



ユーティリティ・ガイド

Adaptive Server[®] Enterprise

15.7 ESD #2

ドキュメント ID : DC32971-01-1572-01

改訂 : 2012 年 7 月

Copyright © 2012 by Sybase, Inc. All rights reserved.

このマニュアルは Sybase ソフトウェアの付属マニュアルであり、新しいマニュアルまたはテクニカル・ノートで特に示されないかぎり、後続のリリースにも付属します。このマニュアルの内容は予告なしに変更されることがあります。このマニュアルに記載されているソフトウェアはライセンス契約に基づいて提供されるものであり、無断で使用することはできません。

このマニュアルの内容を弊社の書面による事前許可を得ずに、電子的、機械的、手作業、光学的、またはその他のいかなる手段によっても、複製、転載、翻訳することを禁じます。

Sybase の商標は、the Sybase trademarks page (<http://www.sybase.com/detail?id=1011207>) で確認できます。Sybase およびこのリストに掲載されている商標は、米国法人 Sybase, Inc. の商標です。® は、米国における登録商標であることを示します。

このマニュアルに記載されている SAP、その他の SAP 製品、サービス、および関連するロゴは、ドイツおよびその他の国における SAP AG の商標または登録商標です。

Java および Java 関連の商標は、米国およびその他の国における Oracle およびその関連会社の商標または登録商標です。

Unicode と Unicode のロゴは、Unicode, Inc. の登録商標です。

IBM および Tivoli は、International Business Machines Corporation の米国およびその他の国における登録商標です。

このマニュアルに記載されている上記以外の社名および製品名は、当該各社の商標または登録商標の場合があります。

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

目次

第 1 章	dataserver を使用したサーバの構築	1
	概要.....	1
	新しいマスタ・デバイスの構築.....	2
	dataserver を使用する場合の環境.....	3
	新しい Adaptive Server を構築する例.....	3
	既存の Adaptive Server の起動.....	7
	ページ・サイズの大きいサーバへのアップグレード.....	7
	現在のサーバ制限値の表示.....	7
第 2 章	コマンド・ラインからの Interactive isql の使用	9
	作業を始める前に.....	9
	isql の起動と停止.....	9
	isql による Transact-SQL の使用方法.....	10
	isql 出力のフォーマット.....	11
	入力の訂正.....	12
	出力を制御する set オプション.....	13
	コマンド・ターミネータの変更.....	13
	コマンド・ターミネータ値とパフォーマンス統計値との関係.....	14
	ネットワーク・パケット・サイズの設定.....	15
	入出力ファイル.....	16
	UNIX コマンド・ラインのリダイレクト.....	16
第 3 章	グラフィック・モードでの Interactive SQL の使用	17
	Interactive SQL の起動.....	18
	主な Interactive SQL ウィンドウ.....	19
	[プラン] ダイアログ・タブ.....	20
	Interactive SQL ツールバー.....	21
	複数のウィンドウのオープン.....	21
	キーボード・ショートカット.....	22
	Interactive SQL を使用したデータの表示.....	23
	Interactive SQL でのテーブル値の編集.....	23
	Interactive SQL の結果セットからローをコピー.....	24
	Interactive SQL の結果セットからテーブル値を編集.....	25

Interactive SQL の結果セットからデータベースへのローの挿入	25
Interactive SQL を使用したデータベースからのローの削除	26
Interactive SQL での SQL 文	27
Interactive SQL コマンドのキャンセル	27
複数の文の結合	28
テーブル、カラム、プロシージャの検索	29
コマンドの再呼び出し	30
コマンドのログ	30
Interactive SQL の設定	31
[一般] ダイアログ・ボックス	31
[結果] ダイアログ・ボックス	33
[インポート/エクスポート] ダイアログ・ボックス	34
[メッセージ] ダイアログ・ボックス	34
[エディタ]	35
[クエリエディタ]	36
コマンド・ファイルの実行	36
ファイルへの出力の書き込み	37
コマンド・ファイルの実行	37
コマンド・ファイルの保存、ロード、実行	38
Interactive SQL での SQL エスケープ構文	39
Interactive SQL コマンド	40
第 4 章 bcp を使用した Adaptive Server とのデータの転送	43
データの転送方法	44
bcp を使用したデータのインポートとエクスポート	44
bcp のモード	46
bcp の要件	47
ファイルへのデータのコピー	48
例	49
bcp のパフォーマンス	49
高速、高速にログを取る、または低速 bcp の使用	50
高速および高速にログを取る bcp を使用したでのコピー・イン	54
分割されたテーブルへのバルク・コピー	54
パラレル・バルク・コピーを使った特定の分割へのデータのコピー	58
暗号化データのバルク・コピー	65
bcp オプションの使用	67
デフォルト・フォーマットの使用	67
コマンド・ラインでのターミネータの変更	69

対話型 bcp のデフォルトの変更	70
bcp プロンプトに対する応答	70
ファイル記憶タイプ	72
プレフィクス長	74
フィールド長	75
フィールド・ターミネータとロー・ターミネータ	78
フォーマット・ファイル	81
bcp フォーマット・ファイルの要素	82
データを対話形式でコピー・アウトする例	85
フィールド長を指定してデータをコピー・アウトする例	86
デリミタを指定してデータをコピー・アウトする例	87
データを対話形式でコピー・インする例	88
フィールド長を指定してデータをコピー・インする例	88
デリミタを指定してデータをコピー・インする例	91
フォーマット・ファイルを指定してデータをコピー・ インする例	91
bcp と代替言語	92
初期化文字列に対するサポート	93
bcp とロー・レベルのアクセス・ルール	93
コピー・インとバッチ・ファイル	94
リカバリ性の向上	94
バッチと分割されたテーブル	95
text または image データのコピー・アウト	96
ネットワーク・パケット・サイズの指定	96
コピー・インとエラー・ファイル	97
コピー・アウトとエラー・ファイル	98
デフォルト、ルール、トリガのデータの整合性	98
デフォルトとデータ型	99
ルールとトリガ	99
bcp と他のユーティリティの相違点	99

第 5 章

dscp の使用	101
dscp について	101
dscp セッションの使用	102
サーバ・エントリの操作	103
サーバ・エントリの追加と変更	103
サーバ・エントリのコピー	106
サーバ・エントリの表示	107
サーバ・エントリの削除	108
dscp の終了	108
dscp ユーティリティ・コマンドのクイック・リファレンス	109

第 6 章	dsedit の使用	111
	dsedit について	111
	dsedit の起動	111
	編集セッションのオープン	113
	サーバ・エントリの追加、表示、編集	116
	Windows でのサーバ・エントリの変更	116
	UNIX プラットフォームでのサーバ・エントリの変更	119
	サーバ・エントリのコピー	122
	dsedit に関するトラブルシューティング	123
第 7 章	ユーティリティ・コマンド・リファレンス	125
	はじめる前に	125
	* r ユーティリティ	128
	ユーティリティ・クイック・リファレンス	128
	インストール・ユーティリティと設定ユーティリティ	129
	言語、文字セット、ソート順の設定ユーティリティ	130
	サーバ起動ユーティリティ	130
	データベース作成ユーティリティと操作ユーティリティ	131
	情報収集ユーティリティ	131
	チューニング・ユーティリティ	132
	クラスタ管理ユーティリティ	132
	backupserver	133
	bcp	139
	buildmaster	158
	certauth	159
	certpk12	163
	certreq	166
	charset	170
	cobpre	171
	cpre	172
	dataserver	173
	ddlgen	183
	defncopy	209
	dscp	215
	dsedit	216
	extractjava	217
	installjava	220
	isql	225
	langinstall	238
	optdiag	241
	preupgrade	249
	pwdcrypt	255
	qrmutil	256
	qptune	262

showserver	269
sqldbgr	270
sqlloc	276
sqllocres	278
sqlsrvr	279
sqlupgrade	287
sqlupgraderes	288
srvbuild	289
srvbuildres	291
startserver	293
sybdiag	296
sybmigrate	303
sybtspasswd	308
xpserver	309

第 8 章

sybcluster を使用したクラスタの管理	311
sybcluster	312
add backupserver	319
add instance	320
connect	322
create backupserver	324
create cluster	326
create xpserver	328
deploy plugin	329
diagnose cluster	331
diagnose instance	333
disconnect	334
drop backupserver	335
drop cluster	336
drop instance	337
drop xpserver	338
exit	339
help	340
localize	341
quit	343
set backupserver	344
set cluster	345
set instance	346
set xpserver port	347
show agents	348
show backupserver config	350
show cluster	351
show instance	354
show membership mode	357

show session	358	
show xpserver	360	
shutdown cluster.....	361	
shutdown instance.....	362	
start cluster	363	
start instance	364	
upgrade server	365	
use.....	368	
第 9 章	マイグレーション・ユーティリティ	369
概要	369	
sybmigrate の利点	369	
sybmigrate でマイグレートできるデータ	370	
sybmigrate でマイグレートできないデータ	372	
作業を始める前に	373	
依存性	374	
インストール	374	
パーミッション	374	
プラットフォーム	375	
環境設定	375	
プロキシ・テーブルのマイグレート	376	
マイグレーション・プロセス	376	
マイグレーション・プロセスの概要	377	
マイグレーション前の考慮事項	377	
パフォーマンスを向上させる設定とチューニング	381	
回避すべきエラー	383	
マイグレーション用従属オブジェクトの自動選択	384	
アーカイブ・データベースのマイグレート	384	
sybmigrate の起動	385	
GUI モード	390	
リソース・ファイル・モード	398	
暗号化カラムに対する sybmigrate の使用	406	
マイグレーション後のアクティビティ	407	
Replication Server ドメインのデータベースのマイグレート	408	
マイグレーション前の手順	408	
マイグレーション後の手順	409	
ワイド・データをサポートするデータベースのマイ グレート	415	
制限事項	416	
トラブルシューティングとエラー・メッセージ	418	
索引	423	

dataserver を使用したサーバの構築

トピック	ページ
概要	1
新しいマスタ・デバイスの構築	2

Adaptive Server[®] では、マスタ・デバイスの構築に `buildmaster` バイナリを使用しません。代わりに、`dataserver` バイナリに `buildmaster` の機能が組み込まれました。この章では、新しいサーバを構築するための `dataserver` の使用方法について説明します。

注意 Windows の `dataserver` バイナリは `sqlsrvr.exe` です。Windows プラットフォームを使用している場合は、この章の `dataserver` への参照をすべて `sqlsrvr` に置き換えてください。

`dataserver` の構文の詳細については、[「dataserver」 \(173 ページ\)](#) を参照してください。`sqlsrvr` の構文の詳細については、[「sqlsrvr」 \(279 ページ\)](#) を参照してください。

概要

`dataserver` コマンドを使用すると、論理ページのサイズが 2K、4K、8K、16K のマスタ・デバイスと `master` データベースを作成できます。論理ページ・サイズを大きくすると、より大きなローを作成でき、1 ページ分の読み込みによってアクセスできるデータの量が増えるので、パフォーマンスが向上します。たとえば、論理ページのサイズが 16K、8K のページは、それぞれ 2K ページの 8 倍、4 倍の量のデータを保持できます。

論理ページ・サイズはサーバ全体の設定であるため、同じサーバの中に、異なった論理ページ・サイズのデータベースを作成することはできません。テーブルのサイズはすべて、ロー・サイズがサーバの現在のページ・サイズを超えないように設定する必要があります。つまり、ローは複数のページにまたがることはできません。

新しいマスタ・デバイスの構築

この項では、`dataserver` ユーティリティを使って、新しいマスタ・デバイスを作成するためのプロセスを説明します。マスタ・デバイスは、`dataserver` の **build** モードを使って構築します。マスタ・デバイスを構築したら、サーバを停止します。次に、**start** モードでサーバを手動で起動してください。これ以降は、マスタ・デバイスを再構築しなくても、必要なときにいつでも **Adaptive Server** を起動、停止、再起動できます。

注意 マスタ・デバイスを構築するときは、`config` ブロック用にさらに 8K 必要であることを考慮してください。

Adaptive Server では次の 3 種類のページ・サイズを使用します。

- 論理ページ・サイズ — データベース・オブジェクトを構築するときのページ。データベースとそれに関連する任意のオブジェクトは、同じ論理ページ・サイズを使用します。論理ページ・サイズには、2K、4K、8K、16K を使用できます。
- 仮想ページ・サイズ — ディスク・レベルでの物理ページの割り付け。常に 2K ページです。ディスク I/O はすべて、仮想ページ・サイズの倍数で実行されます。
- メモリ・ページ・サイズ — **Adaptive Server** 内で割り付け、管理されるメモリ。メモリ・ページ・サイズの単位は常に 2K ページです。

次は、`dataserver` を使って新しいマスタ・デバイスを作成するための構文です。

```
dataserver -d device_name
...
    b [master_device_size [k|K|m|M|g|G]
    [-z logical_page_size [k|K]
-h
```

各パラメータの説明は、次のとおりです。

`-d device_name` — `master` データベースのデバイスのフル・パス名です。`master` データベースのデバイスは、**Adaptive Server** を起動するユーザが書き込み可能である必要があります。デフォルトの `master` データベース・デバイス名は、`d_master` です。

