に読み込んで起動する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                      処理が正常終了した

1                      処理が異常終了した

### 注意

- OLTP 層の全ノードで実行する。

### 関連

digntterm

## 2.2.7 digntmod(電文保証定義変更コマンド)

### 名前

digntmod - 電文保証機能の環境定義を変更する

### 書式

```
digntmod [-i RsndInvl] [-c SendCnt] [-r RcvDelTime] [-s SeqDelTime]
digntmod -h
```

### 説明

電文保証機能の環境定義を与えられた値に変更する。  
パラメータは必ず1つ以上指定すること。

### オプション

#### -i

再送時間間隔を与えられた RsndInvl の値に変更する。省略可。  
RsndInvl の単位は秒であり、1～9999 までの値とする。

#### -c

最大送信回数を与えられた SendCnt の値に変更する。省略可。  
SendCnt の単位は回数であり、1～524287 までの値とする。

#### -r

受信電文の保存時間を与えられた RcvDelTime の値に変更する。省略可。  
RcvDelTime の単位は分であり、1～14400 までの値とする。  
なお、送信元の再送処理がリトライオーバーするまでに掛かる時間を超える値を設定すること。

#### -s

順序性保証グループ情報の保持時間を与えられた SeqDelTime の値に変更する。省略可。  
SeqDelTime の単位は分であり、1～2147483647 までの値とする。

#### -h

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した
1	処理が異常終了した

### 注意

- OLTP 層の設定を変更したいノードで実行する。
- 電文保証機能をウォームで再起動した場合、本設定値は引き継がれる。
- 再送時間間隔と最大送信回数の変更値は、本コマンド投入前に登録した保証電文には適用されない。

### 関連

digntref

## 2.2.8 digntref(電文保証照会コマンド)

### 名前

digntref - 電文保証機能の各種情報を照会する。

### 書式

```
digntref -d ctl
digntref -d cnt [ -D { im | db } ] [ -c ]
digntref -d msg [ -D { im | db } ] [ -k MSGKEY ] [ -u SIZE ]
digntref -h
```

### 説明

電文保証機能の情報を照会する。-d オプションで照会モードを指定し、オプションごとに追加で照会対象の絞り込み等を行うことができる。

### オプション

**-d { ctl | cnt | msg }**

照会モードを指定する。(必須)

ctl : 現在の動作パラメータを表示する。

cnt : 送信が完了していない保証電文数を表示する。

msg : 個々の保証電文の情報を表示する。

**-D { im | db }**

アクセス先の DB を指定する。照会モードが cnt, msg の場合に指定可能である。

省略した場合は両方の情報を照会する。

im : メモリキャッシュ(全 MAPID が対象)

db : OracleDB

**-c**

情報を CSV 形式で出力する。指定しない場合は内容を整形して表示する。

出力イメージについては後述の実行例を参照。

**-k MSGKEY**

照会したい保証電文の識別子を指定する。

MSGKEY は 16 進数表記で 1～32 バイトの領域を表現する(先頭に 0x は付けない)。指定が 32 バイトに満たない場合、先頭から指定されたサイズ分のデータが一致する全ての保証電文が表示対象となる。

MSGKEY に指定する文字数が奇数の場合はパラメータエラーとなる。

**-u SIZE**

保証電文のダンプ出力サイズを指定する。照会モードが msg の場合に指定可能である。

ユーザデータ領域の先頭から SIZE で指定されたサイズ(バイト)の領域が表示される。省略された場合、128 が指定されたものとして動作する。

指定サイズより電文サイズが小さい場合は、電文サイズ分の領域を表示する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0 処理が正常終了した

1 処理が異常終了した

#### 注意

- OLTP ノードでのみ実行可能

#### 関連

`digntinit`, `digntterm`

実行例

動作パラメータ表示(digntref -d ctl)

+===== MESSAGE GUARANTEE DISPLAY =====+ 2016/01/01 00:00:00 ①				
+----- SYSTEM CONFIGURATION INFORMATION -----+				
RSNDINVL(sec)	= 100	①		
SENDMAX	= 10	②		
RSNDCHKINVL(sec)	= 10	③		
DELCHKINVL(min)	= 5	④		
RGNTTIME(min)	= 30	⑤		
SEQDELTIME(min)	= 60	⑥		
INITEXIT	= USERPRCINIT	⑦		
TERMEXIT	= USERPRCTERM	⑧		
JUDGEEXIT	= USERJUDGEEXIT	⑨		
OVERCONAME	= RETRYOVER	⑩		
OVERTXID	= TXID01	⑪		
+----- DAEMON STATUS INFORMATION -----+				
①	②			
DAEMON	STATUS			
-----				
RESEND	ACTIVE			
RCVRSP	ACTIVE			
MSGDEL	ACTIVE			
+----- TIMER INFORMATION -----+				
①	②	③	④	⑤
TPM	RSD(IM)	RSD(DB)	DEL(IM)	DEL(DB)
-----				
TPBASE1	YES	YES	NO	NO
TPBASE2	YES	NO	YES	YES
+===== END OF DISPLAY =====+				

① コマンド実行時刻

+----- SYSTEM CONFIGURATION INFORMATION -----+  
電文保証機能の動作パラメータ等を表示する。

- ① 再送処理間隔(秒)
- ② 最大送信回数
- ③ 再送処理起動インターバル(秒)
- ④ 受信電文情報削除処理起動インターバル(分)
- ⑤ 受信電文情報保持時間(分)
- ⑥ 処理済み順序性保証グループ情報保持時間(分)
- ⑦ 電文保証 TPP プロセス初期化出口関数名
- ⑧ 電文保証 TPP プロセス終了出口関数名
- ⑨ 保証電文送信有無判定出口関数名
- ⑩ リトライオーバーCO 名
- ⑪ リトライオーバーTXID

+----- DAEMON STATUS INFORMATION -----+  
電文保証機能の制御デーモンのステータスを表示する。

- ① デーモン種別
  - RESEND : 保証電文再送処理 TPP
  - RCVRSP : 保証応答受信 TPP
  - MSGDEL : 受信電文情報削除 TPP
- ② ステータス
  - ACTIVE : 動作中

+----- TIMER COUNT INFORMATION -----+  
制御デーモン起動用タイマの登録情報を表示する。  
YES と表示されているものが、定期処理をおこなっている。

- ① TP モニタ名

- ② 保証電文再送処理 TPP 起動タイマ登録有無(メモリキャッシュ処理用)
- ③ 保証電文再送処理 TPP 起動タイマ登録有無(OracleDB 処理用)
- ④ 受信電文情報削除 TPP 起動タイマ登録有無(メモリキャッシュ処理用)
- ⑤ 受信電文情報削除 TPP 起動タイマ登録有無(OracleDB 処理用)

保証電文滞留数表示(digntref -d cnt)

+===== MESSAGE GUARANTEE DISPLAY =====+ 2016/01/01 00:00:00 ①		
+----- UNFINISHED MESSAGE COUNT INFORMATION -----+		
---[ DBTYPE = IM ]----- ①		
②	③	④
LSYSTEM	ACCESSPOINT	MSGCNT
-----		
LSYS01	ACSP01	10
	ACSP02	20
LSYS02	ACSP03	0
	ACSP04	0
[IN-SYSTEM]	---	15
-----		
TOTAL		45
---[ DBTYPE = DB ]-----		
LSYSTEM	ACCESSPOINT	MSGCNT
-----		
:		
+===== END OF DISPLAY =====+		

① コマンド実行時刻

+----- UNFINISHED MESSAGE COUNT INFORMATION -----+  
処理が完了していない保証電文数を宛先アクセスポイントごとに表示する。

- ① DB 種別  
IM : メモリキャッシュ  
DB : OracleDB
- ② 論理システム名 ([IN-SYSTEM]は論理システム内送信)
- ③ 宛先アクセスポイント名 (論理システム内送信では、”---“を表示)
- ④ 保証電文数

保証電文滞留数表示[CSV 出力] (digntref -d cnt -c)

TIME, DBTYPE, LSYSTEM, ACCESSPOINT, MSGCNT	①
2016/11/29 21:22:57, IM, LSYS01, ACSP01, 10	②
2016/11/29 21:22:57, IM, LSYS01, ACSP02, 20	
:	
2016/11/29 21:22:57, IM, [IN-SYSTEM], , 15	
2016/11/29 21:22:57, IM, [TOTAL], , 45	
2016/11/29 21:22:57, DB, LSYS01, ACSP01, 10	
:	
2016/11/29 21:22:57, DB, [TOTAL], , 1000	

- ① 列タイトル行
- ② データ行  
(「コマンド実行時刻, DB 種別, 論理システム名, アクセスポイント名, 保証電文数」の形式)  
[IN-SYSTEM]は論理システム内送信、[TOTAL]はDB 種別ごとの総数を表す。アクセスポイント名は空欄となる。

保証電文情報表示(digntref -d msg)

+===== MESSAGE GUARANTEE DISPLAY =====		2016/01/01 00:00:00	①
+----- MESSAGE INFORMATION -----+			
---[No. 1 ]-----			①
DBTYPE	= DB		②
MAPID	= ---		③
RGSET	= RGSET01		④
MSGID			⑤
0x0000000000000000		: FE00 FFFF FFFF FFFF FFFF FFFF FFFF	: .....
0x0000000000000010		: FFFF FFFF FFFF FFFF FFFF FFFF FFFF	: .....
ACCESSPOINT	= ACSP01		⑥
RESENDINVL	= 30		⑦
SENDMAX	= 5		⑧
SENCNT	= 1		⑨
NEXTSEND	= 2016/01/01 23:59:59		⑩
SEQGROUP			⑪
0x0000000000000000		: 5345 5147 524F 5550 3031 0000 0000 0000	: SEQGROUP01.....
0x0000000000000010		: 0000 0000 0000 0000 0000 0000 0000 0000	: .....
SEQNO	= 100		⑫
MSGSIZE	= 512		⑬
MSGINFO(128 BYTES FROM TOP)			⑭
0x0000000000000000		: FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF	: .....
0x0000000000000010		: FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF	: .....
0x0000000000000020		: FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF	: .....
=====		: ====	: =====
0x0000000000000070		: FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF	: .....

① コマンド実行時刻

+----- MESSAGE INFORMATION -----+

保証電文情報を表示する。

- ① 件数
- ② DB 種別
  - IM : メモリキャッシュ
  - DB : OracleDB
- ③ MAPID(DB 種別が DB の場合は表示されない)
- ④ RGSET 名(DB 種別が IM の場合は表示されない)
- ⑤ 電文識別子(32 バイトの領域を 16 進表示)
- ⑥ 宛先アクセスポイント名(論理システム内送信の場合は表示されない。)
- ⑦ 再送間隔
- ⑧ 最大送信回数
- ⑨ 現在の送信回数
- ⑩ 次回送信時刻(未決定時は----/--/-- --:--:-- が表示される。)
- ⑪ 順序性保証グループ ID(順序性保証なしの場合は表示されない。)
- ⑫ 順序性保証グループ通番(順序性保証なしの場合は表示されない。)  
(滞留がなくなってから一定時間出力がないと 1 にリセットされるため、必ずしも一意の通番ではない)
- ⑬ ユーザ電文サイズ
- ⑭ ユーザ電文情報(電文の先頭から、パラメータで指定されたバイト数分を表示)  
直前の 16 バイトと同じ内容が連続する場合、====を使用して省略表示する。

## 2.2.9 digntterm(電文保証終了コマンド)

### 名前

digntterm - 電文保証機能を停止する

### 書式

```
digntterm
digntterm -h
```

### 説明

電文保証機能を停止する。

### オプション

**-h**  
ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した
1	処理が異常終了した

### 注意

- OLTP 層の全ノードで実行する。

### 関連

digntinit

## 2.2.10 dilspathctrl (論理システム間パス接続切断コマンド)

### 名前

dilspathctrl - 論理システム間パス(常時接続)の接続・切断を指示する

### 書式

```
dilspathctrl -o { open | close | accept | reject } -a acsp [ -M TPM ] [ -T terminal ] [ -F ]
```

```
dilspathctrl -o { open | close | accept | reject } -a acsp [ -M TPM ] [ -d { all | send |  
recv } ] [ -F ]
```

```
dilspathctrl -h
```

### 説明

L パス(常時接続)の接続・切断を指示する。

### オプション

**-o { open | close | accept | reject }**

接続・切断を指定する。(必須)

open : アクセスポイントとの接続を指示する。

close : アクセスポイントとの切断を指示する。

accept : アクセスポイントとの接続を許可する。

reject : アクセスポイントとの接続を拒否する。

**-a acsp**

接続先のアクセスポイントを指定する。(必須)

**-M TPMonitor**

操作対象の TPM を指定する。

省略した場合、論理ノード内の全ての TPM が対象となる。

**-T terminal**

操作対象の端末名を指定する。

省略した場合、アクセスポイントに属する全ての端末が対象となる。

**-d { all | send | recv }**

接続・切断を行う端末の種類を選択する。省略時には all が指定される。

all : アクセスポイントに属するすべての端末を対象とする

send : 送信用端末を対象とする

recv : 受信用端末を対象とする

アクセスポイントが 1 パスの場合、send, recv のいずれを指定しても all と同じ動作となる。

**-F**

対象の端末が既に接続済あるいは切断済の状態であっても、接続操作あるいは切断操作を実施する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0 処理が正常終了した

1 処理が異常終了した

#### 注意

- 本コマンドが正常終了した場合、端末に対する接続・切断の指示が正常に行えたことを示す。接続・切断の結果は利用者インタフェース パス接続切断同期通知 C0 で通知される。
- 本コマンドは、AP ノード、OLTP ノードに投入可能である。

#### 関連

パス接続切断同期通知 C0

## 2.2.11 dilspathref (論理システム間パス照会コマンド)

### 名前

dilspathref- 論理システム間パス (常時接続) の状態を照会する

### 書式

```
dilspathref [-l LsName ] [ -a AcsP ] [ -M TPM ] [-d { acsp | term } ] [ -c ]
```

```
dilspathref -h
```

### 説明

論理システム間パス (常時接続) の接続、切断状態を照会する。オプションにより端末毎の状態を照会できる。

### オプション

**-l LsName**

照会する論理システムを指定する。

省略した場合、すべての論理システムの状態を照会する。

**-a AcsP**

照会するアクセスポイントを指定する。

省略した場合、すべてのアクセスポイントの状態を照会する。

**-M TPM**

照会する TPM 名を指定する。コマンドを実行した論理ノード内に存在すること。

省略した場合、コマンドを実行した論理ノード内にあるすべての TPM の状態を照会する。

**-d { acsp | term }**

照会するパス状態の粒度を指定する。省略可。

省略時は **acsp** が指定されたものとして動作する。

**acsp** : アクセスポイント毎のパス状態を照会する。

アクセスポイント配下の端末が 1 つでも接続済状態であれば活性と表示される。

**term** : 端末毎の接続状態を照会する。

**-c**

照会情報を CSV 形式で出力する。省略可。

省略した場合は整形された状態で出力する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0 処理が正常終了した

1 処理が異常終了した

### 関連

dilspathctrl dilspathrefstats

実行例～アクセスポイント毎の情報照会（-d acsp）

+===== LS-PATH STATUS DISPLAY =====+ 2017/01/23 04:56:07				①
LOGSYSTEM = LSYSTEM01				②
LNODE = LNODE01				
SOURCE____ DESTINATION_____				③
TPM_____	LOGSYSTEM_____	ACCESSPOINT_____	STATUS_____	④
TPM01	LSYS01	AcsP0101	OPEN	
		Acsp0102	OPEN	
	LSYS02	AcsP0201	OPEN	
		Acsp0202	CLOSE	
		Acsp0203	OPEN	
TPM02	LSYS01	AcsP0101	OPEN (R)	
		Acsp0102	CLOSE (R)	
	LSYS02	AcsP0201	CLOSE	
		Acsp0202	CLOSE	
		Acsp0203	CLOSE	
	LSYS03	AcsP0301	OPEN (2way)	
		Acsp0302	OPEN (send) (R)	
+===== END OF DISPLAY =====+				

①コマンド実行時刻

②コマンド実行環境

LOGSYSTEM : 論理システム名

LNODE : 論理ノード名

③区分

SOURCE : 起点側

DESTINATION : 接続先側

④各種情報

No.	カラム表示名	内容
1	LNODENAME	論理ノード名
2	TPM	TPM 名
3	LOGSYSTEM	通信先論理システム名
4	ACCESSPOINT	通信先アクセスポイント名
5	STATUS	パス状態 OPEN : 活性(1 パス) OPEN(2way) : 活性(2 パス) OPEN(send) : 送信端末のみ活性 OPEN(recv) : 受信端末のみ活性 CLOSE : 非活性  接続可否状態(パス状態の後に表示) (R) : 全端末接続拒否

実行例～端末毎の情報照会 (-d term)

+===== LS-PATH STATUS DISPLAY =====+ 2017/01/23 04:56:07						①
LSYSTEM = LSYSTEM01						②
LNODE = LNODE01						
SOURCE		DESTINATION				③
TPM	LOGSYSTEM	ACCESSPOINT	STATUS	TERMINAL	STATUS	④
TPM01	LSYS01	AcsP0101	OPEN	LSYS01_AcsP0101_TRM01	OPEN	
				LSYS01_AcsP0101_TRM02	OPEN	
				LSYS01_AcsP0101_TRM03	OPEN	
		AcsP0102	OPEN	LSYS01_AcsP0102_TRM01	OPEN	
				LSYS01_AcsP0102_TRM02	TROUBLE	
				LSYS01_AcsP0102_TRM03	TROUBLE	
	LSYS02	AcsP0201	OPEN	LSYS02_AcsP0201_TRM01	OPEN	
				LSYS02_AcsP0202_TRM01	CLOSE	
		AcsP0202	CLOSE	LSYS02_AcsP0202_TRM02	CLOSE (R)	
				LSYS02_AcsP0202_TRM03	CLOSE (R)	
		AcsP0203	OPEN	LSYS02_AcsP0203_TRM01	OPEN	
				LSYS02_AcsP0203_TRM02	OPEN (R)	
TPM02	LSYS01	AcsP0101	OPEN (R)	LSYS01_AcsP0101_TRM01	OPEN (R)	
				LSYS01_AcsP0101_TRM02	OPEN (R)	
				LSYS01_AcsP0101_TRM03	OPEN (R)	
		AcsP0102	CLOSE (R)	LSYS01_AcsP0102_TRM01	CLOSE (R)	
				LSYS01_AcsP0102_TRM02	CLOSE (R)	
				LSYS01_AcsP0102_TRM03	CLOSE (R)	
	LSYS02	AcsP0201	CLOSE	LSYS02_AcsP0201_TRM01	CLOSE	
		AcsP0202	CLOSE	LSYS02_AcsP0202_TRM01	CLOSE	
		AcsP0203	CLOSE	LSYS02_AcsP0203_TRM01	CLOSE	
	LSYS03	AcsP0301	OPEN (2way)	LSYS03_AcsP0301_TRM01	OPEN	
				LSYS03_AcsP0301_TRM02	OPEN	
		AcsP0302	OPEN (send)	LSYS03_AcsP0302_TRM01	OPEN	
				LSYS03_AcsP0302_TRM02	CLOSE	
+===== END OF DISPLAY =====+						

①コマンド実行時刻

②コマンド実行環境

LSYSTEM : 論理システム名

LNODE : 論理ノード名

③区分

SOURCE : 起点側

DESTINATION : 接続先側

④各種情報

No.	カラム表示名	内容
1	LNODENAME	論理ノード名
2	TPM	TPM 名
3	LOGSYSTEM	通信先論理システム名
4	ACCESSPOINT	通信先アクセスポイント名
5	STATUS (ACCSESPOINT の後)	<p>パス状態</p> <p>OPEN : 活性(1 パス)</p> <p>OPEN(2way) : 活性(2 パス)</p> <p>OPEN(send) : 送信端末のみ活性</p> <p>OPEN(recv) : 受信端末のみ活性</p> <p>CLOSE : 非活性</p> <p>接続可否状態(パス状態の後に表示)</p> <p>(R) : 全端末接続拒否</p>
6	TERMINAL	端末名
7	STATUS (TERMINAL の後)	<p>端末状態</p> <p>OPEN : 接続</p> <p>OPENING : 接続処理中</p> <p>CLOSING : 切断処理中</p> <p>CLOSE : 切断</p> <p>TROUBLE : 障害</p> <p>RECOVERING : 障害復旧中</p> <p>接続可否状態(パス状態の後に表示)</p> <p>(R) : 接続拒否</p>

実行例～CSV 形式 (-d tpmonitor -c)

TIME, SRC-LNODENAME, SRC-TPM, DEST-LSYS, DEST-ACCESSPOINT, ACSP-STATUS
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0101, OPEN
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0102, OPEN
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0201, OPEN
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0202, CLOSE
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0203, OPEN
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0101, OPEN (R)
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0102, CLOSE (R)
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0201, CLOSE
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0202, CLOSE
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0203, CLOSE
2017/01/23 04:56:07, LNODE01, TPM02, LSYS03, AcsP0301, OPEN (2way)
2017/01/23 04:56:07, LNODE01, TPM02, LSYS03, AcsP0302, OPEN (send)

①各種情報

No.	カラム表示名	内容
1	TIME	コマンド実行時刻
2	SRC-LNODENAME	論理ノード名
3	SRC-TPM	TPM 名
5	DEST-LSYS	接続先論理システム名
6	DEST-ACCESSPOINT	接続先アクセスポイント名
7	ACSP-STATUS	パス状態 OPEN           : 活性(1 パス) OPEN (2way)   : 活性(2 パス) OPEN (send)   : 送信端末のみ活性 OPEN (recv)   : 受信端末のみ活性 CLOSE          : 非活性  接続可否状態(パス状態の後に表示) (R)   : 全端末接続拒否

実行例～端末毎の情報照会 CSV 形式(-d term -c)

TIME, SRC-LNODENAME, SRC-TPM, DEST-LSYS, DEST-ACCESSPOINT, ACSP-STATUS, TERMINAL, TERMINAL-STATUS ①									
2017/01/23	04:56:07	LNODE01	TPM01	LSYS01	AcsP0101	OPEN	LSYS01_AcsP0101_TRM01	OPEN	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS01	AcsP0101	OPEN	LSYS01_AcsP0101_TRM02	OPEN	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS01	AcsP0101	OPEN	LSYS01_AcsP0101_TRM03	OPEN	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS01	AcsP0102	OPEN	LSYS01_AcsP0102_TRM01	OPEN	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS01	AcsP0102	OPEN	LSYS01_AcsP0102_TRM02	TROUBLE	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS01	AcsP0102	OPEN	LSYS01_AcsP0102_TRM03	TROUBLE	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS02	AcsP0201	OPEN	LSYS02_AcsP0201_TRM01	OPEN	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS02	AcsP0202	CLOSE	LSYS02_AcsP0202_TRM01	CLOSE	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS02	AcsP0202	CLOSE	LSYS02_AcsP0202_TRM02	CLOSE (R)	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS02	AcsP0202	CLOSE	LSYS02_AcsP0202_TRM03	CLOSE (R)	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS02	AcsP0203	OPEN	LSYS02_AcsP0203_TRM01	OPEN	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS02	AcsP0203	OPEN	LSYS02_AcsP0203_TRM02	OPEN (R)	
2017/01/23	04:56:07	LNODE01	TPM01	LSYS02	AcsP0203	OPEN	LSYS02_AcsP0203_TRM03	OPEN (R)	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS01	AcsP0101	OPEN (R)	LSYS01_AcsP0101_TRM01	OPEN (R)	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS01	AcsP0101	OPEN (R)	LSYS01_AcsP0101_TRM02	OPEN (R)	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS01	AcsP0101	OPEN (R)	LSYS01_AcsP0101_TRM03	OPEN (R)	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS01	AcsP0102	CLOSE (R)	LSYS01_AcsP0102_TRM01	CLOSE (R)	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS01	AcsP0102	CLOSE (R)	LSYS01_AcsP0102_TRM02	CLOSE (R)	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS01	AcsP0102	CLOSE (R)	LSYS01_AcsP0102_TRM03	CLOSE (R)	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS02	AcsP0201	CLOSE	LSYS02_AcsP0201_TRM01	CLOSE	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS02	AcsP0202	CLOSE	LSYS02_AcsP0202_TRM01	CLOSE	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS02	AcsP0203	CLOSE	LSYS02_AcsP0203_TRM01	CLOSE	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS03	AcsP0301	OPEN	LSYS03_AcsP0301_TRM01	OPEN	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS03	AcsP0301	OPEN	LSYS03_AcsP0301_TRM02	OPEN	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS03	AcsP0302	OPEN	LSYS03_AcsP0302_TRM01	OPEN	
2017/01/23	04:56:07	LNODE01	TPM02	LSYS03	AcsP0302	OPEN	LSYS03_AcsP0302_TRM02	OPEN	

①各種情報

No.	カラム表示名	内容
1	TIME	コマンド実行時刻
2	SRC-LNODENAME	論理ノード名
3	SRC-TPM	TPM 名
5	DEST-LSYS	接続先論理システム名
6	DEST-ACCESSPOINT	接続先アクセスポイント名
7	ACSP-STATUS	パス状態  OPEN           : 活性(1 パス) OPEN(2way)   : 活性(2 パス) OPEN(send)   : 送信端末のみ活性 OPEN(recv)   : 受信端末のみ活性 CLOSE         : 非活性  接続可否状態(パス状態の後に表示) (R)   : 全端末接続拒否
8	TERMINAL	端末名
9	TERMINAL-STATUS	端末の状態  OPEN           : 接続 OPENING       : 接続処理中 CLOSING       : 切断処理中 CLOSE         : 切断 TROUBLE       : 障害 RECOVERING   : 障害復旧中  接続可否状態(パス状態の後に表示) (R)   : 全端末接続拒否

## 2.2.12 dilspathrefstats (論理システム間パス統計情報照会コマンド)

### 名前

dilspathrefstats- 論理システム間パス(常時接続)の統計情報を照会する

### 書式

```
dilspathrefstats [-l LsName ] [ -a AcsP ] [ -M TPM ] [-d { acsp | term } ] [ -c ]
```

```
dilspathrefstats -h
```

### 説明

論理システム間パス(常時接続)の統計情報を照会する。オプションにより端末毎の状態を照会できる。

### オプション

**-l LsName**

照会する論理システムを指定する。

省略した場合、すべての論理システムの状態を照会する。

**-a AcsP**

照会するアクセスポイントを指定する。

省略した場合、すべてのアクセスポイントの状態を照会する。

**-M TPM**

照会する TPMonitor を指定する。TPM コマンドを実行した論理ノード内に存在すること。

省略した場合、コマンドを実行した論理ノード内にあるすべての TPM の状態を照会する。

**-d { acsp | term }**

照会するパス状態の粒度を指定する。省略可。

省略時は **acsp** が指定されたものとして動作する。

**acsp** : アクセスポイント毎のパスの統計情報を照会する。

**term** : 端末毎の統計情報を照会する。

**-c**

照会情報を CSV 形式で出力する。省略可。

省略した場合は整形された状態で出力する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0 処理が正常終了した

1 処理が異常終了した

### 注意

- TPM が停止した場合アクセスポイントの統計情報は 0 となる。再起動後は TPM 再起動後からの統計値が表示される。

### 関連

dilspathref

実行例

+===== LS-PATH STATUS DISPLAY =====+ 2017/01/23 04:56:07					①
LSYSTEM = LSYSTEM01					②
LNODE = LNODE01					
SOURCE DESTINATION					③
TPM	LOGSYSTEM	ACCESSPOINT	SENT	WAIT	④
TPM01	LSYS01	AcsP0101	123	0	
		Acsp0102	456	0	
	LSYS02	AcsP0201	789	10	
		Acsp0202	123	0	
		Acsp0203	456	3	
TPM02	LSYS01	AcsP0101	789	0	
		Acsp0102	123	1	
	LSYS02	AcsP0201	456	0	
		Acsp0202	789	5	
		Acsp0203	123	2	
	LSYS03	AcsP0301	456	1	
		AcsP0302	789	1	
+===== END OF DISPLAY =====+					

- ①コマンド実行時刻
- ②コマンド実行環境

LSYSTEM : 論理システム名

LNODE : 論理ノード名
- ③区分

SOURCE : 起点側

DESTINATION : 接続先側
- ④各種情報

No.	カラム表示名	内容
1	LNODENAME	論理ノード名
2	TPM	TPM 名
3	LOGSYSTEM	通信先論理システム名
4	ACCESSPOINT	通信先アクセスポイント名
5	SENT	累計送信電文数
6	WAIT	滞留電文数

実行例～端末毎の情報照会（-d term）

+===== LS-PATH STATUS DISPLAY =====+ 2017/01/23 04:56:07						①	
LSYSTEM = LSYSTEM01						②	
LNODE = LNODE01							
SOURCE DESTINATION						③	
TPM	LOGSYSTEM	ACCESSPOINT	TERMINAL	SENT	WAIT	④	
TPM01	LSYS01	AcsP0101	LSYS01_AcsP0101_TRM01	41	0		
			LSYS01_AcsP0101_TRM02	41	0		
			LSYS01_AcsP0101_TRM03	41	0		
		AcsP0102	LSYS01_AcsP0102_TRM01	152	0		
			LSYS01_AcsP0102_TRM02	152	0		
			LSYS01_AcsP0102_TRM03	152	0		
		LSYS02	AcsP0201	LSYS02_AcsP0201_TRM01	394	5	
				LSYS02_AcsP0201_TRM02	395	5	
				LSYS02_AcsP0201_TRM03	395	5	
	AcsP0202		LSYS02_AcsP0202_TRM01	41	0		
			LSYS02_AcsP0202_TRM02	41	0		
			LSYS02_AcsP0202_TRM03	41	0		
	AcsP0203	LSYS02_AcsP0203_TRM01	152	1			
		LSYS02_AcsP0203_TRM02	152	1			
		LSYS02_AcsP0203_TRM03	152	1			
TPM02		LSYS01	AcsP0101	LSYS01_AcsP0101_TRM01	263	0	
				LSYS01_AcsP0101_TRM02	263	0	
				LSYS01_AcsP0101_TRM03	263	0	
	AcsP0102		LSYS01_AcsP0102_TRM01	41	0		
			LSYS01_AcsP0102_TRM02	41	0		
			LSYS01_AcsP0102_TRM03	41	1		
	LSYS02		AcsP0201	LSYS02_AcsP0201_TRM01	228	0	
				LSYS02_AcsP0201_TRM02	228	0	
				LSYS02_AcsP0201_TRM03	228	0	
		AcsP0202	LSYS02_AcsP0202_TRM01	263	2		
			LSYS02_AcsP0202_TRM02	263	1		
			LSYS02_AcsP0202_TRM03	263	2		
	AcsP0203	LSYS02_AcsP0203_TRM01	41	1			
		LSYS02_AcsP0203_TRM02	41	1			
		LSYS02_AcsP0203_TRM03	41	0			
LSYS03		AcsP0301	LSYS03_AcsP0301_TRM01	228	1		
			LSYS03_AcsP0301_TRM02	228	0		
			LSYS03_AcsP0301_TRM03	228	0		
	AcsP0302	LSYS03_AcsP0302_TRM01	395	0			
		LSYS03_AcsP0302_TRM02	394	1			
		LSYS03_AcsP0302_TRM03	394	1			
+===== END OF DISPLAY =====+							

- ①コマンド実行時刻
- ②コマンド実行環境
- LOGSYSTEM : 論理システム名
- LNODE : 論理ノード名
- ③区分
- SOURCE : 起点側
- DESTINATION : 接続先側
- ④各種情報

No.	カラム表示名	内容
1	LNODENAME	論理ノード名
2	TPM	TPM 名
3	LOGSYSTEM	通信先論理システム名
4	ACCESSPOINT	通信先アクセスポイント名
6	TERMINAL	端末名
5	SENT	累計送信電文数
6	WAIT	滞留電文数

実行例～CSV形式（-d tpmonitor -c）

TIME, SRC-LNODENAME, SRC-TPM, DEST-LSYS, DEST-ACCESSPOINT, SENT, WAIT
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0101, 123, 0
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0102, 456, 0
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0201, 789, 10
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0202, 123, 0
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0203, 456, 3
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0101, 789, 0
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0102, 123, 1
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0201, 456, 0
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0202, 789, 5
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0203, 123, 2
2017/01/23 04:56:07, LNODE01, TPM02, LSYS03, AcsP0301, 456, 1
2017/01/23 04:56:07, LNODE01, TPM02, LSYS03, AcsP0302, 789, 1

①各種情報

No.	カラム表示名	内容
1	TIME	コマンド実行時刻
2	SRC-LNODENAME	論理ノード名
3	SRC-TPM	TPM 名
4	DEST-LSYS	接続先論理システム名
5	DEST-ACCESSPOINT	接続先アクセスポイント名
6	SENT	累計送信電文数
7	WAIT	滞留電文数

実行例～端末毎の情報照会 CSV 形式(-d term -c)

TIME, SRC-LNODENAME, SRC-TPM, DEST-LSYS, DEST-ACCESSPOINT, TERMINAL, SENT, WAIT	①
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0101, LSYS01_AcsP0101_TRM01, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0101, LSYS01_AcsP0101_TRM02, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0101, LSYS01_AcsP0101_TRM03, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0102, LSYS01_AcsP0102_TRM01, 152, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0102, LSYS01_AcsP0102_TRM02, 152, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS01, AcsP0102, LSYS01_AcsP0102_TRM03, 152, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0201, LSYS02_AcsP0201_TRM01, 394, 5	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0201, LSYS02_AcsP0201_TRM02, 395, 5	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0202, LSYS02_AcsP0202_TRM01, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0202, LSYS02_AcsP0202_TRM02, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0202, LSYS02_AcsP0202_TRM03, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0203, LSYS02_AcsP0203_TRM01, 152, 1	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0203, LSYS02_AcsP0203_TRM02, 152, 1	
2017/01/23 04:56:07, LNODE01, TPM01, LSYS02, AcsP0203, LSYS02_AcsP0203_TRM03, 152, 1	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0101, LSYS01_AcsP0101_TRM01, 263, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0101, LSYS01_AcsP0101_TRM02, 263, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0101, LSYS01_AcsP0101_TRM03, 263, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0102, LSYS01_AcsP0102_TRM01, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0102, LSYS01_AcsP0102_TRM02, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS01, AcsP0102, LSYS01_AcsP0102_TRM03, 41, 1	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0201, LSYS02_AcsP0201_TRM01, 228, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0201, LSYS02_AcsP0201_TRM02, 228, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0202, LSYS02_AcsP0202_TRM01, 263, 2	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0202, LSYS02_AcsP0202_TRM02, 263, 1	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0202, LSYS02_AcsP0202_TRM03, 263, 2	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0203, LSYS02_AcsP0203_TRM01, 41, 1	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0203, LSYS02_AcsP0203_TRM02, 41, 1	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS02, AcsP0203, LSYS02_AcsP0203_TRM03, 41, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS03, AcsP0301, LSYS03_AcsP0301_TRM01, 228, 1	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS03, AcsP0301, LSYS03_AcsP0301_TRM02, 228, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS03, AcsP0302, LSYS03_AcsP0302_TRM01, 395, 0	
2017/01/23 04:56:07, LNODE01, TPM02, LSYS03, AcsP0302, LSYS03_AcsP0302_TRM02, 394, 1	

①各種情報

No.	カラム表示名	内容
1	TIME	コマンド実行時刻
2	SRC-LNODENAME	論理ノード名
3	SRC-TPM	TPM 名
4	DEST-LSYS	接続先論理システム名
5	DEST-ACCESSPOINT	接続先アクセスポイント名
6	TERMINAL	端末名
7	SENT	累計送信電文数
8	WAIT	滞留電文数

## 2.2.13 dincmbhchkctrlinit (DB ヘルスチェックデーモン起動コマンド)

### 名前

dincmbhchkctrlinit - DB ヘルスチェックデーモンを起動する。

### 書式

```
dincmbhchkctrlinit [-t Interval] [-m MaxThread]
dincmbhchkctrlinit -h
```

### 説明

DB ヘルスチェックデーモンを起動する。

### オプション

#### -t Interval

DB ヘルスチェックデーモンの起動監視時間(秒)を 1～65535 の範囲で指定する。省略した場合は 10 秒となる。

#### -m MaxThread

DB ノードの場合に、実表更新スレッドの最大スレッド数を 1～65535 の範囲で指定する。省略した場合は 10 となる。

DB ノード上の DB ヘルスチェックデーモンでは、実表更新スレッドで DB アクセスの可否を監視しているが、DB が無応答の場合には実表更新スレッドを新規に起動し、既に実行済みの実表更新スレッドは停止させるよう制御する。

無応答のままスレッドが停止できない場合に使用するリソースを抑制するため、同時に起動できる最大スレッド数を **-m** で指定する。

#### -h

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

- |   |            |
|---|------------|
| 0 | 処理が正常終了した。 |
| 1 | 処理が異常終了した。 |

### 注意

このコマンドは、AP ノード、OLTP ノード、DB ノードに投入可能である。

## 2.2.14 dincmdbhchkctrlterm (DB ヘルスチェックデーモン停止コマンド)

### 名前

dincmdbhchkctrlterm - DB のヘルスチェックデーモンを停止する。

### 書式

```
dincmdbhchkctrlterm  
dincmdbhchkctrlterm -h
```

### 説明

DB ヘルスチェックデーモンを停止する。

### オプション

**-h**  
ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。

### 注意

このコマンドは、AP ノード、OLTP ノード、DB ノードに投入可能である。

## 2.2.15 dincmdbinit (DB 監視機能初期化コマンド)

### 名前

dincmdbinit - DB 監視機能を起動する。

### 書式

```
dincmdbinit [-w | -c]  
dincmdbinit -h
```

### 説明

DB 監視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う。

### オプション

**-w**

ウォーム起動する。(省略時の既定値)

**-c**

コールド起動する。

**-h**

ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0                      処理が正常終了した。

1                      処理が異常終了した。

### 注意

このコマンドは、AP ノード、OLTP ノード、DB ノードに投入可能である。

### 関連

dincmdbterm

# 2.2.16 dincmdbstat (DB 状態照会コマンド)

## 名前

dincmdbstat - DB の状態を照会する。

## 書式

dincmdbstat [-r | -g]  
dincmdbstat -h

## 説明

DB 監視機能が管理している情報を照会する。

## オプション

- r  
リソースグループと対応するデータベース・インスタンスの一覧を表示する。
- g  
インスタンスグループの状態一覧を表示する。
- h  
ヘルプを表示する。

## カラムの説明

以下に、dincmdbstat が表示するカラムの見出しと各カラムの意味を説明する。

オプションを指定しない場合は、各カラムが以下の順番に従って左から右へプリントされる。

NODENAME	コマンドを投入したノードの名前	
NO	通番	
INSTANCE	インスタンス(ネット・サービス名)	
STATUS	DB 状態	
	NORMAL	正常
	ABNORMAL	異常
	BLOCK	閉塞
	RECONFIG	再構成中
	UNKNOWN	不定

-r オプションを指定した場合は、各カラムが以下の順番に従って左から右へプリントされる。

NODENAME	コマンドを投入したノードの名前	
RGSETNAME	リソースグループセット名	
INSTANCE	現在使用中のインスタンス(ネット・サービス名)	
INSTANCEGRP	インスタンスグループ名	
STATUS	DB 状態	
	NORMAL	正常
	ABNORMAL	異常
	BLOCK	閉塞
	RECONFIG	再構成中
	UNKNOWN	不定