`-b` — `dataserver` がビルド・モードにあり、新しいマスタ・デバイスを作成していることを表します。また、マスタ・デバイスのサイズも表します。デバイスのサイズに単位 (`k`、`m`、`g`) を指定していない場合、`dataserver` はデバイスのサイズを仮想ページ・サイズであると想定します。仮想ページのサイズは常に `2K` です。次に例を示します。

- `-b 51204` — `51,204` 仮想ページ (`100.0078125MB`) のデバイスを指定します。
- `-b 100M` — `100MB` のデバイスを指定します。

`-z` — 論理ページ・サイズを指定します。常に `2K`、`4K`、`8K`、`16K` を指定します。このパラメータは、ビルド・フェーズ間のオプションで、起動モードでは無視されます。ビルド・モードで `-z` パラメータを指定していない場合、マスタ・デバイスは `2K` 論理ページで構築されます。

`-h` — `dataserver` コマンドの構文を出力します。

`dataserver` パラメータとその定義に関する全リストについては、[「`dataserver`」 \(173 ページ\)](#) を参照してください。

`dataserver` を使用する場合の環境

`dataserver` プログラムとともに Adaptive Server を起動した場合、Adaptive Server の実行環境は、次の値によって決定されます。

- `-c configuration_file` に指定された設定ファイル
- デフォルトの設定ファイルである `servername.cfg` (`-c` パラメータを指定しなかった場合)
- デフォルト値 (`-c configuration_file` と `servername.cfg` のどちらも指定しなかった場合)

これらの設定パラメータの詳細については、『システム管理ガイド』の「第 5 章 設定パラメータ」を参照してください。

新しい Adaptive Server を構築する例

新しい Adaptive Server を作成するには、`-b` オプションと `-z` オプションを使用して `dataserver` を発行します。次に例を示します。

- デフォルトの論理ページ・サイズ (`2K`) を使用して `100MB` のマスタ・デバイスを構築し、サーバを起動します。

```
dataserver -d /var/sybase/masterdb.dat -b100M -sMASTER2K
```

- デフォルトの論理ページ・サイズ(4K)を使用して 100MB のマスタ・デバイスを構築します。

```
dataserver -d /var/sybase/masterdb.dat -b100M -z4K -sMASTER4K
```

- サイズが 2K の 102,400 仮想ページのマスタ・デバイスを構築し、8K の論理ページ・サイズを使用するデータベースを作成して、サーバを起動します。

```
dataserver -d /var/sybase/masterdb.dat -b102400 -z8K -sMASTER8K
```

要求された全領域 (102,400 x 2K = 200MB) が、指定した論理ページ・サイズを使用して必要なすべてのシステム・データベースを構築できるだけの領域に満たない場合は、エラー・メッセージが作成され、プロセスは失敗します。

例 次は、`personnel2k` という、2K の論理ページ・サイズを使用して 200MB のデバイスを構築する `dataserver` の出力例です。

```
dataserver -d /var/sybase/personnel2k.dat -b200M -z2k -sPERSONNEL2K
```

ファイルを指定しないと、`dataserver` はデフォルトの設定ファイルを使用します。

```
00:00000:00000:2001/04/16 10:24:31.73 kernel Warning:Using default file
'/var/sybase/PERSONNEL2K.cfg' since a configuration file was not specified.
Specify a configuration file name in the RUNSERVER file to avoid this
message.
```

独自の設定ファイルを指定する場合は、`dataserver` の `-c` パラメータを使用します。詳細については、『システム管理ガイド』の「第 5 章 設定パラメータ」を参照してください。

Adaptive Server では、既存のバージョンの Adaptive Server がインストールされているかどうかに関係なく、インストールはすべてアップグレードとして行います。このため、`dataserver` を実行すると、出力結果は次のようになります。

```
00:00000:00001:2001/04/16 10:24:32.63 server Database 'master' appears to
be at an older revision than the present installation; SQL Server will assess
it, and upgrade it as required.
```

```
00:00000:00001:2001/04/16 10:24:32.66 server Database 'master':beginning
upgrade step [ID 1]:Initialize disk and create empty allocation units
on master device.
```

```
00:00000:00001:2001/04/16 10:24:340.74 server Database 'master':beginning
upgrade step [ID 2]:Bootstrap basic system catalogs in database.
```

dataserer は master データベースの作成を続けます。このデータベースには systypes、sysobjects、sysusages などのテーブルがすべて含まれます。

```
00:00000:00001:2001/04/16 10:24:35.21 server Database 'master':beginning
upgrade step [ID 3]:creating index (table systypes, index ncsystypes)
```

```
00:00000:00001:2001/04/16 10:24:350.36 server Database 'master':beginning
upgrade step [ID 4]:creating index (table sysobjects, index
ncsysobjects)
```

```
00:00000:00001:2001/04/16 10:24:350.44 server Database 'master':beginning
upgrade step [ID 20]:creating table (table sysusages)
```

[...]

dataserer は master データベースを作成すると、次に model データベースを作成します。

[...]

```
00:00000:00001:2001/04/16 10:24:43.14 server Database 'model' appears to
be at an older revision than the present installation; SQL Server will assess
it, and upgrade it as required.
```

```
00:00000:00001:2001/04/16 10:24:43.14 server Database 'model':beginning
upgrade step [ID 1]:Initialize disk and create empty allocation units
on master device.
```

```
00:00000:00001:2001/04/16 10:24:430.83 server Database 'model':beginning
upgrade step [ID 2]:Bootstrap basic system catalogs in database.
```

```
00:00000:00001:2001/04/16 10:24:430.89 server Database 'model':beginning
upgrade step [ID 3]:creating index (table systypes, index ncsystypes)
```

```
00:00000:00001:2001/04/16 10:24:430.91 server Database 'model':beginning
upgrade step [ID 4]:creating index (table sysobjects, index
ncsysobjects)
```

[...]

dataserer は model データベースを作成すると、次に tempdb データベースと sybssystemdb データベースを作成します。

[...]

```
00:00000:00001:2001/04/16 10:24:45.23 server CREATE DATABASE:allocating
1024 logical pages (2.0 megabytes) on disk 'master'.
```

```
00:00000:00001:2001/04/16 10:24:46.79 server Database sybssystemdb
successfully created.
```

```
[...]
```

サーバがデフォルトのソート順を変更して停止すると、**dataserver** は正常に終了します。

```
[...]
```

```
00:00000:00001:2001/04/16 10:24:47.23 server Now loading SQL Server's new
default sort order and character set
```

```
[...]
```

```
00:00000:00001:2001/04/16 10:24:47.31 server Default Sort Order
successfully changed.
```

```
00:00000:00001:2001/04/16 10:24:47.37 server SQL Server shutdown after
verifying System Indexes.
```

```
00:00000:00001:2001/04/16 10:24:47.37 kernel ueshutdown:exiting
```

エラー・メッセージ **dataserver** が正常に終了しないと、マスタ・デバイスのサーバを起動できないため、次のエラー・メッセージが表示されます。

```
00:00000:00001:2001/04/16 19:02:39.53 kernel Use license file
/var/sybase/SYSAM-1_0/licenses/license.dat.
```

```
00:00000:00001:2001/04/16 19:02:39.54 kernel The master device's
configuration area appears to be corrupt.The server needs this data to boot,
and so cannot continue.The server will shut down.
```

共有セグメントを割り付けてサーバを起動できないようにするオプションを記述したユーザ指定の設定ファイルを使用して **dataserver** を実行した場合、**dataserver** の実行は失敗してエラー・メッセージが表示され、マスタ・デバイスのサーバを起動できません。

```
00:00000:00001:2001/04/16 19:04:01.11 kernel Use license file
/var/sybase/SYSAM-1_0/licenses/license.dat.
```

```
00:00000:00000:2001/02/09 19:04:01.25 kernel Using config area from primary
master device.
```

```
00:00000:00001:2001/04/16 19:04:01.36 server The value of the 'max
total_memory' parameter (33792) defined in the configuration file is not high
enough to set the other parameter values specified in the configuration file.
'max total_memory' should be greater than the logical memory '34343'.
```

既存の Adaptive Server の起動

❖ 既存の Adaptive Server の起動

- -b オプションと -z オプションを指定せずに `dataserver` を発行します。

```
dataserver -d /sybase/masterdb.dat
```

ページ・サイズの大きいサーバへのアップグレード

バージョン 12.5 以前の Adaptive Servers では、2K の論理ページ・サイズを使用していました。アップグレードでは、インストールのページ・サイズは変更できません。つまり、現在の Adaptive Server で 2K の論理ページを使用している場合、2K の論理ページを使用している Adaptive Server しかアップグレードできません。

ただし、以前のバージョンの Adaptive Server から 2K の論理ページを使用したデータベースをマイグレートすることはできます。

`sybmigrate` データ・マイグレーション・ツールの使用方法については、[第 9 章「マイグレーション・ユーティリティ」](#)を参照してください。

現在のサーバ制限値の表示

Adaptive Server の制限値に関する情報を表示するには、次の手順に従います。

- `dbcc serverlimits` を実行し、サーバの論理ページ・サイズを出力に含めます。たとえば、次のように入力します。

```
dbcc serverlimits
```

- 文字列 “logical page size” をエラー・ログの中で探します。
- グローバル変数 `@@maxpagesize` に、サーバの論理ページ・サイズが表示されます。isql プロンプトで、次のように入力します。

```
select @@maxpage size
-----
8192
```


コマンド・ラインからの Interactive isql の使用

この章では、コマンド・ライン SQL ユーティリティである isql について説明します。

トピック	ページ
作業を始める前に	9
isql の起動と停止	9
isql による Transact-SQL の使用方法	10
コマンド・ターミネータの変更	13
コマンド・ターミネータ値とパフォーマンス統計値との関係	14
ネットワーク・パケット・サイズの設定	15
入出力ファイル	16

isql の構文の詳細については、「[isql](#) (225 ページ)」を参照してください。

作業を始める前に

Open Client のバージョン 11.1 以降を実行中で、外部の Sybase 設定ファイルを使用している場合は、次の文を設定ファイルに追加すれば isql を使用できるようになります。

```
[isql]
```

isql の起動と停止

❖ isql の起動

- 1 オペレーティング・システムのプロンプトに対して次のコマンドを入力します。

```
isql
```

- 2 プロンプトが表示されたら、パスワードを入力します。

入力したパスワードは画面には表示されません。次のような isql プロンプトが表示されます。

```
1>
```

これで、Transact-SQL コマンドを発行できます。

❖ isql の停止

- 1 行に次のコマンドのいずれかを単独で入力します。

```
quit  
exit
```

isql による Transact-SQL の使用方法

isql は、Transact-SQL コマンドを Adaptive Server に送信し、結果をフォーマットしてから、標準出力に出力します。isql 文には最大サイズの制限はありません。Transact-SQL の使用の詳細については、『Transact-SQL ユーザーズ・ガイド』を参照してください。

注意 オペレーティング・システムから isql ユーティリティ・プログラムを使用して Transact-SQL を直接使用するには、Adaptive Server のアカウント (ログイン名) が必要です。

Transact-SQL コマンドを実行するには、改行してデフォルトのコマンド・ターミネータの “go” を入力してください。

次に例を示します。

```
isql  
Password:  
  
1> use pubs2  
2> go  
1> select *  
2> from authors  
3> where city = "Oakland"  
4> go
```

isql 出力のフォーマット

isql 出力の幅は文字セット拡張または文字幅に基づいて調整され、そこには最大可能バイトの出力カラムが表示されます。たとえば、UTF8 文字セットの場合、各文字には最大で 4 バイト使用されるため、出力カラムの幅は文字数に 4 を掛けたものになります。ただし、出力カラムの幅はカラムの定義値より大きくすることはできず、カラムの幅は次の式によって計算されます。

$$\text{Min}(\text{character_number} \times \text{max_character_width}, \text{column_defined_width})$$

たとえば、カラム `co11` が `varchar(10)` として定義されている場合、`left(col1, 2)` は幅として 8 (1 文字あたり 4 バイト) を返します。`left(col1,5)` は幅つまり 10 を返します。この場合、式は $5 \times 4 = 20$ になりますが、値は定義済みの長さより大きくできません。

表 2-1 では、isql 出力のフォーマットを変更するコマンド・ライン・オプションを説明します。

表 2-1 : isql のフォーマット・オプション

オプション	デフォルト	意味
<code>-h headers</code>	1	カラム見出しの間に出力するローの数
<code>-s colseparator</code>	シングル・スペース	カラム・セパレータ文字を変更
<code>-w columnwidth</code>	80 文字	行の長さを変更
<code>-e</code>		isql に対して発行された各コマンドを出力に含める
<code>-n</code>		番号とプロンプト記号を削除

この例では、クエリの結果は `output` と呼ばれるファイルに格納されます。

```
isql -User_name -Ppassword -Sserver -e -n -o output
use pubs2
go
select *
from authors
where city = "Oakland"
go-
quit
```

`output` の内容を表示するには、次のコマンドを入力します。

- Windows の場合 :


```
type output
```
- UNIX プラットフォームの場合 :

```
cat output

select *
from authors
where city = "Oakland"

au_id      au_lname  au_fname  phone      address
  city      state  country  postalcode
-----
-----
213-46-8915 Green      Marjorie 415 986-7020 309 63rd St. #411
Oakland    CA    USA    94618
274-80-9391 Straight  Dick      415 834-2919 5420 College Av.
Oakland    CA    USA    94609
724-08-9931 Stringer  Dirk      415 843-2991 5420 Telegraph Av.
Oakland    CA    USA    94609
724-80-9391 MacFeather Stearns 415 354-7128 44 Upland Hts.
Oakland    CA    USA    94612
756-30-7391 Karsen   Livia    415 534-9219 5720 McAuley St.
Oakland    CA    USA    94609
```

注意 コマンド・ターミネータは、*output* ファイルには含まれません。

入力の訂正

Transact-SQL コマンドの入力中にミスをした場合は、次のいずれかの方法で訂正してください。

- [Ctrl] キーを押しながら [C] キーを押すか、“reset” という語を 1 行に単独で入力します。これで、クエリ・バッファがクリアされて isql プロンプトに戻ります。
- テキスト・エディタの名前を 1 行に単独で入力します。これで、テキスト・ファイルが開き、クエリを編集できるようになります。ファイルに書き込んで保存すると、isql に戻り、訂正後のクエリが表示されます。“go” と入力してクエリを実行してください。

出力を制御する set オプション

表 2-2 は、Transact-SQL の出力に影響する set オプションのリストです。詳細については、『ASE リファレンス・マニュアル』の「set」を参照してください。

表 2-2 : Transact-SQL の出力に影響する set オプション

set オプション	デフォルト	意味
char_convert	Off	Adaptive Server とクライアント間の文字セット変換のオン/オフを行う。また、サーバの文字セットとクライアントの文字セット間の変換を開始する。
fipsflagger	Off	SQL92 のエントリ・レベルの範囲を超える Transact-SQL 拡張機能が使用されたときに警告する。このオプションをオンにしても、SQL 拡張機能は無効にならない。ANSI SQL 以外のコマンドを発行すると、処理は完了する。
flushmessage	Off	生成されたメッセージを送信する。
language	us_english	システム・メッセージの言語を設定する。
nocount	Off	影響されたローの数をレポートしないようにする。
noexec	Off	各クエリをコンパイルするが、実行はしない。showplan とともに使用されることが多い。
parseonly	Off	クエリのコンパイルおよび実行を行わずに、クエリの構文をチェックし、エラー・メッセージを返す。
showplan	Off	クエリの処理プランの記述を生成する。ストアド・プロシージャまたはトリガの中で使用するときは結果を出力しない。
statistics io statistics time	Off	各実行のあとに、パフォーマンスの統計値を表示する。
statistics subquerycache	Off	キャッシュのヒット数、ミス数、各サブクエリのサブクエリ・キャッシュ内のローの数を表示する。
textsize	32K	返される text データまたは image データのバイト数を制御する。

コマンド・ターミナータの変更

コマンド・ターミナータ引数 (-c) を指定すると、ユーザ独自のターミナータ記号を選択できます。go は、このオプションのデフォルト値です。コマンド・ターミナータは必ず、前にブランクやタブを入れずに、左づめで入力してください。

たとえば、ピリオドをコマンド・ターミナータとして使用するには、次のように isql を呼び出します。

```
isql -c.
```

このコマンド・ターミネータを使用した `isql` セッションの例を次に示します。

```
1> select name from sysusers
2> .
name
-----
sandy
kim
leslie
(3 rows affected)
```

スクリプトで `isql` コマンド・ターミネータ・オプションを使用するには、あらかじめ次の計画をしておく必要があります。

- `installmaster` などの Adaptive Server が提供するスクリプトでは、“go” を使用します。これらのスクリプトを使用するセッションでは、コマンド・ターミネータを変更しないでください。
- ユーザ自身のスクリプトにもすでに “go” が入っている可能性があります。別のコマンド・ターミネータを使用する場合は、スクリプト中のコマンド・ターミネータも更新してください。

コマンド・ターミネータ値とパフォーマンス統計値との関係

`isql` には、パフォーマンス統計値オプション (-p) があります。

たとえば、この構文では次のような統計値が返されます。

```
isql -p
1> select * from sysobjects
2> go

Execution Time (ms.):1000   Clock Time (ms.):1000
1 xact:
```

これは、1つのトランザクションに 1000 ミリ秒かかったことを表しています。クロック・タイム値はトランザクション全体を反映していません。つまり、Client-Library™ がクエリを構築したときから、Client-Library が Adaptive Server からの情報を返すまでの時間です。

1つまたは複数のトランザクションの実行に基づいてパフォーマンスの統計値を収集できます。複数のトランザクションについて統計値を収集する場合には、コマンド・ターミネータのあとに数字を指定してください。

たとえば、次のコマンドは `select *` トランザクションを3回実行して、パフォーマンス統計値をレポートするように、Adaptive Server に指示します。

```
isql -p
1> select * from sysobjects
2> go 3

Execution Time (ms.):1000    Clock Time (ms.):1000
Execution Time (ms.):1000    Clock Time (ms.):2000
Execution Time (ms.):1000    Clock Time (ms.):1000

Execution Time (ms.):1000    Clock Time (ms.):4000
3>xact:
```

ネットワーク・パケット・サイズの設定

適正なネットワーク・パケット・サイズを設定することにより、Adaptive Server のパフォーマンスを飛躍的に向上させることができます。

`-A size` オプションは、isql セッションで使用するネットワーク・パケット・サイズを指定します。デフォルト値は 2048 バイトです。現在の isql セッションのパケット・サイズを 4096 バイトに設定するには、次のように入力します。

```
isql -A 4096
```

ネットワーク・パケット・サイズをチェックするには、次のように入力してください。

```
select * from sysprocesses
```

この isql セッションのパケット・サイズ値は、`sysprocesses` テーブルの中の `network_pktsz` という見出しの下に表示されます。

ネットワーク・パケット・サイズの設定の詳細については、『システム管理ガイド』を参照してください。

入出力ファイル

`-i`および`-o`オプションで、コマンド・ラインに入出力ファイルを指定できます。

`isql`には、出力のためのフォーマット・オプションはありません。ただし、`-n`オプションを使用すると、`isql`プロンプトを削除できます。また、他のツールを使用して出力を再フォーマットすることもできます。

`-e`オプションを指定すると、`isql`は入力内容を出力にエコーします。結果の出力ファイルには、クエリとその結果の両方が含まれます。

UNIX コマンド・ラインのリダイレクト

UNIXのリダイレクト記号“<”と“>”の機能は、次のように`-i`オプションと`-o`オプションの機能に似ています。

```
isql -Usa < input > output
```

次のように、`isql`をダイレクトして端末から入力を取得できます。

```
isql -Usa -Ppassword -Sserver_name << EOF > output
use pubs2
go
select * from table
go
EOF
```

“<<EOF”は、“EOF”文字列が来るまで端末から入力を取得するように`isql`に指示します。“EOF”は任意の文字列に置き換えることができます。同様に、次の例では[Ctrl]キーを押しながら[D]キーを押すことによって入力の終了を示しています。

```
isql -Usa << > output
```


グラフィック・モードでの Interactive SQL の使用

この章では、GUI ベースの Interactive SQL ユーティリティを実行し、使用方法について説明します。

トピック	ページ
Interactive SQL の起動	18
主な Interactive SQL ウィンドウ	19
Interactive SQL ツールバー	21
複数のウィンドウのオープン	21
キーボード・ショートカット	22
Interactive SQL を使用したデータの表示	23
Interactive SQL でのテーブル値の編集	23
Interactive SQL での SQL 文	27
Interactive SQL の設定	31
コマンド・ファイルの実行	36
Interactive SQL での SQL エスケープ構文	39
Interactive SQL コマンド	40

Interactive SQL を使用すると、サーバに対して SQL 文の実行、スクリプトの作成、データベースのデータの表示ができます。Interactive SQL は、次の目的で使用できます。

- データベース内の情報をブラウズする。
- アプリケーションで使用することを予定している SQL 文をテストする。
- データをデータベースにロードし、管理作業を実行する。

また、Interactive SQL では、コマンド・ファイルまたはスクリプト・ファイルを実行することもできます。たとえば、データベースに対して実行する繰り返し可能なスクリプトを作成し、Interactive SQL を使用してそれらのスクリプトをバッチで実行できます。

Interactive SQL の起動

[Interactive SQL を開く] メニュー項目は、サーバへの接続をオープンします。ただし、サーバに対してこのメニュー項目を選択すると、Interactive SQL は、そのサーバのデフォルト・データベースへの接続をオープンします。[Interactive SQL を開く] メニューから特定のデータベースを選択すると、選択したデータベースに対して Interactive SQL がオープンします。

❖ Sybase Central から Interactive SQL を起動する

- 次のいずれかの方法を使用できます。
 - Sybase Central でデータベースを選択し、[ファイル]-[Interactive SQL を開く] を選択します。Interactive SQL がデータベースに接続します。また、データベースを右クリックし、[Interactive SQL を開く] を選択することもできます。
 - サーバに接続せずに Interactive SQL を起動するには、[ツール]-[Adaptive Server Enterprise] - [Interactive SQL を開く] を選択します。[接続] ダイアログが表示されます。

❖ コマンド・ラインから Interactive SQL を起動する

コマンド・ラインから Interactive SQL を起動する手順は、使用するオペレーティング・システムによって異なります。

Interactive SQL を単独で起動すると、[接続] ダイアログが表示され、Sybase Central の場合と同じ方法でデータベースに接続できます。

- UNIX では、Sybase Central と Interactive SQL を $$$SYBASE$ にインストールする必要はありません (これらを起動するために、 $$$SYBASE$ が存在する必要もありません)。Sybase Central と Interactive SQL は、 $$$SYBROOT$ (インストーラによって設定された環境変数) にインストールされます。 $$$SYBROOT$ に移動し、次のように入力します。

```
dbisql
```

Windows では、 $%SYBROOT$ ディレクトリに変更し、次のように入力します。

```
dbisql.bat
```

- [接続] ダイアログで、[接続] ダイアログ・ボックスにデータベースへの接続情報を入力し、[OK] をクリックします。

新しい Interactive SQL ウィンドウを開くには、次の手順に従います。

- 1 [ウィンドウ]-[新しいウィンドウ]を選択します。[接続]ダイアログが表示されます。
- 2 [接続]ダイアログで接続オプションを入力し、[OK]をクリックして接続します。

接続情報 (データベース名、ユーザ ID、データベース・サーバなど)は、[SQL 文] ウィンドウ枠の上にあるタイトル・バーに表示されます。

また、SQL メニューの [接続] コマンドと [切断] コマンドを使用するか、[SQL 文] ウィンドウ枠で `connect` 文または `disconnect` 文を実行することにより、データベースに接続したり接続を切断したりできます。

主な Interactive SQL ウィンドウ

Interactive SQL ウィンドウには、次のウィンドウ枠があります。

- [SQL 文] – SQL 文を入力するための領域です。
- [結果] – 実行したコマンドの結果が表示されます。たとえば、SQL 文を使用して、データベース内の特定のデータを検索すると、このウィンドウ枠の [結果] タブに検索条件に一致するカラムとローが表示されます。情報がウィンドウ枠に収まらない場合は、スクロール・バーが自動的に表示されます。[結果] タブに表示された結果セットは編集できます。
- [メッセージ] – データベース・サーバからのメッセージが表示されます。
- [プラン] – SQL 文に関するクエリ・オプティマイザの実行プランが表示されます。詳細については、下記の [\[プラン\]ダイアログ・タブ](#)を参照してください。

ウィンドウのタイトルには接続名が表示されます。Adaptive Server の場合、接続名はサーバ名 (サーバの `interfaces` ファイルのエントリによって指定)、または接続時にユーザが入力したホスト名とポート番号です。

[プラン] ダイアログ・タブ

[プラン] タブには、現在実行中の SQL テキストに関する実行エンジンのプランを GUI で表現したものが表示されます。これは、現在実行中のクエリのパフォーマンスと統計の特性を把握するのに役立ちます。

注意 [プラン] タブが表示されるのは、Adaptive Server バージョン 15.0 以降に接続した場合だけです。

[プラン] タブの上半分には、プランで使用する演算子の論理フローがツリーベースの階層構造で表示されます。各演算子はツリーの個々のノードとして示されます。各演算子のコストは、クエリ・プロセッサが使用するコスト・モデルに基づいています。ツリー内の各演算子ノードのコストは、他のノードに対して相対的に計算されるため、演算子をそのコストに基づいて特定しやすくなります。

各ノードには、各演算子の詳細を示すツールチップ・テキスト (マウスをノード上に移動したときに表示されるテキスト) が含まれているため、演算子の詳細を比較するときにノードを選択する必要はありません。

[プラン] タブには次のタブがあります。

- [詳細] – 次の 2 つのテーブルに演算子の統計情報の詳細が表示されます。
 - [Node Statistics] – すべての演算子の情報が表示されます。ロー・カウント、論理 I/O、物理 I/O などの統計情報が含まれます。
 - [サブツリーの統計] – 下にすべての演算子の総計が表示され、リーフ演算子以外の演算子の情報が示されます。ロー・カウント、論理 I/O、物理 I/O に関する統計情報が含まれます。
- [XML] – 結果セットが XML 出力として表示されます。
- [テキスト] – クエリ・プランのテキスト・バージョン (showplan の出力内容と同じもの) が表示されます。
- [詳細] – 次の内容が含まれます。
 - [抽象クエリ・プラン] – クエリで使用する抽象クエリ・プランが表示されます。