-g オプションを指定した場合は、各カラムが以下の順番に従って左から右へプリントされる。

NODENAME	コマンドを投入したノードの名前	
INSTANCEGRP	インスタンスグループ名	
STATUS	インスタンスグループの全体 DB 状態	
	NORMAL	活性
	ABNORMAL	非活性
	RECONFIG	再構成中
	****	不定

#### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。

#### 注意

このコマンドは、AP ノード、OLTP ノードに投入可能である。

「INSTANCE」「INSTANCEGRP」カラムは、環境定義された DBCTRL-INSTANCE-DBNAME パラメータ、および DBCTRL-INSTANCEGRP-NAME パラメータの文字列全体を出力するため、定義された文字列が長すぎる場合はカラムの表示が複数行になる可能性がある。

## 2.2.17 dincmdbterm (DB 監視機能終了コマンド)

### 名前

dincmdbterm - DB 監視機能を終了する。

### 書式

```
dincmdbterm  
dincmdbterm -h
```

### 説明

DB 監視機能の共有メモリを削除する処理を行う。

### オプション

**-h**  
ヘルプを表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常終了した。
1	処理が異常終了した。

### 注意

このコマンドは、AP ノード、OLTP ノード、DB ノードに投入可能である。

### 関連

dincmdbinit

## 2.2.18 dincpathdmnctrl (パス制御機能デーモン起動停止コマンド)

### 名前

dincpathdmnctrl - パス制御機能の制御デーモン起動・停止を行う

### 書式

```
dincpathdmnctrl { -b | -e }  
dincpathdmnctrl -h
```

### 説明

**dincpathdmnctrl** はパス制御機能の制御デーモン起動および停止を行う。

### オプション

**-b**

制御デーモンを起動する。

**-e**

制御デーモンを停止する。

**-h**

使用方法を表示する。

### 戻り値

0	処理が正常終了した。既に実行済みである。該当ノードで実行する必要がない。
1	処理が異常終了した。

### 注意

本コマンドは、**dincpathinit** および TPBASE 起動後に実行する。

制御デーモンを停止する場合、TPBASE 停止前に実行する。

電文送受信処理中に制御デーモンの停止を実行した場合、処理中の電文は全て破棄される。

該当論理ノードで都度接続プロトコルによる発呼を行わない場合、本コマンドの実行は不要である。

### 関連

dincpathinit, dincpathterm

## 2.2.19 dincmpathinit (パス制御機能初期化)

### 名前

dincmpathinit - 論理システム間パス管理(常時・都度)、論理ノード間パス管理の初期化処理を行う。

### 書式

```
dincmpathinit
dincmpathinit -h
```

### 説明

**dincmpathinit** は論理システム間通信パス管理、論理ノード間通信パス管理各機能の初期化処理を行う。各機能の動作に必要な共有メモリ領域と、論理システム間パス管理(都度)機能の動作に必要なセマフォが確保される。

### オプション

-h  
使用方法を表示する。

### 戻り値

0	処理が正常に終了した。
1	処理が異常終了した。
2	処理が一部失敗した。

### 注意

本コマンドは TPBASE 起動前に実行する。

### 関連

dincmpathterm

## 2.2.20 dincpathterm (パス制御機能終了)

### 名前

dincpathterm - 論理システム間パス管理(常時・都度)・論理ノード間パス管理の終了処理を行う。

### 書式

```
dincpathterm  
dincpathterm -h
```

### 説明

**dincpathterm** は論理システム間通信パス管理、論理ノード間通信パス管理各機能の終了処理を行う。  
**dincpathinit** で確保された共有メモリ領域とセマフォは本コマンドによって解放される。

### オプション

**-h**  
使用方法を表示する。

### 戻り値

0	処理が正常に終了した。
1	処理が異常終了した。
2	処理が一部失敗した。

### 注意

本コマンドは、通信パス管理機能の制御デーモン停止後、TPBASE 停止後に実行する。

### 関連

dincpathinit

## 2.2.21 dinodepathctrl（論理ノード間パス構成端末制御）

### 名前

dinodepathctrl - 論理ノード間パスを構成する端末を制御する

### 書式

```
dinodepathctrl -o { open | close } -n DestNode [ -m DestTPM ] [ -M TPM ] [ -F ]
dinodepathctrl -o { open | close } -T term [ -M TPM ] [ -F ]
dinodepathctrl -o { open | close } -A [ -M TPM ] [ -F ]
dinodepathctrl -h
```

### 説明

**dinodepathctrl** は論理ノード間パスを構成する端末のうち、指定された論理ノード **DestNode** か論理ノード **DestNode** 内の TP モニタ **DestTPM** に接続される端末、あるいは指定された端末 **term** に対し、接続・切断操作を行う。

### オプション

#### -o open

対象の端末に対し、接続操作を行う。

#### -o close

対象の端末に対し、切断操作を行う。

#### -n DestNode

端末の接続先論理ノード名を指定する。

#### -m

端末の接続先 TP モニタ名を指定する。省略時、-n で指定した論理ノード配下の TP モニタ全てが対象となる。

#### -T

接続・切断操作の対象とする端末名を指定する。

#### -M

接続・切断操作を行う自論理ノード側 TP モニタ名を指定する。省略した場合、自論理ノード配下の TP モニタ全てを対象とする。

#### -A

自論理ノードが関係する論理ノード間パスを構成する端末全てを接続・切断操作の対象とする。

#### -F

対象の端末が既に接続済あるいは切断済の状態であっても、接続操作あるいは切断操作を実施する。

#### -h

使用方法を表示する。

### 戻り値

0	処理が正常に終了した。
1	処理が異常終了した。
2	処理が一部失敗した。

### 注意

本コマンドは、TPBASE に対して端末の接続・切断の要求処理を出したことをもって正常と判断する。実際に端末が接続・切断された状態になっているかどうかは、**dinodepathref** 等で確認すること。

**-F** をつけて本コマンドを実行した場合、実際に端末が接続・切断されるまでの間、**dinodepathref** で参照できる端末の状態は更新されない。

ノード停止またはパス切断をする場合は、停止またはパス切断をする前に、対象ノードを閉塞すること。閉塞しない場合は、パス障害と認識して再接続を試みる。

## 関連

dinodepathref

## 2.2.22 dinodepathref (論理ノード間パス情報照会)

### 名前

dinodepathref - 論理ノード間パスの情報を照会する

### 書式

```
dinodepathref [ -M TPM ] [ -p { ap | oltp } ] [ -d { tpmonitor | term } ] [ -c ]  
dinodepathref [ -M TPM ] [ -n DestNode ] [ -d { tpmonitor | term } ] [ -c ]  
dinodepathref -h
```

### 説明

**dinodepathref** は自論理ノードが関係する論理ノード間パスや、それを構成する端末の情報を表示する。

### オプション

#### -M

論理ノード間パス情報の照会対象とする TP モニタ名を指定する。省略可。  
省略時は自論理ノード配下の TP モニタ全てが照会対象となる。

#### -p { ap | oltp }

論理ノード間パスの接続先論理ノードの種別を指定する。省略可。また、**-n** と同時指定不可。  
省略時は全ての種別を照会対象とする。

**ap** : AP ノードを対象とする。

**oltp** : OLTP ノードを対象とする。

#### -n DestNode

照会対象の接続先論理ノードを絞り込む時、その論理ノード名を指定する。省略可。  
また、**-p** と同時指定不可。省略時は全ての論理ノードを照会対象とする。

#### -d { tpmonitor | term }

照会するパス状態の粒度を指定する。省略可。

省略時は **tpmonitor** が指定されたものとして動作する。

**tpmonitor** : TP モニタ毎のパス状態を照会する。

TP モニタ配下の端末が 1 つでも接続済状態であれば活性と表示される。

**term** : 端末毎の接続状態を照会する。

#### -c

照会情報を CSV 形式で出力する。省略可。

省略した場合は整形された状態で出力する。

#### -h

使用方法を表示する。

### 戻り値

0 処理が正常に終了した。

1 処理が異常終了した。

実行例 (-d tpmonitor)

```
+===== T-PATH STATUS DISPLAY =====+ 2017/01/23 04:56:07 ①
LSNAME   = LSYSTEM01 ②
LNODENAME = LNODE01
SOURCE___ DESTINATION_____ ③
TPM_____ TYPE LNODENAME_____ TPM_____ STATUS___ ④
TPM01     AP  LNODE01      TPM02     OPEN
           OLTP LNODE02    TPM01     CLOSE
                        TPM02     OPEN
TPM02     AP  LNODE01      TPM01     OPEN
           OLTP LNODE02    TPM01     CLOSE
                        TPM02     OPEN
+===== END OF DISPLAY =====+
```

①コマンド実行時刻

②自論理ノード情報

No.	表示名	内容
1	LSNAME	論理システム名
2	LNODENAME	論理ノード名

③区分

SOURCE               : 起点側  
DESTINATION         : 接続先側

④各種情報

No.	カラム表示名	内容
1	LNODENAME	論理ノード名
2	TPM	TP モニタ名
3	TYPE	論理ノード種別 AP       : AP ノード OLTP     : OLTP ノード
4	STATUS	パス活性状態 OPEN     : 活性 CLOSE    : 非活性 BLOCK    : 接続先論理ノードが閉塞状態 PREBLOCK : 接続先論理ノードが予閉塞状態

実行例 (-d term)

+===== T-PATH STATUS DISPLAY =====+ 2017/01/23 04:56:07 ①										
LSNAME = LSYSTEM01 ②										
LNODENAME = LNODE01										
SOURCE DESTINATION ③										
TPM	TYPE	LNODENAME	TPM	STATUS	TERMINAL	STATUS ④				
TPM01	AP	LNODE01	TPM02	OPEN	DXTP_LN00101_TPM01_T0001	OPEN				
					DXTP_LN00101_TPM01_T0002	OPEN				
					DXTP_LN00101_TPM01_T0003	OPEN				
	OLTP	LNODE02	TPM01	CLOSE	DXTP_LN00102_TPM00_T0001	CLOSE				
					DXTP_LN00102_TPM00_T0002	OPENING				
					DXTP_LN00102_TPM00_T0003	CLOSE				
			TPM02	OPEN	DXTP_LN00102_TPM01_T0001	OPEN				
					DXTP_LN00102_TPM01_T0002	OPEN				
					DXTP_LN00102_TPM01_T0003	OPEN				
	TPM02	AP	LNODE01	TPM01	OPEN	DXTP_LN00101_TPM00_T0001	OPEN			
						DXTP_LN00101_TPM00_T0002	OPEN			
						DXTP_LN00101_TPM00_T0003	OPEN			
OLTP		LNODE02	TPM01	CLOSE	DXTP_LN00102_TPM00_T0001	CLOSE				
					DXTP_LN00102_TPM00_T0002	OPENING				
					DXTP_LN00102_TPM00_T0003	CLOSE				
		TPM02	OPEN	DXTP_LN00102_TPM01_T0001	OPEN					
				DXTP_LN00102_TPM01_T0002	OPEN					
				DXTP_LN00102_TPM01_T0003	OPEN					
+===== END OF DISPLAY =====+										

①コマンド実行時刻

②自論理ノード情報

No.	表示名	内容
1	LSNAME	論理システム名
2	LNODENAME	論理ノード名

③区分

SOURCE : 起点側  
DESTINATION : 接続先側

④各種情報

No.	カラム表示名	内容
1	LNODENAME	論理ノード名
2	TPM	TP モニタ名
3	TYPE	論理ノード種別 AP : AP ノード OLTP : OLTP ノード
4	STATUS (TPMONITER の直後)	TP モニタ毎のパス活性状態 OPEN : 活性 CLOSE : 非活性 BLOCK : 接続先論理ノードが閉塞状態 PREBLOCK : 接続先論理ノードが予閉塞状態
5	TERMINAL	端末名
6	STATUS (TERMINAL の直後)	端末の状態 OPEN : 接続 OPENING : 接続処理中 CLOSING : 切断処理中 CLOSE : 切断 TROUBLE : 障害 RECOVERING : 障害復旧中

実行例 (-d tpmonitor -c)

TIME, LSNAME, SRC-LNODENAME, SRC-TPM, DEST-TYPE, DEST-LNODENAME, DEST-TPM, TPATH-STATUS ①
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, AP, LNODE01, TPM02, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM01, CLOSE
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM02, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, AP, LNODE01, TPM01, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM01, CLOSE
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM02, OPEN

①各種情報

No.	カラム表示名	内容
1	TIME	コマンド実行時刻
2	LSNAME	論理システム名
3	SRC-LNODENAME	起点側論理ノード名
4	SRC-TPM	起点側 TP モニタ名
5	DEST-TYPE	論理ノード種別 AP       : AP ノード OLTP     : OLTP ノード
6	DEST-LNODENAME	接続先論理ノード名
7	DEST-TPM	接続先 TP モニタ名
8	TPATH-STATUS	パス活性状態 OPEN     : 活性 CLOSE    : 非活性 BLOCK    : 接続先論理ノードが閉塞状態 PREBLOCK : 接続先論理ノードが予閉塞状態

# 実行例 (-d term -c)

```

TIME, LSNAME, SRC-LNODENAME, SRC-TPM, DEST-TYPE, DEST-LNODENAME, DEST-TPM, TPATH-STATUS, TERMINAL, TERMINAL-STATUS ①
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, AP, LNODE01, TPM02, OPEN, DXTP_LN00101_TPM01_T0001, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, AP, LNODE01, TPM02, OPEN, DXTP_LN00101_TPM01_T0002, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, AP, LNODE01, TPM02, OPEN, DXTP_LN00101_TPM01_T0003, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM01, CLOSE, DXTP_LN00102_TPM00_T0001, CLOSE
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM01, CLOSE, DXTP_LN00102_TPM00_T0002, OPENING
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM01, CLOSE, DXTP_LN00102_TPM00_T0003, CLOSE
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM02, OPEN, DXTP_LN00102_TPM01_T0001, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM02, OPEN, DXTP_LN00102_TPM01_T0002, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM02, OPEN, DXTP_LN00102_TPM01_T0003, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, AP, LNODE01, TPM01, OPEN, DXTP_LN00101_TPM00_T0001, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, AP, LNODE01, TPM01, OPEN, DXTP_LN00101_TPM00_T0002, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, AP, LNODE01, TPM01, OPEN, DXTP_LN00101_TPM00_T0003, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM01, CLOSE, DXTP_LN00102_TPM00_T0001, CLOSE
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM01, CLOSE, DXTP_LN00102_TPM00_T0002, OPENING
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM01, CLOSE, DXTP_LN00102_TPM00_T0003, CLOSE
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM02, OPEN, DXTP_LN00102_TPM01_T0001, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM02, OPEN, DXTP_LN00102_TPM01_T0002, OPEN
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM02, OPEN, DXTP_LN00102_TPM01_T0003, OPEN

```

## ①各種情報

No.	カラム表示名	内容
1	TIME	コマンド実行時刻
2	LSNAME	論理システム名
3	SRC-LNODENAME	起点側論理ノード名
4	SRC-TPM	起点側 TP モニタ名
5	DEST-TYPE	論理ノード種別 AP : AP ノード OLTP : OLTP ノード
6	DEST-LNODENAME	接続先論理ノード名
7	DEST-TPM	接続先 TP モニタ名
8	TPATH-STATUS	TP モニタ毎のパス活性状態 OPEN : 活性 CLOSE : 非活性 BLOCK : 接続先論理ノードが閉塞状態 PREBLOCK : 接続先論理ノードが予閉塞状態
9	TERMINAL	端末名
10	TERMINAL-STATUS	端末の状態 OPEN : 接続 OPENING : 接続処理中 CLOSING : 切断処理中 CLOSE : 切断 TROUBLE : 障害 RECOVERING : 障害復旧中

## 2. 2. 23      **dinodepathrefstats**（論理ノード間パス統計情報照会）

### 名前

**dinodepathrefstats** - 論理ノード間パスの統計情報を取得する

### 書式

```
dinodepathrefstats [ -M TPM ] [ -p { ap | oltp } ] [ -d { tpmonitor | term } ] [ -c ]
dinodepathrefstats [ -M TPM ] [ -n DestNode ] [ -d { tpmonitor | term } ] [ -c ]
dinodepathrefstats -h
```

### 説明

**dinodepathrefstats** は自論理ノードが関係する論理ノード間パス、あるいはそれを構成する端末を通じて送信された電文の数と滞留している電文の数を統計情報として表示する。

### オプション

**-M**

論理ノード間パス統計情報の照会対象とする TP モニタ名を指定する。省略可。  
省略時は自論理ノード配下の TP モニタ全てが照会対象となる。

**-p { ap | oltp }**

論理ノード間パスの接続先論理ノードの種別を指定する。省略可。  
省略時は全ての種別を照会対象とする。

**ap**       : AP ノードを対象とする。

**oltp**     : OLTP ノードを対象とする。

**-n DestNode**

照会対象の接続先論理ノードを絞り込む時、その論理ノード名を指定する。省略可。  
また、**-p** と同時指定不可。省略時は全ての論理ノードを照会対象とする。

**-d { tpmonitor | term }**

照会する統計情報の粒度を指定する。省略可。

省略時は **tpmonitor** が指定されたものとして動作する。

**tpmonitor**       : TP モニタ毎の統計情報を表示する。TP モニタ配下の全端末の統計情報の合算。

**term**            : 端末毎の統計情報を表示する。

**-c**

照会情報を CSV 形式で出力する。省略可。  
省略した場合は整形された状態で出力する。

**-h**

使用方法を表示する。

### 戻り値

0                    処理が正常に終了した。

1                    処理が異常終了した。

### 注意

出力される送信電文数は、TP モニタ起動時点からの累計数である。表示をリセットする手段はない。

実行例 (-d tpmonitor)

+===== T-PATH STATISTICS DISPLAY =====+ 2017/01/23 04:56:07 ①					
LSNAME = LSYSTEM01 ②					
LNODENAME = LNODE01					
SOURCE DESTINATION ③					
TPM	TYPE	LNODENAME	TPM	SENT	WAIT ④
TPM01	AP	LNODE01	TPM02	123	0
	OLTP	LNODE02	TPM01	456	0
			TPM02	789	2
TPM02	AP	LNODE01	TPM01	123	0
	OLTP	LNODE02	TPM01	456	1
			TPM02	789	2
+===== END OF DISPLAY =====+					

①コマンド実行時刻

②自論理ノード情報

No.	表示名	内容
1	LSNAME	論理システム名
2	LNODENAME	論理ノード名

③区分

SOURCE : 起点側  
DESTINATION : 接続先側

④各種情報

No.	カラム表示名	内容
1	TPM	TP モニタ名
2	LNODENAME	論理ノード名
3	TYPE	論理ノード種別
		AP : AP ノード OLTP : OLTP ノード
4	SENT	累計送信電文数
5	WAIT	滞留電文数

実行例 (-d term)

+===== T-PATH STATISTICS DISPLAY =====+ 2017/01/23 04:56:07 ①									
LSNAME = LSYSTEM01 ②									
LNODENAME = LNODE01									
SOURCE DESTINATION ③									
TPM	TYPE	LNODENAME	TPM	SENT	WAIT	TERMINAL	SENT	WAIT	④
TPM01	AP	LNODE01	TPM02	123	0	DXTP_LN00101_TPM01_T0001	41	0	
						DXTP_LN00101_TPM01_T0002	41	0	
						DXTP_LN00101_TPM01_T0003	41	0	
	OLTP	LNODE02	TPM01	456	0	DXTP_LN00102_TPM00_T0001	152	0	
						DXTP_LN00102_TPM00_T0002	152	0	
						DXTP_LN00102_TPM00_T0003	152	0	
			TPM02	789	2	DXTP_LN00102_TPM01_T0001	263	1	
						DXTP_LN00102_TPM01_T0002	263	1	
						DXTP_LN00102_TPM01_T0003	263	0	
TPM02	AP	LNODE01	TPM01	123	0	DXTP_LN00101_TPM00_T0001	41	0	
						DXTP_LN00101_TPM00_T0002	41	0	
						DXTP_LN00101_TPM00_T0003	41	0	
	OLTP	LNODE02	TPM01	456	1	DXTP_LN00102_TPM00_T0001	152	0	
						DXTP_LN00102_TPM00_T0002	152	1	
						DXTP_LN00102_TPM00_T0003	152	0	
			TPM02	789	2	DXTP_LN00102_TPM01_T0001	263	0	
						DXTP_LN00102_TPM01_T0002	263	0	
						DXTP_LN00102_TPM01_T0003	263	2	
+===== END OF DISPLAY =====+									

①コマンド実行時刻

②自論理ノード情報

No.	表示名	内容
1	LSNAME	論理システム名
2	LNODENAME	論理ノード名

③区分

SOURCE : 起点側  
DESTINATION : 接続先側

④各種情報

No.	カラム表示名	内容
1	TPM	TP モニタ名
2	LNODENAME	論理ノード名
3	TYPE	論理ノード種別 AP : AP ノード OLTP : OLTP ノード
4	SENT (TPM の次)	TP モニタ毎の累計送信電文数
5	WAIT (TPM の次)	TP モニタ毎の滞留電文数
6	TERMINAL	端末名
7	SENT (TERMINAL の次)	端末毎の累計送信電文数
8	WAIT (TERMINAL の次)	端末毎の滞留電文数

実行例 (-d tpmonitor -c)

TIME, LSNAME, SRC-LNODENAME, SRC-TPM, DEST-TYPE, DEST-LNODENAME, DEST-TPM, TPATH-SENT, TPATH-WAIT ①
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, AP, LNODE01, TPM02, 123, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM01, 456, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM02, 789, 2
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, AP, LNODE01, TPM01, 123, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM01, 456, 1
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM02, 789, 2

①各種情報

No.	カラム表示名	内容
1	TIME	コマンド実行時刻
2	LSNAME	論理システム名
3	SRC-LNODENAME	起点側論理ノード名
4	SRC-TPM	起点側 TP モニタ名
5	DEST-TYPE	論理ノード種別 AP       : AP ノード OLTP     : OLTP ノード
6	DEST-LNODENAME	接続先論理ノード名
7	DEST-TPM	接続先 TP モニタ名
8	TPATH-SENT	TP モニタ毎の累計送信電文数
9	TPATH-WAIT	TP モニタ毎の滞留電文数

# 実行例 (-d term -c)

```

TIME, LSNAME, SRC-LNODENAME, SRC-TPM, DEST-TYPE, DEST-LNODENAME, DEST-TPM, TPATH-SENT, TPATH-WAIT, TERMINAL, TERMINAL-SENT, TERMINAL-WAIT
①
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, AP, LNODE01, TPM02, 123, 0, DXTP_LN00101_TPM01_T0001, 41, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, AP, LNODE01, TPM02, 123, 0, DXTP_LN00101_TPM01_T0002, 41, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, AP, LNODE01, TPM02, 123, 0, DXTP_LN00101_TPM01_T0003, 41, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM01, 456, 0, DXTP_LN00102_TPM00_T0001, 152, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM01, 456, 0, DXTP_LN00102_TPM00_T0002, 152, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM01, 456, 0, DXTP_LN00102_TPM00_T0003, 152, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM02, 789, 2, DXTP_LN00102_TPM01_T0001, 263, 1
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM02, 789, 2, DXTP_LN00102_TPM01_T0002, 263, 1
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM01, OLTP, LNODE02, TPM02, 789, 2, DXTP_LN00102_TPM01_T0003, 263, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, AP, LNODE01, TPM01, 123, 0, DXTP_LN00101_TPM00_T0001, 41, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, AP, LNODE01, TPM01, 123, 0, DXTP_LN00101_TPM00_T0002, 41, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, AP, LNODE01, TPM01, 123, 0, DXTP_LN00101_TPM00_T0003, 41, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM01, 456, 1, DXTP_LN00102_TPM00_T0001, 152, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM01, 456, 1, DXTP_LN00102_TPM00_T0002, 152, 1
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM01, 456, 1, DXTP_LN00102_TPM00_T0003, 152, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM02, 789, 2, DXTP_LN00102_TPM01_T0001, 263, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM02, 789, 2, DXTP_LN00102_TPM01_T0002, 263, 0
2017/01/23 04:56:07, LSYSTEM01, LNODE01, TPM02, OLTP, LNODE02, TPM02, 789, 2, DXTP_LN00102_TPM01_T0003, 263, 2

```

## ①各種情報

No.	カラム表示名	内容
1	TIME	コマンド実行時刻
2	LSNAME	論理システム名
3	SRC-LNODENAME	起点側論理ノード名
4	SRC-TPM	起点側 TP モニタ名
5	DEST-TYPE	論理ノード種別 AP : AP ノード OLTP : OLTP ノード
6	DEST-LNODENAME	接続先論理ノード名
7	DEST-TPM	接続先 TP モニタ名
8	TPATH-SENT	TP モニタ毎の累計送信電文数
9	TPATH-WAIT	TP モニタ毎の滞留電文数
10	TERMINAL	端末名
11	TERMINAL-SENT	端末毎の累計送信電文数
12	TERMINAL-WAIT	端末毎の滞留電文数

## 2.2.24 diotcref (都度接続管理機能 定義照会コマンド)

### 名前

diotcref - 都度接続管理機能の定義情報を表示する。

### 書式

```
diotcref
diotcref -s
diotcref -a AccessPoint
diotcref -l LsName
diotcref -h
```

### 説明

都度接続管理機能の定義情報、および、都度接続管理機能で扱う宛先情報を表示する。  
オプションを省略した場合、全ての情報が表示される。

### オプション

**-s**

起動状況情報と共通定義情報のみを出力する。

**-a AccessPoint**

指定した宛先アクセスポイントの定義情報のみを出力する。

**-l LsName**

指定した論理システム配下にある宛先アクセスポイントの定義情報のみを出力する。

**-h**

使用方法を表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

- |   |                    |
|---|--------------------|
| 0 | 処理が正常終了した。         |
| 1 | 処理が異常終了した。         |
| 2 | 指定した対象の情報が存在しなかった。 |

### 注意

外部論理システムに都度接続プロトコルを使用する宛先が存在しない場合は、表示対象が存在しない旨が画面出力される。

dincpathinit を実行している必要がある。

### 関連

dincpathinit, dincpathterm

実行例

```
+===== ONE-TIME CONNECTION MANAGEMENT DISPLAY =====+ 2017/09/22 00:30:01 ①
LSNAME      = Ls00      ②
LNODENAME   = OLTP01    ③
+----- STATUS INFORMATION -----+ ④
STATUS      = ACT
WATCH DAEMON PID = 15375
+----- COMMON INFORMATION -----+ ⑤
DAEMON PROCESS NUM = 4
DAEMON PROCESS TOTAL = 8
RELAY THREAD NUM   = 1
SEND THREAD NUM    = 3
RECEIVE THREAD NUM = 3
WATCH INTERVAL     = 5
WATCH RETRY NUM    = 3
AUTO STOP MODE     = NO
ALERT UNLOCK LEVEL = 90 %
ACCEPT QUEUE SIZE  = 100
DAEMON INIT EXIT   = InitExit
DAEMON TERM EXIT   = TermExit
LISTENER INIT EXIT = LsnrInitExit
LISTENER TERM EXIT = LsnrTermExit
+----- DESTINATION ACCESS POINT INFORMATION -----+ ⑥
LSNAME      = Ls01
ACCESS POINT NAME = AcsPoint01
MAXCONNECT  = 10
+----- PROTOCOL INFORMATION -----+ ⑦
PROTOCOL NAME = Protocol01
LISTENER MAIN EXIT = LsnrMainExit
HEADER LENGTH = 128
+----- DESTINATION LISTENER INFORMATION -----+ ⑧
NAME_          IPADDR_          PORT_
Lsnr01         192.168.100.200  20001
Lsnr02         192.168.90.201   20001
:
+----- MESSAGE CONTROL INFORMATION -----+ ⑨
GROUP  STHRLIMIT  RTHRLIMIT  SQUESIZE  RQESIZE  TIMEOUT  LSNRWAIT  ERREXIT_
1      3          3          300        500      60      AUTO   ErrExit01
2      2          2          200        500      120     5      ErrExit02
:
+----- ACCESS POINT INFORMATION -----+
LSNAME      = Ls02
ACCESS POINT NAME = AcsPoint02
:
+===== END OF DISPLAY =====+
```

- ① コマンド実行時刻
- ② 自論理システム名
- ③ 自論理ノード名
- ④ 都度接続管理機能 稼働状況
  - STATUS 制御デーモン起動状態
  - ACT 起動中
  - TERM 停止中
  - WATCH DAEMON PID 監視デーモンプロセス ID（未起動の場合は”---“）
- ⑤ 都度接続管理機能 共通定義情報
  - DAEMON PROCESS NUM ITPBASE あたりの都度接続送信デーモン多重度
  - DAEMON PROCESS TOTAL 論理ノード内の都度接続送信デーモン総プロセス数
  - RELAY THREAD NUM 1 デーモンプロセスの中継スレッド多重度
  - SEND THREAD NUM 1 デーモンプロセスの送信スレッド多重度
  - RECEIVE THREAD NUM 1 デーモンプロセスの受信スレッド多重度
  - WATCH INTERVAL 都度接続管理機能の監視間隔
  - WATCH RETRY NUM 都度接続送信デーモン再起動リトライ回数
  - AUTO STOP MODE 監視デーモン自動停止モード
  - NO 自動停止しない
  - DAEMON いずれかの都度接続送信デーモンの再起動がリトライオーバーした時

	TPBASE	いずれかの TPBASE 配下の都度接続送信デーモンが全滅した時
	NODE	論理ノード内の都度接続送信デーモンが全滅した時
ALERT UNLOCK LEVEL		滞留件数閾値超過警告メッセージの出力抑止解除レベル (%)
ACCEPT QUEUE SIZE		業務アプリケーション(TPP)から要求を受け付ける(listen())ソケットのバックログ
DAEMON INIT EXIT		送受信エラー初期化出口名 (未定義時は"---"が出力される)
DAEMON TERM EXIT		送受信エラー終了出口名 (未定義時は"---"が出力される)
LISTENER INIT EXIT		電文種別決定初期化出口名 (未定義時は"---"が出力される)
LISTENER TERM EXIT		電文種別決定終了出口名 (未定義時は"---"が出力される)
⑥ 宛先アクセスポイント情報		
LSNAME		論理システム名
ACCESS POINT NAME		宛先アクセスポイント名
MAXCONNECT		同時接続上限数
⑦ プロトコル情報		
PROTOCOL NAME		プロトコル名
LISTENER MAIN EXIT		電文種別決定出口名
HEADER LENGTH		ユーザヘッダ長
⑧ 宛先アクセスポイント配下リスナ情報		
NAME		リスナ名
IPADDR		IP アドレス
PORT		ポート番号
⑨ 宛先アクセスポイント配下定義グループ情報		
GROUP		定義グループ番号
STHRLIMIT		送信スレッド占有上限数
RTHRLIMIT		受信スレッド占有上限数
SQUESIZE		送信待ちキューサイズ
RQUESIZE		応答受信待ちキューサイズ
TIMEOUT		送受信タイムアウト既定値(秒)
LSNRWAIT		1 リスナあたりの最大送信待ち時間(秒)
		自動計算(定義省略時)の場合は"AUTO"が表示される
ERREXIT		送受信エラー出口名

## 2.2.25 diotcrefstats (都度接続管理機能 統計情報照会コマンド)

### 名前

diotcrefstats - 都度接続管理機能の統計情報を出力する。

### 書式

```
diotcrefstats [-g] [-c] [-v] [-z]
diotcrefstats [-g] [-c] [-v] [-z] -l LsName
diotcrefstats [-g] [-c] [-v] [-z] -a AccessPoint
diotcrefstats -h
```

### 説明

都度接続プロトコルで発呼した電文の統計情報を表示する。

### オプション

#### -l LsName

指定した論理システム配下にある宛先アクセスポイントの処理件数を表示する。  
省略した場合は全ての宛先アクセスポイントが表示対象になる。

#### -a AccessPoint

指定した宛先アクセスポイントの処理件数を表示する。  
省略した場合は全ての宛先アクセスポイントが表示対象になる。

#### -g

定義グループ単位の処理件数を表示する。  
省略した場合は、宛先アクセスポイント配下の全定義グループの処理件数を合算して表示する。

#### -c

CSV 形式で出力する。

#### -v

詳細な件数項目を出力する。

#### -z

画面表示後に処理件数(累積)をリセットする。  
-l または -a オプションを使用した場合、表示対象の処理件数のみがリセットされる。

#### -h

使用方法を表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

- |   |                    |
|---|--------------------|
| 0 | 処理が正常終了した。         |
| 1 | 処理が異常終了した。         |
| 2 | 指定した対象の情報が存在しなかった。 |

### 注意

外部論理システムに都度接続プロトコルを使用する宛先が存在しない場合は、表示対象が存在しない旨が画面出力される。

dincpathinit を実行している必要がある。

電文送信処理が動作中に本コマンドを実行した場合、表示件数はタイミングによって誤差が生じる。

## 関連

dincmpathinit, dincmpathterm

## 実行例(オプションなし)

```
+===== ONE-TIME CONNECTION COUNTER DISPLAY ===== 2017/09/22 00:30:01 ①
LSNAME      = Ls00      ②
LNODENAME    = OLTP01   ③
LSNAME_____ ACCESSPOINT_____ STARTTIME_____ EXECUTING_ DONE_____ RESPONSE_ ERRORRESP_ DISCARD_ ④
LS01         AcsPoint01    2017/09/21 21:30:00         10      4228      2830        10         3
LS02         AcsPoint02    2017/09/21 21:30:00          0         0         0          0         0
+===== END OF DISPLAY =====
```

## 実行例(-g)

```
+===== ONE-TIME CONNECTION COUNTER DISPLAY ===== 2017/09/22 00:30:01 ①
LSNAME      = Ls00      ②
LNODENAME    = OLTP01   ③
LSNAME_____ ACCESSPOINT_____ GROUP STARTTIME_____ EXECUTING_ DONE_____ RESPONSE_ ERRORRESP_ DISCARD_ ④
LS01         AcsPoint01      1 2017/09/21 21:30:00         10      2830      2830        10         0
                2 2017/09/21 21:30:00          0      1398         0          0         3
LS02         AcsPoint02      1 2017/09/21 21:30:00          0         0         0          0         0
+===== END OF DISPLAY =====
```

## 実行例(-c)

```
LSNAME, ACCESSPOINT, STARTTIME, ENDTIME, EXECUTING, DONE, RESPONSE, ERRORRESP, DISCARD ④
LS01, AcsPoint01, 2017/09/21 21:30:00, 2017/09/22 00:30:01, 10, 4228, 2830, 10, 3
LS02, AcsPoint02, 2017/09/21 21:30:00, 2017/09/22 00:30:01, 0, 0, 0, 0, 0
```

## 実行例(-c -g)

```
LSNAME, ACCESSPOINT, GROUP, STARTTIME, ENDTIME, EXECUTING, DONE, RESPONSE, ERRORRESP, DISCARD ④
LS01, AcsPoint01, 1, 2017/09/21 21:30:00, 2017/09/22 00:30:01, 10, 2830, 2830, 10, 0
LS01, AcsPoint01, 2, 2017/09/21 21:30:00, 2017/09/22 00:30:01, 0, 1398, 0, 0, 3
LS02, AcsPoint02, 1, 2017/09/21 21:30:00, 2017/09/22 00:30:01, 0, 0, 0, 0, 0
```

## 実行例(-c -g -v)

```
LSNAME, ACCESSPOINT, GROUP, STARTTIME, ENDTIME, EXECUTING, EXECSEND, EXECRECV, DONE, RESPONSE, ERRORRESP, DISCARD, PRETMOUT, CONNERR, CONNTMOU
T, SENDERR, SENDTMOUT, RECVERR, RECVTMOUT, STHRLIMIT, RTHRLIMIT ④
LS01, AcsPoint01, 1, 2017/09/21 21:30:00, 2017/09/22 00:30:01, 10, 1, 9, 2830, 2830, 10, 0, 3, 0, 1, 0, 0, 5, 1, 30, 5
LS01, AcsPoint01, 2, 2017/09/21 21:30:00, 2017/09/22 00:30:01, 0, 0, 0, 1398, 0, 0, 3, 3, 0, 0, 0, 0, 0, 0, 1300, 0
LS02, AcsPoint02, 1, 2017/09/21 21:30:00, 2017/09/22 00:30:01, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
```

- ① コマンド実行時刻
- ② 自論理システム名
- ③ 自論理ノード名

## ④ 各種情報

No.	カラム表示名	表示条件	内容
1	LSNAME	常時	宛先アクセスポイントの論理システム名
2	ACCESSPOINT	常時	宛先アクセスポイント名
3	GROUP	-g 指定	定義グループ番号
4	STARTTIME	常時	計測開始時刻
5	ENDTIME	-c 指定	コマンド実行時刻(①と同じ値)
6	EXECUTING	常時	デーモン内で仕掛中(宛先からの応答待ち数含む)の電文件数 送信スレッドで要求電文の受信が未完了の電文は計上されない 本件数は-z 指定を行ってもリセットはされない
7	EXECSND	-v 指定	EXECUTING の件数のうち、宛先へ送信中の件数
8	EXECRCV	-v 指定	EXECUTING の件数のうち、宛先から受信(応答待ち含む)の件数
9	DONE	常時	正常終了した電文の件数(累積)
10	RESPONSE	常時	正常終了した電文のうち、宛先から応答電文を受信した件数(累積)
11	ERRORRESP	常時	エラー応答を送信した件数(累積)
12	DISCARD	常時	要求電文または応答電文を破棄した件数(累積)
13	PRETMOUT	-v 指定	送信前タイムアウト発生件数(累積)
14	CONNERROR	-v 指定	接続エラー発生件数(累積)
15	CONNTMOUT	-v 指定	接続タイムアウト発生件数(累積)
16	SENDERERROR	-v 指定	送信エラー発生件数(累積)
17	SENDTMOUT	-v 指定	送信タイムアウト発生件数(累積)
18	RCVERROR	-v 指定	受信エラー発生件数(累積)
19	RCVTMOUT	-v 指定	受信タイムアウト発生件数(累積)
20	STHRLIMIT	-v 指定	送信スレッド占有数制限による送信保留発生件数(累積)
21	RTHRLIMIT	-v 指定	受信スレッド占有数制限による受信保留発生件数(累積)

## 2.3 メモリキャッシュ

### 2.3.1 コマンド一覧

diimblock	インメモリサーバの閉塞／閉塞解除を行う
diimcheck	全ノードの環境定義情報が一致しているかをチェックする
diimchg	メモリキャッシュの環境定義を動的に置換する
diimchghash	diimchg で展開された新ハッシュ値を有効にする
diimctrl	インメモリサーバの起動または停止を行う
diiminit	メモリキャッシュの起動を行う
diimmod	メモリキャッシュの動作パラメータを変更する
diimnotify	メモリキャッシュ機能に通知を行う
diimperfref	メモリキャッシュの性能情報を照会する
diimperfreset	メモリキャッシュの統計情報を初期化する
diimquegc	IMS キューを解放する
diimref	メモリキャッシュの情報を照会する
diimtblopenclose	表のオープン／クローズを行う
diimterm	メモリキャッシュの停止を行う
ditamswap	TAM のマスタ切替を行う
ditamswitch	TAM のスレーブをマスタに昇格する

## 2.3.2 diimblock(インメモリサーバ閉塞コマンド)

### 名前

diimblock - インメモリサーバの閉塞／閉塞解除を行う

### 書式

```
diimblock -a [-A] [-T TIMEOUT]
diimblock -a -r REPLICATION_GROUP_ID [-T TIMEOUT]
diimblock -a -m MAP_ID [-T TIMEOUT]
diimblock -b [-A] [-T TIMEOUT]
diimblock -b -r REPLICATION_GROUP_ID [-T TIMEOUT]
diimblock -b -m MAP_ID [-T TIMEOUT]
diimblock -h
```