- [リソースの使用率]— スレッドの数や補助セッション記述子 (SDES。スキャンを追跡するために、テーブル・スキャンごとにセッション記述子が1つ必要) など、プランで使用するリソースについて説明します。
- [コスト]— 論理 I/O、物理 I/O、CPU 使用率など、プランに関連するコストがリストされます。
- [オプティマイザ・メソッド]— クエリの実行時間、最初のプランの実行時間、評価対象のプランの数、有効だったプランの数、使用されたプロシージャ・キャッシュの量など、クエリ・プランの統計情報がリストされます。
- [オプティマイザの統計]— テーブルに対して `update statistics` を最後に実行した時間、欠落したヒストグラム・ステップ、ステップの密度がリストされます。

Interactive SQL ツールバー

Interactive SQL ツールバーは、[Interactive SQL] ウィンドウの上部に表示されます。このツールバーのボタンを使用して、次の操作を実行できます。

- 履歴リストの現在位置の直前にある実行済みの SQL 文を呼び出す。
- これまでに実行した SQL 文を最大 50 個までリストして表示する。
- 履歴リストの現在位置の直後にある実行済みの SQL 文を呼び出す。
- [SQL 文] ウィンドウ枠に現在表示されている SQL 文を実行する。
- 現在の SQL 文の実行に割り込む。

複数のウィンドウのオープン

複数の Interactive SQL ウィンドウを開くことができます。各ウィンドウは、個々の接続に対応しています。異なるサーバ上にある 2 つ以上のデータベースに同時に接続したり、1 つのデータベースへの同時接続をオープンしたりできます。

キーボード・ショートカット

Interactive SQL には、以下のキーボード・ショートカットがあります。

ファンクション	説明
Alt + F4	Interactive SQL を終了します。
Alt + 左矢印	履歴リストにある前の SQL 文を表示します。
Alt + 右矢印	履歴リストにある次の SQL 文を表示します。
Ctrl + C	選択したローとカラム見出しをクリップボードにコピーします。
Ctrl + End	現在のウィンドウ枠の最下部に移動します。
Ctrl + F6	開いている Interactive SQL ウィンドウ間を移動します。
Ctrl + H	現在のセッションで実行した SQL 文の履歴を表示します。
Ctrl + Home	現在のウィンドウ枠の最上部に移動します。
Ctrl + N	Interactive SQL ウィンドウの内容をクリアします。
Ctrl + Q	クエリ・エディタを表示します。クエリ・エディタを使用して SQL クエリを作成できます。クエリの作成が終わったら、[OK] をクリックして [SQL 文] ウィンドウ枠にエクスポートします。
Ctrl + S	[SQL 文] ウィンドウ枠の内容を保存します。
Shift + F5	[SQL 文] ウィンドウ枠内の文を実行しないでプランをリフレッシュします。これにより、テーブル・データを変更しないで文のプランを確認できます。
Esc	[SQL 文] ウィンドウ枠をクリアします。
F2	結果セット内の選択したローを編集できます。[Tab] キーを使用してローのカラム間を移動できます。
F5	[SQL 文] ウィンドウ枠内のすべてのテキストを実行します。この操作は、ツールバーの [Execute SQL Statement] ボタンをクリックして実行することもできます。
F7	[テーブル名のルックアップ] ダイアログを表示します。このダイアログでテーブルを検索して選択し、[Enter] キーを押すと、そのテーブル名を [SQL 文] ウィンドウ枠のカーソル位置に挿入できます。また、リストでテーブルを選択し、もう一度 [F7] キーを押すと、そのテーブルのカラムが表示されます。カラムが表示されたら、カラムを選択して [Enter] キーを押すと、そのカラム名を [SQL 文] ウィンドウ枠のカーソル位置に挿入できます。
F8	[プロシージャ名のルックアップ] ダイアログを表示します。このダイアログでプロシージャを検索して選択し、[Enter] キーを押すと、そのプロシージャ名を [SQL 文] ウィンドウ枠のカーソル位置に挿入できます。
F9	[SQL 文] ウィンドウ枠で選択したテキストを実行します。テキストが選択されていない場合は、すべての文が実行されます。
Page down	現在のウィンドウ枠で下へ移動します。
Page up	現在のウィンドウ枠で上へ移動します。

Interactive SQL を使用したデータの表示

Interactive SQL を使用して、データベース内の情報をブラウザできます。

データベース情報は、Interactive SQL の `select` 文を使用して表示できます。文を入力したら、ツールバーの [Execute SQL Statement] ボタンをクリックします。

文を実行すると、[結果] ウィンドウ枠に結果セットが表示されます。テーブルの領域でウィンドウ枠に現在表示されていない領域を表示するには、スクロール・バーを使用します。

❖ **authors** テーブルのすべてのカラムとローをリストする

1 Interactive SQL を起動し、pubs2 データベースに接続します。

2 [SQL 文] ウィンドウ枠で、次のように入力します。

```
select * from authors
```

3 ツールバーの [Execute SQL Statement] ボタンをクリックします。

結果セット内のローを追加、削除、更新できます。

Interactive SQL でのテーブル値の編集

Interactive SQL でクエリを実行した後、結果セットを編集してデータベースを修正できます。また、結果セットからローを選択し、他のアプリケーションで使用できるようにコピーすることもできます。

Interactive SQL では、ローの編集、挿入、削除をサポートしています。これらの操作の結果は、`update`、`insert`、`delete` の各文を実行した場合と同じです。

Interactive SQL でクエリを実行し、[結果] ウィンドウ枠の [結果] タブに結果セットを返してから、ローをコピー、編集、挿入、削除します。結果セットを直接編集すると、Interactive SQL は、データベース・テーブルにその変更を加えるための SQL 文を作成し、実行します。

結果セット内のローまたは値を編集するには、値を修正するテーブルまたはカラムに対する適切なパーミッションが必要です。たとえば、ローを削除するには、そのローが属するテーブルに対する `delete` パーミッションが必要です。

次のような場合には、結果セットの編集に失敗することがあります。

- パーミッションを持っていないローまたはカラムを編集しようとした場合
- プライマリ・キーを持つテーブルからカラムを選択したが、プライマリ・キー・カラムの一部しか選択していない場合
- join の結果セットを編集しようとした場合 (たとえば、結果セットに複数のテーブルのデータがある場合)
- 無効な値を入力した場合 (たとえば、数値カラムに文字列を入力したり、NULL を使用できないカラムに NULL を入力したりした場合)

編集に失敗すると、エラーを説明する Interactive SQL エラー・メッセージが表示され、データベース・テーブルの値は変更されません。

テーブル値を変更したら、commit 文を入力して変更を永続的にします。変更を取り消すには、rollback 文を実行します。

Interactive SQL の結果セットからローをコピー

Interactive SQL の結果セットからローを直接コピーし、他のアプリケーションにペーストできます。ローをコピーすると、カラム見出しもコピーされます。コピーしたデータはカンマで区切られているため、Microsoft Excel などの他のアプリケーションで適切にフォーマットできます。デフォルトでは、コピーしたデータは ASCII フォーマットであり、すべての文字列は一重引用符で囲まれます。結果セット内で選択できるのは連続するローだけです。

❖ Interactive SQL の結果セットからのローのコピー

- 1 コピーするローを選択します。ローを選択するには、次の手順に従います。
 - [Shift] キーを押しながらローをクリックします。
 - [Shift] キーを押しながら、上矢印キーまたは下矢印キーを押します。
- 2 結果セットを右クリックし、ポップアップ・メニューから [コピー] を選択します。また、[Ctrl] キーを押しながら [C] キーを押して、選択したローをコピーすることもできます。

カラム見出しも含め、選択したローがクリップボードにコピーされます。[編集]-[貼り付け]を選択するか、[Ctrl] キーを押しながら [V] キーを押して、他のアプリケーションにペーストします。

Interactive SQL の結果セットからテーブル値を編集

[編集] コマンドを使用すると、ローの個々の値を変更できます。データベース・テーブルに存在するローの一部の値、またはすべての値を変更できます。修正するカラムに対する `update` パーミッションが必要です。結果セットを編集する場合、一度に変更できるのは1つのローの値だけです。

❖ 結果セット内のローを編集する

- 1 編集するローを選択します。
- 2 結果セットを右クリックし、ポップアップ・メニューから [編集] を選択します。[F2] キーを押して結果セットを編集することもできます。
- 3 ローの最初の値に点滅するカーソルが表示されます。
- 4 ローのカラム間でカーソルを移動するには、[Tab] キーを押します。選択したローの値をクリックして、値を編集することもできます。
- 5 新しい値を入力します。
- 6 カラムに無効なデータ型を入力することはできません。たとえば、`int` データ型用に設定されたカラムには、`string` データ型は入力できません。
- 7 `commit` 文を実行して、テーブルへの変更を永続的にします。

Interactive SQL の結果セットからデータベースへのローの挿入

[挿入] コマンドを使用すると、データベース・テーブルに新しいブランクのローが追加されます。[Tab] キーを使用して結果セット内のカラム間を移動し、ローに値を追加します。テーブルに値を追加すると、入力した文字は大文字と小文字が区別されて格納されます。新しいローを追加するには、そのテーブルに対する `insert` パーミッションが必要です (`insert` パーミッションの詳細については、『リファレンス・マニュアル：コマンド』の「`insert`」を参照してください)。

❖ 結果セットに新しいローを挿入する

- 1 結果セットを右クリックし、ポップアップ・メニューから [追加] を選択します。

結果セットに新しいブランクのローが表示され、ローの最初の値に点滅するカーソルが表示されます。

ローのカラム間でカーソルを移動するには、[Tab] キーを押します。選択したローの適切なフィールドをクリックして、値を挿入することもできます。

- 2 新しい値を入力します。

カラムに無効なデータ型を入力することはできません。たとえば、int データ型を受け入れるカラムには、string データ型は入力できません。

- 3 commit 文を実行して、テーブルへの変更を永続的にします。

Interactive SQL を使用したデータベースからのローの削除

[削除] コマンドを使用すると、データベース・テーブルから選択したローが削除されます。ローを削除するには、そのテーブルに対する delete パーミッションが必要です。

❖ 結果セットからのローの削除

- 1 削除するローを選択します。ローを選択するには、次の手順に従います。

- [Shift] キーを押しながらローをクリックします。
- [Shift] キーを押しながら、上矢印キーまたは下矢印キーを押します。

連続していないローを削除する場合は、各ローを個別に削除します。

- 2 結果セットを右クリックし、ポップアップ・メニューから [削除] を選択します。また、[Delete] キーを押して選択したローを削除することもできます。

選択したローがデータベース・テーブルから削除されます。

- 3 commit 文を実行して、テーブルへの変更を永続的にします。

Interactive SQL での SQL 文

すべての SQL 文は、Interactive SQL ウィンドウの上部のウィンドウ枠にコマンドとして入力できます。入力が終わったら、その文を実行します。

❖ SQL 文の実行

1 [すべての SQL 文の実行] ボタンをクリックする。

2 [F5] キーを押す。

[SQL 文] ウィンドウ枠をクリアするには、[編集]-[SQL のクリア] を選択するか、[Esc] キーを押します。

注意 長時間実行クエリを実行している場合は、Interactive SQL によってスプラッシュ画面が表示され、いくつかの診断ヒントが示されます。

Interactive SQL コマンドのキャンセル

コマンドをキャンセルするには、Interactive SQL ツールバーの [SQL 文の中断] ボタンを使用します。

停止操作を実行すると、現在の処理が停止し、次のコマンドを入力するためのプロンプトが表示されます。コマンド・ファイルを処理していた場合は、実行するアクション(コマンド・ファイルの停止、継続、または Interactive SQL の終了)を指定するためのプロンプトが表示されます。これらのアクションは、Interactive SQL の ON_ERROR オプションによって制御できます。

割り込みが検出されると、検出時の状況に応じて 3 種類のエラーのいずれかがレポートされます。割り込みが検出された場合、次のようになります。

- (データベース・サーバではなく) Interactive SQL が要求を処理しているときには、次のメッセージが表示されます。

```
ISQL command terminated by user
```

Interactive SQL はすぐに処理を停止し、現在のデータベース・トランザクションは更新されません。

- データベース・サーバがデータ操作コマンド(select、insert、delete、または update) を処理しているときには、次のメッセージが表示されます。

Statement interrupted by user.

現在のコマンドの効果は未完了のままになりますが、トランザクションの残りの部分は元の状態のまま残ります。

- データベース・サーバがデータ定義コマンド (*create object*、*drop object*、*alter object* など) を処理しているときには、次のメッセージが表示されます。

Terminated by user -- transaction rolled back

データ定義コマンドはすべて、コマンドが起動する前に自動的に **commit** を実行するため、**rollback** は現在のコマンドをキャンセルするだけです。

このメッセージは、データベースを修正するコマンド (*insert*、*update*、*delete*) を実行するバルク・オペレーション・モードでデータベース・サーバが実行されているときにも表示されます。この場合、**rollback** は、現在のコマンドだけでなく、最後の **commit** 以降に実行されたすべての操作をキャンセルします。場合によっては、データベース・サーバが自動 **rollback** を実行するときに、かなりの時間がかかることもあります。

複数の文の結合

Interactive SQL 環境では、複数の文を一度に入力できます。各文の末尾に Transact-SQL コマンドの **go** を入力します。

- 1 [SQL 文] ウィンドウ枠に、次のように **go** で区切って複数の文を入力します。

```
update titles
set price = 21.95
where pub_id = "1389"
go
update titles
set price = price + 2.05
where pub_id = "0736"
go
update titles
set price = price+2.0
where pub_id = "0877"
go
```

- 2 ツールバーの [すべての SQL 文の実行] ボタンをクリックします (または [F9] キーを押します)。3つの文がすべて実行されます。実行後も、コマンドは [SQL 文] ウィンドウ枠に保持されます。このウィンドウ枠をクリアするには、[Esc] キーを押します。

rollback と入力し、文を実行すると、変更をロールバックできます。

テーブル、カラム、プロシージャの検索

注意 この項で説明する機能を使用するには、jConnect for JDBC がインストールされている必要があります。

[ツール]-[テーブル名のルックアップ]または[ツール]-[プロシージャ名のルックアップ]の各ユーティリティを選択して、現在のデータベースに格納されたテーブル、カラム、またはプロシージャの名前を検索し、カーソル位置に挿入します。

[テーブル名のルックアップ]ダイアログと[プロシージャ名のルックアップ]ダイアログで、テーブル、カラム、またはプロシージャの最初の何文字かを入力します。これで、入力したテキストで始まる項目だけを表示するようにリストが限定されます。

SQL の標準ワイルドカード文字である % を使用すると、「任意の文字」を指定できます。すべての項目を表示するには、検索領域をクリアします。

❖ データベース内のテーブル名を検索する

- 1 [ツール]-[テーブル名のルックアップ]を選択します。
- 2 テーブルを検索し、選択します。
- 3 [OK] をクリックして、テーブル名を [SQL 文] ウィンドウ枠に挿入します。

❖ データベース内のカラム名を検索する

- 1 [ツール]-[テーブル名のルックアップ]を選択します。
- 2 カラムを含むテーブルを検索し、選択します。
- 3 [カラムを表示] をクリックします。
- 4 カラムを選択し、[OK] をクリックして、カラム名を [SQL 文] ウィンドウ枠に挿入します。

❖ データベース内のプロシージャ名を検索する

- 1 [ツール]-[プロシージャ名のルックアップ]を選択します。
- 2 プロシージャを検索し、選択します。
- 3 [OK]をクリックして、プロシージャ名を[SQL 文]ウィンドウ枠に挿入します。

コマンドの再呼び出し

コマンドを実行すると、Interactive SQL は自動的にそのコマンドを履歴リストに保存し、現在のセッションが終わるまで保持します。Interactive SQL は、最近使用したコマンドの記録を 50 個まで保持します。[コマンド履歴]ダイアログで、コマンドのリスト全体を表示できます。

[コマンド履歴]ダイアログを表示するには、[Ctrl] キーを押しながら[H] キーを押すか、ツールバーのブック・アイコンを選択します。

最後に使用したコマンドは、リストの一番下に表示されます。コマンドを呼び出すには、コマンドを強調表示にし、[OK] をクリックします。[SQL 文]ウィンドウ枠にコマンドが表示されます。

[コマンド履歴]ダイアログを使用しないでコマンドを再呼び出しすることもできます。ツールバーの矢印を使用して前方または後方のコマンドまでスクロールするか、[Alt] キーを押しながら右矢印キーまたは左矢印キーを押します。

コマンドをテキスト・ファイルに保存し、後続の Interactive SQL セッションで使用できます。

コマンドのログ

Interactive SQL のロギング機能を使用すると、コマンドを実行したとおりに記録できます。Interactive SQL では、ロギング・プロセスを停止するか、現在のセッションを終了するまで記録を続けます。記録されたコマンドは、ログ・ファイルに保管されます。

❖ Interactive SQL コマンドのロギングを開始する

- 1 [SQL] [ロギングの開始]を選択します。
- 2 [保存]ダイアログで、ログ・ファイルの場所と名前を指定します。
- 3 指定したら、[保存]をクリックします。

Interactive SQL コマンドのロギングを停止するには、[SQL]-[ロギングの停止]を選択します。

[SQL 文] ウィンドウ枠に入力してロギングを開始するには、`start logging "c:¥file_name.sql"` と入力し、実行します。

`c:¥file_name.sql` は、ログ・ファイルのパス、名前、拡張子です。ログ・ファイルには、`.sql` 拡張子が必要です。一重引用符を使用するのは、パスに埋め込みスペースが含まれる場合だけです。ロギングを停止するには、`stop logging` と入力し、実行します。

注意 `start logging` コマンドと `stop logging` コマンドは、Transact-SQL コマンドではありません。これらのコマンドは、[Interactive SQL] ダイアログ・ボックス以外では、Adaptive Server でサポートされていません。

ロギングを開始すると、正常に実行されなかったものも含め、実行を試みたすべてのコマンドがログに記録されます。

Interactive SQL の設定

Interactive SQL は、[オプション] ダイアログで設定できます。このダイアログでは、コマンド、外観、インポート/エクスポート機能、メッセージを設定できます。

オプションを選択したら、[OK] または [Make Permanent] を選択します。[Make Permanent] を選択すると、選択した設定で Interactive SQL が起動します。

各オプションは、GUI を使用するか、`set option` 文を使用して設定できます。

[オプション] ダイアログを表示するには、[ツール]-[オプション] を選択します。

[一般] ダイアログ・ボックス

[一般] ダイアログ・ボックスは、次の要素で構成されます。

- [コミット] — トランザクションをいつコミットするかを選択します。トランザクションを自動的にコミットできるのは、各文を実行した後か、Interactive SQL セッションを終了したときだけです。

また、適切な時点で `commit` コマンドを明示的に入力することによって、手動でコミットすることもできます。デフォルトの動作では、Interactive SQL の終了時にトランザクションがコミットされます。

- [コマンド・ファイル] – エラー発生時の Interactive SQL の動作を指定します。エラーが発生したときの動作は次のとおりです。
 - [継続] – Interactive SQL は [結果] ウィンドウ枠にエラー・メッセージを表示しますが、終了しません。問題を解決した後、コマンドを再発行してください。
 - [終了] – エラーが発生すると、Interactive SQL は終了します。
 - [通知して続行] – Interactive SQL はダイアログ・ボックスにエラー・メッセージを表示し、エラーについて説明しますが、終了しません。
 - [通知して終了] – Interactive SQL はダイアログ・ボックスにエラー・メッセージを表示してエラーについて説明し、終了します。
 - [通知して停止] – Interactive SQL はエラー・メッセージを表示し、エラーについて説明します。
 - [プロンプト] – デフォルト設定です。Interactive SQL は、継続するかどうかをユーザに尋ねるメッセージ・ボックスを表示します。
 - [停止] – Interactive SQL は、[結果] ウィンドウ枠にエラー・メッセージを表示します。問題を解決した後、コマンドを再発行してください。

[コマンド・ファイルをログに出力] – ロギングを有効にしている場合、このオプションを選択すると、対話形式で入力された SQL 文とともに、スクリプト・ファイル (またはコマンド・ファイル) から実行された SQL 文がログにコピーされます。このオプションを無効にすると、ロギングを開始したときに、対話形式で入力した SQL 文だけがログにコピーされます。

フォルダ

ブラウザがファイルの検索を開始するディレクトリを指定します。`[Last folder used]` または `[Current folder]` のいずれかを選択します。

[結果] ダイアログ・ボックス

[結果] ダイアログ・ボックスは、次の要素で構成されます。

- [NULL 値の代替文字] — データをブラウズするときに、テーブル・カラムに null をどのように表示するかを指定します。デフォルト設定は (NULL) です。
- [表示できるローの最大数] — 表示するローの数を制限します。デフォルト設定は 500 です。
- [トランケーションの長さ] — Interactive SQL の [結果] ウィンドウ枠の各カラムに表示する文字数を制限します。デフォルト設定は 30 です。
- [Show multiple result sets] — 複数の結果セットの表示を有効／無効にします。たとえば、複数の select 文を含むプロシージャを作成したときに、この機能を使用できます。このオプションを有効にすると、プロシージャの呼び出し時に [結果] ウィンドウ枠の個別のタブに各結果セットが表示されます。

jConnect ドライバを使用している場合に複数の結果セットを表示することを選択すると、Interactive SQL は結果セット全体を取得するまで待ってからローを表示する必要があります。このため、大量の結果セットの場合、待機時間が長くなることがあります。このオプションは、デフォルトではオフになっています。

- [ロー数の表示] — 結果セットにロー数を表示する場合はオンにします。
- [結果の自動再フェッチ] — insert 文、update 文、または delete 文の実行後に、最新の結果セットを自動的に再生成する Interactive SQL の機能を有効または無効にします。たとえば、[結果] ウィンドウ枠の [結果] タブに更新の影響を受けるローが表示されているときにテーブルを更新する場合、このオプションを選択すると、Interactive SQL は [結果] タブを自動的にリフレッシュして実行された変更を反映します。このオプションは、デフォルトではオンになっています。
- [コンソール・モード] — コンソールに結果セットをどのように表示するか (最新の結果セットだけを表示する、すべての結果セットを表示する、または結果セットを表示しない) を選択します。
- [フォント] — 結果セットに使用するフォントを選択します。

[インポート／エクスポート] ダイアログ・ボックス

[インポート／エクスポート] ダイアログ・ボックスは、次の要素で構成されます。

- [デフォルトのエクスポートフォーマット] – エクスポート時に使用するデフォルト・ファイル・フォーマットを選択します。このフォーマットは、[保存] ダイアログの [ファイルの種類] フィールドで自動的に選択されますが、他のフォーマットを選択することもできます。フォーマットが指定されていない場合に、Interactive SQL が output 文を解釈するときにも、このデフォルトが使用されます。デフォルト設定は ASCII です。
- [デフォルトのインポートフォーマット] – インポート時に使用するデフォルト・ファイル・フォーマットを選択します。このフォーマットは、[開く] ダイアログの [ファイルの種類] フィールドで自動的に選択されますが、他のフォーマットを選択することもできます。フォーマットが指定されていない場合に、Interactive SQL が input 文を解釈するときにも、このデフォルトが使用されます。デフォルト設定は ASCII です。
- [ASCIIOptions] – ASCII フォーマットでデータをインポートまたはエクスポートするときに、フィールド・セパレータ、引用符文字列、エスケープ文字、デフォルトのコード化データ型に使用するデフォルトの記号を指定します。デフォルト設定では、フィールド・セパレータはカンマ (,)、引用符文字列はアポストロフィ (')、エスケープ文字は円記号 (¥) です。

デフォルトでは、Interactive SQL はサーバのデフォルト・データ型を使用します。

[メッセージ] ダイアログ・ボックス

[メッセージ] ダイアログ・ボックスは、次の要素で構成されます。

- [SQL 文の実行時間を計測] – 文の実行に要する時間を計測する Interactive SQL の機能を有効／無効にします。このオプションを有効にすると (デフォルト)、[メッセージ] ウィンドウ枠に時間が表示されます。

- [個別のメッセージウィンドウ枠を表示] – データベース・サーバからの情報を表示する場所を指定します。デフォルトでは、メッセージは[結果]ウィンドウ枠の[メッセージ]タブに表示されます。このオプションを選択すると、データベース・サーバの情報は、Interactive SQL ウィンドウの別の[メッセージ]ウィンドウ枠に表示されます。
- [メッセージウィンドウ枠のデフォルトの行数] – [メッセージ]ウィンドウ枠の初期の高さ(行単位)を指定します。デフォルトは7行です。

[エディタ]

[エディタ] ダイアログ・ボックスは、次の要素で構成されます。

- [エディタ] – スクロール・バーのスタイル設定(垂直、水平、両方)を選択します。
- [タブ] – このウィンドウ枠では、SQL テキストでタブをどのように使用するかを指定します。
 - [Tab size] – 各タブを構成するスペースの数を入力します。
 - [Indent size] – 各インデントに使用するスペースの数を入力します。
 - [Tab] ラジオ・ボタン – [Insert spaces] を選択すると、SQL テキストをインデントするときに、タブをスペースに変換します。[Keep tabs] を選択すると、SQL テキストをインデントするときに、タブをスペースとして保持します。
 - [自動インデント] – 次のオプションがあります。
 - SQL テキストを自動的にインデントしない場合は[なし]を選択します。
 - タブとインデントのデフォルト設定を使用する場合は[Default]を選択します。
 - Interactive SQL で SQL テキストを自動的にインデントする場合は[Smart]を選択します。左大カッコをインデントする場合は[Indent open brace]を選択し、右大カッコをインデントする場合は[Indent closing brace]を選択します。Interactive SQL では、ボタンの下にあるウィンドウに、これらのオプションを選択したときにテキストにどのように影響するかが表示されます。

- [フォーマット] – SQL テキストの外観を指定します。
 - [言語の要素] – リストの中で強調表示するテキストの種類 (キーワード、コメント、文字列など) を選択します。
 - [フォアグラウンド] – テキストの前景色を選択します。
 - [バックグラウンド] – テキストの背景色を選択します。
- [スタイル]
 - [フォントサイズ] – フォント・サイズを選択します。
 - [脱字記号の色] – 脱字記号の色を指定します。
 - [すべてリセット] – すべてのスタイルを元の選択内容に戻します。
- [印刷] – 印刷ジョブを次のようにカスタマイズします。
 - [ヘッダ] – ヘッダ・テキストを入力します。
 - [フッタ] – フッタ・テキストを入力します。
 - [フォントサイズ] – フォント・サイズを選択します。