### 説明

**diimblock** は、インメモリサーバの閉塞または閉塞解除を行う。

### オプション

**-a**

閉塞解除する。

**-b**

通常モードで閉塞する。

**-A**

全レプリケーショングループの IMS アクセスサーバを閉塞／閉塞解除する。(既定値)

**-r REPLICATION\_GROUP\_ID**

特定レプリケーショングループの IMS アクセスサーバを閉塞／閉塞解除する場合に、対象の IMS アクセスサーバが属するレプリケーショングループの ID を指定する。

**-T TIMEOUT**

コマンドのタイムアウト時間を指定する。(秒単位:0～2147483647、0 は無制限、既定値:30)

**-m MAP\_ID**

特定 MAP の IMS アクセススレッドを閉塞／閉塞解除する場合に、対象の IMS アクセススレッドが属する MAP の ID を指定する。

**-h**

使用方法を表示する。

### 戻り値

diimblock は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- アクセス中に閉塞すると、アクセスエラーとなり、エラーとなったサービスは、全ての MAP に対し、アクセスが出来なくなる。

## 2.3.3 diimcheck(インメモリサーバ所在情報整合性チェックコマンド)

### 名前

diimcheck - 全ノードのインメモリサーバ所在管理テーブルの情報が一致しているかをチェックする

### 書式

diimcheck [-T TIMEOUT]

diimcheck -h

### 説明

**diimcheck** は、全ノードのインメモリサーバ所在管理テーブルが一致しているかをチェックする。

一部ノードの環境動的置換漏れなどの運用ミスを防ぐため、**diimchghash** 実行前に **diimcheck** を実行することを推奨する。

### オプション

**-T TIMEOUT**

コマンドのタイムアウト時間を指定する。(秒単位:0~2147483647、0は無制限、既定値:0)

**-h**

使用方法を表示する。

### カラムの説明

以下に、diimcheck が表示するカラムの見出しと各カラムの意味を説明する。各カラムは以下の順番に従って左から右へプリントされる。

LNODE      論理ノード名

RESULT      比較結果

OK          コマンドを実行したノードとインメモリサーバ所在管理テーブルが一致

NG          コマンドを実行したノードとインメモリサーバ所在管理テーブルが不一致

### 戻り値

diimcheck は、以下のいずれかの終了ステータスを戻して終了する。

0                  正常終了。ノード間で環境定義の不整合なし。

1                  異常終了

2                  警告終了。ノード間で環境定義の不整合あり。

### 注意

- 本コマンドは、OLTP ノードに投入可能である。

### 関連

diimchghash

## 2.3.4 diimchg(メモリキャッシュ環境定義置換コマンド)

### 名前

diimchg - メモリキャッシュの環境定義を動的に置換する

### 書式

```
diimchg
diimchg -d
diimchg -h
```

### 説明

**diimchg** はインメモリサーバおよびインメモリサーバ所在管理の環境定義情報を動的変更する。

### オプション

**-d**

SG 定義から削除されたエントリを共有メモリから削除する。

**-h**

使用方法を表示する。

### 戻り値

diimchg は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了

### 注意

- 構成情報変更時は、インメモリサーバ所在管理が存在する全ノードで diimchg を実行する必要がある。
- MAP とハッシュ値の対応は変更されない。ハッシュ値変更や MAP 追加を行う場合は、全ノードの diimchg 実行完了後に、diimchghash により、ハッシュ値切替を行う必要がある。
- SG 定義から MAP 情報や TAM 所在情報を削除した場合、パラメータ省略の diimchg を実行し、該当エントリを完全に使用しないことを確認後に diimchg -d を実行する必要がある。
- diimchg -d では、SG 情報の読み込みを行わないため、前回の diimchg 後に変更された情報は反映されない。

### 関連

diimchghash, diimcheck

## 2.3.5 diimchghash(ハッシュ値切替コマンド)

### 名前

diimchghash - diimchg で展開された新ハッシュ値を有効にする

### 書式

diimchghash [-T TIMEOUT]

diimchghash -h

### 説明

**diimchghash** は diimchg で動的変更されたハッシュ値を有効にする。

### オプション

**-T TIMEOUT**

コマンドのタイムアウト時間を指定する。(秒単位:0~2147483647、0は無制限、既定値:30)

**-h**

使用方法を表示する。

### 戻り値

diimchghash は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了

### 注意

- ハッシュ値変更時は、全 OLTP ノードで diimchghash を実行する必要がある。
- diimchghash は自ノードにマスタがあるレプリケーショングループのハッシュ値切替を行い、全ノードへ展開する。
- diimchghash 実行前に、diimcheck により、ノード間のテーブル整合性チェックを行うことを推奨する。

### 関連

diimchg, diimcheck

## 2.3.6 diimctrl (インメモリサーバ起動停止コマンド)

### 名前

diimctrl - インメモリサーバの起動または停止を行う

### 書式

```
diimctrl -b [-A] [-i INTERVAL] [-c COUNT] [-T TIMEOUT] [-j {on|off}]
diimctrl -b [-A] -t [-i INTERVAL] [-c COUNT] [-T TIMEOUT]
diimctrl -b -r REPLICATION_GROUP_ID [-i INTERVAL] [-c COUNT] [-T TIMEOUT] [-j {on|off}]
diimctrl -b -r REPLICATION_GROUP_ID -t [-i INTERVAL] [-c COUNT] [-T TIMEOUT]
diimctrl -b -r REPLICATION_GROUP_ID -M [-T TIMEOUT] [-j {on|off}]
diimctrl -b -r REPLICATION_GROUP_ID -M -t [-T TIMEOUT]
diimctrl -b -m MAP_ID [-T TIMEOUT]
diimctrl -b -B [-T TIMEOUT]
diimctrl -B [-T TIMEOUT]
diimctrl -e [-A] [-s] [-T TIMEOUT]
diimctrl -e -r REPLICATION_GROUP_ID [-s] [-T TIMEOUT]
diimctrl -e -m MAP_ID [-T TIMEOUT]
diimctrl -e -B [-T TIMEOUT]
diimctrl -h
```

### 説明

**diimctrl** は、インメモリサーバの起動または停止を行う。起動時に、対象レプリケーショングループの TAM が未起動の場合は、TAM の起動も行う。

### オプション

#### -b

インメモリサーバを起動する。TAM が未起動の場合は、TAM を起動後にインメモリサーバ起動を行う。

#### -e

インメモリサーバを停止する。

#### -A

全レプリケーショングループの IMS アクセスサーバを起動または停止する。(既定値)

#### -t

TAM インスタンスの起動のみを行う。

本オプションを省略した場合は、IMS アクセスサーバも起動する。

#### -s

停止対象がスレーブの場合に、TAM インスタンスの停止も行う。

本オプションを省略した場合、およびマスタの場合は、IMS アクセスサーバのみ停止する。

#### -r REPLICATION\_GROUP\_ID

特定レプリケーショングループの IMS アクセスサーバの起動／停止を行う場合に、対象の IMS アクセスサーバが属するレプリケーショングループの ID を指定する。

#### -m MAP\_ID

特定 MAP の IMS アクセススレッドの起動／停止を行う場合に、対象の IMS アクセススレッドが属する MAP の ID を指定する。

## **-M**

マスタを定義しているノードの TAM が起動できない場合に、スレーブとして定義されている自ノードの TAM および IMS アクセスサーバをマスタとして起動する場合に指定する。

## **-B**

-b と合わせて指定された場合、IMS ブリッジサーバを起動または停止する。

-e と合わせて指定された場合、全ての IMS ブリッジサーバを停止する。

-b と -e を省略した場合、環境定義に定義されている IMS ブリッジサーバ数と同数になるよう IMS ブリッジサーバを起動または停止する。

## **-i INTERVAL**

スレーブ TAM 起動時の起動状態確認間隔を指定する。(100 ミリ秒単位:1~1000、既定値:10)

## **-c COUNT**

スレーブ TAM 起動時の起動状態確認回数を指定する。(1~2147483647、既定値:30)

## **-T TIMEOUT**

コマンドのデーモンからの応答タイムアウト時間を指定する。(秒単位:0~2147483647、0 は無制限、既定値:30)

0 以外が指定された場合、(INTERVAL×COUNT)+TIMEOUT がタイムアウト時間になる。

## **-j**

on を指定した場合、TAM の更新ログを蓄積する MAP の更新ログを開始する。(既定値)

off を指定した場合、TAM の更新ログを蓄積する MAP の更新ログを開始しない。

## **-h**

使用方法を表示する。

## 戻り値

diimctrl は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

## 注意

- -m 指定は、対象 MAP の IMS アクセスサーバが未起動の場合はエラーとなる。
- -b 指定時に TAM インスタンスが未起動の場合は、TAM インスタンスを起動後にインメモリサーバを起動する。
- -M 指定は、指定したレプリケーショングループのマスタが起動済みの場合はエラーとなる。
- -e 指定時は、TAM インスタンスおよびレプリケーション機構の停止は行わない。
- 環境定義に定義されている IMS ブリッジサーバ数より起動中の IMS ブリッジサーバが少ない場合、diimctrl -b -B で不足分の IMS ブリッジサーバを起動することができる。
- 環境定義に定義されている IMS ブリッジサーバ数より起動中の IMS ブリッジサーバが多い場合、diimctrl -B で余分な IMS ブリッジサーバを停止することができる。ただし、停止されるブリッジサーバのキューに滞留している要求が 0 になっていることを diimref -B で確認後に投入する必要がある。
- ノード障害発生後に diiminit、diimctrl により再起動したノードは、マスタが存在しないノードとなり、ノード間の処理が不均等となる。ノード間の処理均等化を図るためには、ditamswap を使用してマスタ切替を

行う必要がある。

- `-T` に 0 を指定した場合であっても、TAM コマンドからの応答が `INTERVAL×COUNT` を超えた場合はタイムアウトでコマンドが異常終了する。
- アクセスサーバが起動済みの状態で、`-j on` 指定の本コマンドを実行した場合、更新ログの開始のみを実行する。
- `-t` と `-j off` は同時に指定できない。

## 関連

`diiminit`, `diimterm`

## 2.3.7 diiminit(メモリキャッシュ起動コマンド)

### 名前

diiminit - メモリキャッシュの起動を行う

### 書式

```
diiminit [-w] [-p EXIT_PARAMETER] [-T TIMEOUT] [-j {on|off}]
diiminit [-w] [-p EXIT_PARAMETER] -t [-T TIMEOUT]
diiminit -c [-p EXIT_PARAMETER] [-T TIMEOUT] [-j {on|off}]
diiminit -c [-p EXIT_PARAMETER] -t [-T TIMEOUT]
diiminit -h
```

### 説明

**diiminit** は、インメモリサーバ所在管理およびインメモリサーバの共有メモリ作成、インメモリサーバ所在管理デーモンの起動、マスタ TAM インスタンスの起動、インメモリサーバの起動を行う。

### オプション

**-w**

ウォームモードで起動を行う。前回運転時の環境を引き継ぐ。(既定値)

**-c**

コールドモードで起動を行う。新規の環境定義ファイルの値で起動する。

**-p EXIT\_PARAMETER**

環境定義で利用者情報初期値設定関数名が指定されている場合に、利用者情報初期値設定関数へ渡すパラメータを指定する。

**-t**

マスタ TAM インスタンスのみ起動する。  
本オプションを省略した場合、マスタのインメモリサーバも自動起動する。

**-T TIMEOUT**

コマンドのタイムアウト時間を指定する。(秒単位:0~2147483647、0は無制限、既定値:60)

**-j**

on を指定した場合、TAM の更新ログを蓄積する MAP の更新ログを開始する。(既定値)  
off を指定した場合、TAM の更新ログを蓄積する MAP の更新ログを開始しない。

**-h**

使用方法を表示する。

### 戻り値

diiminit は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- AP 層では、TAM インスタンスおよびインメモリサーバの起動は行わない。
- diiminit は、スレーブ TAM の起動は行わない。全 OLTP 層ノードのインメモリサーバ所在管理デーモン起動後に、diimctrl -b コマンドによりスレーブ TAM のインスタンスおよび IMS アクセスサーバを起動する必要

がある。

- TAM の更新ログを蓄積する定義で、`-j off` を指定して起動した場合、`diimctrl -b -j on` により更新ログを開始するまでは、TAM への更新処理を行うことはできない。
- TAM の更新ログを蓄積する定義を行っている時、TAM の更新ログ保存機能は warm モードで起動する。このため、TAM の更新ログを蓄積する定義を行っている場合の初回起動はエラーとなる。初回起動時や更新ログを破棄して良い場合は、`diiminit -j off` で起動、手動で `tamtxlogidstart` コマンドを cold モードで実行後、`diimctrl -b -j on` により更新ログを使用可能な状態にする。
- `-t` と `-j off` は同時に指定できない。

## 関連

`diimctrl`, `diimterm`, `ditamswap`

## 2.3.8 diimmod(メモリキャッシュ動作変更コマンド)

### 名前

diimmod - メモリキャッシュの動作パラメータを変更する

### 書式

```
diimmod -c -T csendinvl=value
diimmod -f [-T hlchkinvl=value] [-T hlchkrty=value]
diimmod -f [-T acschkinvl=value] [-T brschkinvl=value] [-T tamchkinvl=value]
           [-T switchinvl=value] [-T switchrty=value]
diimmod -m MAPID -T commitmode={normal|group}
diimmod -m MAPID -T reqnum=value
diimmod -m MAPID -T tmout=value
diimmod -m MAPID -T quechk={yes|no}
diimmod -m MAPID -T stats={yes|no}
diimmod -m MAPID -T output={yes|no}
diimmod -m MAPID -T bufcycl={yes|no}
diimmod -b -T connectinvl=value
diimmod -b -T stats={yes|no}
diimmod -b -T output={yes|no}
diimmod -b -T bufcycl={yes|no}
diimmod -u -T reqtimeout=value
diimmod -h
```

### 説明

**diimmod** は、インメモリサーバおよびインメモリサーバ所在管理が利用する動作パラメータ情報の変更を行う。

### オプション

**-c**

メモリキャッシュ機能の制御に関する動作変更を行うことを表す。

**-f**

メモリキャッシュ機能の監視に関する動作変更を行うことを表す。

**-m**

MAPID を指定し、MAP 単位の情報に関する動作変更を行うことを表す。

**-b**

ブリッジサーバに関する動作変更を行うことを表す。

**-u**

利用者アプリケーションに関する動作変更を行うことを表す。

**-T**

動作変更内容を指定する。

<b>csendinvl</b>	制御電文再送間隔(1~36000、100 ミリ秒単位で指定)
<b>hlchkinvl</b>	ノード間ヘルスチェック間隔(0~36000、100 ミリ秒単位で指定)
<b>hlchkrty</b>	ノード間ヘルスチェックリトライ回数(0~99)
<b>acschkinvl</b>	アクセスサーバ監視間隔(0~36000、100 ミリ秒単位で指定)
<b>brschkinvl</b>	ブリッジサーバ監視間隔(0~36000、100 ミリ秒単位で指定)

<b>tamchkinvl</b>	TAM インスタンス監視間隔(0～36000、100 ミリ秒単位で指定)
<b>switchinvl</b>	マスタ切替リトライ間隔(1～36000、100 ミリ秒単位で指定)
<b>switchrty</b>	マスタ切替リトライ回数(0～99)
<b>commitmode</b>	コミットモード
<b>reqnum</b>	グループコミットのコミット要求数(2～256)
<b>tmout</b>	グループコミットのタイムアウト値(1～2147483647、ミリ秒単位で指定)
<b>quechk</b>	コミット保留時、IMS キュー滞留チェックするか否か
<b>connectinvl</b>	ブリッジサーバ間の再接続間隔(1～3600、秒単位で指定)
<b>stats</b>	アクセス統計情報を蓄積するか否か
<b>output</b>	アクセス情報ログ蓄積有無
<b>bufcycl</b>	アクセス情報ログの蓄積領域をサイクリックに使用するか否か
<b>reqtimeout</b>	利用者アプリケーションの応答監視タイマ値(0～2147483647、ミリ秒単位で指定)

**-h**

使用方法を表示する。

## 戻り値

diimmod は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了
- 2      警告終了

## 注意

- -T オプションは 1 つ以上指定する必要がある。
- -T オプションは、-c、-f、-m、-b、-u オプションの前に指定することはできない。

## 2.3.9 diimnotify(メモリキャッシュ通知コマンド)

### 名前

diimnotify - メモリキャッシュ機能に通知を行う

### 書式

```
diimnotify -M lnode_fault -n LNODENAME
diimnotify -h
```

### 説明

指定された情報をインメモリサーバ所在管理デーモンに通知する。

### オプション

**-M**

通知する障害の種類を指定する。

**lnode\_fault**      論理ノード障害

**-n LNODENAME**

障害対象とする論理ノード名を指定する。

**-h**

使用方法を表示する。

### 戻り値

diimnotify は、以下のいずれかの終了ステータスを戻して終了する。

0                      正常終了

1                      異常終了

### 注意

本コマンドは任意の OLTP ノード 1 台に投入する。

本コマンドは利用者側でノード障害監視を行う場合に、検出した障害を通知するためのコマンドである。そのため、環境定義 (IMENV 節-%NODEFAULT-HLTHCHKINVL) にてノード障害監視を有効にしている場合、本コマンドを投入する必要はない。

本コマンドは、障害情報をインメモリサーバ所在管理デーモンに通知するが、その後のマスタ切替処理等の完了は待ち合わせない。

## 2.3.10 diimperfref(メモリキャッシュ性能情報照会コマンド)

### 名前

diimperfref - メモリキャッシュの性能情報を照会する

### 書式

```
diimperfref -S -M -m MAP_ID [-R]
diimperfref -S -B
diimperfref -L -M -m MAP_ID [-o OUTPUT_FILE]
diimperfref -L -B [-o OUTPUT_FILE]
diimperfref -h
```

### 説明

**diimperfref** は、インメモリサーバ機能の性能情報照会を行う。

### オプション

**-S**

統計情報を表示する。

**-L**

アクセスログ情報を CSV 形式で表示する。

**-M**

MAP 単位の情報を表示する。

**-B**

ブリッジサーバの情報を表示する。

**-R**

次回照会用に、統計情報をリセットする。

**-m MAP\_ID**

指定した MAP\_ID に該当する MAP の情報を表示する。

**-o OUTPUT\_FILE**

指定したファイルへアクセスログの情報を出力する。  
省略時には標準出力に出力する。

**-h**

使用方法を表示する。

### 戻り値

diimperfref は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了

### 注意

- -L 指定時、アクセスログ情報が 1 件もない場合は、何も出力されない。
- -S -M 指定時、稼働統計の情報は diimperfref の -R 指定もしくは diimperfreset でリセットしない限り引き継がれるため、アクセス統計情報採取が OFF になっていても、過去に蓄積された情報が出力される。
- ただし、稼働率に関しては diimperfref もしくは diimperfreset から diimperfref までの期間のデータが出

力される。そのため、上記の期間でアクセス統計情報採取の状態が OFF であった場合には、稼働率は 0%が出力される。

実行例

MAP 統計情報照会 (diimperfref -S -M -m MAP\_ID)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①				
+----- MAP PERFORMANCE INFORMATION DISPLAY -----+				
MAPID	= 1	①		
STATUS	= NORMAL	②		
WAITCNT	= 5	③		
DEADLOCKCNT	= 1	④		
GRPCOMMIT (COUNT)	= 0	⑤		
GRPCOMMIT (TIME)	= 0	⑥		
GRPCOMMIT (QUEUE)	= 0	⑦		
BUSYRATE:				
THREAD	RATE	START-TIME	END-TIME ⑧	
TT	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	
AT	44%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	
-----				
AT1	40%			
AT2	60%			
AT3	50%			
AT4	30%			
AT5	0%			
AT6	80%			
AT7	60%			
AT8	50%			
AT9	40%			
AT10	50%			
AT11	40%			
AT12	60%			
AT13	70%			
AT14	20%			
AT15	20%			
AT16	30%			
TYPE	AVG [usec]	MIN [usec]	MAX [usec]	REQUEST
⑨	⑩	⑪	⑫	⑬
READ1 (NOLOCK)	200	150	300	400
READ1 (LOCK)	200	150	300	400
READ	200	150	300	400
WRITE	200	150	300	400
REWRITE	200	150	300	400
DELETE	200	150	300	400
DELETEx1 (READ)	200	150	300	400
DELETEx1 (DELETE)	200	150	300	400
TRUNCATE_PRE	200	150	300	400
TRUNCATE	200	150	300	400
COMMIT	5200	1150	10300	1400
GRPCOMMIT	10000	2000	15000	3000
ROLLBACK	200	150	300	400
+===== END OF DISPLAY =====+				

① コマンド実行時刻

+----- MAP PERFORMANCE INFORMATION DISPLAY -----+

MAP の統計情報を表示

① MAPID

② MAP に対応するインメモリサーバの状態

NORMAL	稼動状態
STOP	停止状態
STARTING	起動中
STOPPING	停止中

③ ロック待ち発生回数

④ デッドロック検出回数

⑤ グループコミット実行回数(コミット数に達したことによるグループコミット実行回数)

⑥ グループコミット実行回数(制限時間に達したことによるグループコミット実行回数)

⑦ グループコミット実行回数(キューなしに検出によるグループコミット実行回数)

⑧ 稼働率と稼働率算出対象時刻

THREAD	TT: トランザクションスレッド AT: 全アクセススレッド平均 AT1~16: 各アクセススレッド別(多重度に応じて出力)
RATE	稼働率(最大 100%)
START-TIME	稼働率測定期間の開始時刻
END-TIME	稼働率測定期間の終了時刻

⑨ アクセス要求種別

READ1 (NOLOCK)	キー指定レコード読込(排他なし)
READ1 (LOCK)	キー指定レコード読込(排他あり)
READ	複数レコード読込
WRITE	レコード追加
REWRITE	レコード更新
DELETE	レコード削除
DELETEx1 (READ)	キー指定レコード削除(読込)
DELETEx1 (DELETE)	キー指定レコード削除(削除)
TRUNCATE_PRE	全レコード削除事前処理
TRUNCATE	全レコード削除
COMMIT	コミット
GRPCOMMIT	グループコミット
ROLLBACK	ロールバック

⑩ 平均処理時間(μs)

⑪ 最小処理時間(μs)

⑫ 最大処理時間(μs)

⑬ アクセス要求回数

ブリッジ統計情報照会(diimperfref -S -B)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①						
+----- BRIDGE PERFORMANCE INFORMATION DISPLAY -----+						
SRVID	=	1	①			
STATUS	=	NORMAL	②			
TYPE	BUSYRATE				CNT	MSGSIZE
③	④				⑤	⑥
BRIDGE (CLI)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	3200	-	
SEND (CLI)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	1600	210000	
RECEIVE (CLI)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	1600	210000	
BRIDGE (SRV)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	3200	-	
SEND (SRV)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	1600	210000	
RECEIVE (SRV)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	1600	210000	
+----- BRIDGE PERFORMANCE INFORMATION DISPLAY -----+						
SRVID	=	2				
STATUS	=	NORMAL				
TYPE	BUSYRATE				CNT	MSGSIZE
BRIDGE (CLI)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	3200	-	
SEND (CLI)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	1600	210000	
RECEIVE (CLI)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	1600	210000	
BRIDGE (SRV)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	3200	-	
SEND (SRV)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	1600	210000	
RECEIVE (SRV)	60%	2011/09/11 18:50:00.000	2011/09/11 19:00:00.000	1600	210000	
+===== END OF DISPLAY =====+						

① コマンド実行時刻

+----- BRIDGE PERFORMANCE INFORMATION DISPLAY -----+

ブリッジサーバの統計情報を表示

① サーバ ID

② ブリッジサーバの状態

NORMAL	稼動状態
STOP	停止状態
STARTING	起動中
STOPPING	停止中

③ ブリッジサーバ機能種別

BRIDGE (CLI)	クライアント側ブリッジ
SEND (CLI)	クライアント側送信
RECEIVE (CLI)	クライアント側受信
BRIDGE (SRV)	サーバ側ブリッジ
SEND (SRV)	サーバ側送信
RECEIVE (SRV)	サーバ側受信

④ 稼働率と稼働率算出対象時刻

⑤ 処理件数

⑥ 送受信電文サイズ

MAP アクセスログ照会(diimperfref -L -M -m MAP\_ID)

CSV 形式で以下の情報を出力する

No.	項目名 (ヘッダ)	出力内容
1	MAPID (MAPID)	MAPID
2	スレッド識別番号 (THREAD)	MAP 内のスレッド識別番号 0                    トランザクションスレッド 1～16                アクセススレッド
3	処理状態 (STATUS)	処理状態 EXEC                処理中 FINISH               処理済
4	開始時刻 (STARTTIME)	開始時刻
5	終了時刻 (ENDTIME)	終了時刻 処理状態が処理済の場合にのみ出力する
6	処理時間 (TAT)	処理時間(μ s) 処理状態が処理済の場合にのみ出力する
7	要求元ノード ID (LNODEID)	要求元のノード ID
8	要求元プロセス ID (PID)	要求元のプロセス ID
9	要求元スレッド ID (TID)	要求元のスレッド ID
10	要求元サービス通番 (SERVICENO)	要求元のサービス通番
11	処理種別 (TYPE)	READ1_NOLOCK        キー指定レコード読込(排他なし) READ1_LOCK           キー指定レコード読込(排他あり) READ                   複数レコード読込 WRITE                   レコード追加 REWRITE                レコード更新 DELETE                  レコード削除 DELETEX1_READ        キー指定レコード削除(読込) DELETEX1_DELETE      キー指定レコード削除(削除) TRUNCATE_PRE          全レコード削除事前処理 TRUNCATE               全レコード削除 COMMIT                  コミット GROUP_COMMIT(COUNT)   コミット数に達したことによるグループコミット GROUP_COMMIT(TIME)   制限時間に達したことによるグループコミット GROUP_COMMIT(QUEUE) キューなしに検出によるグループコミット ROLLBACK               ロールバック SMQ_SEND               要求元プロセスへの応答電文送信(メッセージバッファ経由あり) SMQ_NOTIFY            要求元プロセスへの応答電文送信(メッセージバッファ経由なし)

No.	項目名 (ヘッダ)	出力内容
		LOCKSRCH            ロックリストエントリ検索処理 TamReadByKey       KVS の TamReadByKey 処理 TamRewrite          KVS の TamRewrite 処理 TamDelete           KVS の TamDelete 処理 TamWrite            KVS の TamWrite 処理 TamTruncate        KVS の TamTruncate 処理 TamNReadEX        KVS の TamNReadEX 処理 TamNWrite          KVS の TamNWrite 処理 TamTxStart         KVS の TamTxStart 処理 TamCommit1        KVS の TamCommit1 処理 TamCommit2        KVS の TamCommit2 処理 TamRollback        KVS の TamRollback 処理 TamPreCommit2     KVS の TamPreCommit2 処理
12	ユーザレコード件数 (USERRECCNT)	ユーザレコード件数 複数レコード読込の場合に、何件のレコードを取得したか情報を得るための値
13	KVS レコード件数 (KVSRECCNT)	KVS のレコード件数 可変長レコードの場合に、KVS で何件のレコードとして扱ったか情報を得るための値
14	グループコミット纏め数 (GRPCOMMITCNT)	グループコミットで纏めたコミット数
15	論理表 ID (LTABLEID)	論理表 ID
16	論理表名 (LTABLENAME)	論理表名 本コマンド実行時に、論理表 ID から変換する
17	物理表名 (PTABLENAME)	物理表名 本コマンド実行時に、論理表 ID と MAPID から変換する
18	処理結果 (RESULT)	処理結果 処理状態が処理済の場合にのみ出力する

ブリッジアクセスログ照会(diimperfref -L -B)

CSV 形式で以下の情報を出力する

No.	項目名 (ヘッダ)	出力内容
1	サーバ ID (SRVID)	ブリッジサーバ ID
2	ブリッジサーバ内スレッド 種別 (BRIDGETYPE)	ブリッジサーバ内スレッド種別 SRV_BRIDGE           サーバ側ブリッジスレッド SRV_SEND             サーバ側送信スレッド SRV_RECEIVE          サーバ側受信スレッド CLI_BRIDGE           利用者側ブリッジスレッド CLI_SEND             利用者側送信スレッド CLI_RECEIVE          利用者側受信スレッド
3	処理状態 (STATUS)	処理状態 EXEC                処理中 FINISH              処理済
4	開始時刻 (STARTTIME)	開始時刻
5	終了時刻 (ENDTIME)	終了時刻 処理状態が処理済の場合にのみ出力する
6	処理時間 (TAT)	処理時間( $\mu$ s ) 処理状態が処理済の場合にのみ出力する
7	要求元ノード ID (LNODEID)	要求元のノード ID
8	要求元プロセス ID (PID)	要求元のプロセス ID
9	要求元スレッド ID (TID)	要求元のスレッド ID
10	要求元サービス通番 (SERVICENO)	要求元のサービス通番
11	要求先ノード ID (DSTLNODEID)	要求先のノード ID
12	要求先 MAP (DSTMAP)	要求先の MAPID
13	障害発生状態 (TROUBLE)	障害発生状態 空白                正常時 FAILED              障害時
14	送受信サイズ (SIZE)	送受信サイズ
15	処理結果 (RESULT)	処理結果 処理状態が処理済の場合にのみ出力する

## 2.3.11 diimperfreset(メモリキャッシュ統計情報初期化コマンド)

### 名前

diimperfreset - メモリキャッシュの統計情報を初期化する

### 書式

```
diimperfreset -M -m MAP_ID
diimperfreset -B [-s SRV_ID]
diimperfreset -h
```

### 説明

**diimperfreset** は、インメモリサーバ機能の統計情報の初期化を行う。

### オプション

**-M**

MAP 指定の統計情報初期化処理を行う。

**-B**

ブリッジサーバの統計情報初期化処理を行う。

**-m MAP\_ID**

指定した MAP\_ID に該当する MAP の統計情報を初期化する。

**-s SRV\_ID**

指定した SRV\_ID に該当するブリッジサーバの統計情報を初期化する。  
省略時には全ブリッジサーバの統計情報を初期化する。

**-h**

使用方法を表示する。

### 戻り値

diimperfreset は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了 全ての統計情報初期化処理が失敗
2	警告終了 一部の統計情報初期化処理が失敗

## 2.3.12 diimquegc (IMS キュー解放コマンド)

### 名前

diimquegc - IMS キューを解放する

### 書式

diimquegc  
diimquegc -h

### 説明

**diimquegc** は、メモリキャッシュ機能により使用中として管理されているが、実際には利用者が存在せず未使用の状態となっている IMS キューが存在する場合、IMS キューおよび管理情報の解放を行う。

### オプション

-h

使用方法を表示する。

### 戻り値

diimquegc は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了

## 2.3.13 diimref(メモリキャッシュ情報照会コマンド)

### 名前

diimref - メモリキャッシュの情報を照会する

### 書式

```
diimref -N
diimref -R [-d]
diimref -T
diimref -M [-v] [-m MAP_ID]
diimref -M [-v] [-A]
diimref -L [-k] [-d] [-t LOGICAL_TABLE_NAME] [-A]
diimref -B [-v] [-A]
diimref -S
diimref -H
diimref -h
```

### 説明

**diimref** は、インメモリサーバ所在管理テーブルの情報照会を行う。

### オプション

**-N**

ノード情報を表示する。

**-R**

レプリケーショングループ情報を表示する。

**-T**

TAM の配置情報を表示する。

**-M**

MAP に関する情報を表示する。

**-L**

論理表に関する情報を表示する。

**-B**

ブリッジサーバに関する情報を表示する。

**-S**

IMENV 節で定義される動作パラメータの情報を表示する。

**-H**

MAP 毎に割り当てるハッシュ値の範囲に関する情報を表示する。

**-A**

全情報を表示対象とする。

-M 指定時に-A を省略した場合は、他ノードのみに定義されている MAP 情報は表示対象外である。

-L 指定時に-A を省略した場合は、DIOSA 制御表は表示対象外である。

-B 指定時に-A を省略した場合は、環境定義で定義された多重度数のサーバ情報のみを出力する。

-B 指定時に-A を指定した場合は、最大多重度(16 多重)のサーバ情報を出力する。

**-m**

指定した MAPID に該当する MAP の情報を表示する。

**-t**

指定した論理表名に該当する論理表の情報を表示する。

**-d**

レプリケーショングループ (-R オプション指定)、または論理表 (-L オプション指定) に関連する MAP の情報を表示する。

**-v**

詳細情報を表示する。

**-k**

レコードのキー情報を表示する。

**-h**

使用方法を表示する。

## 戻り値

diimref は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了

## 注意

- 「diimref -M」の項目「SMQID」は 5 桁、「MSGQNUM」は 8 桁、「OVERFLOW」は 8 桁の表示となっている。桁を超えた値となった場合でも表示を行うが、見出しに対して表示がずれる。表示がずれた場合はスペース文字を区切りとして並び順に判断する必要がある。

実行例

ノード情報照会(diimref -N)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①		
+----- NODE INFORMATION DISPLAY -----+		
LNODE	STATUS	BRIDGE-SERVER
①	②	③
oltp1	STOP	STOP
oltp2	NORMAL	NORMAL
* oltp3	STOP	STOP
+===== END OF DISPLAY =====+		

① コマンド実行時刻

+----- NODE INFORMATION DISPLAY -----+

ノードごとの稼動情報を表示

① 論理ノード名(削除予定の場合は直前に"\*"が表示される。)

② ノード状態

NORMAL	正常稼動
STOP	停止
TROUBLE	障害
TPATH-TRBL	Tパス障害

③ ブリッジサーバ状態(ノード状態が TROUBLE の場合は無効)

NORMAL	正常稼動
STOP	停止
TROUBLE	障害

レプリケーショングループ情報照会(diimref -R -d)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①		
+----- REPLICATION-GROUP INFORMATION DISPLAY -----+		
REPGRP	STATUS	MAPID
①	②	③
1*	STOP	1*, 11*, 12*, 13*
2	NORMAL	21, 22, 23
3	STOP	31, 32*, 33
+===== END OF DISPLAY =====+		

① コマンド実行時刻

+----- REPLICATION-GROUP INFORMATION DISPLAY -----+

レプリケーショングループ ID ごとの稼動情報を表示

① レプリケーショングループ ID(削除予定の場合は直後に"\*"が表示される。)

② レプリケーショングループ状態

NORMAL	正常稼動
STOP	停止
TROUBLE	障害
SWITCH_ABN	障害によるマスタ切替中
SWITCH_PLN	計画マスタ切替中

③ MAPID(-d 省略時は表示されない。削除予定の MAP には直後に"\*"が表示される。)

TAM の配置情報照会(diimref -T)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①						
+----- TAM LOCATION INFORMATION DISPLAY -----+						
REPGRP = 1 * INSTANCE = INS001						
①	②					
LNODE	TYPE	TYPE (SG)	TAM-STATUS	SW-STATUS	VNODE	VNODE (SG)
③	④	⑤	⑥	⑦	⑧	⑨
LNOL01	MASTER	MASTER	NORMAL	MASTER	TAMLN001MASTER	TAMLN001MASTER
LNOL02	SLAVE	SLAVE	NORMAL	READY	TAMLN001SLAVE02	TAMLN001SLAVE02
LNOL03	SLAVE	SLAVE	NORMAL	READY	TAMLN001SLAVE03	TAMLN001SLAVE03
REPGRP = 2 INSTANCE = INS002						
LNODE	TYPE	TYPE (SG)	TAM-STATUS	SW-STATUS	VNODE	VNODE (SG)
LNOL01	SLAVE	SLAVE	NORMAL	READY	TAMLN002SLAVE01	TAMLN002SLAVE01
LNOL02	MASTER	MASTER	NORMAL	MASTER	TAMLN002MASTER	TAMLN002MASTER
*LNOL03	SLAVE	SLAVE	NORMAL	READY	TAMLN002SLAVE03	TAMLN002SLAVE03
+===== END OF DISPLAY =====+						

① コマンド実行時刻

+----- TAM LOCATION INFORMATION DISPLAY -----+

TAM の配置情報を表示

- ① レプリケーショングループ ID（削除予定の場合は直後に”\*”が表示される。）
- ② TAM のインスタンス名
- ③ 論理ノード名（削除予定の場合は直前に”\*”が表示される。）
- ④ 現在のマスタ／スレーブ種別
  - MASTER           マスタ
  - SLAVE           スレーブ
- ⑤ SG 定義のマスタ／スレーブ種別
  - MASTER           マスタ
  - SLAVE           スレーブ
- ⑥ TAM の状態
  - NORMAL           正常稼動
  - STOP           停止
  - TROUBLE           障害
  - STARTING           起動処理中
  - STOPPING           停止処理中
- ⑦ マスタ切替先の可否状態
  - READY           切替可能
  - STOP           切替不可
  - MASTER           マスタノード
  - TROUBLE           障害
- ⑧ TAM の論理ノード名。TAM が正常稼動中の場合のみ表示する。
- ⑨ SG 定義の TAM 論理ノード名

MAP 情報照会(diimref -M -A)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①											
+----- MAP INFORMATION -----+											
				-----AT-----			-----TT-----				
MAPID	STATUS	REPGRP	TYPE	SMQID	MSGQNUM	OVERFLOW	SMQID	MSGQNUM	OVERFLOW	MASTER	BLOCK
①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪	⑫
1	NORMAL	10	MASTER	11	0	0	21	0	0	NORMAL/UPD	OFF
2	NORMAL	10	MASTER	12	0	0	22	0	0	NORMAL/UPD	OFF
8	NORMAL	255	MASTER	9	0	0	29	0	0	NORMAL/UPD	OFF
30	NORMAL	10	MASTER	13	0	0	23	0	0	NORMAL/UPD	OFF
50	NORMAL	10	MASTER	14	0	0	24	0	0	NORMAL/UPD	OFF
9999	NORMAL	20	MASTER	8	0	0	28	0	0	NORMAL/UPD	OFF
2147483647	NORMAL	255	MASTER	10	0	0	20	0	0	NORMAL/UPD	OFF
+===== END OF DISPLAY =====+											

① コマンド実行時刻

+----- MAP INFORMATION -----+

MAP ごとの稼動情報を表示

① MAPID

② MAP に対応するインメモリサーバの状態

NORMAL	稼動状態
STOP	停止状態
STARTING	起動中
STOPPING	停止中
---	(他ノードのみに定義されている MAP 情報の場合)

③ レプリケーショングループ ID

④ マスタ/スレーブ種別

MASTER	マスタ
SLAVE	スレーブ
---	(他ノードのみに定義されている MAP 情報の場合)

⑤ アクセススレッド用キュー(SMQ)の ID

⑥ アクセススレッド用キュー(SMQ)に滞留しているアクセス要求数

⑦ アクセススレッド用キュー(SMQ)のバッファ確保に失敗した回数

⑧ トランザクションスレッド用キュー(SMQ)の ID

⑨ トランザクションスレッド用キュー(SMQ)に滞留しているアクセス要求数

⑩ トランザクションスレッド用キュー(SMQ)のバッファ確保に失敗した回数

※⑤～⑩は、インメモリサーバが停止している場合は、---が表示される。

⑪ MAP に対応するマスタノードのインメモリサーバの状態

NORMAL/UPD	稼動状態
NORMAL/REF	稼動状態(更新不可)
STOP	停止状態

⑫ MAP に対応する閉塞状態

OFF	閉塞中でない状態
ON	閉塞中

MAP 情報照会(diimref -M -v)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①											
+----- MAP INFORMATION -----+											
				-----AT-----			-----TT-----				
MAPID	STATUS	REPGRP	TYPE	SMQID	MSGQNUM	OVERFLOW	SMQID	MSGQNUM	OVERFLOW	MASTER	BLOCK
①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪	⑫
1	NORMAL	10	MASTER	11	0	0	21	0	0	NORMAL/UPD	OFF
+----- LOGICAL TABLE INFORMATION -----+											
TABLEID		= 10	①								
LTABLENAME		= LTABLE_010	②								
PTABLENAME		= PTABLE_010_01	③								
PTABLESTATUS		= OPEN	④								
+===== END OF DISPLAY =====+											