[クエリエディタ]

[クエリエディタ] ダイアログ・ボックスは、次の要素で構成されます。

- [完全修飾テーブル名と完全修飾カラム] – このオプションを選択すると、Interactive SQL で、テーブル名の前に所有者名が付けられ (例: `dbo.authors`)、カラム名の前には所有者名とテーブル名が付けられます (例: `dbo.titles.price`)。
- [引用符名] – このオプションを選択すると、予約語との競合を回避するために、テーブル名とカラム名が自動的に引用符で囲まれます。
- [起動時にテーブルのリストを取得] – このオプションを選択すると、クエリ・エディタの起動時に、データベース内のテーブルのリストを自動的に取得します。

コマンド・ファイルの実行

この項では、一連のコマンドで構成されるファイルを処理する方法について説明します。

ファイルへの出力の書き込み

Interactive SQL では、各コマンドの出力は、次のコマンドが実行されるまで [結果] ウィンドウ枠に保持されます。データの記録を保持する場合は、各文の出力を個別のファイルに保存できます。

statement1 と statement2 が 2 つの select 文である場合、次のようにそれぞれ file1 と file2 に出力できます。

```
Statement1; OUTPUT TO file1
go
statement2; OUTPUT TO file2
go
```

たとえば、次のコマンドによってクエリの結果が保存されます。

```
select * from titles
go
output to "C:\My Documents\Employees.txt"
```

コマンド・ファイルの実行

コマンド・ファイルは、次の方法で実行できます。

- Interactive SQL の read コマンドを使用してコマンド・ファイルを実行する。次の文は、temp.sql ファイルを実行します。

```
read temp.SQL
```

- コマンド・ファイルを [SQL 文] ウィンドウ枠にロードし、このウィンドウ枠から直接実行する。

コマンド・ファイルを [SQL 文] ウィンドウ枠に再度ロードするには、[ファイル]-[開く]を選択します。ファイル名を入力するよう要求されたら、*transfer.sql* と入力します。

- Interactive SQL のコマンド・ライン引数としてコマンド・ファイルを指定する。

Interactive SQL の [SQL 文] ウィンドウ枠の行数は、500 行に制限されています。コマンド・ファイルがこれよりも大きい場合は、大きなファイルを処理する汎用エディタの機能を使用し、read コマンドを使用して Interactive SQL にインポートしてください。Interactive SQL には、読み込むことができる行数に制限はありません。

コマンド・ファイルの保存、ロード、実行

今後の Interactive SQL セッションで使用できるように、[SQL 文] ウィンドウ枠に現在表示されているコマンドを保存できます。コマンドを保存するファイルは、コマンド・ファイルと呼ばれます。

コマンド・ファイルは、SQL 文を含むテキスト・ファイルです。コマンド・ファイルは、好みのエディタを使用して作成できます。実行する SQL 文とともに、コメント行を含めることができます。一般に、このようなコマンド・ファイルはスクリプトと呼ばれます。

新しいセッションを開始したときに、コマンド・ファイルの内容を [SQL 文] ウィンドウ枠にロードしたり、内容をすぐに実行したりできます。

❖ コマンドを [SQL 文] ウィンドウ枠からファイルに保存する

- 1 [ファイル]-[保存]を選択します。
- 2 [保存]ダイアログで、ファイルの場所、名前、フォーマットを指定します。指定したら、[保存]をクリックします。

❖ コマンドをファイルから [SQL 文] ウィンドウ枠にロードする

- 1 [ファイル]-[開く]を選択します。
- 2 [開く]ダイアログで、ファイルを検索し、選択します。選択したら、[開く]をクリックします。

❖ コマンド・ファイルをすぐに実行する

- 1 [ファイル]-[スクリプトの実行]を選択します。
- 2 [スクリプトの実行]メニュー項目は、read 文に相当します。たとえば、[SQL 文] ウィンドウ枠に、以下のように入力してコマンド・ファイルを実行することもできます。c: ¥filename.sql は、ファイルのパス、名前、拡張子です。この例に示す一重引用符は、パスにスペースが含まれている場合にのみ必要です。

```
READ "c: ¥filename.sql"
```

- 3 [開く]ダイアログで、ファイルを検索し、選択します。選択したら、[開く]をクリックします。

[スクリプトの実行]メニュー項目は、READ 文に相当します。たとえば、[SQL 文] ウィンドウ枠に、以下のように入力してコマンド・ファイルを実行することもできます。c: ¥filename.sql は、ファイルのパス、名前、拡張子です。この例に示す一重引用符は、パスにスペースが含まれている場合にのみ必要です。

```
READ 'c: ¥filename.sql'
```

Interactive SQL での SQL エスケープ構文

Interactive SQL では、JDBC エスケープ構文をサポートしています。このエスケープ構文を使用すると、使用しているデータベース管理システムに関係なく、Interactive SQL からストアド・プロシージャを呼び出すことができます。エスケープ構文の一般的な形式は次のとおりです。

```
{{ keyword parameters }}
```

大カッコは必ず二重にしてください。この二重のカッコは、Interactive SQL に固有です。連続するカッコの間にはスペースを入れないでください。つまり、"{{" は使用できますが、"{{ " は使用できません。また、文中に改行文字は使用できません。Interactive SQL ではストアド・プロシージャを実行しないため、エスケープ構文をストアド・プロシージャで使用することはできません。

エスケープ構文を使用すると、JDBC ドライバによって実装される関数ライブラリにアクセスできます。このライブラリには、**number**、**string**、**time**、**date**、**system** の各関数が含まれています。

たとえば、データベース管理システムに依存しない方法で現在のユーザの名前を取得するには、次のように入力します。

```
select {{ fn user() }}
```

使用できる関数は、使用している JDBC ドライバによって異なります。jConnect がサポートする数値関数は、次のとおりです。

abs	cos	log10	sign
acos	cot	pi	sin
asin	degrees	power	sqrt
atan	exp	radians	tan
atan2	floor	rand	
ceiling	log	round	

jConnect がサポートする文字列関数は、次のとおりです。

ascii	difference	repeat	space
char	lcase	right	substring
concat	length	soundex	ucase

jConnect がサポートするシステム関数は、次のとおりです。

database	ifnull	user	convert
----------	--------	------	---------

jConnect がサポートするシステム関数は、次のとおりです。

curdate	dayofweek	monthname	timestampadd
curtime	hour	now	timestampdiff
dayname	minute	quarter	year
dayofmonth	month	second	

エスケープ構文を使用している文は、Adaptive Server Anywhere、Adaptive Server Enterprise、Oracle、SQL Server、または Interactive SQL から接続している別のデータベース管理システムで動作します。たとえば、SQL エスケープ構文を使用して、sa_db_info プロシージャを持つデータベース・プロパティを取得するには、Interactive SQL の [SQL 文] ウィンドウ枠に次のように入力します。

```
((CANN sa_db_info(1)))
```

Interactive SQL コマンド

Interactive SQL には、Interactive SQL 画面の上部のウィンドウ枠に入力する一連のコマンドがあります。これらのコマンドは Interactive SQL のみで使用されるもので、Adaptive Server に送信され、実行されることはありません。Interactive SQL で使用できるコマンドは、次のとおりです。

- clear — Interactive SQL のウィンドウ枠をクリアします。
- configure — Interactive SQL の [オプション] ダイアログを開きます。
- connect — データベースへの接続を確立します。
- disconnect — データベースへの現在の接続を切断します。
- exit — Interactive SQL を終了します。
- input — 外部ファイルまたはキーボードからデータベース・テーブルにデータをインポートします。
- output — 外部ファイルまたはキーボードからデータベース・テーブルにデータをインポートします。
- parameters — Interactive SQL コマンド・ファイルに対するパラメータを指定します。
- read — Interactive SQL 文をファイルから読み込みます。
- set connection — 現在のデータベース接続を別のサーバに変更します。
- set option — Interactive SQL オプションの値を変更します。
- start logging — 実行された SQL 文のログ・ファイルへの記録を開始します。

- **stop logging** —現在のセッションで SQL 文のロギングの停止に使用されます。
- **system** — Interactive SQL 内から実行ファイルを起動します。

『リファレンス・マニュアル：コマンド』の「DBISQL コマンドの使用」を参照してください。

bcp を使用した Adaptive Server とのデータの転送

この章では、バルク・コピー・ユーティリティである *bcp* を使用した、Adaptive Server とオペレーティング・システム・ファイルとの間でのデータの転送方法について説明します。

トピック	ページ
データの転送方法	44
<i>bcp</i> の要件	47
<i>bcp</i> のモード	46
ファイルへのデータのコピー	48
<i>bcp</i> のパフォーマンス	49
暗号化データのバルク・コピー	65
<i>bcp</i> オプションの使用	67
対話型 <i>bcp</i> のデフォルトの変更	70
フォーマット・ファイル	81
データを対話形式でコピー・アウトする例	85
データを対話形式でコピー・インする例	88
<i>bcp</i> と代替言語	92
初期化文字列に対するサポート	93
<i>bcp</i> とロー・レベルのアクセス・ルール	93
コピー・インとバッチ・ファイル	94
text または image データのコピー・アウト	96
ネットワーク・パケット・サイズの指定	96
コピー・インとエラー・ファイル	97
コピー・アウトとエラー・ファイル	98
デフォルト、ルール、トリガのデータの整合性	98
<i>bcp</i> と他のユーティリティの相違点	99

bcp は、データベース・テーブルまたはビューと、オペレーティング・システム・ファイル間での、便利で高速なデータ転送方法を提供します。**bcp** はさまざまなフォーマットでのファイルの読み込みや書き込みが可能です。ファイルからコピー・インする場合、**bcp** はデータを既存のデータベース・テーブルに挿入します。ファイルにコピー・アウトする場合は、**bcp** はファイルの以前の内容を上書きします。

bcp の構文の詳細については、「**bcp**」(139 ページ) を参照してください。

15.0.3 よりも前のバージョンでは、ノンクラスタード・インデックスまたはトリガを持つテーブルで高速 **bcp** を実行できませんでした。Cluster Edition 15.0.3 以降のバージョンには、この制限はありません。

データの転送方法

Adaptive Server データベースとの間でデータを転送するには、次のような方法があります。

- **bcp** をオペレーティング・システムからスタンドアロン・プログラムとして使用する方法。この方法については、この章で説明します。
- Client-Library で、バルク・ライブラリ・ルーチン呼び出す方法。Client-Library の詳細については、『Open Client/Server Common Libraries リファレンス・マニュアル』を参照してください。

bcp を使用したデータのインポートとエクスポート

Transact-SQL コマンドでは、データをバルク転送できないため、大量のデータを転送する場合は、**bcp** を使用します。**bcp** では、以下の操作ができます。

- 別のデータベース管理システムのレコードなど、以前に別のプログラムで使用していたデータのインポート。これが、**bcp** の最も一般的な使用法です。

bcp を使用するには、インポートするレコードを格納するファイルをあらかじめ作成しておきます。基本的な手順は次のとおりです。

- a 転送するデータをオペレーティング・システム・ファイルに変換します。
- b **bcp** をオペレーティング・システムのコマンド・ラインから実行します。

- Adaptive Server 間、またはオペレーティング・システム・ファイルを生成する他のデータ・ソースと Adaptive Server との間でのテーブルの移動。
- ビューからのデータのコピー・アウト。bcp を使用してビューの外へデータをコピーするための構文の詳細については、「[bcp](#)」(139 ページ) を参照してください。

注意 bcp を使用してビューにデータを「コピー・イン」することはできません。

- スプレッドシート・プログラムなどの他のプログラムで使用するためのデータの転送。データを転送する基本的な手順は次のとおりです。
 - a bcp は、データを Adaptive Server からオペレーティング・システム・ファイルに移動し、他のプログラムはそこからデータをインポートします。
 - b そのプログラムがデータを使用し終わったら、そのデータをオペレーティング・システム・ファイルにコピーします。それから bcp を使用して Adaptive Server にコピーします。

Adaptive Server は、「ターミネータ (カラムを区切るのに使用する文字)」かフィールドの長さがデータ・ファイルに指定されていれば、任意の文字フォーマットまたはバイナリ・フォーマットのデータを受け入れることができます。

bcp は次のように動作するため、転送で使用されるテーブルの構造が同一である必要はありません。

- ファイルからインポートするときには、データを既存のデータベース・テーブルに追加します。
- ファイルにエクスポートするときには、ファイルの既存の内容を上書きします。

転送が完了すると、bcp は以下の情報を表示します。

- 正常にコピーされたデータのロー数
- コピーできなかったローがある場合はその数
- コピーの総所要時間
- 1 つのローの平均コピー所要時間 (ミリ秒単位)
- 1 秒あたりにコピーされたロー数

bcp が正常に実行されると、リターン・ステータス 0 が表示されます。一般にリターン・ステータスは、`/usr/include/sys/` ディレクトリの `errno.h` ファイルにリストされたエラーに対応するオペレーティング・システム・レベルのエラーを反映しています。

bcp のモード

bcp in には、3 つのモードがあります。

- 低速 bcp – 各ローの挿入のログを取る。1 つ以上のインデックスのあるテーブルに使用される。
- 高速 bcp – ページ割り付けのログだけを取る。インデックスのないテーブルに、可能なかぎり高速にデータをコピーする。ノンクラスタード・インデックスのあるテーブルには高速 bcp を使用できる。
- 完全なログを取る高速 bcp – 各ローの完全なログを提供する。インデックスを持つテーブルまたは複写されたテーブルで高速 bcp を使用できる。

実際のコピー・タスクにどちらの bcp モードが最適かは、次の情報に基づいて判断します。

- データのコピー先となるテーブルのサイズ
- コピーするデータの量
- テーブル内のインデックスの数
- テーブルが複写されるかどうか
- インデックスの再作成に使用できる予備データベース・デバイスの領域

注意 高速 bcp を使用した場合はパフォーマンスが向上します。また、低速 bcp を使用した場合はデータのリカバリ性が向上します。完全なログを取る高速 bcp は、両方の組み合わせを提供します。

インデックス・タイプに応じて bcp によって使用されるモードは、次のとおりです。

テーブル・プロパティ	ロギングありの bulkcopy がオンの場合の bcp モード	ロギングなしの bulkcopy がオンの場合の bcp モード
クラスタード・インデックス。	低速モード	低速モード
インデックスまたはトリガを持つがクラスタード・インデックスは持たない複写テーブル	高速モード	低速モード
ノンクラスタード・インデックス、トリガなし	高速モード	低速モード
トリガ、インデックスなし	高速モード	高速モード
トリガを持つノンクラスタード・インデックス	高速モード	高速モード
インデックスなし、トリガなし、複写なし	高速モード	高速モード

bcp の要件

bcp を使用するには、以下の基本的なデータ情報を渡し、転送するデータとそのデータにアクセスするためのコマンドを準備する必要があります。

必要な基本情報

Adaptive Server とオペレーティング・システム・ファイルとの間でデータを転送するには、次の情報を入力する必要があります。

- データベースと、テーブルまたはビューの名前
- オペレーティング・システム・ファイルの名前
- 転送の方向 (コピー・インまたはコピー・アウト)

他にも、bcp を使用して、記憶タイプ、記憶長、各カラムのターミネータを任意に変更できます。

パーミッション

bcp を使用するには、Adaptive Server アカウントと、データベースのテーブルまたはビューおよび転送に使用するオペレーティング・システム・ファイルに対する適切なパーミッションが必要です。

- テーブルにデータをコピー・インするには、そのテーブルに対する `insert` と `select` の各パーミッションが必要です。
- テーブルをオペレーティング・システム・ファイルにコピーするには、次のテーブルに対する `select` パーミッションが必要です。
 - コピーするテーブル
 - `sysobjects`

- syscolumns
- sysindexes

転送前の作業

bcp in を使用するには、コマンドと転送するデータを準備しておきます。

- 高速または高速にログを取る bcp を使用するには、select into/bulkcopy/pllsort を true に設定します。たとえば、pubs2 データベースに対してこのオプションを設定するには、次のように入力します。

```
sp_dboption pubs2, "select into/bulkcopy/pllsort", true
```

詳細については、「[bcp のモード](#)」(46 ページ) を参照。

追加事項：

- Open Client のバージョン 11.1 以降を実行中で、外部の Sybase 設定ファイルを使用している場合は、次の文を設定ファイルに追加すれば bcp を使用できます。

```
[BCP]
```

- \$SYBASE 環境変数を Adaptive Server の現在のバージョンのロケーションに設定してから、bcp を使用してください。
- 以前のバージョンの bcp を使用するには、ocs.cfg ファイルの [bcp] セクションで CS_BEHAVIOR プロパティを設定します。

```
[bcp]
```

```
CS_BEHAVIOR = CS_BEHAVIOR_100
```

CS_BEHAVIOR を CS_BEHAVIOR_100 に設定していなければ、bcp 11.1 以降の機能を使用できます。

ファイルへのデータのコピー

bcp を使用すると、テーブルからオペレーティング・システム・ファイルにデータをコピーできます。テーブル名と1つ以上のパーティションをソースとして指定できます。データは以下にコピーできます。

- すべてのパーティションを1つのファイル
- 各パーティションごとに1つのファイル

コピー先のファイル名を指定していない場合、Adaptive Server はパーティション名に基づいてファイル名を作成します。

例

以下の例では、ptn1、ptn2、ptn3 の3つのパーティションに分割された bigtable からオペレーティング・システム・ファイルにデータをコピーします。

bigtable 内のデータを file1 にコピーするには、次のように入力します。

```
bcp mydb..bigtable out file1
```

ptn1、ptn2、ptn3 から file2 にデータをコピーするには、次のように入力します。

```
bcp mydb..bigtable partition ptn1, ptn2, ptn3 out file2
```

ptn1 と ptn2 からデータ・ファイル ptn1.dat と ptn2.dat にデータをコピーするには、次のように入力します。

```
bcp mydb..bigtable partition ptn1, ptn2 out ptn1.dat,  
ptn2.dat
```

次の例でも、ptn1 と ptn2 から ptn1.dat と ptn2.dat にデータをコピーします。次のように入力します。

```
bcp mydb..bigtable partition ptn1, ptn2 out
```

bcp の完全な構文と使用方法については、「[第7章 ユーティリティ・コマンド・リファレンス](#)」を参照してください。

bcp のパフォーマンス

bcp のパフォーマンスを向上させるには、次のような方法があります。

- 高速にログを取る bcp を使用します。
- 分割されたテーブルを使用します。分割されたテーブルに対して複数の bcp セッションを使用することにより、データのコピーに必要な時間を大幅に削減できます。ただし、このようなパフォーマンスの向上は、低速 bcp よりも高速および高速にログを取る bcp において顕著となります。
- bcp を並列モードで使用することによって、パフォーマンスは大幅に向上します。パラレル・バルク・コピーを使用すると、分割間でバランスのとれたデータ配分ができます。詳細については、「[パラレル・バルク・コピーを使った特定の分割へのデータのコピー](#)」(58 ページ)を参照してください。

注意 bcp はターゲット・テーブル上のトリガを起動しません。

高速、高速にログを取る、または低速 bcp の使用

テーブルにインデックスがあると、転送速度に影響を及ぼします。インデックスのあるテーブルに対して高速にログを取る bcp を明示的に指定しない限り、bcp では自動的に低速モードが使用されます。低速モードでは、データ挿入がトランザクション・ログに記録されます。これらの挿入の記録により、トランザクション・ログが非常に大きくなる可能性があります。

データが大きくなりすぎないようにし、障害発生時にデータベースを完全な形にリカバリできるようにするには、dump transaction を使ってログのバックアップを取ります。

デフォルトでは、新しく作成されたデータベースの select into/bulkcopy/pllsort オプションは false (無効) です。このオプションを無効にすると、bcp では自動的に低速モードが使用されます。高速 bcp と高速にログを取る bcp では、いずれも select into/bulkcopy/pllsort オプションを true に設定する必要があります。今後のデータベース用にデフォルトの設定を変更するには、model データベースでこのオプションをオンにします。

注意 インデックスのあるテーブルでデータをコピー・アウトまたはコピー・インするときには、select into/bulkcopy/pllsort オプションを true に設定する必要はありません。インデックスのあるテーブルは常に低速 bcp でコピーされ、すべての挿入がログに記録されます。

select into/bulkcopy/pllsort オプションがオンの場合は、トランザクション・ログをダンプできません。dump transaction を発行すると、エラー・メッセージが表示され、代わりに dump database を使用するよう指示されます。

警告！ データベースをダンプしてから select into/bulkcopy/pllsort フラグをオフに設定してください。ログが取られていないデータをデータベースに挿入し、dump database を実行する前に dump transaction を実行した場合は、そのデータをリカバリすることはできません。

高速 bcp の実行後は、Adaptive Server によって dump transaction が禁止されます。代わりに、dump database を使用する必要があります。低速 bcp は最小限にしかログに記録されない演算なので、Adaptive Server では、データベース内で select into/bulkcopy/pllsort を true または false に設定しても、低速 bcp を実行した後に dump transaction を発行できます。

Adaptive Serverが低速
bcp を実行する場合

Adaptive Server では、次の場合に低速 bcp が使用されます。

- `sp_dboption 'select into/bulkcopy/pllsort'` がオフの場合。
- `sp_dboption 'select into/bulkcopy/pllsort'` がオンで、テーブルが全ページ・ロック・スキームを使用し、テーブルにクラスタード・インデックスがある場合。
- `sp_dboption 'select into/bulkcopy/pllsort'` がオンで、テーブルにユニークでノンクラスタード・インデックスがある場合。

ユニークなインデックス上で `ignore_dup_key` オプションが有効な場合、高速 bcp を実行すると、重複キーを持つローを挿入したときにテーブルとインデックスの整合性が失われた状態になる。この不整合を回避するために、Adaptive Server は低速 bcp を実行する。

- テーブルにノンクラスタード・インデックスまたはトリガがあり、テーブルが複写対象としてマーク付けされているかデータベースがウォーム・スタンバイとして使用される場合。

高速 bcp は挿入のログを記録しないので、Adaptive Server が高速 bcp を使用した場合は、bcp によってコピーされたローを、問題があった場合に複写サイトでリカバリできない。これらの場合、Adaptive Server は、旧バージョンの動作を使用するように記述されたアプリケーションとの互換性を維持するために低速 bcp を使用する。

Adaptive Server が高
速 bcp を実行する場合

Adaptive Server が高速 bcp を使用するのには、次の場合です（いずれの場合においても、`sp_dboption Eeselect into/bulkcopy/pllsortEf` が有効になっており、テーブルにクラスタード・インデックスはありません）。

- 高速にログを取る bcp を明示的に指定していない場合。
- テーブルに、ユニークでないノンクラスタード・インデックスがある場合。Adaptive Server は、インデックスの更新およびページの割り付けのみのログを記録する。テーブルへの挿入のログは記録しない。
- テーブルにトリガがある場合。ただし、bcp はターゲット・テーブルのトリガを起動しない。
- テーブルに、クラスタード・インデックスが設定されているデータロー・ロック・スキームまたはデータページ・ロック・スキームがある場合。

テーブルにノンクラスタード・インデックスまたはトリガがあるが `sp_dboption 'select into/bulkcopy/pllsort'` が有効になっていない場合は、Adaptive Server は低速 bcp を使用します。

高速 bcp は、dump database の実行中は処理速度が低下します。

高速 bcp は、ページ割り付けだけをログに記録します。データをコピー・インする場合、データベース・テーブルにインデックスがなければ、bcp は最高速度で機能します。

高速 bcp を使用してデータを挿入した場合、高速 bcp ではデータの挿入はログに記録されないため、トランザクション・ログをデバイスにバックアップ (dump) することはできません。変更内容がログに記録されていないため、バックアップ・データが存在せず、データをリカバリすることはできません。バックアップ要求 (dump transaction) を行うと、エラー・メッセージが表示され、代わりに dump database を使用するように指示されます。この制約は、dump database が正常に終了するまで効力があります。dump database と dump transaction の詳細については、『システム管理ガイド』と『ASE リファレンス・マニュアル』を参照してください。

bcp 最適化は Adaptive Server によって実行され、Open Client バージョン 15.0 以降を使用する必要はありません。

Adaptive Server が高速にログを取る bcp を実行する場合

高速にログを取る bcp は、インデックスとトリガを含むテーブルで使用できます (bcp はターゲット・テーブルのトリガを起動しません)。高速にログを取る bcp は、挿入をテーブルに記録します。

セッションに対して高速にログを取る bcp を設定するには、set logbulkcopy {on | off} コマンドを使用します。set logbulkcopy {on | off} は、--initstring '*Transact-SQL_command*' パラメータに含めることができます。このパラメータは、データを転送する前に Transact-SQL コマンドを Adaptive Server に送信します。たとえば、次の例では、titles.txt データを pubs2..titles テーブルに転送するときにロギングを有効にすることができます。

```
bcp pubs2..titles in titles.txt --initstring 'set logbulkcopy on'
```

インデックスを持つテーブルのコピー

bcp プログラムは、データと関連のあるインデックスを持たないテーブルにデータをロードするために最適化されています。このプログラムは、できる限り高速に、インデックスのないテーブルにデータをロードします。高速にログを取る bcp は、すべてのデータ変更をテーブルに記録します。

1 つ以上のインデックスのあるテーブルにデータをコピーする場合は、高速および高速にログを取る **bcp** を使用できます。これには、**create table** 文の一意性制約を使用して暗黙的に作成されたインデックスも含まれます。しかし、**bcp** は、テーブルに定義された他の整合性制約は適用しません。

注意 低速 **bcp** では、**bcp** によってインデックスのあるテーブルへの挿入がログに記録されるため、ログのサイズが非常に大きくなる場合があります。バルク・コピーが完了してから、**dump database** を使用してデータベースのバックアップを行い、**dump transaction** を使ってログをトランケートしてください。