① コマンド実行時刻

+----- MAP INFORMATION -----+

MAP ごとの稼動情報を表示

① MAPID

② MAP に対応するインメモリサーバの状態

NORMAL	稼動状態
STOP	停止状態
STARTING	起動中
STOPPING	停止中
---	(他ノードのみに定義されている MAP 情報の場合)

③ レプリケーショングループ ID

④ マスタ／スレーブ種別

MASTER	マスタ
SLAVE	スレーブ
---	(他ノードのみに定義されている MAP 情報の場合)

⑤ アクセススレッド用キュー(SMQ)の ID

⑥ アクセススレッド用キュー(SMQ)に滞留しているアクセス要求数

⑦ アクセススレッド用キュー(SMQ)のバッファ確保に失敗した回数

⑧ トランザクションスレッド用キュー(SMQ)の ID

⑨ トランザクションスレッド用キュー(SMQ)に滞留しているアクセス要求数

⑩ トランザクションスレッド用キュー(SMQ)のバッファ確保に失敗した回数

※⑤～⑩は、インメモリサーバが停止している場合は、---が表示される。

⑪ MAP に対応するマスタノードのインメモリサーバの状態

NORMAL/UPD	稼動状態
NORMAL/REF	稼動状態(更新不可)
STOP	停止状態

⑫ MAP に対応する閉塞状態

OFF	閉塞中でない状態
ON	閉塞中

+----- LOGICAL TABLE INFORMATION -----+

MAP に分散されている表の情報を表示

① 論理表 ID

② 論理表名

③ 物理表名

④ 物理表の状態

OPEN	MAP 内の全スレッドがオープンしている状態
------	------------------------

CLOSE	MAP 内の全スレッドがクローズ状態
OPEN/CLOSE	オープン、クローズしているスレッドが混在している状態

論理表情報照会(diimref -L -k -d)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①		
+----- TABLE INFORMATION -----+		
TABLEID	= 10	①
LTABLENAME	= LTABLE_010	②
RECTYPE	= VARLEN	③
MAPDIST	= 1	④
TABLETYPE	= 0	⑤
+----- PRIMARY KEY INFORMATION -----+		
KEYNAME	= pk1	①
TOTALKEYSIZE	= 20	②
+----- PRIMARY KEY DEFINITION INFORMATION -----+		
OFFSET	KEYSIZE	KVSOFFSET
①	②	③
0	20	24
+----- SECONDARY KEY INFORMATION -----+		
KEYNAME	= sk1	①
TOTALKEYSIZE	= 12	②
+----- SECONDARY KEY DEFINITION INFORMATION -----+		
OFFSET	KEYSIZE	KVSOFFSET
①	②	③
20	12	60
+----- MAP DISTRIBUTED INFORMATION -----+		
MAPID	= 1	①
STATUS	= NORMAL	②
PTABLENAME	= PTABLE_010_01	③
PTABLESTATUS	= OPEN	④
+===== END OF DISPLAY =====+		

① コマンド実行時刻

+----- TABLE INFORMATION -----+

論理表に関する情報を表示

- ① 論理表 ID
- ② 論理表名
- ③ レコード形式
  - FIXED 固定長レコード
  - VARLEN 可変長レコード
- ④ 分散先の MAP 数
- ⑤ 論理表種別(1～9) ※：-A 指定時は論理表種別(0)の情報も表示

+----- PRIMARY KEY INFORMATION -----+

プライマリキーの情報を表示

- ① プライマリキーのインデックス名
- ② プライマリキーの長さ

+----- PRIMARY KEY DEFINITION INFORMATION -----+

プライマリキーを構成する項目に関する情報を表示

- ① ユーザレコードにおける先頭からのオフセット
- ② 長さ
- ③ KVS レコードにおける先頭からのオフセット

+----- SECONDARY KEY INFORMATION -----+

セカンダリキーの情報を表示

- ① セカンダリキーのインデックス名
- ② セカンダリキーの長さ

+----- SECONDARY KEY DEFINITION INFORMATION -----+

セカンダリキーを構成する項目に関する情報を表示

- ① ユーザレコードにおける先頭からのオフセット
- ② 長さ
- ③ KVS レコードにおける先頭からのオフセット

+----- MAP DISTRIBUTED INFORMATION -----+

分散先の MAP の情報を表示

- ① MAPID

- ② MAP に対応するインメモリサーバの状態

NORMAL	稼動状態
STOP	停止状態
STARTING	起動中
STOPPING	停止中

- ③ 物理表名

- ④ 物理表の状態

OPEN	MAP 内の全スレッドがオープンしている状態
CLOSE	MAP 内の全スレッドがクローズ状態
OPEN/CLOSE	オープン、クローズしているスレッドが混在している状態

ブリッジサーバ情報照会(diimref -B)

+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①				
+----- BRIDGE MULTI INFORMATION -----+				
BRIDGE-MULTI = 4 ①				
+----- BRIDGE INFORMATION -----+				
SERVERID	STATUS	SMQID	MSGQNUM	OVERFLOW
①	②	③	④	⑤
1	NORMAL	10	0	12
2	NORMAL	5	3	1
3	STOP			
4	NORMAL	130	1	10
+===== END OF DISPLAY =====+				

① コマンド実行時刻

+----- BRIDGE MULTI INFORMATION -----+

① ブリッジ多重度

+----- BRIDGE INFORMATION -----+

ブリッジサーバごとの稼動情報を表示

① ブリッジサーバ ID

② ブリッジサーバ状態

NORMAL	正常稼動
STOP	停止
STARTING	起動中
STOPPING	停止中
TROUBLE	障害

③ キュー(SMQ)の ID(ブリッジサーバが停止している場合は、---が表示される)

④ キュー(SMQ)に滞留しているアクセス要求数(ブリッジサーバが停止している場合は、---が表示される)

⑤ キュー(SMQ)のバッファ確保に失敗した回数(ブリッジサーバが停止している場合は、---が表示される)

ブリッジサーバ情報照会(diimref -B -v)

```
+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①
+----- BRIDGE MULTI INFORMATION -----+
BRIDGE-MULTI = 4 ①
+----- BRIDGE INFORMATION -----+
  ①          ②
SERVERID    = 1          STATUS    = NORMAL
+----- CLIENT INFORMATION -----+
  ①          ②          ③
SMQID(BRG)  = 1          MSGQNUM(BRG) = 0          OVERFLOW(BRG) = 0
  ④          ⑤          ⑥
SMQID(SND)  = 2          MSGQNUM(SND) = 3          OVERFLOW(SND) = 1
+----- SERVER INFORMATION -----+
  ①          ②          ③
SMQID(RCV)  = 3          MSGQNUM(RCV) = 0          OVERFLOW(RCV) = 1
  ④          ⑤          ⑥
SMQID(SND)  = 4          MSGQNUM(SND) = 1          OVERFLOW(SND) = 5
+----- DESTINATION NODE INFORMATION -----+
  NODEID  NODESTATUS  CONNSTATUS
    ①      ②          ③
      1   NORMAL    CONNECT
      2   NORMAL    CONNECT
+===== END OF DISPLAY =====+
```

① コマンド実行時刻

+----- BRIDGE MULTI INFORMATION -----+

① ブリッジ多重度

+----- BRIDGE INFORMATION -----+

ブリッジサーバごとの稼動情報を表示

① ブリッジサーバ ID

② ブリッジサーバ状態

NORMAL	正常稼動
STOP	停止
STARTING	起動中
STOPPING	停止中
TROUBLE	障害

+----- CLIENT INFORMATION -----+

利用者側キュー(SMQ)の稼動情報を表示(ブリッジサーバが停止している場合は、---が表示される)

- ① ブリッジスレッドキュー(SMQ)の ID
- ② ブリッジスレッドキュー(SMQ)に滞留しているアクセス要求数
- ③ ブリッジスレッドキュー(SMQ)のバッファ確保に失敗した回数
- ④ 送信スレッドキュー(SMQ)の ID
- ⑤ 送信スレッドキュー(SMQ)に滞留しているアクセス要求数
- ⑥ 送信スレッドキュー(SMQ)のバッファ確保に失敗した回数

+----- SERVER INFORMATION -----+

サーバ側キュー(SMQ)の稼動情報を表示(ブリッジサーバが停止している場合は、---が表示される)

- ① 受信スレッドキュー(SMQ)の ID
- ② 受信スレッドキュー(SMQ)に滞留しているアクセス要求数
- ③ 受信スレッドキュー(SMQ)のバッファ確保に失敗した回数
- ④ 送信スレッドキュー(SMQ)の ID
- ⑤ 送信スレッドキュー(SMQ)に滞留しているアクセス要求数

⑥ 送信スレッドキュー(SMQ)のバッファ確保に失敗した回数

+----- DESTINATION NODE INFORMATION -----+

ノード間の情報を表示

① ノード ID

② 相手先ノード状態

NORMAL	正常稼動
STOP	停止
TROUBLE	障害

③ 接続状態

CONNECT	接続
DISCONNECT	切断

## IMENV 節定義情報照会(diimref -S)

```

===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①
+----- COMMON SG INFORMATION DISPLAY -----+
SWITCH CTRLSENDINVL ALLOWCOMMIT RESTTXIDBLK
 ①          ②          ③          ④
AUTO          10  REPGTHALF  YES
+----- NODEFAULT SG INFORMATION DISPLAY -----+
HLTHCHKINVL HLTHCHKRETRY TPATHCHK DOWNCNT DOWNTIME
 ①          ②          ③          ④          ⑤
          10          3  YES          GTHALF          5
+----- SERVERFAULT SG INFORMATION DISPLAY -----+
CHKINVL DELAYLIMIT MSTRESTART SLVRESTART RESTARTTIME
 ①          ②          ③          ④          ⑤
          10          5000          0          20          30
+----- BRIDGEFAULT SG INFORMATION DISPLAY -----+
CHKINVL RESTART RESTARTTIME
 ①          ②          ③
          10          20          30
+----- TAMFAULT SG INFORMATION DISPLAY -----+
CHKINVL SWITCHRETRY SWITCHINVL
 ①          ②          ③
          10          99          10
+----- BRIDGE SG INFORMATION DISPLAY -----+
MULTI CONNECTINVL IMQUEBUFSZ IMQUEUNIT STATS ACSLOGOUTPUT ACSLOGBUFSIZE ACSLOGBUFCYCL
 ①          ②          ③          ④          ⑤          ⑥          ⑦          ⑧
          1          10          1024          8  YES          YES          1024  YES
+----- USERAP SG INFORMATION DISPLAY -----+
HASHEXIT REQTIMEOUT IMQUEBUFSZ IMQUEUNIT USERAREA USERAREAEXIT
 ①          ②          ③          ④          ⑤          ⑥
hashfunc          30          128          8          16 userareainitfunc
+----- MAP SG INFORMATION DISPLAY -----+
MAPID MLT TXTBLSIZE IMQUEBUFSZ IMQUEUNIT JNL STATS COMMIT GROUPCOMMIT ACCESSLOG
 ①          ②          ③          ④          ⑤          ⑥          ⑦          ⑧          ⑨          ⑩          ⑪          ⑫          ⑬          ⑭
11          1          1000          1024          8  YES  NO          NORMAL          5  YES          1  YES          1024  YES
12*          1          2000          2048          8  YES  NO          NORMAL          5  YES          1  YES          256  NO
21          8          1000          1024          80  NO  YES          GROUP          10  NO          0  YES          2048  YES
31          16          500          256          80  NO  NO          GROUP          256  YES          50  NO          1024  YES
===== END OF DISPLAY =====+

```

### ① コマンド実行時刻

+----- COMMON SG INFORMATION DISPLAY -----+

## %COMMON 項の動作パラメータ情報を表示

### ① 障害発生時のマスタ切替

AUTO	自動的にマスタ切替を行う
MANUAL	自動的にマスタ切替を行わない

② メモリキャッシュ機能が送信する制御電文の再送間隔(100 ミリ秒単位)

### ③ コミットを正常終了するかの判断方法

REPGTHALF	レプリケーションが半数より多く成功していればコミットを正常終了する
REPGEONE	レプリケーションが一つでも成功していればコミットを正常終了する
REPNOTHING	レプリケーションが一つも成功していなくてもコミットを正常終了する

④ マスタ切替時に AP 層の電文保留用トランザクションを閉塞する制御を自動的に行うか否か

YES	電文保留用トランザクションの閉塞および閉塞解除を行う
NO	電文保留用トランザクションの閉塞および閉塞解除を行わない

```
+----- NODEFAULT SG INFORMATION DISPLAY -----+
```

① AP 層-OLTP 層間で行うヘルスチェック処理の間隔(100 ミリ秒単位)

## ② AP 層-OLTP 層間のヘルスチェック処理失敗時のリトライ回数

③ AP 層において T パス状態を参照し、T パス障害検出時にノード障害とみなすか否か

YES	Tパス状態を参照し、ノード障害判定に利用する
NO	Tパス状態を参照しない

- ④ AP 層が検出した OLTP 層障害によって、OLTP 層のノード障害とする条件
- |        |   |
|--------|---|
| ALL    | 全 AP 層で障害を検出した場合                                  |
| GTHALF | 過半数の AP 層でノード障害を検出した後、DOWNTIME で指定した間隔で状態変化が無い場合  |
| GEHALF | 半数以上の AP 層でノード障害を検出した後、DOWNTIME で指定した間隔で状態変化が無い場合 |
| ANY    | 任意の 1AP 層でノード障害を検出した後、DOWNTIME で指定した間隔で状態変化が無い場合  |

- ⑤ DOWNCNT で指定した条件を満たし、ノード障害とするまでの間隔(100 ミリ秒単位)

+----- SERVERFAULT SG INFORMATION DISPLAY -----+

- ① アクセスサーバの滞留電文数を監視する間隔(100 ミリ秒単位)
- ② アクセスサーバの滞留電文数上限しきい値
- ③ マスタ TAM インスタンスに対応するアクセスサーバ異常終了時の再起動回数
- ④ スレーブ TAM インスタンスに対応するアクセスサーバ異常終了時の再起動回数
- ⑤ アクセスサーバが再起動したあと正常に起動したと判定する時間(秒単位)

+----- BRIDGEFAULT SG INFORMATION DISPLAY -----+

- ① ブリッジサーバのプロセス起動状態を監視する間隔(100 ミリ秒単位)
- ② ブリッジサーバ異常終了時の再起動回数
- ③ ブリッジサーバが再起動したあと正常に起動したと判定する時間(秒単位)

+----- TAMFAULT SG INFORMATION DISPLAY -----+

- ① TAM インスタンスの稼動状態を監視する間隔(100 ミリ秒単位)
- ② TAM のマスタ切替失敗時のリトライ回数
- ③ TAM のマスタ切替失敗時にリトライする間隔(100 ミリ秒単位)

+----- BRIDGE SG INFORMATION DISPLAY -----+

- ① ブリッジサーバの多重度
- ② ブリッジサーバ間の再接続間隔(秒単位)
- ③ ブリッジサーバが結果を返却するための IMS キューに使用するメモリサイズ(K バイト単位)
- ④ ブリッジサーバが結果を返却するための IMS キューに使用するブロックサイズ(バイト単位)
- ⑤ アクセス統計情報を蓄積するか否か

YES	アクセス統計情報を蓄積する
NO	アクセス統計情報を蓄積しない

- ⑥ アクセス情報ログを蓄積するか否か
 

YES	アクセス情報ログの蓄積を行う
NO	アクセス情報ログの蓄積を行わない
- ⑦ アクセス情報ログを蓄積する領域のサイズ(K バイト単位)
- ⑧ アクセス情報ログの蓄積領域をサイクリックに使用するか否か
 

YES	サイクリックに使用する
NO	一杯になったら停止する

+----- USERAP SG INFORMATION DISPLAY -----+

- ① メインキーをハッシュ値に変換するためのハッシュ関数名
- ② 利用者アプリケーションがアクセス要求を行ってからアクセス結果を受け取るまでの監視タイマ値(ミリ秒単位)
- ③ IMS にアクセス要求するための IMS キューに使用する共有メモリバッファのメモリサイズ(K バイト単位)
- ④ IMS にアクセス要求するための IMS キューに使用する共有メモリバッファのブロックサイズ(バイト単位)
- ⑤ 利用者情報領域の領域長(バイト単位)
- ⑥ 利用者情報領域の初期値を設定する関数名

+----- MAP SG INFORMATION DISPLAY -----+

- ① MAP ID(SG 定義では削除されているが共有メモリから削除されていない MAP の場合は直後に“\*”が表示される。)
- ② アクセスサーバのアクセススレッド多重度
- ③ トランザクション管理テーブル(コミットするまでの更新中レコードを管理)に格納できるレコード数
- ④ アクセスサーバが結果を返却するための IMS キューに使用するメモリサイズ(K バイト単位)
- ⑤ アクセスサーバが結果を返却するための IMS キューに使用するブロックサイズ(バイト単位)

- ⑥ TAM の更新ログを蓄積するか否か
  - YES      更新ログを蓄積する
  - NO        更新ログを蓄積しない
- ⑦ アクセス統計情報を蓄積するか否か
  - YES      アクセス統計情報を蓄積する
  - NO        アクセス統計情報を蓄積しない
- ⑧ コミット方式
  - NORMAL 通常コミット
  - GROUP   グループコミット
- ⑨ グループ化するコミット要求数(コミット方式がグループコミットの場合のみ有効)
- ⑩ IMS キュー判定有無(コミット方式がグループコミットの場合のみ有効)
  - YES      IMS キューが空となった時点でコミット数、制限時間を満たしてなくてもグループコミットを実行する
  - NO        IMS キューの状態に関わらずコミット数または制限時間の条件を達成するまで待つ
- ⑪ コミット要求数の合計がグループ化するコミット要求数に満たない場合にコミット要求の到着を待つ時間  
(ミリ秒単位、コミット方式がグループコミットの場合のみ有効)
- ⑫ アクセス情報ログを蓄積するか否か
  - YES      アクセス情報ログの蓄積を行う
  - NO        アクセス情報ログの蓄積を行わない
- ⑬ アクセス情報ログを蓄積する領域のサイズ(K バイト単位)
- ⑭ アクセス情報ログの蓄積領域をサイクリックに使用するか否か
  - YES      サイクリックに使用する
  - NO        一杯になったら停止する

MAP 毎に割り当てるハッシュ値の範囲に関する情報照会(diimref -H)

```
+===== IN-MEMORY SERVER CONTROL INFORMATION DISPLAY =====+ 2011/09/11 19:00:00 ①
+----- HASHRANGE INFORMATION DISPLAY -----+
MAPID = 11 ①
HASHRANGE      = 1001-1009, 2001-2009, 3001-3009, 4001-4009, 5001-5009, 6001-6009, 7001-7009, 8001-8009, 9001-9009, 10001-10009 ②
HASHRANGE (SG) = 91001-91009, 92001-92009, 93001-93009 ②
MAPID = 12
HASHRANGE      = 1011-1019, 2101-2109
HASHRANGE (SG) = 55011-55019
MAPID = 21
HASHRANGE      = 51-100, 151-200
HASHRANGE (SG) = 1-200, 501-800
+===== END OF DISPLAY =====+
```

① コマンド実行時刻

+----- HASHRANGE SG INFORMATION DISPLAY -----+

- ① MAP ID (削除予定の場合は直後に“\*”が表示される。)
- ② 現在有効なハッシュ値範囲(開始ハッシュ値-終了ハッシュ値, 開始ハッシュ値-終了ハッシュ値, …)
- ③ SG 定義のハッシュ値範囲(開始ハッシュ値-終了ハッシュ値, 開始ハッシュ値-終了ハッシュ値, …)

## 2.3.14 diimtblopenclose (表制御コマンド)

### 名前

diimtblopenclose - 表のオープン／クローズを行う

### 書式

```
diimtblopenclose [-m MAP_ID] -o {LOGICAL_TABLE_NAME}...
diimtblopenclose [-m MAP_ID] -c {LOGICAL_TABLE_NAME}...
diimtblopenclose -h
```

### 説明

**diimtblopenclose** は、**MAP\_ID** で指定した MAP にある **LOGICAL\_TABLE\_NAME** に指定した表のオープンおよびクローズを行う。

### オプション

**-m MAP\_ID**

MAP を指定する。省略した場合は **diimtblopenclose** を実行したノードの全 MAP が対象となる。

**-o**

表をオープンする場合に指定する。

**-c**

表をクローズする場合に指定する。

**LOGICAL\_TABLE\_NAME ...**

オープン、または、クローズ対象の論理表名を指定する。複数指定する場合は、論理表名をスペース区切りで指定する。

**-h**

使用方法を表示する。

### 戻り値

**diimtblopenclose** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了 全ての表で処理失敗
2	警告終了 一部の表で処理失敗

### 注意

- 論理表名は 255 文字以内、最大 1023 個まで指定可能とする。
- **LOGICAL\_TABLE\_NAME ...** は、必ず **-m**、**-c**、**-o** よりも後ろに指定する必要がある。
- **-m** が指定され、対象 MAP の IMS アクセスサーバが未起動の場合は、異常終了する。
- **-m** が指定され、指定された論理表がその MAP に定義されていなかった場合は、エラーとなる。
- **diimtblopenclose** は、**diimctrl** によりスレーブの TAM インスタンス、およびインメモリサーバを起動している最中に実行しないこと。実行した場合は、スレーブの TAM インスタンス、およびインメモリサーバの起動が失敗する可能性がある。
- **-m** を指定しない場合で、かつ、1 レプリケーショングループ配下に大量の MAP を定義している場合 (おおよそ 64 以上) はコマンドがタイムアウトする可能性がある。その場合は **-m** を指定して MAP 単位にコマンドを実行しなおす必要がある。
- **-m** が省略された場合は、指定された論理表が定義されている MAP に対してのみ処理される。

例) 論理表と MAP 分散例

	MAP#1	MAP#2	MAP#3
論理表 A	○	○	○
論理表 B	○	○	○
論理表 C	○		

オプション指定例	処理結果
-m 2 -o A C	異常終了(論理表未定義エラー)
-o A C (-m 省略)	MAP#1 の論理表 A, C をオープン MAP#2 の論理表 A をオープン MAP#3 の論理表 A をオープン

## 2.3.15 diimterm(メモリキャッシュ停止コマンド)

### 名前

diimterm - メモリキャッシュの停止を行う

### 書式

```
diimterm [-f] [-T TIMEOUT]
diimterm -h
```

### 説明

**diimterm** は、インメモリサーバの停止、TAM インスタンスの停止、インメモリサーバ所在管理デーモンの停止、インメモリサーバ所在管理およびインメモリサーバの共有メモリ削除を行う。

### オプション

**-f**

強制モードで停止する。ノード内にマスタの TAM が存在する場合は、マスタの TAM インスタンスは停止しない。

**-T TIMEOUT**

コマンドのタイムアウト時間を指定する。(秒単位:0~2147483647、0 は無制限、既定値:30)

**-h**

使用方法を表示する。

### 戻り値

diimterm は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- OLTP 層のメモリキャッシュデーモン計画停止時は、ditamswap でノード内の全マスタをスレーブに切替後、-f オプション省略の diimterm を実行する。
- f オプションを省略した場合、ノード内にマスタの TAM が存在する場合は、異常終了する。

### 関連

diiminit, ditamswap

## 2.3.16 ditamswap(マスタ切替コマンド)

### 名前

ditamswap - TAM のマスタ切替を行う

### 書式

```
ditamswap [-A] [-s] [-T TIMEOUT] [-j {on|off}]
ditamswap -r REPLICATION_GROUP_ID [-n LNODENAME] [-T TIMEOUT] [-j {on|off}]
ditamswap -r REPLICATION_GROUP_ID -s [-T TIMEOUT] [-j {on|off}]
ditamswap -h
```

### 説明

TAM のマスターサーバとスレーブサーバを切り替える。  
本コマンドは、マスタの TAM が存在するノードで投入する。

### オプション

#### -A

自ノードがマスタの全 TAM をスレーブに切り替える。  
新マスタのノードは、インメモリサーバ所在管理が自動的に決定する。

#### -s

新マスタを環境定義でマスタとして定義しているノードとする。  
自ノード、または正常稼動していないノードがマスタとして定義しているレプリケーショングループは処理対象外となる。

#### -r REPLICATION\_GROUP\_ID

自ノードにマスタが存在する特定の TAM をスレーブに切り替える場合に、対象の TAM が属するレプリケーショングループの ID を指定する。

#### -n LNODENAME

新マスタとするノードの論理ノード名を指定する。  
-n 省略時は、新マスタのノードは、インメモリサーバ所在管理が自動的に決定する。

#### -T TIMEOUT

コマンドのタイムアウト時間を指定する。(秒単位:0～2147483647、0 は無制限、既定値:30)

#### -j

on を指定した場合、切替先で TAM の更新ログを蓄積する MAP の更新ログを開始する。(既定値)  
off を指定した場合、切替先で TAM の更新ログを蓄積する MAP の更新ログを開始しない。

#### -h

使用方法を表示する。

### 戻り値

ditamswap は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- インメモリサーバ所在管理が自動的に新マスタを決定する際は、既存のマスタが最も少ないノードを選択する。
- TAM の更新ログを蓄積する定義で、`-j off` を指定した場合、`diimctrl -b -j on` により更新ログを開始するまでは、TAM への更新処理を行うことはできない。

## 関連

`diimterm`

## 2.3.17 ditamswitch(マスタ昇格コマンド)

### 名前

ditamswitch - TAM のスレーブをマスタに昇格する

### 書式

ditamswitch -r REPLICATION\_GROUP\_ID [-T TIMEOUT]

ditamswitch -h

### 説明

TAM のマスタが障害となった場合に、スレーブをマスタに昇格させる。

本コマンドは、昇格させるスレーブの TAM が存在するノードで投入する。

### オプション

**-r REPLICATION\_GROUP\_ID**

自ノードにスレーブが存在する特定 TAM をマスタに切り替える場合に、対象の TAM が属するレプリケーショングループの ID を指定する。

**-T TIMEOUT**

コマンドのタイムアウト時間を指定する。(秒単位:0~2147483647、0 は無制限、既定値:30)

**-h**

使用方法を表示する。

### 戻り値

ditamswitch は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了
- 2      警告終了

### 注意

- 自動的に障害検出する定義を行っている場合は、本コマンドを投入する必要はない。
- 本コマンドは、自動的に障害検出する定義を行わず、サービスガード等を使用したパッケージ切替による稼働一待機構成とする場合に、パッケージ切替時に自動実行するコマンドとして定義する。
- TAM の更新ログを蓄積する MAP の更新ログは開始しない。

## 2.4 データストア基盤

### 2.4.1 コマンド一覧

#### (1) ディレード

didltblock	スーパーストリーム、またはユニットの無効化状態を変更する。
didltchg	定義変更で更新した DB の情報を共有メモリに反映する。
didltcreate	ディレード転送用のテーブルを生成する。
didltctrl	ディレード転送の各種デーモンを起動/停止する。
didltdivchg	指定されたスーパーストリームのディビジョン切り替えを行う。
didltinit	ディレード転送機能を起動する。
didltmodreset	動作変更コマンドで変更された値を環境定義値に戻す。
didltpathref	ディレード転送機能のノード間パスの状態を照会する。
didltpooldel	処理済みログデータを削除する。
didltpoolmod	プールファイル制御に関する各種動作変更指示を行う。
didltpoolref	ディレード転送のプールファイル状況を表示する。
didltref	ディレード転送機能のデーモン起動状態や、スーパーストリームの構成情報を照会する。
didltrefdb	DB に管理している情報を出力する。
didltslmmv	スーパーストリームの動作ノードを変更する。
didltslmref	スーパーストリームの所在管理情報を照会する。
didltterm	ディレード転送停止。
didltupd	定義変更をおこなう。

#### (2) センダ

didtsctrl	センダ機能のデーモンを起動/停止する。
didtsdivend	転送中のディビジョンを強制的に終了する。
didtsinit	センダ機能を開始する。
didtsmod	センダの各種動作環境を変更する。
didtsref	センダのデータ転送状況を表示する。
didtsstart	ログデータの送信を開始する。
didtsstop	ログデータの送信を停止する。
didstern	センダ機能を終了する。

#### (3) レシーバ

didtrctrl	レシーバ機能のデーモンを起動/停止する。
didtrdivend	転送中のディビジョンを強制的に終了する。
didtrinit	レシーバ機能を開始する。
didtrmod	レシーバの各種動作環境を変更する。
didtrref	レシーバのデータ転送状況を表示する。
didtrstart	ログデータの受信を開始する。
didtrstop	ログデータの受信を停止する。

didtrterm      レシーバ機能を終了する。

(4)      **ログリーダー**

didtlctrl      ログリーダー機能のデーモンを起動/停止する。

didtldivend      データ処理中のディビジョンを強制的に終了する。

didtlinit      ログリーダー機能を開始する。

didtlmod      ログリーダーの各種動作環境を変更する。

didtlref      ログリーダーのデータ処理状況を表示する。

didtlstart      ログリーダーのデータ処理を開始する。

didtlstop      ログリーダーのデータ処理を停止する。

didtlterm      ログリーダー機能を終了する。

## 2.4.2 didltblock(ディレード無効化状態変更)

### 名前

didltblock - スーパーストリーム、またはユニットの無効化状態を変更する。

### 書式

```
didltblock {-a|-b} -s SuperStreamName
didltblock {-a|-b} -s SuperStreamName -F {sender|receiver|reader} -u UnitName
didltblock -h
```

### 説明

**didltblock** コマンドは、スーパーストリームやスーパーストリーム配下のユニットの無効化状態を変更する。ユニット指定の場合、上位スーパーストリーム、ユニット種別の指定も必須である。スーパーストリームは、配下の全ユニットが処理停止状態(INIT、INACT、DIVEND)、ユニットは、指定したユニットが処理停止状態(INIT、INACT、DIVEND)の場合に無効化することが可能である。コマンド実行ノードで動作中のスーパーストリームのみ指定可能である。

### オプション

**-a**

指定対象の無効化を解除する。

**-b**

指定対象を無効化する。

**-s**

スーパーストリーム名を指定する。

**-F**

無効化状態変更対象とするユニットの種別を指定する。

<b>sender</b>	センダ
<b>receiver</b>	レシーバ
<b>reader</b>	ログリーダー

**-u**

ユニット名を指定する。

**-h**

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

### 関連

didltinit, didltterm

## 2.4.3 didltchg(ディレード定義反映)

### 名前

didltchg - 環境定義変更を共有メモリに反映する

### 書式

```
didltchg [-A [-M force]] [-C nocheck]
didltchg [-A [-M force]] -o
didltchg -h
```

### 説明

ディレード転送機能の環境定義(DELAYED 節)、または共通の環境定義(DIOSAMAP/SYSMAP 節)変更時、各ノードの共有メモリ情報を更新するためのコマンドである。環境定義を変更した場合、まず定義変更コマンド(didltupd)を実行して DB の情報を更新後、本コマンドで共有メモリの情報を更新する必要があるが、通常は定義変更コマンド(didltupd)に連動して自動的に実行されるため、定義変更をおこなった場合でも、通常は本コマンドを実行する必要はない。

定義変更実行時にディレード転送停止中のノードでは、次回起動時に確実に環境定義から起動するため、共有メモリの引き継ぎファイルを削除する。(ウォーム指定されてもコールドモードで起動する。)

### オプション

#### -A

全ノードに定義反映コマンドを配信する。  
指定しない場合、自ノード上で反映処理をおこなう。

#### -M

エラー発生時のモードを指定する。  
指定しない場合、反映中にエラーが発生した場合、残りのノードへの反映をおこなわずに異常終了する。  
**force**            反映中にエラーが発生しても全ノードへ反映をおこなう。

#### -C

定義反映済みであるかのチェックを行うかどうかを指定する。  
指定しない場合、環境定義オブジェクト(DELAYED 節)と共有メモリの作成日時をチェックし、日時が異なる場合のみ、共有メモリ更新を行う。  
**nocheck**            定義反映済みであるかのチェックを行わず、強制的に共有メモリを更新する。

#### -o

最新の環境定義オブジェクト(DIOSAMAP/SYSMAP 節)で、通信制御機能の制御情報を更新する。  
指定しない場合、ディレード転送定義を反映する。

#### -h

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了
2	自ノードの反映処理は完了したが、追加/削除された機能の制御デーモン起動/停止処理でエラーが発生した。 強制モードで実行中、一部のノードでエラーが発生した。

### 注意

- コマンド実行ノード上で、コマンド配信デーモンが起動している必要がある。

関連

didltinit, didltterm, didltupd

## 2.4.4 didltcreate(ディレード定義生成)

### 名前

didltcreate - ディレード転送用のテーブルを生成する

### 書式

```
didltcreate [-D {im|db [-I InstanceGroupName]]] [-r]
didltcreate [-D {im|db [-I InstanceGroupName]]] [-s SuperStreamName] [-r]
didltcreate -h
```

### 説明

**didltcreate** コマンドは、環境定義 DELAYED 節の情報を読み込み、ディレード転送の各機能(DSAM, センダ、レシーバ、ログリーダー)の動作に必要な制御テーブルを作成する。定義生成は、初回のディレード転送起動(didltinit)実行前に必ず実行する必要がある。

メモリキャッシュと Oracle の両方が動作するシステムでは、両方にアクセス可能なノードでコマンドを実行すると、両方のデータベースに制御テーブルを作成する。両方にアクセス可能なノードが存在しない場合、それぞれにアクセス可能なノードで1回ずつコマンドを実行する必要がある。

Oracle 障害中でメモリキャッシュの定義生成のみを実行したい場合や、障害復旧後に Oracle のみ定義生成をおこないたい場合などのために、-D オプションで定義生成をおこなうデータベースを指定することが可能である。

また、-D db 指定時は、すべてのインスタンスグループが正常である必要があるが、-I オプションでインスタンスグループ名まで指定することで、指定されたインスタンスグループのテーブルのみが生成されるため、一部のインスタンスグループが障害中の場合でも、正常なインスタンスグループに対してのみ定義生成をおこなうことが可能である。

### オプション

**-D**

処理対象を指定する。(省略可)

**im**                      メモリキャッシュ

**db**                      Oracle

省略した場合、コマンド実行ノードでアクセス可能なデータベースに対して定義生成をおこなう。

**-I**

インスタンスグループを指定する。(省略可)

-D db を指定した場合のみ指定可能。省略した場合、全てのインスタンスグループが生成対象となる。

**-s**

指定したスーパーストリームのみを生成とする。

**-r**

定義生成中に中断した状態でも、強制的に定義生成を行う。

スーパーストリーム指定ありの場合、現在の定義情報が取得できなくても、強制的に定義生成を行う。

**-h**

使用方法を表示する。

### 戻り値

0                      正常終了

1                      異常終了

### 注意

- コマンド実行ノード上で、コマンド配信デーモンが起動している必要がある。
- 定義生成対象にメモリキャッシュが含まれる場合、IMS 所在管理が起動している必要がある。
- 本コマンド実行前に、環境定義生成コマンド(diirmadd または diirmrep)でディレード転送の SG オブジェクトを生成しておく必要がある。
- 論理システム内の 1 ノードでも、センダ、レシーバ、ログリーダのいずれかが開始コマンド(didtsinit、didtrinit、didtlinit)投入済みの状態の場合、スーパーストリームを指定しない定義生成を実行することはできない。
- Oracle 版スーパーストリームを定義した環境で、ディレード起動中に、スーパーストリームを指定しない定義生成を実行した場合、DB ノード上で定義反映コマンドを実行する必要がある。
- スーパーストリームを指定した定義生成は、事前に全件(スーパーストリーム指定なし)の定義生成を実行しておく必要がある。
- スーパーストリームを指定した定義生成は、スーパーストリームの動作ノードで実行する必要がある。  
他のノードで実行した場合、動作ノードで起動中のセンダ/レシーバ/ログリーダの実行デーモンが停止するため、処理を再開するためには、デーモン起動停止コマンド(didtsctrl/didtrctrl/didtlctrl)を実行して実行デーモンを再起動する必要がある。
- 既に運用中の状態で定義生成をおこなうと、スーパーストリームの指定有無に関わらず、動作変更コマンドで変更したパラメータは、全て環境定義に定義された値にリセットされる。ただし、無効化状態については、スーパーストリーム指定の定義生成を実施しても生成前の状態を引き継ぐ。
- スーパーストリーム指定の定義生成は、インスタンスグループにデフォルト RGSET が含まれないインスタンスグループを指定する場合でも、デフォルト RGSET が含まれるインスタンスグループに接続できる状態でのみ実行可能である。
- 定義パターンと定義生成対象の対応は以下の通りである。

[定義パターン]

No.	定義内容	POOLFILE	USERDATA
1	削除デーモン用定義(IM)	—	—
2	削除デーモン用定義(DB)	—	—
3	スーパーストリーム定義	IM	IM
4	スーパーストリーム定義	IM	DB
5	スーパーストリーム定義	IM	DB(別)
6	スーパーストリーム定義	DB	DB
7	スーパーストリーム定義	DB	DB(別)

※ DB(別)とは、デフォルト RGSET を含まないインスタンスグループの DB を指す。

[定義生成対象]

コマンドパラメータ指定	定義パターン						
	1	2	3	4	5	6	7
なし	D	D	PSRL	PSRL	PSRL	PSRL	PSRL
-D im	D	—	PSRL	PSR	PSR	—	—
-D db	—	D	—	L	L	PSRL	PSRL
-D db -I INS	—	D	—	L	—	PSRL	—
-D db -I INS(別)	—	—	—	—	L	—	L

— ： 定義生成されない

D ： 削除デーモン用定義

P : プールファイル定義  
S : センダユニット定義  
R : レシーバユニット定義  
L : ログリーダユニット定義

※ IM/DB にアクセス可能なノードでコマンドを実行しているものとする。

## 関連

didltinit, diirmadd, diirmrep

## 2.4.5 didltctrl (デーモン起動停止)

### 名前

didltctrl - ディレード転送の各種デーモンを起動/停止する

### 書式

```
didltctrl [-b] [-F {locationm|locationa|pathc|pathd|del|sender|receiver|reader}]
didltctrl -e [-M force] [-F {locationm|locationa|pathc|pathd|del|sender|receiver|reader}]
didltctrl -h
```

### 説明

**didltctrl** コマンドは、ディレード転送起動時に起動される各種デーモンの起動/停止をおこなう。実行デーモンのような、ディレード起動時に起動されないデーモンについては、本コマンドで起動/停止できない。制御デーモンには、ストリーム所在(マネージャ)、ストリーム所在(エージェント)、通信デーモン、振り分けデーモン、削除デーモン、管理デーモン(センダ/レシーバ/ログリーダー)が存在する。

### オプション

**-b**

コマンド実行ノードで起動対象のデーモンで、未起動状態のデーモンを起動する。(省略時の規定値)

**-e**

コマンド実行ノードで起動対象のデーモンで、起動状態のデーモンを停止する。

**-M**

デーモンを強制停止する場合に指定する。

**force**                      制御デーモンを強制停止する。

**-F**

起動/停止対象のデーモンを以下の中から指定する。指定しない場合、そのノードで動作するすべてのデーモンが対象となる。

<b>locationm</b>	ストリーム所在(マネージャ)
<b>locationa</b>	ストリーム所在(エージェント)
<b>pathc</b>	通信デーモン
<b>pathd</b>	振り分けデーモン
<b>del</b>	削除デーモン
<b>sender</b>	センダ管理デーモン
<b>receiver</b>	レシーバ管理デーモン
<b>reader</b>	ログリーダー管理デーモン

**-h**

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了(すべての制御デーモンの起動停止に失敗した。)
2	警告終了(一部の制御デーモンの起動停止に失敗した。)

### 注意

- ディレード転送の各制御デーモンは、ディレード転送起動コマンド(didltinit)、ディレード転送停止コマンド(didltterm)に連動して起動停止するため、通常は実行する必要はない。

- 自ノードで動作中の Oracle スーパーストリームが存在する状態でデーモンを停止しても、スーパーストリームは他ノードへ移動しない。移動させたい場合は、スーパーストリーム所在変更コマンドを実行すること。
- センダ/レシーバ/ログリーダの管理デーモンは、機能開始コマンド(didtsinit, didtrinit, didtlinit)が実行されている場合のみ起動停止対象となる。

## 関連

didltinit, didltterm, didltslmmv

## 2.4.6 didltdivchg(ディビジョン切り替え)

### 名前

didltdivchg - 指定されたスーパーストリームのディビジョン切り替えを行う

### 書式

didltdivchg -s SuperStreamName

didltdivchg -h

### 説明

**didltdivchg** は指定されたスーパーストリームのディビジョン切り替えを行う。

コマンドは、POOLTYPE=WRITER を定義したスーパーストリームに対してのみ実行可能である。

### オプション

**-s**

切替を行うスーパーストリームを指定する。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

以下の終了ステータスが返される。

0                      正常終了

1                      異常終了

### 関連

didltnit

## 2.4.7 didltinit(ディレード起動)

### 名前

didltinit - ディレード転送機能を起動する。

### 書式

didltinit [-M create]

didltinit -h

### 説明

ディレード転送機能を環境定義 (DELAYED 節) または引き継ぎファイルを使用して初期化し、ノードごとに必要な、ディレード転送機能 (ディレード共通、DSAM) の制御デーモンを起動する。

制御デーモンには、ストリーム所在 (マネージャ)、ストリーム所在 (エージェント)、通信デーモン、振り分けデーモン、削除デーモンが存在する。

### オプション

**-M**

起動モードを指定する。

指定しない場合、引継ぎファイルを使用して設定情報を構築する。

**create**                      環境定義 (DELAYED 節) オブジェクトを使用して設定情報を構築する。

**-h**

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了
2	警告終了 (初期化完了したが、デーモン起動に失敗した。)

### 注意

- DIOSA/XTP が起動している必要がある。
- 初回のディレード転送起動時には、定義生成コマンド (didltcreate) を実行しておく必要がある。
- 制御デーモンの起動に失敗した場合、制御デーモン起動コマンド (didltctrl -b) で制御デーモンを起動する必要がある。
- 引継ぎファイルは、当コマンドを -M create オプションで実行した場合、または定義反映コマンド実行時に作成する。

### 関連

didltcreate, didltctrl, didltterm, didltchg

## 2.4.8 didltmodreset(ディレード動作変更リセット)

### 名前

didltmodreset - 動作変更コマンドで変更されたパラメータを環境定義値に戻す。

### 書式

```
didltmodreset [-D {im|db [-I InstanceGroupName]]]
```

```
didltmodreset -h
```

### 説明

**didltmodreset** コマンドは、ディレード転送の各機能(ディレード共通、DSAM、センダ、レシーバ、ログリーダー)の動作変更コマンドで変更されたパラメータを、全て環境定義で定義された値に戻す。

ただし以下の情報は除く。

- ・ディビジョン ID / 通番 / ユーザ通番
- ・相手制御電文通番
- ・相手論理システム名

メモリキャッシュと Oracle の両方が動作するシステムでは、両方にアクセス可能なノードでコマンドを実行すると、両方のデータベースの動作パラメータをリセットする。両方にアクセス可能なノードが存在しない場合、それぞれにアクセス可能なノードで1回ずつコマンドを実行する必要がある。

Oracle 障害中でメモリキャッシュのリセットのみを実行したい場合や、障害復旧後に Oracle のみリセットをおこないたい場合などのために、-D オプションで動作変更リセットをおこなうデータベースを指定することが可能である。また、-D db 指定時は、-I オプションでインスタンスグループ名まで指定することも可能である。-I が指定された場合、指定されたインスタンスグループの動作変更パラメータのみがリセットされる。

### オプション

-D

処理対象を指定する。(省略可)

**im**                      メモリキャッシュ

**db**                      Oracle

省略した場合、コマンド実行ノードでアクセス可能なデータベースに対して動作変更リセット処理おこなう。

-I

インスタンスグループを指定する。(省略可)

-D db を指定した場合のみ指定可能。省略した場合、全てのインスタンスグループの動作パラメータがリセットされる。

-h

使用方法を表示する。

### 戻り値

0                      正常終了

1                      異常終了

### 注意

- ・ 処理対象にメモリキャッシュが含まれる場合、IMS 所在管理が起動している必要がある。

- ディレード起動中にコマンドを実行した場合、リセットされた定義値が即時反映されないパラメータもある。その場合、ディレード転送を再起動する必要がある。起動した状態でパラメータの変更をおこないたい場合は、動作変更コマンドを実行すること。
- 定義パターンと動作変更リセット対象の対応は以下の通りである。

[定義パターン]

No.	アクセス先 DB	
	POOLFILE	USERDATA
1	IM	IM
2	IM	DB
3	IM	DB (別)
4	DB	DB
5	DB	DB (別)

※ DB (別) とは、デフォルト RGSET を含まないインスタンスグループの DB を指す。

コマンドパラメータ指定	定義パターン				
	1	2	3	4	5
なし	SRL	SRL	SRL	SRL	SRL
-D im	SRL	SR	SR	—	—
-D db	—	—	—	SRL	SRL
-D db -I INS	—	—	—	SRL	SR
-D db -I INS (別)	—	—	—	—	L

- ： 処理対象外
- S ： センダユニットの動作変更パラメータをリセット
- R ： レシーバユニットの動作変更パラメータをリセット
- L ： ログリーダユニットの動作変更パラメータをリセット
- ※ IM/DB にアクセス可能なノードでコマンドを実行しているものとする。

関連

didltpoolmod, didtsmod, didtrmod, didtlmod

## 2.4.9 didltpathref (ディレード転送パス状態照会)

### 名前

didltpathref - ディレード転送機能のノード間パスの状態を照会する

### 書式

didltpathref [-h]

### 説明

ディレード転送機能に定義されているノード間の通信パス状態を表示する。

### オプション

-h

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了

実行例

ノード間パス状態照会(didltpathref)

```
+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①
LSNAME      = LS1                               ②
LNODENAME    = oltp1                             ③

+----- CONFIGURATION INFORMATION -----+
MAXTHR_C     = 20                               ①
MINTHR_C     = 10                               ②
MAXTHR_D     = 20                               ③
MINTHR_D     = 10                               ④
RCHKINVL     = 60                               ⑤
HCHKINVL     = 0                                ⑥
DLSIDLETIME  = 60                               ⑦

+----- COMMUNICATION DAEMON INFORMATION -----+
+----- DLS PATH INFORMATION -----+
①          ②          ③          ④
DLSNAME_____ IPADDR_____ PORT_  STATUS
LS1          _____      _____
LS1          10.164.01.100    11111  OPEN
LS2          _____      _____
LS2          10.164.02.100    11111  OPEN
:
:

+----- NODE PATH INFORMATION -----+
①          ②
LNODENAME_____ STATUS
oltp1        OPEN
oltp2        CLOSE
:
:

+----- DISPATCH DAEMON INFORMATION -----+
+----- PATH INFORMATION -----+
①          ②          ③
LNODENAME_____ DLSNAME_____ STATUS
ap1          _____      _____
ap1          LS1            OPEN
ap1          LS2            IDLE
ap2          _____      _____
ap2          LS1            CLOSE
ap2          LS2            CLOSE
:
:

+===== END OF DISPLAY =====+
```

+===== DELAYED TRANSFER DISPLAY =====+

- ① コマンド実行時刻
- ② 自論理システム名
- ③ 自ノード名

+----- CONFIGURATION INFORMATION -----+

環境定義情報表示

- ① 通信デーモン送受信処理最大スレッド数
- ② 通信デーモン送受信処理最小スレッド数
- ③ 振り分けデーモン送受信処理最大スレッド数
- ④ 振り分けデーモン送受信処理最小スレッド数
- ⑤ 復旧監視間隔
- ⑥ ヘルスチェック間隔
- ⑦ 相手論理システム通信対象外判定時間

+----- COMMUNICATION DAEMON INFORMATION -----+

通信デーモンの管理情報を表示 (AP ノードでのみ表示される)

+----- DLS PATH INFORMATION -----+

相手論理システムの通信状態を表示

① 相手論理システム名

② IP アドレス

③ ポート番号

IP アドレス、ポート番号がハイフン”---“の場合は、①の論理システム全体としての通信状態を示す

IP アドレス、ポート番号が設定されている場合は、接続先ごとの通信状態を示す

④ 接続状態

OPEN : 通信可能

CLOSE : 通信不可

IDLE : 一定時間以上未使用 (通信可能とみなす)

+----- NODE PATH INFORMATION -----+

自論理システム内の振り分けデーモンとの通信状態を表示

① 論理ノード名

② 接続状態

OPEN : 通信可能

CLOSE : 通信不可

+----- DISPATCH DAEMON INFORMATION -----+

振り分けデーモンの管理情報を表示 (振り分けデーモンが動作するノードでのみ表示される)

+----- PATH INFORMATION -----+

① 論理ノード名

② 相手論理システム名

ハイフン”---“の場合は、①の論理ノードの通信デーモンとの通信状態を示す

論理システム名が設定されている場合は、①の論理ノードと相手論理システムとの通信状態を示す

③ 接続状態

OPEN : 通信可能

CLOSE : 通信不可

IDLE : 一定時間以上未使用 (通信可能とみなす)

## 2.4.10 didltpooldel (ログデータ削除)

### 名前

didltpooldel - 処理済みログデータ削除

### 書式

didltpooldel -s SuperStreamName [-M all]

didltpooldel -h

### 説明

didltpooldel は、処理済みとなったログデータを削除する。削除はスタックファイル単位に行われる。

削除するスタック数は、環境定義の削除スタック数に従うが、-M all を指定することで、削除可能なスタックファイルを全て削除することも可能である。

-s オプションで指定するスーパーストリームは、コマンド実行ノードで動作中である必要がある。

### オプション

-s

指定したスーパーストリームのログデータを削除する。

-M

削除モードを指定する。

省略した場合は、処理済となったスタックファイル全てを削除対象とせず、ディレード転送の環境定義に指定された削除スタックファイル数 (DELAYED 節 SUPERSTREAM 項 DELSTACKCNT パラメータ) 分のスタックファイルを削除対象とする。

all                      処理済となったスタックファイル全てを削除対象とする。

-h

使用方法を表示する。

### 戻り値

以下の終了ステータスが返される。

0                          正常終了

1                          異常終了

## 2.4.11 didltpoolmod(プールファイル動作変更)

### 名前

didltpoolmod - プールファイル制御に関する各種動作変更指示を行う

### 書式

```
didltpoolmod -T divid=value -T datano=value -T userdatano=value -s SuperStreamName
didltpoolmod -h
```

### 説明

**didltpoolmod** はプールファイル制御に関する各種動作変更指示を行う。以下の動作パラメータが変更可能である。

- ・開始ディビジョン番号・開始通番の変更

### オプション

**-T**

動作種別を指定する。

<b>divid</b>	開始ディビジョン ID(1 以上)
<b>datano</b>	開始通番(0 以上)
<b>userdatano</b>	開始利用者通番(0 以上)

**-s**

指定したスーパーストリームを変更対象とする。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

以下の終了ステータスが返される。

0	正常終了
1	異常終了

### 注意

- ・ **-T** オプションで指定するパラメータは、“=”の両辺に空白を指定してはならない。
- ・ **-T** オプション **divid**、**datano**、**userdatano** 指定の場合は、定義生成コマンド(didltcreate)で生成された制御 DB に対して変更内容を反映するため、先に定義生成コマンドで制御 DB を生成しておく必要がある。

### 関連

didltnit

## 2.4.12 didltpoolref (プールファイル状況照会)

### 名前

didltpoolref - ディレード転送のプールファイル状況を表示する

### 書式

```
didltpoolref -d stack -s SuperStreamName
didltpoolref -d pool -s SuperStreamName [-v]
didltpoolref -d swap
didltpoolref -d del
didltpoolref -h
```

### 説明

**didltpoolref** はディレード転送のプールファイル状況を表示する。以下の情報が照会可能である。

- ・スタックファイル情報表示
- ・プールファイル登録状況表示
- ・スワップ状況表示
- ・削除デーモン状況表示

### オプション

**-d**

表示種別を指定する。

<b>stack</b>	スタックファイル情報表示
<b>pool</b>	プールファイル登録状況表示
<b>swap</b>	スワップ状況表示
<b>del</b>	削除デーモン状況表示

**-s**

表示するスーパーストリーム名を指定する。

**-v**

詳細情報を表示する。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

以下の終了ステータスが返される。

0	正常終了
1	異常終了

### 注意

- ・スワップ状況表示で表示されるのは、コマンド実行ノードのノード種別で動作するスーパーストリームのみである。

実行例

スタック情報照会 (TAM 版スーパーストリーム) (didltpoolref -d stack -s スーパーストリーム名)

+===== DELAYED TRANSFER DISPLAY =====+										2011/09/22 22:22:22 ①													
+----- SUPERSTREAM INFORMATION -----+																							
NAME		= TAM0000000001_1										①											
POOLFILEID		= 0										②											
POOLTYPE		= WRITER										③											
MTNSTATUS		= NORMAL										④											
SWAPSIZE		= 100										⑤											
BLKSTATUS		= ACT										⑥											
DELTYPE		= AUTO										⑦		DELSTACKCNT		= 2		⑧					
PUTSTACKID		= 1										⑨		GETSTACKID		= 1		⑩					
+----- STACK INFORMATION -----+																							
NO		STACKID		STATUS_		STACKDATASIZE																	
①		②		④		⑤																	
01		1		USE		0																	
02		2		INIT		---																	
03		3		INIT		---																	
04		4		INIT		---																	
05		5		INIT		---																	
+===== END OF DISPLAY =====+																							

スタック情報照会 (Oracle 版スーパーストリーム) (didltpoolref -d stack -s スーパーストリーム名)

```
+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①
+----- SUPERSTREAM INFORMATION -----+
NAME           = ORA_SPST_1           ①
POOLFILEID     = --                   ②
POOLTYPE       = WRITER                ③
MTNSTATUS      = NORMAL                ④
SWAPSIZE       = 100                  ⑤
BLKSTATUS      = ACT                   ⑥
DELTYPE        = AUTO                  ⑦  DELSTACKCNT      = 2           ⑧
PUTSTACKID     = 1                     ⑨  GETSTACKID       = 1           ⑩

+----- STACK INFORMATION -----+
NO  STACKID  PARTID_____  STATUS_  STACKDATASIZE
①      ②      ③              ④              ⑤
01      1    ORA_SPST_1_01    USE        0
02      2    ORA_SPST_1_02    INIT        ---
03      3    ORA_SPST_1_03    INIT        ---
04      4    ORA_SPST_1_04    INIT        ---
05      5    ORA_SPST_1_05    INIT        ---

+===== END OF DISPLAY =====+
```

- ① コマンド実行時刻
- +----- SUPERSTREAM INFORMATION -----+
- スーパーストリーム情報表示
- ① スーパーストリーム名
- ② プールファイル ID (TAM 版スーパーストリームの場合のみ表示)
- ③ プールファイル種別
- WRITER : ライタ
- RECEIVER : レシーバ
- ④ メンテナンス状態
- NORMAL : 通常状態
- CREATE : 定義生成中
- UPDATE : 定義変更中
- UPDATECHK : 定義変更チェック中
- ⑤ スワップサイズ (単位は MByte)
- ⑥ 無効化状態
- ACT : 通常状態
- BLOCK : 無効化中
- ⑦ データ削除方式
- AUTO : 自動削除
- ⑧ 削除スタック数

- ⑨ 書込み先スタック ID
- ⑩ 最古読込み中スタック ID

+----- STACK INFORMATION -----+

スタック情報表示

- ① スワップ順番
- ② スタック ID
- ③ パーティション ID(Oracle 版スーパーストリームの場合のみ表示)
- ④ 使用状態
  - INIT : 初期化済み
  - USE : 使用中
  - FREE : 処理済み(未初期化)
  - INITING : 初期化中
- ⑤ スタックに格納されているログデータ合計サイズ(単位は MByte)

プールファイル情報照会 (didltpoolref -d pool -s スーパーストリーム名 [-v])

```
+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①
+----- SUPERSTREAM INFORMATION -----+
NAME          = TAM0000000001_1 ①
RGSETNAME     = --- ②
POOLFILEID    = 0 ③
POOLTYPE      = WRITER ④
MTNSTATUS     = NORMAL ⑤
POOLFULLFLAG  = NORMAL ⑥
BLKSTATUS     = ACT ⑦
+----- CURRENT INFORMATION -----+
DIVID_____ FIXNO_____ USERFIXNO_ FIXSTACKID PUTNO_____ USERPUTNO_ PUTSTACKID
 ①           ②           ③           ④           ⑤           ⑥           ⑦
    1         0         0         1         0         0         1
+----- HISTORY INFORMATION -----+
DIVID_____ FIXNO_____ USERFIXNO_ FIXSTACKID PUTNO_____ USERPUTNO_ PUTSTACKID
 ①           ②           ③           ④           ⑤           ⑥           ⑦
    2         222       222         1         222       222         1
    3         333       333         1         333       333         1
+----- STACK INFORMATION -----+
DIVID_____ STACKID_ DATA-FROM_ DATA-TO_ SWAPCNT_____
 ①           ②           ③           ④           ⑤
    1         1         1         ---         0
    2         1         1         222         0
    3         1         1         333         0
+----- FINISH NO INFORMATION -----+
TYPE  UNITNAME_____ DIVID_____ FINISHNO_ BLKSTATUS
①    ②           ③           ④    ⑤
S     SND01           1         0  ACT
L     LRD01           1         0  ACT
+===== END OF DISPLAY =====+
```

① コマンド実行時刻

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報表示

① スーパーストリーム名

② リソースグループセット名 (Oracle 版スーパーストリームの場合のみ表示)

③ プールファイル ID (TAM 版スーパーストリームの場合のみ表示)

④ プールファイル種別

WRITER : ライタ

RECEIVER : レシーバ

⑤ メンテナンス状態

NORMAL : 通常状態

CREATE : 定義生成中

UPDATE : 定義変更中

UPDATECHK : 定義変更チェック中

⑥ プールファイル満杯状態フラグ

NORMAL : 通常状態

FULL : 満杯状態 (PART と表示された場合も同様)

⑦ 無効化状態

ACT : 通常状態

BLOCK : 無効化中

+----- CURRENT (HISTORY) INFORMATION -----+

最新ディビジョン (過去ディビジョン) ストリーム情報表示 (過去ディビジョンの情報は -v 指定時のみ表示)

① ディビジョン ID

② データの確定通番

③ ユーザデータの確定通番

④ 確定スタック ID

⑤ データの最大登録通番

⑥ ユーザデータの最大登録通番

⑦ 書込み先スタック ID

+----- STACK INFORMATION -----+

スタック情報表示(～v 指定時のみ表示)

- ① デイビジョン ID
- ② スタック ID
- ③ 開始通番
- ④ 終了通番(最新スタックの場合、終了通番は表示されない)
- ⑤ スワップ回数

+----- FINISH NO INFORMATION -----+

ユニットごとの処理済通番情報表示(～v 指定時のみ表示)

- ① ユニット種別
  - S           : センダユニット
  - L           : ログリーダユニット
- ② ユニット名
- ③ デイビジョン ID(無効化中の場合表示されない)
- ④ 処理済通番(無効化中の場合表示されない)
- ⑤ 無効化状態
  - ACT        : 通常状態
  - BLOCK     : 無効化中

スワップ情報照会 (didltpoolref -d swap)

+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①										
+----- SUPERSTREAM INFORMATION -----+										
TYPE	NAME_____	POOLFILEID	ACC	BLKSTATUS	MTNSTATUS	PUTSTACKID	SWAPCNT__	SWAPSTS	DELSTS	
①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	
W	TAM0000000001_1	0	IM	ACT	NORMAL	1	0	NORMAL	NORMAL	
W	TAM0000000011_1	0	IM	ACT	NORMAL	1	0	NORMAL	NORMAL	
+===== END OF DISPLAY =====+										

① コマンド実行時刻

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報表示

① プールファイル種別

W : ライタ

R : レシーバ

② スーパーストリーム名

③ プールファイル ID (TAM 版スーパーストリームのみ表示)

④ プールファイルの DB 種別

IM : TAM

DB : Oracle

⑤ 無効化状態

ACT : 通常状態

BLOCK : 無効化中

⑥ メンテナンス状態フラグ

NORMAL : 通常状態

CREATE : 定義生成中

UPDATE : 定義変更中

UPDATECHK : 定義変更チェック中

⑦ 書き込み先スタック ID

⑧ スワップ回数

⑨ スワップ状態

NORMAL : 通常状態

SWAP : スワップ中

⑩ 削除状態

NORMAL : 通常状態

DEL : 削除処理中

# 削除デーモン情報照会(didltpoolref -d del)

+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①		
+----- DELETE DAEMON INFORMATION -----+		
DMNSTATUS	= INACT	①
DELINVL_IM	= 10	②
DELINVL_DB	= 10	③
CONNSTATUS_IM	= CONN	④
CONNSTATUS_DB	= CONN	⑤
EXEETIME_IM	= 2011/09/22 22:22:00	⑥
EXEETIME_DB	= 2011/09/22 22:22:00	⑦
+===== END OF DISPLAY =====+		

## ① コマンド実行時刻

+----- DELETE DAEMON INFORMATION -----+

## 削除デーモン情報表示

### ① 削除デーモンステータス

- INIT : デーモン停止中
- INACT : デーモン起動中(監視インターバルによる待機中)
- ACT : デーモン処理中(削除処理実行中)

### ② TAM データ削除実行間隔(単位は分)

### ③ Oracle データ削除実行間隔(単位は分)

### ④ TAM 接続状態

### ⑤ Oracle 接続状態

- INIT : 初期状態 / 接続対象外
- CONN : 接続中
- DISCONN : 切断中

### ⑥ TAM データ削除前回実行時刻

### ⑦ Oracle データ削除前回実行時刻

## 2.4.13 didltref(ディレード状態照会)

### 名前

didltref - ディレード転送機能のデーモン起動状態や、スーパーストリームの構成情報を照会する。

### 書式

```
didltref -d {all|config} [-s SuperStreamName]
didltref -d system
didltref -d location
didltref -h
```

### 説明

ディレード転送全機能(センダ、レシーバ、ログリーダー、DSAM)の制御デーモンに関するプロセス情報やスーパーストリームやストリームに関する構成情報等を表示する。

### オプション

**-d**

表示する情報を指定する。

<b>all</b>	全ての情報を表示する。
<b>config</b>	構成情報を表示する。
<b>system</b>	システム情報を表示する。
<b>location</b>	スーパーストリームの動作ノード一覧を表示する。

**-s**

指定したスーパーストリームの情報を表示する。

省略した場合、すべてのスーパーストリームの情報を表示する。

**-h**

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了
2	警告終了

### 注意

- **-d location** で Oracle 版のスーパーストリーム動作ノードが表示されるのは、コマンドを実行したノードが、ストリーム所在管理エージェントデーモンが動作するノードの場合のみである。TAM 版のスーパーストリーム動作ノードについては、全ての AP ノード、OLTP ノードで表示される。

### 関連

didltinit, didltterm

実行例

構成情報照会(didltref -d config)

```
+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①
+----- SYSTEM INFORMATION -----+
LSID      = 1                      ①
LSNAME    = LS1                    ②
LNODETYPE = OLTP                    ③
LNODENAME = oltp1                  ④
+----- CONFIGURATION INFORMATION -----+
REV       = 1                      ①
CREATETIME = 2011/09/22 22:22:00  ②
+----- TOTAL UNIT NUMBER -----+
SENDER    (TAM)      = 1            ①
SENDER    (ORACLE)   = 0            ②
RECEIVER  (TAM)      = 1            ③
RECEIVER  (ORACLE)   = 0            ④
LOGREADER (TAM)      = 1            ⑤
LOGREADER (ORACLE)   = 0            ⑥
LOGREADER (TAM-ORACLE) = 2          ⑦
+----- SUPERSTREAM INFORMATION -----+
NAME      = TAM0000000001_1        ①
ALIAS     = TAM0000000001_1        ②
MAPID     = 10                     ③
RGSETNAME = ---                     ④
POOLFILE  = TAM                     ⑤
POOLTYPE  = WRITER                  ⑥
POOLSTATUS = NORMAL                 ⑦
SENDMSGSIZE = 32000                 ⑧
COMPRESS  = YES                     ⑨
+----- UNIT INFORMATION -----+
TYPE      NAME      MAPID      RGSET
 ①        ②          ③        ④
SENDER    sender1    10      ---
LOGREADER logreader1 10      ---
LOGREADER logreader2 ---    RGSET01
      :
      :
+===== END OF DISPLAY =====+
```

① コマンド実行時刻

+----- SYSTEM INFORMATION -----+

共通のシステム情報表示

① 論理システム ID

② 論理システム名

③ 論理ノード種別

④ 論理ノード名

+----- CONFIGURATION INFORMATION -----+

構成の全体情報表示

① 定義リビジョン

② 作成日時

+----- TOTAL UNIT NUMBER -----+

ユニット数情報表示

① センダユニット総数(TAM)

② センダユニット総数(Oracle)

③ レシーバユニット総数(TAM)

④ レシーバユニット総数(Oracle)

⑤ ログリーダーユニット総数(TAM)

⑥ ログリーダーユニット総数(Oracle)

⑦ ログリーダーユニット総数(TAM-Oracle)

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報表示

- ① スーパーストリーム名
- ② エイリアス名
- ③ スーパーストリーム ID
- ④ MAPID (ログデータ格納先が TAM の場合に値を表示)
- ⑤ リソースグループセット名 (ログデータ格納先が ORACLE の場合に値を表示)
- ⑥ ログデータ格納先
  - TAM : TAM
  - ORACLE : Oracle
- ⑦ プールファイル種別
  - WRITER : ライタ
  - RECEIVER : レシーバ
- ⑧ プールファイル状態
  - NORMAL : 通常状態
  - FULL : 満杯状態
- ⑨ TAM スタックファイル数 (ログデータ格納先が TAM の場合に値を表示)
- ⑩ 最大ログデータ長 (最大ユーザ電文長)
- ⑪ ログデータ圧縮指定
  - YES : 圧縮する
  - NO : 圧縮しない

+----- UNIT INFORMATION -----+

スーパーストリーム配下のユニット情報表示

- ① ユニット種別
- ② ユニット名
- ③ MAPID (センダ、レシーバのログデータ格納先、ログリーダーのユーザデータ格納先が TAM の場合のみ)
- ④ リソースグループセット名 (センダ、レシーバのログデータ格納先、ログリーダーのユーザデータ格納先が Oracle の場合のみ)

システム情報照会 (didltref -d system)

+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①				
+----- SYSTEM INFORMATION -----+				
LSID	= 1			①
LSNAME	= LS1			②
LNODETYPE	= OLTP			③
LNODENAME	= oltp1			④
+----- SYSTEM CONTROL INFORMATION -----+				
FAILOVER	= NO			①
DMNRESTINVL	= 5			②
DMNDOWNCNT	= 5			③
DMNDOWNTIME	= 60			④
RCVRYCHKINVL	= 60			⑤
+----- DAEMON LOCATION -----+				
SUPERSTREAM (TAM)	= OLTP			
SUPERSTREAM (ORACLE)	= AP			
PATH-C	= AP			
PATH-D	= AP/OLTP			
LOCATION-M	= DB			
LOCATION-A	= AP			
+----- DAEMON LIST -----+				
FUNCTYPE	PID	PRCNAME	STARTTIME	DMNSTATUS
①	②	③	④	⑤
PATH-C	---	didltpathcommd	---	---
PATH-D	9632	didltpathdispd	2011/09/22 11:11:11	AVAIL
LOCATION-M	---	didltslmmgrd	---	---
LOCATION-A	---	didltslmagtd	---	---
SENDER	9635	didtsdmnmngd	2011/09/22 11:11:11	AVAIL
RECEIVER	9646	didtrdmnmngd	2011/09/22 11:11:11	AVAIL
LOGREADER	9650	didtldnmngd	2011/09/22 11:11:11	AVAIL
DEL	---	didltpooldeld	---	UNAVAIL
+===== END OF DISPLAY =====+				

① コマンド実行時刻

+----- SYSTEM INFORMATION -----+

共通のシステム情報表示

- ① 論理システム ID
- ② 論理システム名
- ③ 論理ノード種別
- ④ 論理ノード名

+----- SYSTEM CONTROL INFORMATION -----+

ディレード共通の制御情報

- ① 死活監視リトライオーバー時のフェールオーバー指定
- ② 実行デーモン再起動のリトライ間隔
- ③ 実行デーモン再起動中止判定回数
- ④ 実行デーモン再起動中止判定時間
- ⑤ 障害復旧監視間隔

+----- DAEMON LOCATION -----+

動作ノード情報

TAM 版、Oracle 版のスーパーストリーム、および各デーモンが動作する論理ノード種別を表示する。  
複数の論理ノード種別で起動する場合、"/"で区切って複数表示する。

+----- DAEMON LIST -----+

[各デーモンの起動状態]

- ① デーモン種別
  - PATH-C : 通信制御 通信デーモン
  - PATH-D : 通信制御 振り分けデーモン
  - LOCATION-M : ストリーム所在管理マネージャデーモン
  - LOCATION-A : ストリーム所在管理エージェントデーモン

- SENDER

:

センダ管理デーモン
- RECEIVER

:

レシーバ管理デーモン
- LOGREADER

:

ログリーダー管理デーモン
- DEL

:

ログデータ削除デーモン
- ② デーモンプロセス ID
- ③ デーモンプロセス名
- ④ デーモンプロセス起動日時
- ⑤ デーモンプロセス状態
- AVAIL

:

起動中
- UNAVAIL

:

停止中
- :

コマンド実行ノードで起動対象外

スーパーストリーム動作ノード情報(didl1tref -d location)

```
+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①
+----- SYSTEM INFORMATION -----+
LSID      = 1                      ①
LSNAME    = LS1                    ②
LNODETYPE = AP                     ③
LNODENAME = ap1                    ④
+----- SUPERSTREAM LOCATION INFORMATION -----+
NAME_____ LNODENAME_____ POOLFILE  MAPID_____
 ①          ②              ③          ④
SPST_IM      oltp1          TAM          1
SPST_DB      ap1            ORACLE       ---
:
+===== END OF DISPLAY =====+
```

- ① コマンド実行時刻
- +----- SYSTEM INFORMATION -----+
- 共通のシステム情報表示
- ① 論理システム ID
- ② 論理システム名
- ③ 論理ノード種別
- ④ 論理ノード名
- +----- SUPERSTREAM LOCATION INFORMATION -----+
- スーパーストリーム動作ノード情報表示
- ① スーパーストリーム名
- ② 動作ノード名
- ③ ログデータ格納先
- TAM

:

TAM
- ORACLE

:

Oracle
- ④ MAPID(ログデータ格納先が TAM の場合のみ表示)

## 2.4.14 didltrefdb(ディレード定義リスト出力)

### 名前

didltrefdb - DB に管理している情報を出力する。

### 書式

```
didltrefdb [-F {sender|receiver|reader|dsam|location}] [-d {system|config}] [-v]
didltrefdb [-F {sender|receiver|reader|dsam}] -s SuperStreamName [-v]
didltrefdb -h
```

### 説明

ディレード転送全機能(センダ、レシーバ、ログリーダー、DSAM、ストリーム所在管理)の制御 DB 内の各定義情報を表示する。

### オプション

#### -F

表示対象とする機能を指定する。

<b>sender</b>	センダ情報
<b>receiver</b>	レシーバ情報
<b>reader</b>	ログリーダー情報
<b>dsam</b>	DSAM 情報(既定値)
<b>location</b>	ストリーム所在管理機能

#### -d

表示対象を指定する。

本オプションは、-F で dsam を指定した場合のみ指定可能である。

指定がない場合、システム情報、構成情報を全て表示する。

<b>system</b>	システム情報
<b>config</b>	構成情報

#### -s

指定したスーパーストリームの定義情報を表示する。

#### -v

詳細出力を行う。

#### -h

使用方法を表示する。

### 戻り値

0	正常終了
1	異常終了
2	警告終了(表示対象なし)

### 注意

- 定義生成コマンド(didltcreate)が実行されている必要がある。

## 2.4.15 didltslmmv(スーパーストリーム所在変更)

### 名前

didltslmmv - スーパーストリームの動作ノードを変更する

### 書式

```
didltslmmv -M spst -n LNodeName -s SuperStreamName
didltslmmv -M node -n LNodeName
didltslmmv -h
```

### 説明

**didltslmmv** は、ログデータ格納先が Oracle と定義されたスーパーストリームの動作ノードを変更する。**-M** オプションで変更モードを指定する。変更モードに **spst** を指定した場合は、**SuperStreamName** で指定したスーパーストリームの動作ノードを **LNodeName** で指定したノードに変更する。変更モードに **node** を指定した場合は、**LNodeName** で指定したノードが動作ノードであるスーパーストリーム全てを、他の正常なノードで動作するように変更する。動作ノードの変更は、他のノードの割り当て数が均等になるように自動的に決定される。

### オプション

#### **-M**

スーパーストリーム所在変更の変更モードを指定する。

**spst**                    指定したスーパーストリームの動作ノードを指定したノードに変更する。

**node**                   指定したノードが動作ノードであるスーパーストリーム全てを、他の正常なノードで動作するように変更する。

#### **-n LNodeName**

変更対象になるノードを **LNodeName** に指定する。**spst** モードおよび **node** モード使用時に必ず指定する。

#### **-s SuperStreamName**

変更対象になるスーパーストリームを **SuperStreamName** に指定する。**spst** モード使用時に必ず指定する。

#### **-h**

使用方法を表示する。

### 戻り値

**didltslmmv** は、以下のいずれかの終了ステータスを戻して終了する。

0                        正常終了

1                        異常終了

### 注意

**didltslmmv** は DB ノード かつ ストリーム所在管理マネージャデーモン(メイン)が起動しているノードでのみ実行することができる。

**spst** モード使用時に **LNodeName** に指定できるノードは、DELAYED 節 COMMON 項の DBLAYER で指定したノード属性と一致する論理ノードの中で稼働中のノードのみである。稼働中のノードとは、状態が **ACTIVE** のノードである。ノードの状態は **didltslmref** コマンドで参照できる。

### 関連

**didltslmref**

## 2.4.16 didltslmref(スーパーストリーム所在管理情報照会)

### 名前

didltslmref - スーパーストリームの所在管理情報を照会する。

### 書式

didltslmref [-v]

didltslmref -h

### 説明

**didltslmref** は、ログデータ格納先が Oracle と定義されたスーパーストリームの動作ノード一覧を表示する。DB ノードかつ ストリーム所在管理マネージャデーモンが起動しているノードで実行した場合は、スーパーストリームの動作ノード一覧に加えて、ストリーム所在管理エージェントデーモンが動作するノードの状態および割り当てられたスーパーストリーム数(メインマネージャ起動ノードの場合のみ)を一覧表示する。

### オプション

-v

スーパーストリーム動作ノード一覧情報およびノード別スーパーストリーム数の詳細情報を表示する。  
ストリーム所在管理マネージャデーモンが起動している DB ノードでのみ有効となる。

-h

使用方法を表示する。

### 戻り値

**didltslmref** は、以下のいずれかの終了ステータスを戻して終了する。

0                      正常終了

1                      異常終了

### 注意

- **didltslmref** はストリーム所在管理マネージャデーモンおよびエージェントデーモンが動作するノードでのみ実行することができる。スーパーストリームの動作ノード一覧表示は、ストリーム所在管理エージェントデーモンの動作するノードで実行した場合はメモリに保持している情報を表示し、ストリーム所在管理マネージャデーモンの動作するノードで実行した場合はデータベースに保持している情報を表示する。
- デーモン起動直後に、エージェント～マネージャ間接続前の実施や、データベースに保持している情報を取得する前の状態で照会を実施した場合、テーブルリビジョンが"---"で表示される。
- 動作ノード決定前の状態で照会を実施した場合、動作ノードが"---"で表示される。

### 関連

didltslmmv

実行例

スーパーストリーム所在管理情報照会 AP/OLTP ノード(didltslmref)

```
+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①
+----- SUPERSTREAM LOCATION INFORMATION -----+
TBLREV = 1                                     ①
②          ③
NAME_____ LNODENAME_____
SPST1      oltp1
SPST2      oltp2
SPST3      oltp1
      :
      :
+===== END OF DISPLAY =====+
```

+===== DELAYED TRANSFER DISPLAY =====+

① コマンド実行時刻

+----- SUPERSTREAM LOCATION INFORMATION -----+

スーパーストリームの動作ノード一覧表示

① テーブルリビジョン

② スーパーストリーム名

③ 動作ノード名

スーパーストリーム所在管理情報照会 DB ノード(didltslmref)

```
+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①
+----- SUPERSTREAM LOCATION INFORMATION -----+
      ①②
TBLREV = 1[1]
③          ④
NAME_____ LNODENAME_____
SPST1      oltp1
SPST2      oltp2
SPST3      oltp1
      :
      :
+----- NODE STATUS INFORMATION -----+
①          ②          ③
LNODENAME_____ STATUS__ SUPERSTREAMNUM
ap1          ACTIVE          0
*oltp1       ACTIVE          6
*oltp2       INACTIVE        6
*oltp3       TROUBLE         0
      :
      :
+===== END OF DISPLAY =====+
```

+===== DELAYED TRANSFER DISPLAY =====+

① コマンド実行時刻

+----- SUPERSTREAM LOCATION INFORMATION -----+

スーパーストリームの動作ノード一覧表示

① テーブルリビジョン(DB 上の値)

② テーブルリビジョン(ストリーム所在管理マネージャデーモンが保持している値)

③ スーパーストリーム名

④ 動作ノード名

+----- NODE STATUS INFORMATION -----+

ノード状態一覧表示

① ノード名(スーパーストリーム割り当て対象のノードにはノード名の先頭に“\*”が付加される)

② ノードの稼働状況

ACTIVE : 正常に稼働している

INACTIVE : 停止中

TROUBLE : 障害中

③割り当てられているスーパーストリーム数

## スーパーストリーム所在管理情報照会 DB ノード 詳細指定(didltslmref -v)

```

+===== DELAYED TRANSFER DISPLAY =====+ 2011/09/22 22:22:22 ①
+----- SUPERSTREAM LOCATION INFORMATION -----+
      ①②
TBLREV = 1[1]
      ③          ④      ⑤
NAME_____ GROUP LNODENAME_____
SPST1          1 oltp1
SPST2          1 oltp2
SPST3          2 oltp1
      :
      :
+----- NODE STATUS INFORMATION -----+
      ①          ②      ③
LNODENAME_____ STATUS__ SUPERSTREAMNUM_____
      TOTAL [ 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16]
ap1            ACTIVE      0   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
*oltp1         ACTIVE      6   1  1  0  0  0  2  2  0  0  0  0  0  0  0  0
*oltp2         INACTIVE     6   1  1  0  0  0  2  1  0  0  0  0  0  0  0  1
*oltp3         TROUBLE      0   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
      :
      :
+===== END OF DISPLAY =====+

```

+===== DELAYED TRANSFER DISPLAY =====+

### ① コマンド実行時刻

+----- SUPERSTREAM LOCATION INFORMATION -----+

## スーパーストリームの動作ノード一覧表示

① テーブルリビジョン(DB 上の値)

② テーブルリビジョン(ストリーム所在管理マネージャデーモンが保持している値)

③ スーパーストリーム名

④ スーパーストリームグループ番号

⑤ 動作ノード名

```

+----- NODE STATUS INFORMATION -----+

```

ノード状態一覧表示

① ノード名(スーパーストリーム割り当て対象のノードにはノード名の先頭に“\*”が付加される)

## ② ノードの稼働状況

ACTIVE : 正常に稼働している

INACTIVE : 停止中

TROUBLE : 障害中

### ③割り当てられているスーパーストリーム数

TOTAL : 全体数

「01-16」： スーパーストリームグループ別 割当スーパーストリーム数

## 2.4.17 didltterm(ディレード停止)

### 名前

didltterm - ディレード転送停止

### 書式

didltterm [-M force]

didltterm -h

### 説明

ディレード転送機能を終了し、ディレード転送機能(ディレード共通、DSAM)の制御デーモンを停止する。

ディレード停止は、センダ、レシーバ、ログリーダー起動中の場合エラーとなる。

### オプション

**-M**

制御デーモンの停止モードを指定する。

省略した場合は通常停止モードとなる。

**force**                    制御デーモンを強制停止後、ディレード転送を停止する。

**-h**

使用方法を表示する。

### 戻り値

0                            正常終了

1                            異常終了

### 注意

- 自ノードで動作中の Oracle スーパーストリームが存在する状態でディレード停止しても、スーパーストリームは他ノードへ移動しない。移動させたい場合は、スーパーストリーム所在変更コマンドを実行すること。

### 関連

didltinit

## 2.4.18 didltupd(ディレード定義変更)

### 名前

didltupd - 定義変更をおこなう。

### 書式

```
didltupd [-M force] [-D im] [-D db [-I InstanceGroupName]...] [-r]
didltupd -h
```

### 説明

ディレード転送機能の環境定義(DELAYED 節)に対する変更内容を、メモリキャッシュと Oracle の制御 DB へ反映する。

制御 DB への反映後、定義反映コマンド(didlitchg)を自動発行し、論理システム内の各ノードで、共有メモリへの反映も行う。

メモリキャッシュと Oracle の両方が動作するシステムでは、両方にアクセス可能なノードでコマンドを実行すると、両方のデータベースの制御テーブルを変更する。両方にアクセス可能なノードが存在しない場合、それぞれにアクセス可能なノードで1回ずつコマンドを実行する必要がある。

Oracle 障害中でメモリキャッシュの定義変更のみを実行したい場合や、障害復旧後に Oracle のみ定義変更をおこないたい場合などのために、-D オプションで定義変更をおこなうデータベースを指定することが可能である。また、-D db 指定時は、-I オプションでインスタンスグループ名まで指定することも可能である。-I が指定された場合、指定されたインスタンスグループ上のテーブルのみが変更される。ただし、プールファイル格納先やユーザデータ更新先を変更するような定義変更をおこなう場合、-D オプションを指定してはならない。

### オプション

#### -M

定義変更可否をチェックして不可と判定された場合のモードを指定する。

指定しない場合、チェックした結果が定義変更不可の場合は異常終了する。

**force**                      チェックした結果が定義変更不可の場合でも変更処理をおこなう。

#### -D

処理対象を指定する。

**im**                          メモリキャッシュ

**db**                          Oracle

省略した場合、コマンド実行ノードでアクセス可能なデータベースに対して定義変更処理をおこなう。

#### -I

インスタンスグループを指定する。(16 個まで指定可能、省略可)

-D db を指定した場合のみ指定可能。省略した場合、全てのインスタンスグループが変更対象となる。

指定する場合、デフォルトリソースグループセットが属するインスタンスグループは、必ず指定する必要がある。

#### -r

メンテナンス中の場合も、強制的に定義変更をおこなう。

#### -h

使用方法を表示する。

戻り値

0	正常終了
1	異常終了
2	警告終了(定義反映処理でエラーが発生した。)

注意

- コマンド実行ノード上で、コマンド配信デーモンが起動している必要がある。
- 処理対象にメモリキャッシュが含まれる場合、IMS 所在管理が起動している必要がある。
- 本コマンドは、定義生成コマンド(didlcreate)で生成された制御 DB に対して変更内容を反映するため、先に定義生成コマンドで制御 DB を生成しておく必要がある。
- センダ、レシーバ、ログリーダのいずれかが開始コマンド(didtsinit、didtrinit、didtlinit)投入済みの状態の場合、定義変更を実行することはできない。
- 本コマンドを実行すると、ディレード転送の各機能の動作変更コマンド(didltpoolmod, didtsmod, didtrmod, didtlmod)で変更されていたパラメータは、全て環境定義(DELAYED 節)に指定された定義値に戻る。ただし、スーパーストリーム、ユニットの無効化状態、センダユニット/レシーバユニットの相手論理システム名については、定義変更前の状態を引き継ぐ。
- ディレード転送全体の定義(DELAYED 節-COMMON 項)、通信制御機能の定義(DELAYED 節-PATHCTRL 項)、およびストリーム所在管理機能の定義(DELAYED 節-LOCATIONCTRL 項)のみを変更した場合、本コマンドを実行しても各ノードの定義は変更されない。その場合は、定義反映コマンド(didlchg)を全ノード反映のオプション(-A)で実行する必要がある。
- 論理ノード未起動などにより定義反映コマンド(didlchg)のコマンド配信が失敗した場合、失敗したノードでディレード起動する際には-M create オプション(環境定義(DELAYED 節)オブジェクトを使用して設定情報を構築)で起動する必要がある。
- 定義パターンと定義変更対象の対応は以下の通りである。

[定義パターン]

No.	定義内容	POOLFILE	USERDATA
1	削除デーモン用定義(IM)	—	—
2	削除デーモン用定義(DB)	—	—
3	スーパーストリーム定義	IM	IM
4	スーパーストリーム定義	IM	DB
5	スーパーストリーム定義	IM	DB(別)
6	スーパーストリーム定義	DB	DB
7	スーパーストリーム定義	DB	DB(別)

※ DB(別)とは、デフォルト RGSET を含まないインスタンスグループの DB を指す。

[定義変更対象]



## 2.4.19 didtctrl (ログリーダーデーモン起動停止)

### 名前

didtctrl - ログリーダー機能の実行デーモンを起動/停止する

### 書式

```
didtctrl
didtctrl -h
```

### 説明

**didtctrl** は、ログリーダー実行デーモンの起動と停止を行う。

コマンドを実行すると、コマンド実行ノードで動作すべきスーパーストリームを確認し、必要なログリーダー実行デーモンの起動処理および、不要となったログリーダー実行デーモンの停止処理をおこなう。

### オプション

-h

使用方法を表示する。

### 戻り値

**didtctrl** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtlinit, didtlterm

## 2. 4. 20     **didtldivend(ログリーダー強制ディビジョン終了)**

### 名前

didtldivend - データ処理中のディビジョンを強制的に終了する

### 書式

```
didtldivend -s SuperStreamName [-u UnitName]
didtldivend -h
```

### 説明

**didtldivend** は、指定されたログデータを処理するログリーダーのスーパーストリーム、もしくは指定されたユニットのデータ処理中ディビジョンを強制的に終了させ、次のディビジョンのデータ処理を可能にする。本コマンドにより強制的にディビジョンを終了させると、指定対象における現在のディビジョンのログデータは未処理のログデータも含め、すべて破棄される。

本コマンドの実行には、ディビジョン終了対象のスーパーストリーム状態が初期状態 (INIT)、データ処理停止状態 (INACT)、またはディビジョン終了状態 (DIVEND) である必要がある。但し、すべてのスーパーストリーム状態が初期状態 (INIT)、またはディビジョン終了状態 (DIVEND) の場合は、既にディビジョンが終了している状態 (未開始の状態) と判断し、本コマンドを実行することはできない。

また、対象となるスーパーストリームには、終了させるディビジョンの次のディビジョンが存在している必要がある。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリーム、およびユニットのみが対象となる。

### オプション

**-s**  
指定したスーパーストリームを対象とする。

**-u**  
指定したユニットを対象とする。

**-h**  
使用方法を表示する。

### 戻り値

**didtldivend** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit,   didtlinit,   didtlstop,   didtrdivend

## 2.4.21 didtlinit(ログリーダー開始)

### 名前

didtlinit - ログリーダー機能を開始する

### 書式

```
didtlinit
didtlinit -h
```

### 説明

**didtlinit** は、ログリーダー機能が動作するために必要な環境を生成し、ログリーダー管理デーモン、およびログリーダー実行デーモンの起動を行う。

### オプション

**-h**  
使用方法を表示する。

### 戻り値

**didtlinit** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtlterm

## 2.4.22 didtlmod(ログリーダ動作変更)

### 名前

didtlmod - ログリーダの各種動作環境を変更する

### 書式

```
didtlmod -T dataperlot=value {-A|-s SuperStreamName [-u UnitName]}
didtlmod -T divid=value -T datano=value -T userdatano=value -s SuperStreamName [-u UnitName]
didtlmod -h
```

### 説明

**didtlmod** は、ログリーダの各種動作環境を変更する。

ディビジョン ID(**divid**)、処理開始通番(**datano**)、および処理開始利用者通番(**userdatano**)を変更する場合は、変更対象のスーパーストリーム状態がデータ処理停止状態(INIT, INACT, DIVEND)である必要がある。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリーム、およびユニットのみが対象となる。

### オプション

#### -T

動作変更内容を指定する。指定するパラメータにおいて、“=”の前後に空白を指定してはならない。

<b>dataperlot</b>	コミット係数(1ロット内で処理するログデータ数)
<b>divid</b>	ディビジョン ID(16進数、1以上)
<b>datano</b>	処理開始通番(0以上)
<b>userdatano</b>	処理開始利用者通番(0以上)

#### -A

全スーパーストリームに関する動作環境を変更する。

#### -s

指定したスーパーストリームに関する動作環境を対象とする。

#### -u

指定したユニットに関する動作環境を変更する。

#### -h

使用方法を表示する。

### 戻り値

**didtlmod** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtlinit

## 2. 4. 23 didtlref (ログリーダ状況照会)

### 名前

didtlref - ログリーダのデータ処理状況を表示する。

### 書式

```
didtlref -d system
didtlref -p
didtlref -A [-l] [-v] [-b]
didtlref -s SuperStreamName [-l] [-v]
didtlref -s SuperStreamName -u UnitName [-v]
didtlref -h
```

### 説明

**didtlref** は、ログリーダのデータ処理状況を表示する。指定したオプションによって出力フォーマットは異なる。

**-d system** 指定では、コマンド実行ノードの論理ノード種別で動作するユニットの実行デーモン情報が対象となる。それ以外のオプションでは、コマンド実行ノードで動作中のスーパーストリーム、およびユニットのみが対象となる。

### オプション

**-d**

表示する情報を指定する。

**system** システム情報を表示する。

**-p**

性能情報を表示する。

**-A**

全スーパーストリームを対象とする。

**-s**

指定したスーパーストリームを対象とする。

**-u**

指定したユニットを対象とする。

**-l**

リスト形式で表示する。

**-v**

詳細情報を表示する。

**-b**

無効化状態のスーパーストリームも表示する。

**-h**

使用方法を表示する。

### 戻り値

**didtlref** は、以下のいずれかの終了ステータスを戻して終了する。

0 正常終了

- |   |      |
|---|------|
| 1 | 異常終了 |
| 2 | 警告終了 |

#### 注意

- ログリーダー開始直後の実行デーモン起動前や、メンテナンスコマンドによる DB 更新結果を共有メモリに反映する前の状態で、ユニット情報を照会した場合、通常表示、リスト表示におけるほとんどの項目が“---”で表示される。

#### 関連

didltinit, didtlinit

実行例

システム情報表示( didtlref -d system )

```
+===== DELAYED TRANSFER (LOGREADER) DISPLAY =====+ 2012/03/26 00:00:00 ①
LSNAME = LS11 ②          LNODENAME = oltp1 ③
+----- SYSTEM INFORMATION -----+
STATUS = ACT ①          STARTTIME = 2012/03/26 00:00:00 ②
+----- COMMAND INFORMATION -----+
CMDREQNUM = 0 ①
+----- TIMER INFORMATION -----+
DMNRESTINVL = 5 ①
DMNDOWNCNT = 5 ②
DMNDOWNTIME = 60 ③
RCVRYCHKINVL = 60 ④
+----- EXEC DAEMON INFORMATION -----+
SPSTNAME = SPST1 ①
UNITNAME = UNIT1 ②
PRCTARGET = YES ③          STATUS = ACT ④
SPSTBLOCK = ACT ⑤          UNITBLOCK = ACT ⑥
TRBLSTATUS = NONE ⑦        RESTARTCNT = 0 ⑧
PROCID = 11489 ⑨          STARTTIME = 2012/03/26 00:00:00 ⑩
+----- EXEC DAEMON INFORMATION -----+
:
+===== END OF DISPLAY =====+
```

+===== DELAYED TRANSFER (LOGREADER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SYSTEM INFORMATION -----+

管理デーモン情報

- ① 管理デーモンステータス
  - ACT : 起動中
  - TERM : 停止中
  - ACTING : 起動処理中
  - TERMINATING : 停止処理中
- ② 管理デーモン起動時刻

+----- COMMAND INFORMATION -----+

コマンド処理情報

- ① 処理中のコマンド要求数

+----- TIMER INFORMATION -----+

タイマ情報

- ① デーモン再起動間隔
- ② デーモン再起動リトライ回数
- ③ デーモン再起動リトライオーバー判定時間
- ④ 障害時復旧監視間隔

+----- EXEC DAEMON INFORMATION -----+

実行デーモン情報(実行デーモン数分繰り返す)

- ① スーパーストリーム名
- ② ユニット名
- ③ 起動対象フラグ
  - YES : 起動対象
  - NO : 起動対象外
- ④ 実行デーモンステータス
  - ACT : 起動中
  - TERM : 停止中

- TERMINATING : 停止処理中
- ⑤ スーパーストリーム無効化状態
- ACT : 有効
- BLOCK : 無効
- ⑥ ユニット無効化状態
- ACT : 有効
- BLOCK : 無効
- ⑦ 障害ステータス
- NONE : 通常状態
- DBTROUBLE : DB 障害
- SWITCH : マスタ切替検出
- DUPLICATE : 二重起動検出
- NORECORD : 制御レコードなし
- RETRYOVER : リトライオーバー
- OTHER : その他異常
- ⑧ 再起動回数
- ⑨ プロセス ID
- ⑩ 実行デーモン起動時刻

性能情報表示( didtlref -p )

+===== DELAYED TRANSFER (LOGREADER) DISPLAY =====+ 2012/03/26 00:00:00 ①											
LSNAME = LS11 ②		LNODENAME = oltp1 ③									
+----- PERFORMANCE INFORMATION -----+											
①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪	⑫
SPSTNAME	UNITNAME	PERF	APPROCMIN	APPROCMAX	APPROCAVG	COMMITMIN	COMMITMAX	COMMITAVG	CPUUSRMIN	CPUUSRMAX	CPUUSRAVG
⑬	⑭	⑮									
CPUSYSMIN CPUSYSMAX CPUSYSAVG											
SPST1	UNIT1	YES	0	0	0	0	0	0	0	0	0
	0	0	0								
SPST2	UNIT2	YES	0	0	0	0	0	0	0	0	0
	0	0	0								
SPST3	UNIT3	NO	---	---	---	---	---	---	---	---	---
---	---	---	---								
SPST3	UNIT4	NO	---	---	---	---	---	---	---	---	---
---	---	---	---								
+===== END OF DISPLAY =====+											

+===== DELAYED TRANSFER (LOGREADER) DISPLAY =====+

① コマンド実行時刻

② 論理システム名

③ 論理ノード名

+----- PERFORMANCE INFORMATION -----+

性能情報(ユニット数分表示)

① スーパーストリーム名

② ユニット名

③ 性能情報収集有無

YES : 収集する

NO : 収集しない

④ AP 処理時間(最小)

⑤ AP 処理時間(最大)

⑥ AP 処理時間(平均)

⑦ コミット処理時間(最小)

⑧ コミット処理時間(最大)

⑨ コミット処理時間(平均)

⑩ CPU 時間[ユーザ](最小)

⑪ CPU 時間[ユーザ](最大)

⑫ CPU 時間[ユーザ](平均)

⑬ CPU 時間[システム](最小)

⑭ CPU 時間[システム](最大)

⑮ CPU 時間[システム](平均)

(④～⑮の単位はマイクロ秒)

リスト表示( didtlref -l { -A [-v] [-b] | -s スーパーストリーム名 [-v] } )

+===== DELAYED TRANSFER (LOGREADER) DISPLAY =====+ 2012/03/26 00:00:00 ①									
LSNAME = LS11 ②		LNODENAME = oltp1 ③							
+----- SUPERSTREAM INFORMATION -----+									
①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩
SPSTNAME_____	UNITNAME_____	DMNSTATUS	STATUS	CURDIVID__	EXECCNT_____	USEREXECCNT__	POOLCNT_____	USERPOOLCNT__	DATALIM
SPST1	UNIT1	ACT	ACT	1	100	30	1000	500	NO
SPST2	UNIT2	ACT	INACT	1	200	90	20	10	NO
SPST3	UNIT3	BLOCK	---	---	---	---	---	---	---
SPST3	UNIT4	BLOCK	---	---	---	---	---	---	---
-----									
⑪					⑫	⑬	⑭	⑮	
TOTALCNT = 4					300	120	1020	510	
+----- SUPERSTREAM INFORMATION (BLOCK) -----+									
①	②								
SPSTNAME_____	UNITNAME_____	DMNSTATUS	STATUS	CURDIVID__	EXECCNT_____	USEREXECCNT__	POOLCNT_____	USERPOOLCNT__	DATALIM
SPST8	UNIT8	---	---	---	---	---	---	---	---
+===== END OF DISPLAY =====+									

+===== DELAYED TRANSFER (LOGREADER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報(無効化解除状態のスーパーストリーム配下のユニット数分表示)

- ① スーパーストリーム名
- ② ユニット名
- ③ 実行デーモン状態
  - ACT : 起動中
  - TERM : 停止中
  - BLOCK : ユニット無効化状態
  - DBTROUBLE : DB 障害
  - SWITCH : マスタ切替検出
  - DUPLICATE : 二重起動検出
  - NORECORD : 制御レコードなし
  - RETRYOVER : リトライオーバー
  - OTHER : その他異常
- ④ ユニットステータス
  - INIT : 初期状態
  - ACT : データ処理中
  - INACT : データ処理停止中
  - DIVEND : ディビジョン終了状態
- ⑤ カレントディビジョン ID
- ⑥ 処理済件数
- ⑦ 処理済ユーザ件数
- ⑧ 滞留データ件数(カレントディビジョン)(-v 指定時のみ出力)
- ⑨ 滞留ユーザデータ件数(カレントディビジョン)(-v 指定時のみ出力)
- ⑩ 処理完了状態
  - YES : 未処理データなし
  - NO : 未処理データあり
- ⑪ 合計ユニット数
- ⑫ 合計処理済み件数
- ⑬ 合計処理済みユーザ件数
- ⑭ 合計滞留データ件数(-v 指定時のみ出力)
- ⑮ 合計滞留ユーザデータ件数(-v 指定時のみ出力)

+----- SUPERSTREAM INFORMATION (BLOCK) -----+

無効化状態のスーパーSTREAM情報(-b 指定時のみ出力)

① スーパーSTREAM名

② ユニット名

通常表示( didtlref { -A [-v] [-b] | -s スーパーストリーム名 [-u ユニット名] [-v] } )

```
+===== DELAYED TRANSFER(LOGREADER) DISPLAY =====+ 2012/03/26 00:00:00 ①
LSNAME = LS11 ②          LNODENAME = oltp1 ③
+----- SUPERSTREAM INFORMATION -----+
SPSTNAME      = SPST1          UNITNAME      = UNIT1
CTLTYPE       = ORACLE         MAPID_CTL     = ---          RGSETNAME_CTL  = RGSET01
POOLTYPE      = ORACLE         MAPID_POOL    = ---          RGSETNAME_POOL = #DEFAULT
DMNSTATUS     = ACT            TRBLSTATUS   = NONE          UNITBLOCK     = ACT
STATUS        = INIT
CURDIVID      = 1              NEXTDIVID   = 0
EXEC CNT      = 0              USEREXEC CNT = 0
EXEC DATANO   = 0              USEREXEC DATANO = 0
POOL CNT      = 0              USERPOOL CNT = 0
TOTALPOOL CNT = 0              TOTALUSERPOOL CNT = 0          DATALIM       = YES
FAILEDNO      = 0              FAILREASON   = NONE          SIGNAL        = NONE
DATAEXEC STATUS = INIT          EXEC REQEST  = NORMAL
DATA PER LOT (SG) = 5 (5 )      CHKINVL    = 10000        DLYTIME       = 0
PERF          = YES            ELPOVER     = MSG          ELPTIME       = 0
ELPRESET      = 0              CPUTIME    = 0          RETRY         = 5
DATABUFSIZE   = 1024
PRCINIT       = UpdPrcInitExit
PRCTERM       = UpdPrcTermExit
TRNINIT       = UpdTrnInitExit
TRNTERM       = UpdTrnTermExit
MAINAP        = LogUpdOraExit
+----- SUPERSTREAM INFORMATION -----+
:
+----- SUPERSTREAM INFORMATION (BLOCK) -----+
SPSTNAME = SPST8 ①          UNITNAME = UNIT8 ②
+===== END OF DISPLAY =====+
```

+===== DELAYED TRANSFER(LOGREADER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報(無効化解除状態のスーパーストリーム配下のユニット数分表示)

SPSTNAME	: スーパーストリーム名
UNITNAME	: ユニット名
CTLTYPE	: 制御 DB の DB 種別(TAM/ORACLE)
MAPID_CTL	: 制御 DB の MAPID(ORACLE の場合"---")
RGSETNAME_CTL	: 制御 DB のリソースグループセット名(TAM の場合"---")
POOLTYPE	: プールファイルの DB 種別(TAM/ORACLE)
MAPID_POOL	: プールファイルの MAPID(ORACLE の場合"---")
RGSETNAME_POOL	: プールファイルのリソースグループセット名(TAM の場合"---")
DMNSTATUS	: 実行デーモン状態
	ACT : 起動中
	TERM : 停止中
	TERMINATING : 停止処理中
TRBLSTATUS	: 障害ステータス
	NONE : 通常状態
	DBTROUBLE : DB 障害
	SWITCH : マスタ切替検出
	DUPLICATE : 二重起動検出
	NORECORD : 制御レコードなし
	RETRYOVER : リトライオーバー
	OTHER : その他異常
UNITBLOCK	: ユニット無効化状態
	ACT : 有効
	BLOCK : 無効
STATUS	: ユニットステータス

	INIT	: 初期状態
	ACT	: データ処理中
	INACT	: データ処理停止中
	DIVEND	: デイビジョン終了状態
CURDIVID	:	カレントデイビジョン ID
NEXTDIVID	:	次デイビジョン ID
EXECNT	:	処理済み件数
USEREXECNT	:	処理済みユーザ件数
EXECDATANO	:	処理済み通番
USEREXECDATANO	:	処理済みユーザ通番
POOLCNT	:	滞留件数(カレントデイビジョン) (-v 指定時のみ出力)
USERPOOLCNT	:	滞留ユーザ件数(カレントデイビジョン) (-v 指定時のみ出力)
TOTALPOOLCNT	:	滞留件数(カレントデイビジョン以降の全デイビジョン合計) (-v 指定時のみ出力)
TOTALUSERPOOLCNT	:	滞留ユーザ件数(カレントデイビジョン以降の全デイビジョン合計) (-v 指定時のみ出力)
DATALIM	:	処理完了状態 (-v 指定時のみ出力)
	YES	: 未処理データなし
	NO	: 未処理データあり
FAILEDNO	:	障害通番 (-v 指定時のみ出力)
FAILREASON	:	障害理由 (-v 指定時のみ出力)
	NONE	: なし
	DIOSA	: DIOSA 処理
	AP	: AP 要求
	SIGNAL	: シグナル
	OTHER	: その他
SIGNAL	:	発生シグナル情報 (-v 指定時のみ出力)
	NONE	: なし
	SIGxxx	: SIGABRT/SIGFPE/SIGBUS/SIGSEGV/SIGTERM/SIGXCPU
	OTHER	: その他
DATAEXECSTATUS	:	データ処理ステータス (-v 指定時のみ出力)
	INIT	: 初期状態
	TRNSTART	: トランザクション初期化
	LOGGETBEFORE	: ログデータ取得前
	LOGGET	: ログデータ取得処理中
	LOGGETAFTER	: ログデータ取得処理後
	APBEFORE	: メイン AP 呼び出し前
	AP	: メイン AP 内
	APAFTER	: メイン AP 呼び出し後
	COMMITBEFORE	: コミット前
	COMMIT	: コミット中
	COMMITAFTER	: コミット後
	TRNEND	: トランザクション終了
EXECREQUEST	:	データ処理要求 (-v 指定時のみ出力)
	NORMAL	: 通常状態
	ROLLBACKRETRY	: ロールバックリトライ
	ABORT	: アボート処理要求
DATAPERLOT	:	ロット化係数(カッコ内は SG 値)
CHKINVL	:	データチェック間隔(ミリ秒)
DLYTIME	:	処理遅延時間
PERF	:	性能情報収集
	YES	: 収集する
	NO	: 収集しない
ELPOVER	:	経過時間監視時間オーバー時動作
	STOP	: 処理停止

	MSG	: メッセージ出力
ELPTIME	:	経過時間監視時間
ELPRESET	:	経過時間監視リセット要求回数
CPUTIME	:	CPU 時間監視時間
RETRY	:	ロールバックリトライ回数
DATABUFSIZE	:	データバッファサイズ
PRCINIT	:	プロセス初期化利用者出口
PRCTERM	:	プロセス終了利用者出口
TRNINIT	:	トランザクション初期化利用者出口
TRNTERM	:	トランザクション終了利用者出口
MAINAP	:	メイン処理

+----- SUPERSTREAM INFORMATION (BLOCK) -----+

無効化状態のスーパーストリーム情報 (-b 指定時および指定されたスーパーストリームが無効化状態の場合のみ出力)

- ① スーパーストリーム名
- ② ユニット名

## 2.4.24 didtlstart(ログリーダーデータ処理開始)

### 名前

didtlstart - ログリーダーのデータ処理を開始する。

### 書式

```
didtlstart -A
didtlstart -s SuperStreamName [-u UnitName] [-M skip]
didtlstart -h
```

### 説明

**didtlstart** は、指定されたユニット、もしくはスーパーストリームのデータ処理を開始する。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリーム、およびユニットのみが対象となる。

### オプション

**-A**

全スーパーストリームのデータ処理を開始する。

**-s**

指定したスーパーストリームのデータ処理を開始する。

**-u**

指定したユニットのデータ処理を開始する。

**-M**

開始モードを指定する。

省略した場合、中断点からデータ処理を再開する。

<b>skip</b>	中断点が障害データの場合、障害データの次の未処理データから処理を再開する。
	中断点が未処理データの場合、中断点からデータ処理を再開する。

**-h**

使用方法を表示する。

### 戻り値

**didtlstart** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtlinit, didtlstop

## 2.4.25 didtlstop(ログリーダーデータ処理停止)

### 名前

didtlstop - ログリーダーのデータ処理を停止する。

### 書式

```
didtlstop -A
didtlstop -s SuperStreamName [-u UnitName]
didtlstop -h
```

### 説明

**didtlstop** は、指定されたユニット、もしくはスーパーストリームのデータ処理を停止する。  
オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリーム、およびユニットのみが対象となる。

### オプション

**-A**

全スーパーストリームのデータ処理を停止する。

**-s**

指定したスーパーストリームのデータ処理を停止する。

**-u**

指定したユニットのデータ処理を停止する。

**-h**

使用方法を表示する。

### 戻り値

**didtlstop** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtlinit, didtlstart

## 2.4.26 didtlterm(ログリーダー終了)

### 名前

didtlterm - ログリーダー機能を終了する。

### 書式

didtlterm [-M {stop|force}]

didtlterm -h

### 説明

**didtlterm** は、ログリーダー管理デーモン、およびログリーダー実行デーモンを停止し、ログリーダーの動作環境を解放して、ログリーダー機能を終了する。

### オプション

#### -M

ログリーダー実行デーモンを停止する際の停止モードを指定する。

省略した場合、スーパーストリームがデータ処理停止状態であることを確認して停止するため、データ処理中のスーパーストリームが存在する場合は、停止することができない。

<b>stop</b>	スーパーストリームがデータ処理停止状態でない場合は、データ処理停止状態にしてから停止する。
-------------	---

<b>force</b>	データ処理状態に関わらず強制的に停止する。
--------------	-----------------------

#### -h

使用方法を表示する。

### 戻り値

**didtlterm** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didltterm, didtlinit, didtlstop

## 2.4.27 didtctrl (レシーバデーモン起動停止)

### 名前

didtctrl - レシーバ機能の実行デーモンを起動/停止する

### 書式

```
didtctrl
didtctrl -h
```

### 説明

**didtctrl** は、レシーバ実行デーモンの起動/停止を行う。

コマンドを実行すると、コマンド実行ノードで動作すべきスーパーストリームを確認し、必要なレシーバ実行デーモンの起動処理および、不要となったレシーバ実行デーモンの停止処理をおこなう。

### オプション

-h

使用方法を表示する。

### 戻り値

**didtctrl** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtrinit, didtrterm

## 2. 4. 28      **didtrdivend**(レシーバ強制ディビジョン終了)

### 名前

**didtrdivend** - 転送中のディビジョンを強制的に終了する

### 書式

**didtrdivend** -s SuperStreamName

**didtrdivend** -h

### 説明

**didtrdivend** は、指定されたスーパーストリームの転送中ディビジョンを強制的に終了させ、次のディビジョンの転送を可能にする。本コマンドにより強制的にディビジョンを終了させると、現在のディビジョンの受信済みログデータは、すべて破棄される。

本コマンドの実行には、終了対象のスーパーストリームの状態がデータ転送停止状態(INACT)である必要がある。初期状態(INIT)、またはディビジョン終了状態(DIVEND)の場合は、既にディビジョンは終了している状態(未開始の状態)と判断し、本コマンドを実行することはできない。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリームのみが対象となる。

### オプション

-s

指定したスーパーストリームのディビジョンを強制的に終了する。

-h

使用方法を表示する。

### 戻り値

**didtrdivend** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- 転送先のセンダも合わせてディビジョン終了コマンドを投入する必要がある。

### 関連

**didltinit**,   **didtrinit**,   **didtrstop**,   **didtsdivend**,   **didltdivchg**

## 2.4.29 didtrinit(レシーバ開始)

### 名前

didtrinit - レシーバ機能を開始する

### 書式

```
didtrinit
didtrinit -h
```

### 説明

**didtrinit** は、レシーバ機能を開始するために必要な環境を生成し、レシーバ管理デーモン、レシーバ実行デーモンを起動する。

### オプション

**-h**  
使用方法を表示する。

### 戻り値

**didtrinit** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtrterm

## 2. 4. 30 didtrmod(レシーバ動作変更)

### 名前

didtrmod - レシーバの各種動作環境を変更する

### 書式

```
didtrmod -T msgpercmt=value {-A|-s SuperStreamName}
didtrmod -T dls=LsName -s SuperStreamName
didtrmod -T divid=value -T datano=value -T userdatano=value -s SuperStreamName
didtrmod -T dctlmsgseqno=value -s SuperStreamName
didtrmod -h
```

### 説明

**didtrmod** は、レシーバの各種動作環境を変更する。

相手論理システム(**dls**)、およびディビジョン ID(**divid**)、受信開始通番(**datano**)、受信開始利用者通番(**userdatano**)を変更する場合は、変更対象スーパーストリームの状態がデータ転送停止状態(INIT, INACT, DIVEND)である必要がある。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリームのみが対象となる。

### オプション

#### -T

動作変更内容を指定する。

<b>msgpercmt</b>	1 コミットあたりの電文受信数 (0~100)
<b>dls</b>	相手論理システム
<b>divid</b>	ディビジョン ID (1 以上)
<b>datano</b>	受信開始通番 (0 以上)
<b>userdatano</b>	受信開始利用者通番 (0 以上)
<b>dctlmsgseqno</b>	相手制御電文通番 (0 以上)

#### -A

全スーパーストリームに関する動作環境を変更する。

#### -s

指定したスーパーストリームに関する動作環境を変更する。

#### -h

使用方法を表示する。

### 戻り値

**didtrmod** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- **-T** オプションで指定するパラメータは、“=”の両辺に空白を指定してはならない。

### 関連

didltinit, didtrinit

## 2.4.31 didtrref(レシーバ状況照会)

### 名前

didtrref - レシーバのデータ転送状況を表示する。

### 書式

```
didtrref -d system
didtrref -A [-l] [-b]
didtrref -s SuperStreamName [-l]
didtrref -h
```

### 説明

**didtrref** は、レシーバのデータ転送状況を表示する。指定したオプションによって出力フォーマットは異なる。

**-d system** 指定では、コマンド実行ノードの論理ノード種別で動作するユニットの実行デーモン情報が対象となる。それ以外のオプションでは、コマンド実行ノードで動作中のスーパーストリーム、およびユニットのみが対象となる。

### オプション

**-d**

表示する情報を指定する。

**system** システム情報を表示する。

**-A**

全スーパーストリームの転送状況を表示する。

**-s**

指定したスーパーストリームの転送状況を表示する。

**-l**

リスト形式で表示する。

**-b**

無効化状態のスーパーストリームも表示する。

**-h**

使用方法を表示する。

### 戻り値

**didtrref** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- レシーバ開始直後の実行デーモン起動前や、メンテナンスコマンドによる DB 更新結果を共有メモリに反映する前の状態で、スーパーストリーム情報を照会した場合、通常表示、リスト表示におけるほとんどの項目が“---”で表示される。

関連

didltinit, didtrinit

実行例

システム情報表示( didtrref -d system )

```
+===== DELAYED TRANSFER(RECEIVER) DISPLAY =====+ 2012/03/26 00:00:00 ①
LSNAME = LS11 ②          LNODENAME = oltp1 ③
+----- SYSTEM INFORMATION -----+
STATUS = ACT ①          STARTTIME = 2012/03/26 00:00:00 ②
+----- COMMAND INFORMATION -----+
CMDREQNUM = 0 ①
+----- TIMER INFORMATION -----+
DMNRESTINVL = 5 ①
DMNDOWNCNT = 5 ②
DMNDOWNTIME = 60 ③
RCVRYCHKINVL = 60 ④
+----- EXEC DAEMON INFORMATION -----+
SPSTNAME = SPST1 ①
UNITNAME = UNIT1 ②
PRCTARGET = YES ③          STATUS = ACT ④
SPSTBLOCK = ACT ⑤          UNITBLOCK = ACT ⑥
TRBLSTATUS = NONE ⑦        RESTARTCNT = 0 ⑧
PROCID = 11489 ⑨          STARTTIME = 2012/03/26 00:00:00 ⑩
+----- EXEC DAEMON INFORMATION -----+
:
+===== END OF DISPLAY =====+
```

+===== DELAYED TRANSFER(RECEIVER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SYSTEM INFORMATION -----+

管理デーモン情報

- ① 管理デーモンステータス
  - ACT : 起動中
  - TERM : 停止中
  - ACTING : 起動処理中
  - TERMINATING : 停止処理中
- ② 管理デーモン起動時刻

+----- COMMAND INFORMATION -----+

コマンド処理情報

- ① 処理中のコマンド要求数

+----- TIMER INFORMATION -----+

タイマ情報

- ① デーモン再起動間隔
- ② デーモン再起動リトライ回数
- ③ デーモン再起動リトライオーバー判定時間
- ④ 障害時復旧監視間隔

+----- EXEC DAEMON INFORMATION -----+

実行デーモン情報(実行デーモン数分繰り返す)

- ① スーパーストリーム名
- ② ユニット名
- ③ 起動対象フラグ
  - YES : 起動対象
  - NO : 起動対象外
- ④ 実行デーモンステータス
  - ACT : 起動中
  - TERM : 停止中

- TERMINATING : 停止処理中
- ⑤ スーパーストリーム無効化状態
- ACT : 有効
- BLOCK : 無効
- ⑥ ユニット無効化状態
- ACT : 有効
- BLOCK : 無効
- ⑦ 障害ステータス
- NONE : 通常状態
- DBTROUBLE : DB 障害
- SWITCH : マスタ切替検出
- DUPLICATE : 二重起動検出
- NORECORD : 制御レコードなし
- RETRYOVER : リトライオーバー
- OTHER : その他異常
- ⑧ 再起動回数
- ⑨ プロセス ID
- ⑩ 実行デーモン起動時刻

リスト表示( didtrref -l [-A [-b] | -s スーパーストリーム名 ] )

+===== DELAYED TRANSFER(RECEIVER) DISPLAY =====+ 2012/03/26 00:00:00 ①						
LSNAME = LS11 ②		LNODENAME = oltp1 ③				
+----- SUPERSTREAM INFORMATION -----+						
①	②	③	④	⑤	⑥	⑦
SPSTNAME_____	UNITNAME_____	DMNSTATUS	STATUS____	CURDIVID__	RECVDATACNT__	DRTYREQCNT__
SPST1	UNIT1	ACT	ACT	1	100	0
SPST2	UNIT2	ACT	INACT	1	200	1
-----						
⑧					⑨	⑩
TOTALCNT = 2					300	1
+----- SUPERSTREAM INFORMATION (BLOCK) -----+						
SPSTNAME_____	UNITNAME_____	DMNSTATUS	STATUS____	CURDIVID__	RECVDATACNT__	DRTYREQCNT__
①	②					
SPST8	UNIT8	---	---	---	---	---
+===== END OF DISPLAY =====+						

+===== DELAYED TRANSFER(RECEIVER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報(無効化解除状態のスーパーストリーム配下のユニット数分表示)

- ① スーパーストリーム名
- ② ユニット名
- ③ 実行デーモン状態

- ACT : 起動中
- TERM : 停止中
- BLOCK : ユニット無効化状態
- DBTROUBLE : DB 障害
- SWITCH : マスタ切替検出
- DUPLICATE : 二重起動検出
- NORECORD : 制御レコードなし
- RETRYOVER : リトライオーバー
- OTHER : その他異常

- ④ ユニットステータス

- INIT : 初期状態
- ACTING : 活性中
- ACT : 活性状態
- INACTING : 非活性中
- INACT : 非活性状態
- STOP : 相手側停止状態
- DIVEND : ディビジョン終了状態

- ⑤ カレントディビジョン ID

- ⑥ 受信済件数

- ⑦ 再送要求回数

+----- SUPERSTREAM INFORMATION (BLOCK) -----+

無効化状態のスーパーストリーム情報(-b 指定時のみ出力)

- ① スーパーストリーム名
- ② ユニット名

通常表示( didtrref [-A [-b] | -s スーパーストリーム名 ] )

+===== DELAYED TRANSFER(RECEIVER) DISPLAY =====+ 2012/03/26 00:00:00 ①									
LSNAME = LS11 ②		LNODENAME = oltp1 ③							
+----- SUPERSTREAM INFORMATION -----+									
SPSTNAME	=	SPST1	UNITNAME	=	UNIT1				
POOLTYPE	=	TAM	MAPID	=	1	RGSETNAME	=	---	
DMNSTATUS	=	ACT	TRBLSTATUS	=	NONE	UNITBLOCK	=	ACT	
STATUS	=	INIT	DLSNAME	=	LS11				
CURDIVID	=	1	DIVID_DSAM	=	1				
NEXTDIVID	=	0	NEXTDIVID_DSAM	=	0				
CRTYCNT	=	0	DRTYREQCNT	=	0				
RECVDATACNT	=	0	RECVDATANO	=	0	USERRECVDATANO	=	0	
RECVFIXCNT	=	0	RECVFIXNO	=	0	USERRECVFIXNO	=	0	
MSGSIZE	=	16384	MSGPERCMT(SG)	=	1	(1	)		
CRTYMAX	=	5	CRTYINVL	=	60				
+----- SUPERSTREAM INFORMATION -----+									
:									
+----- SUPERSTREAM INFORMATION (BLOCK) -----+									
SPSTNAME = SPST8 ①		UNITNAME = UNIT8 ②							
+===== END OF DISPLAY =====+									

+===== DELAYED TRANSFER(RECEIVER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報(無効化解除状態のスーパーストリーム配下のユニット数分表示)

SPSTNAME	:	スーパーストリーム名
UNITNAME	:	ユニット名
POOLTYPE	:	プールファイル(制御 DB も同じ)の DB 種別
		TAM : TAM
		ORACLE : Oracle
MAPID	:	MAPID(ORACLE の場合"---")
RGSETNAME	:	リソースグループセット名(TAM の場合"---")
DMNSTATUS	:	実行デーモン状態
		ACT : 起動中
		TERM : 停止中
		TERMINATING : 停止処理中
TRBLSTATUS	:	障害ステータス
		NONE : 通常状態
		DBTROUBLE : DB 障害
		SWITCH : マスタ切替検出
		DUPLICATE : 二重起動検出
		NORECORD : 制御レコードなし
		RETRYOVER : リトライオーバー
		OTHER : その他異常
UNITBLOCK	:	ユニット無効化状態
		ACT : 有効
		BLOCK : 無効
STATUS	:	ユニットステータス
		INIT : 初期状態
		ACTING : 活性中
		ACT : 活性状態
		INACTING : 非活性中
		INACT : 非活性状態
		STOP : 相手側停止状態
		DIVEND : ディビジョン終了状態
DLSNAME	:	相手論理システム名

CURDIVID	: カレントディビジョン ID
DIVID_DSAM	: DSAM 向けディビジョン ID
NEXTDIVID	: 次ディビジョン ID
NEXTDIVID_DSAM	: DSAM 向け次ディビジョン ID
CRTYCNT	: 制御電文送信回数
DRTYREQCNT	: 再送要求回数
RECVDATACNT	: 受信済み件数
RECVDATANO	: 受信済み通番
USERRECVDATANO	: 受信済みユーザデータ通番
RECVFIXCNT	: 受信済み確定件数
RECVFIXNO	: 受信済み確定通番
USERRECVFIXNO	: 受信済み確定ユーザデータ通番
MSGSIZE	: 電文サイズ
MSGPERCMT	: 1 コミットあたりの電文受信数(カッコ内は SG 値)
CRTYMAX	: 制御電文再送回数
CRTYINVL	: 制御電文再送間隔

+----- SUPERSTREAM INFORMATION (BLOCK) -----+

無効化状態のスーパーSTREAM 情報 (-b 指定時および指定されたスーパーSTREAM が無効化状態の場合のみ出力)

- ① スーパーSTREAM 名
- ② ユニット名

## 2. 4. 32 didtrstart(レシーバ転送開始)

### 名前

didtrstart - ログデータの受信を開始する。

### 書式

```
didtrstart -A
didtrstart -s SuperStreamName
didtrstart -h
```

### 説明

**didtrstart** は、指定されたスーパーストリームのログデータの受信を開始する。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリームのみが対象となる。

### オプション

**-A**

全スーパーストリームの転送を開始する。

**-s**

指定したスーパーストリームの転送を開始する。

**-h**

使用方法を表示する。

### 戻り値

**didtrstart** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtrinit

## 2. 4. 33 didtrstop(レシーバ転送停止)

### 名前

didtrstop - ログデータの受信を停止する。

### 書式

```
didtrstop -A
didtrstop -s SuperStreamName [-M force]
didtrstop -h
```

### 説明

**didtrstop** は、指定されたスーパーストリームのログデータの受信を停止する。  
相手システムへ停止要求電文を送信できない場合は、リトライオーバー後に非活性状態 (INACT) に遷移する。  
強制モード (**-M force**) による転送停止は、非活性中 (INACTING) 状態のストリームに対してのみ有効である。  
オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリームのみが対象となる。

### オプション

**-A**

全スーパーストリームの転送を停止する。

**-s**

指定したスーパーストリームの転送を停止する。

**-M**

転送停止モードを指定する。

省略した場合、相手システムからの応答を待ち合わせてから停止する。

**force**                      相手システムからの応答を待たずに強制的に停止する。

**-h**

使用方法を表示する。

### 戻り値

**didtrstop** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtrinit, didtrstart

## 2.4.34 didtrterm(レシーバ終了)

### 名前

didtrterm - レシーバ機能を終了する。

### 書式

didtrterm [-M {stop|force}]

didtrterm -h

### 説明

**didtrterm** は、レシーバ管理デーモン、レシーバ実行デーモンを停止し、動作環境を解放して、レシーバ機能を終了する。

### オプション

**-M**

終了モードを指定する。

省略した場合、ストリームが転送停止状態であることを確認してレシーバ機能を終了する。転送中のスーパーストリームが存在する場合は、レシーバ機能を終了することができない。

**stop**                    転送中のスーパーストリームに対して転送停止処理を実行してから終了する。

**force**                  転送状態に関わらず終了する。

**-h**

使用方法を表示する。

### 戻り値

**didtrterm** は、以下のいずれかの終了ステータスを戻して終了する。

0                        正常終了

1                        異常終了

### 関連

didltinit, didtrinit, didtrstop

## 2. 4. 35     **didtsctrl (センダデーモン起動停止)**

### 名前

didtsctrl - センダ機能の実行デーモンを起動/停止する

### 書式

```
didtsctrl
didtsctrl -h
```

### 説明

**didtsctrl** は、センダ実行デーモンの起動/停止を行う。

コマンドを実行すると、コマンド実行ノードで動作すべきスーパーストリームを確認し、必要なセンダ実行デーモンの起動処理および、不要となったセンダ実行デーモンの停止処理をおこなう。

### オプション

-h

使用方法を表示する。

### 戻り値

didtsctrl は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit,   didtsinit,   didtsterm

## 2. 4. 36      **didtsdivend(センダ強制ディビジョン終了)**

### 名前

didtsdivend - 転送中のディビジョンを強制的に終了する

### 書式

didtsdivend -s SuperStreamName

didtsdivend -h

### 説明

**didtsdivend** は、指定されたスーパーストリームの転送中ディビジョンを強制的に終了させ、次のディビジョンの転送を可能にする。本コマンドにより強制的にディビジョンを終了させると、現在のディビジョンのログデータは未転送のログデータも含め、すべて破棄される。

本コマンドの実行には、終了対象のスーパーストリームの状態がデータ転送停止状態(INACT)である必要がある。初期状態(INIT)、またはディビジョン終了状態(DIVEND)の場合は、既にディビジョンは終了している状態(未開始の状態)と判断し、本コマンドを実行することはできない。

また、終了させるディビジョンの次のディビジョンが存在する必要があるため、必要に応じて、ディビジョン切替コマンド(didltdivchg)を実行すること。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリームのみが対象となる。

### オプション

-s

指定したスーパーストリームのディビジョンを強制的に終了する。

-h

使用方法を表示する。

### 戻り値

didtsdivend は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- 転送先のレシーバも合わせてディビジョン終了コマンドを投入する必要がある。

### 関連

didltinit,   didtsinit,   didtsstop,   didtrdivend,   didltdivchg

## 2.4.37 didtsinit(センダ開始)

### 名前

didtsinit - センダ機能を開始する

### 書式

```
didtsinit
didtsinit -h
```

### 説明

**didtsinit** は、センダ機能を開始するために必要な環境を生成し、センダ管理デーモン、センダ実行デーモンを起動する。

### オプション

**-h**  
使用方法を表示する。

### 戻り値

didtsinit は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtsterm

## 2. 4. 38 didtsmod(センダ動作変更)

### 名前

didtsmod - センダの各種動作環境を変更する

### 書式

```
didtsmod -T msgdlytime=value {-A|-s SuperStreamName}
didtsmod -T msgdlycnt=value {-A|-s SuperStreamName}
didtsmod -T psendcnt=value {-A|-s SuperStreamName}
didtsmod -T dls=LsName -s SuperStreamName
didtsmod -T divid=value -T datano=value -T userdatano=value -s SuperStreamName
didtsmod -T dctlmsgseqno=value -s SuperStreamName
didtsmod -h
```

### 説明

**didtsmod** は、センダの各種動作環境を変更する。

相手論理システム(**dls**)、およびディビジョン ID(**divid**)、送信開始通番(**datano**)、送信開始利用者通番(**userdatano**)を変更する場合は、変更対象スーパーストリームの状態がデータ転送停止状態(INIT, INACT, DIVEND)である必要がある。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリームのみが対象となる。

### オプション

#### -T

動作変更内容を指定する。

<b>msgdlytime</b>	ログデータ電文送信遅延時間(0~1000 ミリ秒)
<b>msgdlycnt</b>	連続電文送信数(1 以上)
<b>psendcnt</b>	ペーシングあたりの電文数(1 以上)
<b>dls</b>	相手論理システム
<b>divid</b>	ディビジョン ID(1 以上)
<b>datano</b>	送信開始通番(0 以上)
<b>userdatano</b>	送信開始利用者通番(0 以上)
<b>dctlmsgseqno</b>	相手制御電文通番(0 以上)

#### -A

全スーパーストリームに関する動作環境を変更する。

#### -s

指定したスーパーストリームに関する動作環境を変更する。

#### -h

使用方法を表示する。

### 戻り値

didtsmod は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- **-T** オプションで指定するパラメータは、“=”の両辺に空白を指定してはならない。

関連

didltinit, didtsinit

## 2. 4. 39      **didtsref**(センダ状況照会)

### 名前

**didtsref** - センダのデータ転送状況を表示する。

### 書式

```
didtsref -d system
didtsref -A [-l] [-v] [-b]
didtsref -s SuperStreamName [-l] [-v]
didtsref -h
```

### 説明

**didtsref** は、センダのデータ転送状況を表示する。指定したオプションによって出力フォーマットは異なる。

**-d system** 指定では、コマンド実行ノードの論理ノード種別で動作するユニットの実行デーモン情報が対象となる。それ以外のオプションでは、コマンド実行ノードで動作中のスーパーストリーム、およびユニットのみが対象となる。

### オプション

**-d**

表示する情報を指定する。

**system**                  システム情報を表示する。

**-A**

全スーパーストリームの転送状況を表示する。

**-s**

指定したスーパーストリームの転送状況を表示する。

**-l**

リスト形式で表示する。

**-v**

詳細情報を表示する。

**-b**

無効化状態のスーパーストリームも表示する。

**-h**

使用方法を表示する。

### 戻り値

**didtsref** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 注意

- センダ開始直後の実行デーモン起動前や、メンテナンスコマンドによる DB 更新結果を共有メモリに反映する前の状態で、スーパーストリーム情報を照会した場合、通常表示、リスト表示におけるほとんどの項目が“---”で表示される。

関連

didltinit, didtsinit

実行例

システム情報表示( didtsref -d system )

```
+===== DELAYED TRANSFER(SENDER) DISPLAY =====+ 2012/03/26 00:00:00 ①
LSNAME = LS11 ②          LNODENAME = oltp1 ③
+----- SYSTEM INFORMATION -----+
STATUS = ACT ①          STARTTIME = 2012/03/26 00:00:00 ②
+----- COMMAND INFORMATION -----+
CMDREQNUM = 0 ①
+----- TIMER INFORMATION -----+
DMNRESTINVL = 5 ①
DMNDOWNCNT = 5 ②
DMNDOWNTIME = 60 ③
RCVRYCHKINVL = 60 ④
+----- EXEC DAEMON INFORMATION -----+
SPSTNAME = SPST1 ①
UNITNAME = UNIT1 ②
PRCTARGET = YES ③          STATUS = ACT ④
SPSTBLOCK = ACT ⑤          UNITBLOCK = ACT ⑥
TRBLSTATUS = NONE ⑦        RESTARTCNT = 0 ⑧
PROCID = 11489 ⑨          STARTTIME = 2012/03/26 00:00:00 ⑩
+----- EXEC DAEMON INFORMATION -----+
:
+===== END OF DISPLAY =====+
```

+===== DELAYED TRANSFER(SENDER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SYSTEM INFORMATION -----+

管理デーモン情報

- ① 管理デーモンステータス
  - ACT : 起動中
  - TERM : 停止中
  - ACTING : 起動処理中
  - TERMINATING : 停止処理中
- ② 管理デーモン起動時刻

+----- COMMAND INFORMATION -----+

コマンド処理情報

- ① 処理中のコマンド要求数

+----- TIMER INFORMATION -----+

タイマ情報

- ① デーモン再起動間隔
- ② デーモン再起動リトライ回数
- ③ デーモン再起動リトライオーバー判定時間
- ④ 障害時復旧監視間隔

+----- EXEC DAEMON INFORMATION -----+

実行デーモン情報(実行デーモン数分繰り返す)

- ① スーパーストリーム名
- ② ユニット名
- ③ 起動対象フラグ
  - YES : 起動対象
  - NO : 起動対象外
- ④ 実行デーモンステータス
  - ACT : 起動中
  - TERM : 停止中

- TERMINATING : 停止処理中
- ⑤ スーパーストリーム無効化状態
- ACT : 有効
- BLOCK : 無効
- ⑥ ユニット無効化状態
- ACT : 有効
- BLOCK : 無効
- ⑦ 障害ステータス
- NONE : 通常状態
- DBTROUBLE : DB 障害
- SWITCH : マスタ切替検出
- DUPLICATE : 二重起動検出
- NORECORD : 制御レコードなし
- RETRYOVER : リトライオーバー
- OTHER : その他異常
- ⑧ 再起動回数
- ⑨ プロセス ID
- ⑩ 実行デーモン起動時刻

リスト表示( didtsref -l [-A [-v] [-b] | -s スーパーストリーム名 [-v] ] )

+===== DELAYED TRANSFER(SENDER) DISPLAY =====+ 2012/03/26 00:00:00 ①							
LSNAME = LS11 ②		LNODENAME = oltp1 ③					
+----- SUPERSTREAM INFORMATION -----+							
①	②	③	④	⑤	⑥	⑦	⑧
SPSTNAME_____	UNITNAME_____	DMNSTATUS	STATUS____	CURDIVID__	SENDDATACNT__	POOLCNT_____	DATALIM
SPST1	UNIT1	ACT	ACT	1	100	20	NO
SPST2	UNIT2	ACT	INACT	1	200	0	YES
SPST3	UNIT3	BLOCK	---	---	---	---	---
SPST4	UNIT4	BLOCK	---	---	---	---	---
-----							
⑨					⑩	⑪	
TOTALCNT = 2					300	20	
+----- SUPERSTREAM INFORMATION (BLOCK) -----+							
SPSTNAME_____	UNITNAME_____	DMNSTATUS	STATUS____	CURDIVID__	SENDDATACNT__	POOLCNT_____	DATALIM
①	②						
SPST8	UNIT8	---	---	---	---	---	---
+===== END OF DISPLAY =====+							

+===== DELAYED TRANSFER(SENDER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報(無効化解除状態のスーパーストリーム配下のユニット数分表示)

- ① スーパーストリーム名
- ② ユニット名
- ③ 実行デーモン状態
  - ACT : 起動中
  - TERM : 停止中
  - BLOCK : ユニット無効化状態
  - DBTROUBLE : DB 障害
  - SWITCH : マスタ切替検出
  - DUPLICATE : 二重起動検出
  - NORECORD : 制御レコードなし
  - RETRYOVER : リトライオーバー
  - OTHER : その他異常
- ④ ユニットステータス
  - INIT : 初期状態
  - ACTING : 活性中
  - ACT : 活性状態
  - INACTING : 非活性中
  - INACT : 非活性状態
  - STOP : 相手側停止状態
  - DIVENDING : ディビジョン終了中
  - DIVEND : ディビジョン終了状態
- ⑤ カレントディビジョン ID
- ⑥ 送信済み件数
- ⑦ 滞留データ件数(カレントディビジョン) (-v 指定時のみ出力)
- ⑧ 処理完了状態(-v 指定時のみ出力)
  - YES : 未処理データなし
  - NO : 未処理データあり
- ⑨ 合計ユニット数
- ⑩ 合計送信済み件数
- ⑪ 合計滞留データ件数(-v 指定時のみ出力)

+----- SUPERSTREAM INFORMATION (BLOCK) -----+

無効化状態のスーパーSTREAM情報 (-b 指定時のみ出力)

① スーパーSTREAM名

② ユニット名

通常表示( didtsref [-A [-v] [-b] | -s スーパーストリーム名 [-v] ] )

```
+===== DELAYED TRANSFER(SENDER) DISPLAY =====+ 2012/03/26 00:00:00 ①
LSNAME = LS11 ②          LNODENAME = oltp1 ③
+----- SUPERSTREAM INFORMATION -----+
SPSTNAME      = SPST1          UNITNAME      = UNIT1
POOLTYPE      = TAM            MAPID         = 1              RGSETNAME     = ---
DMNSTATUS     = ACT            TRBLSTATUS    = NONE          UNITBLOCK     = ACT
STATUS        = INIT          DLSNAME       = LS11
CURDIVID      = 1             DIVID_DSAM    = 1
NEXTDIVID     = 0             NEXTDIVID_DSAM = 0
CRTYCNT       = 0             PRTYCNT       = 0
SENDDATACNT   = 0             SENDDATANO    = 0              USERSENDDATANO = 0
MSGSIZE       = 16384         CHKINVL      = 1000          PSEND CNT(SG) = 10      (10      )
MSGDLYTIME(SG) = 0      (0      ) MSGDLYCNT(SG) = 1      (1      )
CRTYMAX       = 5             CRTYINVL      = 60
PRTYMAX       = 5             PRTYINVL      = 60
+----- SUPERSTREAM INFORMATION -----+
:
+----- SUPERSTREAM INFORMATION (BLOCK) -----+
SPSTNAME = SPST8 ①          UNITNAME = UNIT8 ②
+===== END OF DISPLAY =====+
```

+===== DELAYED TRANSFER(SENDER) DISPLAY =====+

- ① コマンド実行時刻
- ② 論理システム名
- ③ 論理ノード名

+----- SUPERSTREAM INFORMATION -----+

スーパーストリーム情報(無効化解除状態のスーパーストリーム配下のユニット数分表示)

- SPSTNAME : スーパーストリーム名
- UNITNAME : ユニット名
- POOLTYPE : プールファイル(制御 DB も同じ)の DB 種別
  - TAM : TAM
  - ORACLE : Oracle
- MAPID : MAPID(ORACLE の場合"---")
- RGSETNAME : リソースグループセット名(TAM の場合"---")
- DMNSTATUS : 実行デーモン状態
  - ACT : 起動中
  - TERM : 停止中
  - TERMINATING : 停止処理中
- TRBLSTATUS : 障害ステータス
  - NONE : 通常状態
  - DBTROUBLE : DB 障害
  - SWITCH : マスタ切替検出
  - DUPLICATE : 二重起動検出
  - NORECORD : 制御レコードなし
  - RETRYOVER : リトライオーバー
  - OTHER : その他異常
- UNITBLOCK : ユニット無効化状態
  - ACT : 有効
  - BLOCK : 無効
- STATUS : ユニットステータス
  - INIT : 初期状態
  - ACTING : 活性中
  - ACT : 活性状態
  - INACTING : 非活性中
  - INACT : 非活性状態
  - STOP : 相手側停止状態
  - DIVENDING : デイビジョン終了中

	DIVEND	: デイビジョン終了状態
DLSNAME	:	相手論理システム名
CURDIVID	:	カレントデイビジョン ID
DIVID_DSAM	:	DSAM 向けデイビジョン ID
NEXTDIVID	:	次デイビジョン ID
NEXTDIVID_DSAM	:	DSAM 向け次デイビジョン ID
CRTYCNT	:	制御電文送信回数
PRTYCNT	:	ペーシング要求電文送信回数
SENDACNT	:	送信済み件数
SENDATANO	:	送信済み通番
USERSENDATANO	:	送信済みユーザデータ通番
POOLCNT	:	滞留件数(カレントデイビジョン) (-v 指定時のみ出力)
TOTALPOOLCNT	:	滞留件数(カレントデイビジョン以降の全デイビジョン合計) (-v 指定時のみ出力)
DATALIM	:	処理完了状態 (-v 指定時のみ出力)
	YES	: 未処理データなし
	NO	: 未処理データあり
FIRSTRTYDATANO	:	レシーバからの再送要求に指定された先頭通番 (-v 指定時のみ出力)
LASTRTYDATANO	:	レシーバからの再送要求に指定された最終通番 (-v 指定時のみ出力)
MSGSIZE	:	電文サイズ
CHKINVL	:	データ発生監視間隔
PSENCNT	:	1 ペーシングあたりの送信電文数(カッコ内は SG 値)
MSGDLYTIME	:	電文送信遅延間隔(カッコ内は SG 値)
MSGDLYCNT	:	電文送信遅延電文数(カッコ内は SG 値)
CRTYMAX	:	制御電文再送回数
CRTYINVL	:	制御電文再送間隔
PRTYMAX	:	ペーシング要求電文再送回数
PRTYINVL	:	ペーシング要求電文再送間隔

+----- SUPERSTREAM INFORMATION (BLOCK) -----+

無効化状態のスーパーSTREAM情報 (-b 指定時および指定されたスーパーSTREAMが無効化状態の場合のみ出力)

① スーパーSTREAM名

② ユニット名

## 2. 4. 40      **didtsstart**(センダ転送開始)

### 名前

**didtsstart** - ログデータの送信を開始する。

### 書式

```
didtsstart -A
didtsstart -s SuperStreamName
didtsstart -h
```

### 説明

**didtsstart** は、指定されたスーパーストリームのログデータの送信を開始する。

オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリームのみが対象となる。

### オプション

**-A**

全スーパーストリームの転送を開始する。

**-s**

指定したスーパーストリームの転送を開始する。

**-h**

使用方法を表示する。

### 戻り値

**didtsstart** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit,   didtsinit

## 2. 4. 41 didtsstop(センダ転送停止)

### 名前

didtsstop - ログデータの送信を停止する。

### 書式

```
didtsstop -A
didtsstop -s SuperStreamName [-M force]
didtsstop -h
```

### 説明

**didtsstop** は、指定されたスーパーストリームのログデータの送信を停止する。  
相手システムへ停止要求電文を送信できない場合は、リトライオーバー後に非活性状態 (INACT) に遷移する。  
強制モード (**-M force**) による転送停止は、非活性中 (INACTING) 状態のストリームに対してのみ有効である。  
本コマンドは、コマンド実行したノード上で動作するスーパーストリームのみ対象とする。  
オプションの指定に関わらず、コマンド実行ノードで動作中のスーパーストリームのみが対象となる。

### オプション

**-A**

全スーパーストリームの転送を停止する。

**-s**

指定したスーパーストリームの転送を停止する。

**-M**

転送停止モードを指定する。

省略した場合、相手システムからの応答を待ち合わせてから停止する。

**force**                      相手システムからの応答を待たずに強制的に停止する。

**-h**

使用方法を表示する。

### 戻り値

**didtsstop** は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了
2	警告終了

### 関連

didltinit, didtsinit, didtsstart

## 2.4.42 didtsterm(センダ終了)

### 名前

didtsterm - センダ機能を終了する。

### 書式

didtsterm [-M {stop|force}]

didtsterm -h

### 説明

**didtsterm** は、センダ管理デーモン、センダ実行デーモンを停止し、動作環境を解放して、センダ機能を終了する。

### オプション

**-M**

終了モードを指定する。

省略した場合、ストリームが転送停止状態であることを確認してセンダ機能を終了する。転送中のスーパーストリームが存在する場合は、センダ機能を終了することができない。

**stop**                    転送中のスーパーストリームに対して転送停止処理を実行してから終了する。

**force**                  転送状態に関わらず終了する。

**-h**

使用方法を表示する。

### 戻り値

**didtsterm** は、以下のいずれかの終了ステータスを戻して終了する。

0                        正常終了

1                        異常終了

### 関連

didltinit,   didtsinit,   didtsstop

## 2.5 データ変換・通信オプション

### 2.5.1 コマンド一覧

#### (1) DB アクセス制御

dadacchg	環境定義ファイルの変更を反映する
dadacinit	DB アクセス制御機能の初期化処理を行う
dadacref	DB アクセス制御機能の環境定義情報を照会する
dadacterm	DB アクセス制御機能の終了処理を行う

#### (2) TAM 再配置機能

datrlrefopdata	運用情報および TAM 再配置可否を表示する
datrlupdhash	制御情報に格納されているハッシュ値を更新する
datrlupdopdata	運用情報を更新する

#### (3) データ同期制御機能

dadscblock	スーパーストリームの更新ログ出力抑止状態を変更する。
dadscinfochg	センタ構築時の制御情報を同期させる。
dadsclogref	更新ログの内容を照会する。
dadscref	更新ログ出力が抑止されているスーパーストリームを照会する。

#### (4) セーブロード機能

daolmchkstatus	実行状況の実行状態が RUN のコマンドのプロセスの生存を確認する。
daolmininit	TAM 再配置機能、セーブロード機能共通の初期化処理を行う。
daolmrefstatus	TAM 再配置機能、セーブロード機能のコマンド実行状況を表示する。
daolmterm	TAM 再配置機能、セーブロード機能共通の終了処理を行う。
daslutamload	OracleDB 上のデータを TAM へロードする。
daslutamsave	TAM 上のデータを OracleDB へセーブする。
daslufltamload	TAM 出力ファイル上のデータを TAM へロードする。
daslufltamsave	TAM 上のデータを TAM 出力ファイルへセーブする。

## 2.5.2 dadacchg (環境定義動的変更コマンド)

### 名前

dadacchg - DB アクセス制御の環境定義を動的に変更する

### 書式

dadacchg  
dadacchg -h

### 説明

**dadacchg** は DB アクセス制御の環境定義情報を動的に変更する。

### オプション

-h

使用方法を表示する。

### 戻り値

dadacchg は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了

### 注意

環境定義を動的に変更する時は、DIOSA/XTP の全論理ノード上で dadacchg を実行する必要がある。

### 関連

dadacinit

## 2.5.3 dadacinit(起動コマンド)

### 名前

dadacinit - DB アクセス制御機能の初期化処理を行う

### 書式

dadacinit [-w | -c]

dadacinit -h

### 説明

**dadacinit** は DB アクセス制御の共有メモリを作成して、DB アクセス制御の環境定義を共有メモリ上に展開する。

### オプション

-w

前回運転時の環境を引き継ぐ。(ウォームスタート)(既定値)

-c

最新の環境定義で起動する。(コールドスタート)

-h

使用方法を表示する。

### 戻り値

dadacinit は、以下のいずれかの終了ステータスを戻して終了する。

0                      正常終了

1                      異常終了

### 関連

dadacterm

## 2.5.4 dadacref (環境定義情報照会コマンド)

### 名前

dadacref- DB アクセス制御機能の環境定義情報を照会する

### 書式

```
dadacref [-A | -r REVISION_NO] [-t TABLE_NAME | -t TABLE_ID] [-v]  
dadacref -h
```

### 説明

**dadacref** は、現在稼働中の DB アクセス制御機能の管理テーブルの情報照会を行う。  
パラメータを省略すると最新リビジョン番号の全テーブル情報を表示する。

### オプション

#### -A

全てのリビジョン番号の全テーブル情報を表示する。省略時は最新リビジョン番号の情報を表示する。

#### -r

指定したリビジョン番号の情報を表示する。省略時は最新リビジョン番号の情報を表示する。

#### -t

指定したテーブル名またはテーブル ID の情報を表示する。省略時は全テーブル情報を表示する。

#### -v

全てのリビジョン番号と SQL 雛形ファイルの情報を表示する。

#### -h

使用方法を表示する。

### 戻り値

dadacref は、以下のいずれかの終了ステータスを戻して終了する。

- |   |                                  |
|---|----------------------------------|
| 0 | 正常終了                             |
| 1 | 異常終了                             |
| 2 | 警告終了 (-r または -t 指定時に、指定対象が存在しない) |

### 注意

DB アクセス制御機能が起動している必要がある。

### 関連

dadacinit

実行例

dadacref

+===== DATABASE ACCESS CONTROL INFORMATION DISPLAY =====+ 2013/11/01 18:46:20			①
+===== TABLE SET INFORMATION =====+			
VERSION	: 1		②
+===== TABLE INFORMATION =====+			
TABLE NAME	: USER_TABLE01		③
TABLE ID	: 1		④
TRUNCATE TYPE	: DELETE		⑤
REPLICATION	: ASYNC		⑥
DATE FORMAT	: ---		⑦
DATE NLSPARAM	: ---		⑧
+----- COLUMN INFORMATION -----+			⑨
NAME	TYPE	SIZE	
-----			
DIATC_MAINKEY	BINARY	32	
DIATC_OPNUM	UNSIGNED	4	
DIATC_VERSION	UNSIGNED	4	
MAINKEY	CHAR	16	
DATA_LEN	VARIABLE_SIZE	4	
DATA	CLOB	-	
-----			
TOTAL SIZE:		60	⑩
+----- INDEX INFORMATION -----+			⑪
NAME	COLUMN		
-----			
PK_USER_TABLE01	MAINKEY		
+===== END OF DISPLAY =====+			

## 表示項目の説明

- ① コマンド実行日時
- ② リビジョン番号
- ③ テーブル名
- ④ テーブル ID
- ⑤ OracleDB への全件削除の方法
  - DELETE : DELETE の SQL を発行する
  - TRUNCATE : TRUNCATE の SQL を発行する
  - TEMPLATE : SQL 雛形ファイルで定義した SQL を発行する
- ⑥ レプリケーション種別
  - NONE : レプリケーションを行わない
  - ASYNC : 非同期型レプリケーションを行う
- ⑦ OracleDB の TO\_DATE 関数および TO\_CHAR 関数の第 2 パラメータに指定する日付書式文字列  
環境定義 DACENV 節-TABLE 項の DATEFORMAT パラメータを省略した場合、「---」を表示する。
- ⑧ OracleDB の TO\_DATE 関数および TO\_CHAR 関数の第 3 パラメータに指定する NLS パラメータ  
環境定義 DACENV 節-TABLE 項の DATENLSPARAM パラメータを省略した場合、「---」を表示する。
- ⑨ DACENV の COLUMN 項の情報
  - 「項目名、属性、長さ」の順に表示する
  - 先頭 3 行には、必ず DIOSA の制御情報が表示される。
- ⑩ COLUMN 項の SIZE の合計
  - 固定長項目の合計を表示する。
  - 可変長項目がある場合は、実際のレコード長は可変長項目の長さに応じて大きくなる。
  - ここに表示されるサイズは、DIOSA の制御情報を含んだサイズである。
- ⑪ DACENV の INDEX 項の情報
  - 先頭の情報がプライマリキー情報である。
  - 「インデックス名、項目名」の順に表示する。
  - 複数キー項目から構成される項目名は、複数行で表示する。

**dadacref -v**

```
+===== DATABASE ACCESS CONTROL INFORMATION DISPLAY =====+ 2013/11/01 18:50:25 ①
+===== VERSION INFORMATION =====+ ②
VERSION [1]      : 7
VERSION [2]      : 6
VERSION [3]      : 5
VERSION [4]      : 4

+===== TABLE SET INFORMATION =====+
VERSION          : 7

+===== TABLE INFORMATION =====+
TABLE NAME       : USER_TABLE01
TABLE ID         : 1

<省略>

+----- SQL INFORMATION -----+ ③
INSERT1 = INSERT INTO USER_TABLE02 ( MAINKEY, DATA ) VALUES ( :MAINKEY, :DATA )
INSERT2 = INSERT INTO USER_TABLE03 ( MAINKEY, DATA ) VALUES ( :MAINKEY, :DATA )

UPDATE1 = INSERT INTO USER_TABLE04( MAINKEY, DATA ) VALUES ( :MAINKEY, :DATA )

DELETE1 = DELETE FROM USER_TABLE02 WHERE MAINKEY=:MAINKEY

+===== END OF DISPLAY =====+
```

**表示項目の説明**

- ① コマンド実行日時
- ② 全てのリビジョン番号
- ③ SQL 雛形ファイルに記載した内容  
SQL 雛形ファイルを省略した場合や、当該テーブルに対応する SQL 雛形が定義されていない場合、「Undefined.」を表示する。

## 2.5.5 dadacterm(終了コマンド)

### 名前

dadacterm - DB アクセス制御機能の終了処理を行う

### 書式

dadacterm  
dadacterm -h

### 説明

**dadacterm** は DB アクセス制御の共有メモリを削除する。

### オプション

-h

使用方法を表示する。

### 戻り値

dadacterm は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
1	異常終了

### 注意

起動コマンドを実行した場合、必ず本コマンドを実行すること。

### 関連

dadacinit

## 2.5.6 dadscblock(更新ログ出力抑止状態変更)

### 名前

dadscblock - スーパーストリームの更新ログ出力抑止状態を変更する

### 書式

```
dadscblock -a {-o | -t { MAPID | all }} [-s SuperStreamName]
dadscblock -b {-o | -t MAPID} -s SuperStreamName
dadscblock -h
```

### 説明

**dadscblock** は更新ログ出力抑止に関する動作情報を更新する。スーパーストリーム単位に更新ログ出力抑止状態を変更する。

### オプション

**-a**

抑止解除を行う。

**-b**

抑止登録を行う。

**-o**

Oracle へ更新ログを出力するスーパーストリームを対象とする。

**-t**

メモリテーブルへ更新ログを出力するスーパーストリームを対象とする。  
オプションで対象 MAPID を指定する。全ての MAP を対象とする場合は"all"を指定する。

**-s**

スーパーストリーム名を指定する。  
**-b** を指定した場合は指定必須である。  
**-a** を指定した場合は省略可能である。省略した場合は他オプションの指定範囲内の全てのスーパーストリームを対象とする。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常に終了した。
1	エラーが発生した。
2	対象の一部で処理に失敗した。

### 注意

- **-t** オプションでコマンドを実行する場合、インメモリサーバおよびインメモリサーバ所在管理が開始している必要がある。
- **-t all** 指定時は、アクセスできない MAP への処理は行わず、アクセス可能な全ての MAP に対して処理を行う。一部アクセスできない MAP が存在した場合、本コマンドは警告終了する。

## 2.5.7 dadscinfochg (更新状況同期)

### 名前

dadscinfochg - センタ構築時の制御情報を同期させる

### 書式

```
dadscinfochg -W -F reader -u UnitName -f FileName
dadscinfochg -W -F dsam -f FileName
dadscinfochg -R -F {reader|sender|receiver|dsam} [-u UnitName] -f FileName
dadscinfochg -h
```

### 説明

センタ構築(セーブ／ロード)時の機能間の処理状況の同期を、制御情報ファイルを介して同期させる。

**dadscinfochg** は**-W** で指定した場合、**-F** で指定した機能の処理状況を**-f** で指定した制御情報ファイルに出力する。

**-R** で指定した場合、**-f** で指定した制御情報ファイルの処理状況を**-F** で指定した機能へ反映する。

**-u** で指定した機能のユニットを指定する。**-W** 指定かつ**-F** で **reader** を指定した場合は指定必須である。**-F** で **dsam** を指定した場合は指定不要である。

### オプション

**-W**

制御情報ファイルを作成する。

**-R**

制御情報ファイルを反映する。

**-F**

機能を指定する。

<b>reader</b>	: ログリーダー
<b>sender</b>	: センダ
<b>receiver</b>	: レシーバ
<b>dsam</b>	: D S A M

**-u**

ユニット名を指定する。**-W** 指定かつ**-F** で **reader** を指定した場合は指定必須である。

**-F** で **dsam** を指定した場合は指定不要である。

**-f**

制御情報ファイルを指定する。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常に終了した。
1	エラーが発生した。
2	一部のスーパーストリーム処理でエラーが発生した。

### 注意

- ・**-W** を指定した場合、DIOSA/XTP ディレード転送機能が未起動時でも実行可能である。

- ・ -R を指定した場合、-F で指定した機能が開始している必要がある。

実行例

```
> dadscinfochg -R -F reader -f file
DADSC000 I LS01.node.APP1 dadscinfochg Command started. INPUT=[dadscinfochg -R -F reader -f file]

          ①                      ②
SUPERSTREAM = TAM0000000001_1 ... SUCCESS
SUPERSTREAM = TAM0000000002_1 ... SUCCESS (UNIT BLOCK)
SUPERSTREAM = TAM0000000003_1 ... BLOCK
DADSC014 I LS01.node.APP1 dadscinfochg Finished modifying Delayed Transfer function.
DADSC001 I LS01.node.APP1 dadscinfochg Command terminated normally.
```

表示項目の説明

- ① スーパーストリーム名
- ② 処理結果
  - SUCCESS 正常終了
  - SUCCESS (UNIT BLOCK) 正常終了 (配下に無効化中ユニットあり)
  - BLOCK スーパーストリーム無効化中
  - NOINFO 制御情報ファイルに該当スーパーストリームの情報がない

## 2.5.8 dadsclogref (更新ログ照会)

### 名前

dadsclogref - 更新ログの内容を照会する。

### 書式

```
dadsclogref {-o | -t MAPID} -s SuperStreamName -d DivisionID -n DataNo [-v]  
dadsclogref -h
```

### 説明

指定されたスーパーストリーム名、ディビジョン ID、通番のログデータを読み込み、格納されている更新ログの情報を表示する。DIOSA/XTP データストア (ログリーダ) 停止の原因となった更新ログの情報を照会したい場合、ログリーダ照会コマンドで表示される処理済み通番+1 を、**DataNo** に指定する。

**DataNo** に分割されたログデータの先頭以外の通番を指定した場合、表示されるのはその次の更新ログ情報になるため、必ず先頭の通番を指定すること。

### オプション

**-o**

Oracle の更新ログ転送制御テーブルに登録されているレコードを照会する。

**-t**

メモリテーブルの更新ログ転送制御テーブルに登録されているレコードを照会する。

オプションで対象 MAPID を指定する。指定するのは **-s** で指定するスーパーストリームに対して定義している MAPID である。

**-s SuperStreamName**

スーパーストリーム名を指定する。

**-d DivisionID**

表示対象更新ログのディビジョン ID を指定する。

**-n DataNo**

表示対象更新ログの通番を指定する。

**-v**

詳細表示する場合に指定する。

**-h**

本コマンドの使用方法を表示する。

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常に終了した。
1	エラーが発生した。

### 注意

- ディレード転送機能が起動している必要がある。
- DB アクセス制御機能が起動している必要がある。ただし、Java アプリケーションからの更新ログについては、起動していなくても照会可能である。
- t** オプションでコマンドを実行する場合、インメモリサーバおよびインメモリサーバ所在管理が開始している必要がある。

実行例(C アプリケーションからの更新ログの場合)

+===== DATA REPLICATION INFORMATION DISPLAY =====+ 2012/07/07 12:00:00 ①		
-----		
SuperStream	: TAM0000000001_1	②
Division ID	: 1	③
Data No.	: 1	④
User Data No.	: 1	⑤
-----		
Update Log No.1 [dac]		⑥
[TABLE]	LTABLE	⑦
[TABLE VERSION]	5	⑧
[OPERATION]	INSERT	⑨
[TARGET]	TAM/ORACLE	⑩
[COMMIT]	NO	⑪
[EXEC SQL TEMPLATE]	YES	⑫
[DISTRIBUTE METHOD]	BY_MAINKEY	⑬
[MAPID]	1	⑭
[MAIN KEY]		⑮
	46 00 00 00 00 00 00 00 32 37 31 35 34 32 34 38	F.....27154248
	30 30 30 32 38 39 30 32 00 00 00 00 00 00 00 00	00028902.....
[HASH VALUE]	153	⑯
[OPNUMBER]	15	⑰
[RECORD VERSION]	1	⑱
[PRIMARY KEY]		⑲
	00 00 00 1E	....
[RECORD SIZE]	50	⑳
[RECORD]		㉑
	46 00 00 00 00 00 00 00 32 37 31 35 34 32 34 38	F.....27154248
	30 30 30 32 38 39 30 32 00 00 00 00 00 00 00 00	00028902.....
	30 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00	01.....
	00 01	..
	:	
+===== END OF DISPLAY =====+		

## 表示項目の説明

- ① コマンド実行日時
- ② スーパーストリーム名
- ③ デイビジョン ID
- ④ ログデータ通番
- ⑤ ログデータユーザ通番
- ⑥ 更新ログ通番と種別  
種別([ ]内)には、以下が表示される。  
dac — DB アクセス制御機能が生成するユーザ表更新ログ
- ⑦ 処理対象の表名  
更新種別が COMMIT の場合は"-----"を表示
- ⑧ リビジョン番号
- ⑨ 更新種別( INSERT / UPDATE / DELETE / TRUNCATE / COMMIT )
- ⑩ アクセス対象  
TAM — TAM のみを更新する  
TAM/ORACLE — TAM、Oracle の両方を更新する
- ⑪ OracleDB トランザクション境界フラグ( YES / NO )
- ⑫ SQL 雛形実行フラグ( YES / NO )
- ⑬ 分散キーの指定方法( BY\_MAINKEY / BY\_MAPID )
- ⑭ MAPID
- ⑮ メインキー  
分散キーの指定方法が BY\_MAPID の場合はダンプイメージの表示位置に"-----"を表示
- ⑯ ハッシュ値
- ⑰ 運用番号 (本項目は、互換性のために残されています)
- ⑱ レコード版数
- ⑲ 更新用キー情報  
更新種別が TRUNCATE / COMMIT の場合はダンプイメージの表示位置に"-----"を表示
- ⑳ レコードサイズ
- ㉑ 更新ログの全情報 (-v オプション指定時のみ表示)  
更新種別が COMMIT の場合はダンプイメージの表示位置に"-----"を表示  
更新種別が INSERT、UPDATE の場合は TAM レコードのイメージを表示

実行例（Java アプリケーションからの更新ログの場合）

+===== DATA REPLICATION INFORMATION DISPLAY =====+ 2014/08/17 12:00:00 ①	
-----	
SuperStream : ORA_000000001_1	②
Division ID : 1	③
Data No. : 1	④
User Data No. : 1	⑤
-----	
Update Log No.1 [dac]	⑥
[SQL]	⑦
UPDATE TEST_TABLE SET TEST_ID = ?, TEST_DATE = ? WHERE TEST_ALIAS = ?;	
[PARAM1]	⑧
[TYPE] java.lang.Long	⑨
[SIZE] 8	⑩
[VALUE]	⑪
00 00 00 00 00 4F 8B 00	.....0..
[PARAM2]	
[TYPE] java.sql.Timestamp	
[SIZE] 26	
[VALUE]	
41 44 32 30 31 34 2D 30 38 2D 31 36 20 30 32 3A	AD2014-08-16.02:
34 32 3A 33 32 2E 39 33 32 00	42:32.932.
[PARAM3]	
[TYPE] java.lang.String	
[SIZE] 8	
[VALUE]	
53 5F 58 58 58 5F 59 00	S_XXX_Y.
:	
+===== END OF DISPLAY =====+	

表示項目の説明

- ①～⑥はC アプリケーションからの更新ログの場合と同じ
- ⑦ SQL 文
- ⑧ バインド変数に指定したパラメータ
- ⑨ パラメータの型
- Java の型名が表示される
- ⑩ パラメータのサイズ
- ⑪ パラメータの値(-v オプション指定時のみ表示)
- パラメータの型が整数型の場合はネットワークバイトオーダで表示される
- パラメータのサイズが 0 の場合はダンプイメージの表示位置に"-----"を表示

## 2.5.9 dadscref (更新ログ出力抑止照会)

### 名前

dadscref - 更新ログ出力が抑止されているスーパーストリームを照会する

### 書式

```
dadscref {-o | -t}  
dadscref -h
```

### 説明

**dadscref** は更新ログ転送制御テーブル内を照会し、プールファイルオーバーフローにより更新ログ出力が抑止されているスーパーストリームの情報を照会する。

### オプション

-o

Oracle の更新ログ転送制御テーブルに登録されているレコードを照会する。

-t

メモリテーブルの更新ログ転送制御テーブルに登録されているレコードを照会する。

-h

本コマンドの使用方法を表示する。

### カラムの説明

以下に、dadscref が表示するカラムの見出しと各カラムの意味を説明する。

各カラムが以下の順番に従って左から右へ表示される。

MAP	MAPID
STREAM	ストリーム名

### 戻り値

以下のいずれかの終了ステータスを戻して終了する。

0	処理が正常に終了した。
1	エラーが発生した。
2	一部情報の取得に失敗した。

### 注意

- -t オプションでコマンドを実行する場合、インメモリサーバおよびインメモリサーバ所在管理が開始している必要がある。

実行例

```
dadscref -t

+===== DATA REPLICATION INFORMATION DISPLAY =====+ 2012/07/07 13:00:00 ①
+----- BLOCKED STREAM LIST -----+
MAP_____ STREAM_____
      ②                ③
      1  TAM0000001_00_1
      1  TAM0000001_01_1
     11  #CANNOT READ(0x102010)
+===== END OF DISPLAY =====+
```

```
dadscref -o

+===== DATA REPLICATION INFORMATION DISPLAY =====+ 2012/07/07 13:00:00 ①
+----- BLOCKED STREAM LIST -----+
STREAM_____
      ③
MTN0000001_00_1
ORA_STRM01_1
+===== END OF DISPLAY =====+
```

表示項目の説明

- ① コマンド実行日時
  - ② MAPID
  - ③ ストリーム名
- t 指定時に、ストリーム名に”#CANNOT READ(コード)”と表示された場合、その MAP へアクセスできなかったことを表している。
- コードは MAP の状態を 16 進数で表示しており、各桁の意味は左から順に以下の通りである。
- 1-2 桁：レプリケーショングループ状態
    - 10 正常稼動
    - 20 停止
    - 31 計画マスタ切替中
    - 40 障害
    - 41 障害によるマスタ切替中
  - 3-4 桁：MAP 運用状態
    - 10 正常稼動
    - 20 停止状態
  - 5-6 桁：MAP 閉塞状態
    - 10 正常稼動
    - 30 閉塞状態
- 上記以外のコードが表示された場合は、DIOSA/XTP の API リファレンスマニュアル(t\_diosa\_getmapstatuca)、および DIOSA/XTP のヘッダファイル(diosa.h)の内容からコードの示す状態を確認する。

## 2.5.10 daolmchkstatus (実行状況確認コマンド)

### 名前

daolmchkstatus - 実行状況の実行状態が RUN のコマンドのプロセスの生存を確認する

### 書式

```
daolmchkstatus
daolmchkstatus -h
```

### 説明

daolmchkstatus コマンドは、実行状況を管理しているコマンドのうち、実行状態が RUN のコマンドのプロセスの生存を確認し、プロセスが終了している場合は、実行状態を ERROR に更新する。また、プロセスが生存している場合は、実行状態は変更しないが戻り値によって、実行中のプロセスが存在していた情報を返却する。

### オプション

-h  
    使用法を表示する。

### 戻り値

daolmchkstatus は、以下のいずれかの終了ステータスを戻して終了する。

- 0     正常終了
- 1     異常終了
- 2     実行中のプロセスが存在

### 関連

daolminit, daolmterm

## 2.5.11 daolminit(オンライン中メンテナンス初期化コマンド)

### 名前

daolminit - TAM 再配置機能、セーブロード機能共通の初期化処理を行う

### 書式

```
daolminit
daolminit -h
```

### 説明

daolminit コマンドは、TAM 再配置機能、セーブロード機能共通の初期化処理を行う。

### オプション

-h

使用法を表示する。

### 戻り値

daolminit は、以下のいずれかの終了ステータスを戻して終了する。

0 正常終了

1 異常終了

### 関連

daolmterm

## 2.5.12 daolmrefstatus(実行状況照会コマンド)

### 名前

daolmrefstatus - TAM 再配置機能、セーブロード機能のコマンド実行状況を表示する

### 書式

daolmrefstatus [-s {DONE|ERROR|RUN}]  
daolmrefstatus -h

### 説明

daolmrefstatus コマンドは、TAM 再配置機能、セーブロード機能の下記コマンドの実行状況を表示する。

TAM 再配置機能	ハッシュ値更新コマンド
セーブロード機能	TAM セーブコマンド
	TAM ロードコマンド
	ファイル出力 TAM セーブコマンド
	ファイル入力 TAM ロードコマンド

### オプション

- s  
実行状態が STATUS のコマンドの実行状況のみ表示する。
- h  
使用法を表示する。

### カラムの説明

以下に、daolmrefstatus が表示するカラムの見出しと各カラムの意味をあげる。各カラムは以下の順番に従って左から右へプリントされる。

START DATE	コマンド開始日時
LAST UPDATE DATE	実行状況最終更新日時
PROGRESS	進捗状況 「完了数 / 全体数」の形式で表示 全体数は論理表数を意味する
STATUS	実行状態 DONE          正常終了 ERROR         異常終了 RUN            実行中
NAME	コマンド名
PID	プロセス ID
MAPID	MAP ID

### 戻り値

daolmrefstatus は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了

関連

daolminit, daolmterm

実行例

+===== ONLINE MAINTENANCE STATUS DISPLAY =====+					YYYY/MM/DD HH:MM:SS			
START DATE		LAST UPDATE DATE		PROGRESS	STATUS	NAME	PID	MAPID
2015/08/16 03:50:51		2015/08/16 03:55:44		66/66	DONE	daslutamsave	14352	11
2015/08/16 03:50:51		2015/08/16 03:55:27		66/66	DONE	daslutamsave	14353	12
2015/08/16 04:13:30		2015/08/16 04:17:19		66/66	DONE	datrlupdhash	24902	
2015/08/16 04:22:22		2015/08/16 04:22:28		66/66	DONE	daslufltamsave	38823	12
2015/08/16 04:26:11		2015/08/16 04:26:21		66/66	DONE	daslufltamload	38964	12
+===== END OF DISPLAY =====+								

## 2.5.13 daolmterm(オンライン中メンテナンス終了コマンド)

### 名前

daolmterm - TAM 再配置機能、セーブロード機能共通の終了処理を行う

### 書式

```
daolmterm
daolmterm -h
```

### 説明

daolmterm コマンドは、TAM 再配置機能、セーブロード機能共通の終了処理を行う。

### オプション

-h

使用法を表示する。

### 戻り値

daolmterm は、以下のいずれかの終了ステータスを戻して終了する。

0      正常終了

1      異常終了

### 関連

daolminit

## 2.5.14 daslutamload (TAM ロードコマンド)

### 名前

daslutamload - OracleDB 上のデータを TAM へロードする

### 書式

```
daslutamload -m MAP_ID [-n COMMITNUMBER] [-t TIMEOUT] [-r READSTRECNUM] [-k]  
daslutamload -h
```

### 説明

daslutamload コマンドは、パラメータで指定された MAP の論理表、コンポーネントの OracleDB 上のデータを TAM へロードする。本コマンドは、データアクセスが発生しない状態で実行する必要がある。

### オプション

#### -m

ロード対象とする MAP の MAPID を指定する。

#### -n

COMMITNUMBER 件数毎にコミットを実行する。(数字 0~10000000)

0 を指定した場合は、最後に 1 回だけコミットを実行する。

省略した場合は 100 件ロードする毎にコミットを実行する。

#### -t

タイムアウト時間を分単位で指定する。(数字 0~1440)

指定した時間が経過しても処理が終了しない場合は、処理を中止し警告終了する。

0 を指定、または省略した場合はタイムアウトしない。

#### -r

複数レコード読込時に何件単位にレコードを取得するか読込基準件数を指定する。(数字 1~1000)

省略した場合は、10 件が読込基準件数となる。

#### -k

論理表のデータのうち、ユーザデータ状態管理表にメインキーが登録されているデータのみを TAM にロードする。

#### -h

使用法を表示する。

### 戻り値

daslutamload は、以下のいずれかの終了ステータスを戻して終了する。

- 0 正常終了
- 1 異常終了
- 2 警告終了(タイムアウト)

### 注意

対象 MAP の TAM と OracleDB 上のデータへデータアクセスが発生しない状態で実行する必要がある。  
マスタノードで実行する必要がある。

## 2.5.15 daslutamsave (TAM セーブコマンド)

### 名前

daslutamsave - TAM 上のデータを OracleDB へセーブする

### 書式

```
daslutamsave -m MAP_ID [-n COMMITNUMBER] [-t TIMEOUT] [-r READSTRECNUM]
daslutamsave -h
```

### 説明

daslutamsave コマンドは、パラメータで指定された MAP の論理表、コンポーネントの TAM 上のデータを、OracleDB へセーブする。

### オプション

#### -m

セーブ対象とする MAP の MAPID を指定する。

#### -n

COMMITNUMBER 件数毎にコミットを実行する。(数字 0~10000000)

0 を指定した場合は、最後に 1 回だけコミットを実行する。

省略した場合は 10000 件セーブする毎にコミットを実行する。

#### -t

タイムアウト時間を分単位で指定する。(数字 0~1440)

指定した時間が経過しても処理が終了しない場合は、処理を中止し警告終了する。

0 を指定、または省略した場合はタイムアウトしない。

#### -r

複数レコード読込時に何件単位にレコードを取得するか読込基準件数を指定する。(数字 1~1000)

省略した場合は、10 件が読込基準件数となる。

#### -h

使用法を表示する。

### 戻り値

daslutamsave は、以下のいずれかの終了ステータスを戻して終了する。

- 0 正常終了
- 1 異常終了
- 2 警告終了(タイムアウト)

### 注意

対象 MAP の OracleDB 上のデータへデータアクセスが発生しない状態で実行する必要がある。

## 2.5.16 daslufltamload(ファイル入力 TAM ロードコマンド)

### 名前

daslufltamload - TAM 出力ファイル 上のデータを TAM へロードする

### 書式

```
daslufltamload -m MAP_ID -d DIRECTORY [-f EXIT_FUNCTION] [-n COMMITNUMBER] [-t TIMEOUT]
daslufltamload -h
```

### 説明

daslufltamload コマンドは、パラメータで指定された MAP の論理表の TAM 出力ファイル上のデータを TAM へロードする。

MAP 構成、TAM 構成に変更が発生した場合、出口関数を使用し新しい構成に対応したデータのロードを行う。本コマンドは、TAM へのデータアクセスが発生しない状態で実行する必要がある。

### オプション

**-m**

ロード対象とする MAP の MAPID を指定する。

**-d**

TAM 出力ファイルの格納ディレクトリを指定する。

**-f**

レコード情報と論理表名よりハッシュ値を返す出口関数を指定する。

出口関数はユーザ側で作成する。

出口関数指定の場合、格納ディレクトリ配下の全ての TAM 出力ファイルより、ロード対象のデータを TAM へロードする。

出口関数の指定がない場合、格納ディレクトリ配下の TAM 出力ファイルより、ロード対象のファイルのみを読み込み、データを全て TAM へロードする。

**-n**

COMMITNUMBER 件数毎にコミットを実行する。(数字 0~10000000)

0 を指定した場合は、最後に 1 回だけコミットを実行する。

省略した場合は 100 件ロードする毎にコミットを実行する。

**-t**

タイムアウト時間を分単位で指定する。(数字 0~1440)

指定した時間が経過しても処理が終了しない場合は、処理を中止し警告終了する。

0 を指定、または省略した場合はタイムアウトしない。

**-h**

使用法を表示する。

### 戻り値

daslufltamload は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了
- 2      警告終了(タイムアウト)

#### 注意

対象 MAP の TAM 上のデータへデータアクセスが発生しない状態で実行する必要がある。  
マスタノードで実行する必要がある。

## 2.5.17 daslufltamsave(ファイル出力 TAM セーブコマンド)

### 名前

daslufltamsave - TAM 上のデータを TAM 出力ファイルへセーブする

### 書式

```
daslufltamsave -m MAP_ID -d DIRECTORY [-t TIMEOUT] [-r READSTRECNUM]
daslufltamsave -h
```

### 説明

daslufltamsave コマンドは、構成変更前に TAM 情報のバックアップを作成するため、パラメータで指定された MAP の論理表の TAM 上のデータを、TAM 出力ファイルへセーブする。

### オプション

**-m**

セーブ対象とする MAP の MAPID を指定する。

**-d**

TAM 出力ファイルの格納ディレクトリを指定する。

**-t**

タイムアウト時間を分単位で指定する。(数字 0～1440)

指定した時間が経過しても処理が終了しない場合は、処理を中止し警告終了する。

0 を指定、または省略した場合はタイムアウトしない。

**-r**

複数レコード読込時に何件単位にレコードを取得するか読込基準件数を指定する。(数字 1～1000)

省略した場合は、10 件が読込基準件数となる。

**-h**

使用法を表示する。

### 戻り値

daslufltamsave は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了
- 2      警告終了(タイムアウト)

## 2.5.18 datrlrefopdata(運用情報照会コマンド)

### 名前

datrlrefopdata - 運用情報および TAM 再配置可否を表示する

### 書式

```
datrlrefopdata [-r REPLICATION_GROUP_ID [-m MAP_ID[, MAP_ID2...]]] [-l]
datrlrefopdata -h
```

### 説明

datrlrefopdata コマンドは、パラメータで指定されたレプリケーショングループまたは MAP について、運用情報を表示する。パラメータを省略すると全ての MAP の情報を表示する。

### オプション

- r  
指定されたレプリケーショングループの情報を表示する。-m 省略時はレプリケーショングループの全 MAP の情報を表示する。
- m  
指定された MAP の情報を表示する。表示対象の MAP が複数存在する場合はカンマ区切りで指定する。
- l  
カラムの見出しを表示する。
- h  
使用法を表示する。

### カラムの説明

以下に、datrlrefopdata が表示するカラムの見出しと各カラムの意味をあげる。各カラムは空白区切りで以下の順番に従って左から右へプリントされる。スレーブノード名は可変個のカラムで、複数のスレーブノードが存在する場合は個数分のカラムがプリントされる。

REPGRP	レプリケーショングループ ID
MAP	MAP ID
STATUS	ハッシュ値変更状態。ハッシュ値切替を行った場合にハッシュ値の変更や MAP の追加、削除が行われるかどうかを表す。 NO            変更なし CHG           変更あり ADD           MAP 追加 DEL           MAP 削除
TRL	再配置可否。マスタ切替中などで再配置が行えない場合は不可となる。 OK            可 NG            不可
TCSTATUS	TC 状態。(本項目は、互換性のために残されています)
TCNUMBER	TC 番号。(本項目は、互換性のために残されています)
OPNUMBER	運用番号。(本項目は、互換性のために残されています)
OPMODE	運用モード

	UNUSED	未使用
	TAM	TAM アクセスモード
	SAVE	TAM→Oracle 移行状態
	ORACLE	Oracle アクセスモード
	LOAD	Oracle→TAM 移行状態
MAPCHANGESTATUS	MAP 構成変更状態	
	NORMAL	通常状態
	CHANGE	構成変更中
MASTER	マスタ論理ノード名	
SLAVE	スレーブ論理ノード名（空白区切りで複数表示）	

戻り値

- datrlrefopdata は、以下のいずれかの終了ステータスを戻して終了する。
- 0      正常終了
  - 1      異常終了
  - 2      警告終了(条件付照会時に表示レコード無し、または該当レコード無し)

関連

datrlupdopdata

実行例

```
+===== OPERATIONAL DATA DISPLAY =====+          YYYY/MM/DD HH:MM:SS
REPGRP MAP STATUS TRL TCSTATUS TCNUMBER OPNUMBER OPMODE MAPCHANGESTATUS MASTER SLAVE
1 11 NO OK NORMAL 1 0 TAM NORMAL LNODE-A LNODE-B LNODE-C LNODE-D
1 12 CHG OK NORMAL 1 0 TAM CHANGE LNODE-A LNODE-B LNODE-C LNODE-D
2 21 CHG OK NORMAL 1 0 TAM CHANGE LNODE-B LNODE-C LNODE-D LNODE-A
3 31 ADD OK NORMAL 0 0 UNUSED CHANGE LNODE-C LNODE-D LNODE-A LNODE-B
+===== END OF DISPLAY =====+
```

## 2.5.19 datrlupdhash (ハッシュ値更新コマンド)

### 名前

datrlupdhash - 制御情報に格納されているハッシュ値を更新する

### 書式

<メモリキャッシュが動作する環境の場合>

```
datrlupdhash [-m MAP_ID[,MAP_ID2...]] [-r RGSET] [-n COMMITNUMBER] [-t TIMEOUT] [-k]
```

```
datrlupdhash -h
```

<メモリキャッシュが動作しない環境の場合>

```
datrlupdhash [-r RGSET] [-n COMMITNUMBER] [-t TIMEOUT] [-k] -f HASHFUNCTION
```

```
datrlupdhash -h
```

### 説明

datrlupdhash コマンドは OracleDB 上の制御情報に格納されているハッシュ値を再計算し更新する。TAM 再配置手順、オンライン中 DB リカバリ手順の中で使用する。

詳細は「利用の手引」の「2.3. オンライン中メンテナンス」を参照。

### オプション

**-m**

ハッシュ値更新対象とする MAP の MAPID を指定する。対象の MAP が複数存在する場合はカンマ区切りで指定する。指定した MAP のいずれかに属する論理表を更新対象とする。  
省略した場合は全 MAP の論理表を更新対象とする。

**-r**

DB 接続に使用するリソースグループセット名を指定する。(英数字 1～31 文字)  
省略した場合はデフォルトリソースグループセット名が使用される。

**-n**

COMMITNUMBER 件数毎にコミットを実行する。(数字 0～10000000)  
0 を指定した場合は、最後に 1 回だけコミットを実行する。  
省略した場合は 10000 件ハッシュ値を更新する毎にコミットを実行する。

**-t**

タイムアウト時間を分単位で指定する。(数字 0～1440)  
指定した時間が経過しても処理が終了しない場合は、処理を中止し警告終了する。  
0 を指定、または省略した場合はタイムアウトしない。

**-k**

論理表のデータのうち、ユーザデータ状態管理表にメインキーが登録されているデータのハッシュ値のみを更新する。指定しない場合は、メインキーの登録有無に関わらず、論理表の全てのデータを更新する。

**-f**

ハッシュ関数名を指定する。(英数字 1～30 文字)  
DIOSA/XTP メモリキャッシュを使用しない環境においてハッシュ値を計算する場合に指定する。  
省略した場合はメモリキャッシュの環境定義 (IMENV 節) で定義したハッシュ関数を使用する。

**-h**

使用法を表示する。

### 戻り値

datrlupdhash は、以下のいずれかの終了ステータスを戻して終了する。

- 0      正常終了
- 1      異常終了
- 2      警告終了(タイムアウト)

#### 注意

OracleDB 上のデータヘデータアクセスが発生しない状態で実行する必要がある。

指定した MAP に属する論理表が指定されていない MAP にも分散している場合、指定していない MAP に属するレコードも含めてハッシュ値の更新を行う。

メモリキャッシュが動作する環境では、ハッシュ関数名を指定してはならない。

## 2.5.20 datrlupdopdata (運用情報更新コマンド)

### 名前

datrlupdopdata - 運用情報を更新する

### 書式

```
datrlupdopdata -r REPLICATION_GROUP_ID -m MAP_ID[,MAP_ID2...] [-c] [-o {u|t|s|o|l}]
                    [-t TIMEOUT] [-l {n|c}]
datrlupdopdata -r REPLICATION_GROUP_ID -C [-t TIMEOUT]
datrlupdopdata -h
```

### 説明

datrlupdopdata コマンドは、パラメータで指定された MAP の運用情報を、指定された状態に更新する。

### オプション

#### -r

更新対象とする MAP の、レプリケーショングループ ID を指定する。

#### -m

更新対象とする MAP の、MAP ID を指定する。更新対象の MAP が複数存在する場合はカンマ区切りで指定する。

#### -c

ハッシュ値を切り替える。

#### -o

運用モードを更新する。

u	未使用
t	TAM アクセスモード
s	TAM→Oracle 移行状態
o	Oracle アクセスモード
l	Oracle→TAM 移行状態

#### -l

MAP 構成変更状態を更新する。

n	通常状態
c	構成変更中

#### -C

ハッシュ値を切り替える。TAM と Oracle のデータを同期する必要がない場合に指定する。

#### -t

更新完了応答待ち合わせ時間を秒で指定する。(数字 0～3600)

指定した時間が経過しても処理が終了しない場合は、処理を中止し警告終了する。

0 を指定、または省略した場合はタイムアウトしない。

#### -h

使用法を表示する。

### 戻り値

datrlupdopdata は、以下のいずれかの終了ステータスを戻して終了する。

0	正常終了
---	------

- 1 異常終了
- 2 警告終了(タイムアウト)

#### 注意

-C の指定は、既存データに影響がないハッシュ値を追加する場合以外で実行してはならず、また運用モードが「TAM アクセスモード」、「未使用」以外の MAP が存在する場合、コマンドは異常終了する。  
マスタノードで実行する必要がある。

#### 関連

`datrlrefopdata`

## 第3章 TPP リファレンス

### 3.1 概要

DIOSA/XTP の一部の制御デーモンは、TPBASE の TPP として実行する必要がある。

本章では、TPP に指定するパラメータや、利用上の一般的な規則について説明する。

なお、パラメータとして記述されている内容は、TPBASE の定義ファイル(.ped ファイルの ARGS パラメータ)で指定するパラメータのことである。定義のパスは DIOSA/XTP を /opt/diosa\_xtp にインストールした前提で記している。  
/opt/diosa\_xtp 以外にインストールした場合は、適宜読み替えること。

### 3.2 TPP 一覧

dicoc TPP	C0 の呼び出しやトランザクション制御をおこなう。
digntdel TPP	電文保証の受信情報を削除する。
digntfintpp	電文保証の応答電文を受信する。
digntrsd TPP	電文保証の電文を再送する。
dincmctrl TPP	論理システム間、論理ノード間のパスを管理する。

# 3.3 TPP リファレンス

## 3.3.1 dicoc TPP (C0 制御 TPP)

書式

dicoc TPP

説明

TPBASE の TPP を使ったアプリケーション用フレームワーク。  
オンライン処理データを受信し、C0 の呼び出しやトランザクション制御をおこなう。

オプション

なし

プロセス環境定義ファイルの定義

以下のパラメータを設定する。

PROEPILIB

/opt/diosa\_xtp/lib/libdxtpcocprotpb. so

PROENTRY

di\_coc\_CocPrologue

EPIENTRY

di\_coc\_CocEpilogue

トランザクション定義ファイルの定義

以下のパラメータを設定する。

TPPLIB

/opt/diosa\_xtp/lib/libdxtpcoctrntpb. so

TPPENTRY

di\_coc\_CocTrnsEnt

ABORTEXTLIB

/opt/diosa\_xtp/lib/libdxtpcocabttpb. so

ABORTEXTIT

di\_coc\_CocExceptMain

注意

なし

関連

なし

### 3.3.2 digntdeltp (電文保証 受信情報削除 TPP)

#### 書式

digntdeltp [-i] [-d]

#### 説明

電文保証の受信情報を削除する。

#### オプション

-i -d いずれも指定されない場合は、メモリキャッシュ、Oracle 両方の DB を処理対象にする。

-i

DB としてメモリキャッシュを使用する。

-d

DB として Oracle を使用する。

#### プロセス環境定義ファイルの定義

以下のパラメータを設定する。

**PROEPILIB**

/opt/diosa\_xtp/lib/libdxtpgntdelrcvproepi.so

**PROENTRY**

di\_gnt\_TppDelRcvInit

**EPIENTRY**

di\_gnt\_TppDelRcvTerm

#### トランザクション定義ファイルの定義

以下のパラメータを設定する。

**TPPLIB**

/opt/diosa\_xtp/lib/libdxtpgntdelrcvtrns.so

**TPPENTRY**

di\_gnt\_TppDelRcvMain

**ABORTEXTLIB**

/opt/diosa\_xtp/lib/libdxtpgntdelrcvabort.so

**ABORTEXTIT**

di\_gnt\_TppDelRcvAbort

#### di\_gnt\_TppDelRcvMain 注意

-i -d の指定に関わらず、DIOSA/XTP のライブラリ構成で未インストールを設定した DB は処理対象外となる。

#### 関連

digntfintpp digntrsdtp

### 3.3.3 digntfintpp（電文保証 応答電文受信 TPP）

#### 書式

digntfintpp [-i] [-d]

#### 説明

電文保証の応答電文を処理する。

#### オプション

-i -d いずれも指定されない場合は、メモリキャッシュ、Oracle 両方の DB を処理対象にする。

-i

DB としてメモリキャッシュを使用する。

-d

DB として Oracle を使用する。

#### プロセス環境定義ファイルの定義

以下のパラメータを設定する。

**PROEPILIB**

/opt/diosa\_xtp/lib/libdxtpgntrcvfinproepi.so

**PROENTRY**

di\_gnt\_TppRcvFinInit

**EPIENTRY**

di\_gnt\_TppRcvFinTerm

#### トランザクション定義ファイルの定義

以下のパラメータを設定する。

**TPPLIB**

/opt/diosa\_xtp/lib/libdxtpgntrcvfintrns.so

**TPPENTRY**

di\_gnt\_TppRcvFinMain

**ABORTEXTLIB**

/opt/diosa\_xtp/lib/libdxtpgntrcvfinabort.so

**ABORTEXTIT**

di\_gnt\_TppRcvFinAbort

#### 注意

-i -d の指定に関わらず、DIOSA/XTP のライブラリ構成で未インストールを設定した DB は処理対象外となる。

#### 関連

digntdeltpg digntrsdtpg

### 3.3.4 digntrsdtp (電文保証 電文再送 TPP)

#### 書式

digntrsdtp [-i] [-d]

#### 説明

電文保証の電文を再送する。

#### オプション

-I -d いずれも指定されない場合は、メモリキャッシュ、Oracle 両方の DB を処理対象にする。

-i

DB としてメモリキャッシュを使用する。

-d

DB として Oracle を使用する。

#### プロセス環境定義ファイルの定義

以下のパラメータを設定する。

**PROEPILIB**

/opt/diosa\_xtp/lib/libdxtpgntresendproepi.so

**PROENTRY**

di\_gnt\_TppResendInit

**EPIENTRY**

di\_gnt\_TppResendTerm

#### トランザクション定義ファイルの定義

以下のパラメータを設定する。

**TPPLIB**

/opt/diosa\_xtp/lib/libdxtpgntresendtrns.so

**TPPENTRY**

di\_gnt\_TppResendMain

**ABORTEXTLIB**

/opt/diosa\_xtp/lib/libdxtpgntresendabort.so

**ABORTEXT**

di\_gnt\_TppResendAbort

#### 注意

-i -d の指定に関わらず、DIOSA/XTP のライブラリ構成で未インストールを設定した DB は処理対象外となる。

#### 関連

digntdeltp digntfintpp

### 3.3.5 dincmctrltp (通信制御 TPP)

#### 書式

dincmctrltp

#### 説明

論理システム間、論理ノード間のパスを管理する。

#### オプション

なし

#### プロセス環境定義ファイルの定義

以下のパラメータを設定する。

##### PROEPILIB

/opt/diosa\_xtp/lib/libdxtpnctppproepi.so

##### PROENTRY

di\_ncm\_TppPrologue

##### EPIENTRY

di\_ncm\_TppEpilogue

##### MAXPROCESS

1

#### 注意

通信制御 TPP の多重度は 1 を指定すること。

#### 関連

なし

# 付録A コマンド一覧

## A.1 コマンド呼び出し可能ノード

### A.1.1 アプリケーション実行制御

関数名	論理ノード種別			備考
	AP	OLTP	DB	
diapptrcctrl	○	○	○	
diapptrcfedit	○	○	○	
diapptrcfenv	○	○	○	
diapptrcinit	○	○	○	
diapptrcmedit	○	○	○	
diapptrcmenv	○	○	○	
diapptrcmflush	○	○	○	
diapptrcterm	○	○	○	
dibacexec	○	○	—	
dibcmctrl	○	○	○	
dibcmref	○	○	○	
dibcmsync	○	○	○	
dibcmupd	○	○	○	
dicddchg	○	○	○	
dicddctrl	○	○	○	
dicddhstctl	○	○	○	
dicddhstedit	○	○	○	
dicmdsend	○	○	○	
dicocinit	○	○	—	
dicocmod	○	○	—	
dicocref	○	○	—	
dicocrefsg	○	○	—	
dicocterm	○	○	—	
dicoctxblock	○	○	—	
dicomcatget	○	○	○	
didamctrl	○	○	○	
didbgerr	○	○	○	
didbgerrswap	○	○	○	
didbginit	○	○	○	
didbglog	○	○	○	
didbglogswap	○	○	○	
didbgterm	○	○	○	
didlrchg	○	○	○	
didlrinit	○	○	○	
didlrreflib	○	○	○	
didlrreflm	○	○	○	
dietgctrl	○	○	—	
dietgref	○	○	—	

関数名	論理ノード種別			備考
	AP	OLTP	DB	
diirmadd	○	○	○	
diirmdel	○	○	○	
diirmref	○	○	○	
diirmrep	○	○	○	
dimdrctrl	○	○	○	
dimemchg	○	○	○	
dimemfree	○	○	○	
dimemref	○	○	○	
dimsgdctrl	○	○	○	
dimsgdisp	○	○	○	
dimsglimit	○	○	○	
dimsglvmod	○	○	○	
dimsglvref	○	○	○	
dimsgmtn	○	○	○	
diopschg	○	○	—	
diopsctrl	○	○	—	
diopsedit	○	○	—	
diopsflush	○	○	—	
diopsfput	○	○	—	
diopsgather	○	○	—	
diopsmrg	○	○	—	
distart_com	○	○	○	
distatref	○	○	○	
distop	○	○	○	
ditmcactv	○	○	—	
ditmcdel	○	○	—	
ditmcdmn	○	○	—	
ditmcdmnstop	○	○	—	
ditmchold	○	○	—	
ditmcref	○	○	—	
ditmcset	○	○	—	
ditmsgset	○	○	—	

• ○ … 使用可能    — … 使用不可    △ … 条件付使用可能

A. 1. 2      通信制御

関数名	論理ノード種別			備考
	AP	OLTP	DB	
digntchgdest	—	○	—	
digntdbreset	—	○	—	
digntdel	—	○	—	
digntinit	—	○	—	
digntmod	—	○	—	
digntref	—	○	—	
digntterm	—	○	—	
dilspathctrl	○	○	—	
dilspathref	○	○	—	
dilspathrefstats	○	○	—	
dincmdbhchkctrlinit	○	○	○	
dincmdbhchkctrlterm	○	○	○	
dincmdbinit	○	○	○	
dincmdbstat	○	○	—	
dincmdbterm	○	○	○	
dincmpathdmnctrl	○	○	—	
dincmpathinit	○	○	—	
dincmpathterm	○	○	—	
dinodepathctrl	○	○	—	
dinodepathref	○	○	—	
dinodepathrefstats	○	○	—	
diotcref	○	○	—	
diotcrefstats	○	○	—	

•    ○ … 使用可能    — … 使用不可    △ … 条件付使用可能

A. 1.3      メモリキャッシュ

関数名	論理ノード種別			備考
	AP	OLTP	DB	
diimblock	—	○	—	
diimcheck	○	○	—	
diimchg	○	○	—	
diimchghash	○	○	—	
diimctrl	—	○	—	
diiminit	○	○	—	
diimmod	○	○	—	
diimnotify	—	○	—	
diimperfref	—	○	—	
diimperfreset	—	○	—	
diimquegc	—	○	—	
diimref	○	○	—	
diimtblopenclose	—	○	—	
diimterm	○	○	—	
ditamswap	—	○	—	
ditamswitch	—	○	—	

•    ○ … 使用可能    — … 使用不可    △ … 条件付使用可能

A. 1.4      データストア

関数名	論理ノード種別			備考
	AP	OLTP	DB	
didltblock	○	○	—	
didltchg	○	○	○	
didltcreate	○	○	○	
didltctrl	○	○	○	
didltdivchg	○	○	○	
didltinit	○	○	○	
didltmodreset	○	○	○	
didltpathref	○	○	—	
didltpooldel	○	○	—	
didltpoolmod	○	○	○	
didltpoolref	○	○	—	
didltref	○	○	○	
didltrefdb	○	○	○	
didltslmmv	—	—	○	
didltslmref	○	○	○	
didltterm	○	○	○	
didltupd	○	○	○	
didtlctrl	○	○	—	
didtldivend	○	○	—	
didtlinit	○	○	—	
didtlmod	○	○	—	
didtlref	○	○	—	
didtlstart	○	○	—	
didtlstop	○	○	—	
didtlterm	○	○	—	
didtrctrl	○	○	—	
didtrdivend	○	○	—	
didtrinit	○	○	—	
didtrmod	○	○	—	
didtrref	○	○	—	
didtrstart	○	○	—	
didtrstop	○	○	—	
didtrterm	○	○	—	
didtsctrl	○	○	—	
didtsdivend	○	○	—	
didtsinit	○	○	—	
didtsmod	○	○	—	
didtsref	○	○	—	
didtsstart	○	○	—	
didtsstop	○	○	—	
didstern	○	○	—	

•    ○ … 使用可能    — … 使用不可    △ … 条件付使用可能

A. 1.5      データ変換・通信オプション

関数名	論理ノード種別			備考
	AP	OLTP	DB	
dadacchg	○	○	—	
dadacinit	○	○	—	
dadacref	○	○	—	
dadacterm	○	○	—	
dadscblock	○	○	○	
dadscinfochg	○	○	○	
dadsclogref	○	○	○	
dadscref	○	○	○	
daolmchkstatus	○	○	—	
daolmininit	○	○	—	
daolmrefstatus	○	○	—	
daolmterm	○	○	—	
daslutamload	—	○	—	
daslutamsave	—	○	—	
daslufltamload	—	○	—	
daslufltamsave	—	○	—	
datrlrefopdata	○	○	—	
datrlupdhash	○	○	—	
datrlupdopdata	○	○	—	

•    ○ … 使用可能    — … 使用不可    △ … 条件付使用可能

D I O S A / X T P V2.1  
コマンドリファレンス

2018 年 2 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, 2017, 2018

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。