インデックスがあるテーブルにデータをコピーする場合には、パフォーマンスが大幅に低下します。膨大な数のローをコピー・インする場合は、まず **drop index** (または一意性制約として作成されたインデックスには **alter table**) を使用してすべてのインデックスを削除し、データベース・オプションの設定、テーブルへのデータのコピー、インデックスの再生成を行ってからデータベースをダンプすると、処理速度が上がる場合があります。インデックスの構成に必要なディスク・スペースとして、データに必要な空き領域の約 2.2 倍分を割り当てるようにしてください。

インデックスの削除

大量のローをコピーする場合、データ・サイズの 1.2 倍の領域が必要であり、サーバにはクラスタード・インデックスを再構成するための十分な領域が必要となります。

- 十分な領域がある場合は、**drop index** を使用して、事前にインデックスをすべて削除できます。
- データをソートし、インデックスを構築するために必要な領域がサーバにない場合は、低速 **bcp** を使用してください。

高速および高速にログを取る bcp を使用したでのコピー・イン

表 4-1 に、高速および高速にログを取る bcp を使用してデータを Adaptive Server にコピー・インする手順を示します。

表 4-1: 高速 bcp を使ってデータをコピー・インする手順

手順	作業者
sp_dboption を使用して select into/bulkcopy/pllsort を true の後の「.」は不要。	システム管理者またはデータベース所有者
テーブルのインデックスを再構築するための十分な空き領域があることを確認する。 テーブルのインデックスを削除する。	テーブル所有者
テーブルに対する insert パーミッションがあることを確認する。	テーブル所有者によって許可されたユーザ
bcp でコピーを行う。	insert パーミッションを持つユーザ
インデックスを再構築する。	テーブル所有者
必要に応じて sp_dboption をリセットする。	システム管理者またはデータベース所有者
dump database を使用して、新しく挿入されたデータのバックアップを取る。	システム管理者、オペレータ、またはデータベース所有者
ストアド・プロシージャまたはクエリを実行して、新しくロードされたデータがルールに違反していないかどうかを確認する。	テーブル所有者またはストアド・プロシージャ所有者

分割されたテーブルへのバルク・コピー

ある条件のもとでは、分割されたテーブルに対して複数の bcp セッションを実行することによって bcp のパフォーマンスを大幅に向上させることができます。

テーブルを分割するとロックの競合が減り、I/O が複数のデバイスに分散するため、挿入のパフォーマンスが向上します。分割されたテーブルで bcp のパフォーマンスが向上するのは、主に I/O が分散することによります。

分割されたテーブルに対して bcp セッションを実行する場合、次の点を考慮してください。

- テーブルの分割によってパフォーマンスが向上するのは、テーブルへのバルク・コピー・インを実行する場合です。
- 低速 bcp のパフォーマンスは、テーブルを分割しても向上しません。表 4-1 (54 ページ) で説明したように、すべてのインデックスを削除し、高速または高速にログを取る bcp を使用してください。

- 複数の bcp セッションを実行すると、すぐにネットワーク・トラフィックがボトルネックとなることがあります。このボトルネックを防ぐには、可能なかぎり Adaptive Server にローカル接続してください。

分割されたテーブルにデータをコピーするには、次の2とおりの方法があります。

- コピー先の分割に関係なく、ランダムにデータをコピーする。たとえば、*file1* から *bigtable* にデータをコピーするには、次のように入力します。

```
bcp mydb..bigtable in file1
```

file1、*file2*、*file3* から *bigtable* にデータをコピーするには、次のように入力します。

```
bcp mydb..bigtable in file1, file2, file3
```

- 特定の分割にデータをコピーする。たとえば、*file1* から *ptn1*、*file2* から *ptn2*、*file3* から *ptn3* にデータをコピーするには、次のように入力します。

```
bcp mydb..bigtable partition ptn1, ptn2, ptn3 in  
file1, file2, file3
```

file1 から *bigtable* の最初の分割にデータをコピーするには、次のように入力します。

```
bcp mydb..bigtable:1 in file1
```

テーブルにクラスタード・インデックスがある場合、bcp は低速モードで実行され、このインデックスによってローの配置を制御できます。

bcp の完全な構文と使用方法については、「[第7章 ユーティリティ・コマンド・リファレンス](#)」を参照してください。

❖ データを分割へランダムにコピーする

複数の bcp セッションを使用するときに、分割されたテーブルにランダムにデータをコピーするには、次の手順に従います。

- 1 システムで必要なだけの数の分割と物理デバイスを持つテーブルを設定します。

詳細については、『パフォーマンス&チューニング・シリーズ』およびこのマニュアルの「[パラレル・バルク・コピーを使った特定の分割へのデータのコピー](#)」(58 ページ)を参照してください。

- Adaptive Server が複数の bcp セッションのサポートに十分なロックを持つよう設定されていることを確認します。ロックの設定については、『システム管理ガイド』を参照してください。
- テーブルのインデックスを削除し、高速または高速にログを取る bcp を使用できるようにします。詳細については、「[高速、高速にログを取る、または低速 bcp の使用](#)」(50 ページ)を参照してください。

注意 低速 bcp を使用する場合、インデックスを削除するとパフォーマンスが大幅に向上することがあります。

- 同時に使用する bcp セッションの数と同じ数で同じサイズになるように、bcp 入力ファイルを分割します。
また、`-F first_row` オプションと `-L last_row` オプションを使用して、各「入力ファイル」の先頭と末尾を指定することもできます。
- 個々のファイルを使用して、ローカルの Adaptive Server マシン上で bcp セッションを並行に実行します。
たとえば、UNIX プラットフォームでは、シェル・ウィンドウごとにそれぞれ異なるセッションを実行できます。また、個々の bcp セッションをバックグラウンドで開始することもできます。

分割されたテーブルへのデータのコピー方法の詳細は、「[パラレル・バルク・コピーを使った特定の分割へのデータのコピー](#)」(58 ページ)を参照してください。

dbcc checktable と sp_helpsegment を使用した bcp セッションのモニタリング

bcp セッションで使用するパーティションを指定していない場合、Adaptive Server は複数の bcp セッションをそのテーブルの使用可能なパーティションにランダムに割り当てます。このようにランダム割り当てが行われた場合は、挿入が均等に分散されているかどうかを判断するために、次のいずれかの方法で分割をモニタしてください。

- dbcc checktable コマンドを定期的に行い、各分割の総ページ数をチェックする。
- sp_helpsegment または sp_helppartition を使用する。これらのコマンドはデータベース・オブジェクトをロックしないで同じチェックを行う。

dbcc checktable の詳細については、『システム管理ガイド』を参照してください。sp_helpsegment と sp_helppartition の詳細については、『リファレンス・マニュアル』を参照してください。

テーブル分割の詳細については、『パフォーマンス&チューニング・シリーズ』を参照してください。

ページ割り付けを増加することによるロギングの減少

高速または高速にログを取る bcp を使用する場合、各 bcp in バッチではページ・マネージャが1つ以上のエクステントを割り付ける必要が生じます。また、このような割り付けごとに1つのログ・レコードが生成されます。

設定パラメータ number of pre-allocated extents を使用して、Adaptive Server がページ・マネージャを使用して割り付けるエクステントの数を指定します。

- 設定パラメータ number of pre-allocated extents の有効な値は 0 ～ 32 で、デフォルト値は 2 です。
- 大規模な bcp オペレーションを行うときには、この数を増やし、ページ割り付けでログがいっぱいになるのを防いでください。
- 大規模なエクステント割り付けが行われないようにするには、この値を 0 に設定します。これにより、ページ・マネージャは単一ページ割り付けだけを実行します。

Adaptive Server は実際に必要なページより多くのページを割り付ける場合があるため、領域が限られている場合は、この値を小さくしておいてください。これらのページの割り付けは、バッチ終了時に解除されます。

Adaptive Server バージョン 15.5 以降では、事前割り付けエクステントの最大値が 31 から 32 に増えています。

number of pre-allocated extents パラメータに値 32 を使用することは、設定にとって特別な意味があり、Adaptive Server が内部的に実行する領域の割り付けに影響を及ぼします。number of pre-allocated extents が 32 に設定されると、Adaptive Server はエクステント数分の割り付け単位全体を bcp in や select into などのユーティリティ操作のために予約しようとし、これらの操作では、領域予約の大規模な割り付けスキームが使用されるからです。

number of pre-allocated extents の最大数を使用すると、特に並列実行する場合に、これらのユーティリティのパフォーマンスが大幅に向上します。値 32 を使用すると、ユーティリティを実行する各エンジンが独自の割り付け単位上で他のエンジンから干渉されることなく独立して作業できる可能性が大幅に高くなります。

詳細については、『システム管理ガイド第 1 巻』の「設定パラメータの設定」を参照してください。

パラレル・バルク・コピーを使った特定の分割へのデータのコピー

特定の分割へデータを並列にコピーするには、パラレル・バルク・コピーを使用します。パラレル・バルク・コピーにより、bcp セッション実行時のパフォーマンスは飛躍的に向上します。これは、大規模なバルク・コピー・ジョブを複数のセッションに分割し、同時に実行できるからです。

パラレル・バルク・コピーを使用するには、次の点に注意します。

- コピー先のテーブルは分割してください。
 - `sp_helppartition` を使用してテーブルのパーティションの数を確認します。
 - テーブルが分割されていない場合は、`alter table ... partition` を使用してテーブルを分割します。
- 次のような理由により、コピー先のテーブルにはインデックスを付けないでください。
 - テーブルにクラスタード・インデックスがある場合、そのインデックスがデータの物理的な配置を決定するため、bcp コマンドの分割指定が無視されます。
 - インデックスがある場合、bcp は自動的に高速バルク・コピー・モードではなく低速バルク・コピー・モードとなります。
- テーブルにノンクラスタード・インデックスがある場合、パラレル・バルク・コピーではインデックス・ページのデッドロックが発生しやすくなります。
- 最高のパフォーマンスを発揮させるには、各分割を別個の物理ディスクに保管してください。
- コピー先のテーブルを分割してから、データベースにデータをコピーしてください。

- パラレル・バルク・コピーでは、複数のオペレーティング・システムのファイルからテーブルにコピー・インできます。

すべての種類の分割されたテーブルで、次のコマンドを使用します。

```
bcp tablename partition partition_name in file_name
```

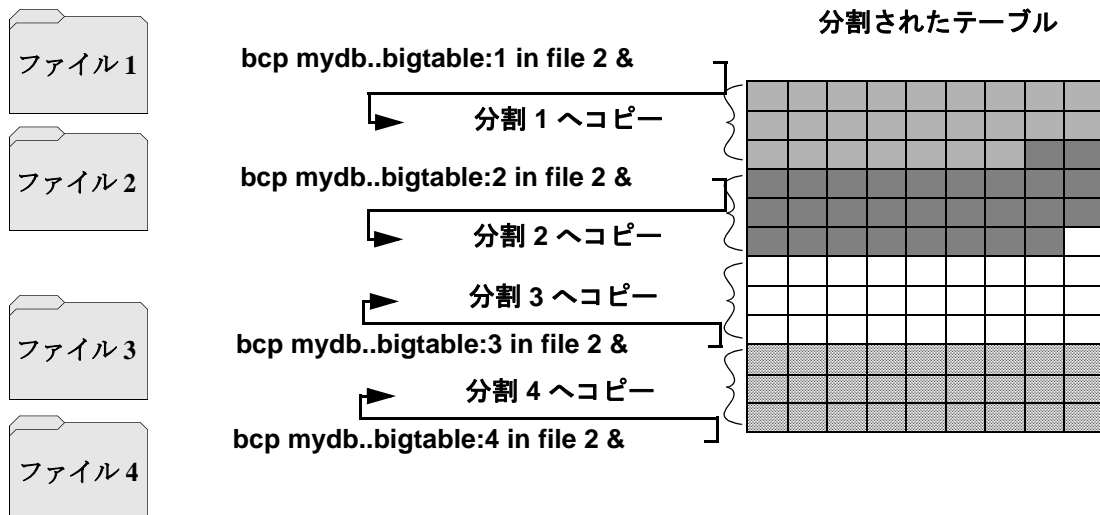
ラウンドロビン方式で分割されたテーブルの場合のみ、次のコマンドを使用できます。

```
bcp tablename partition_number in file_name
```

図 4-1 は、パラレル・バルク・コピーのプロセスを示します。

図 4-1: パラレル・バルク・コピーを使用したラウンドロビン方式で分割されたテーブルへのデータのコピー

小さなファイルに 4 分割された
大きなファイル



テーブルの分割に関する詳細については、『Transact-SQL ユーザーズ・ガイド』を参照してください。

注意 パラレル・バルク・コピーを使用してデータをコピー・アウトするときには、パーティション番号を指定することはできません。パーティション名は指定できます。

bcp in とロック

bcp を使用してテーブルにコピー・インする場合、特にパラレル bcp を使用してテーブルにコピー・インする場合には、コピー・プロセスで次のロックが必要になります。

- テーブルへの意図的な排他ロック
- 各データ・ページまたはデータ・ローへの排他ページ・ロック
- インデックス・ページへの排他ロック (インデックスがある場合)

巨大なテーブルをコピー・インする場合、特に分割されたテーブルへの同時コピーを行う場合には、多数のロックが発生する可能性があります。

ロック数の不足を回避するには、ロックの数を増やします。

- 必要なロックの数を見積もるには、次の式を使用します。

```
# of simultaneous batches * (rows_per_batch / (2016/row_length))
```

- テーブルのローの長さを確認するには、次のクエリを実行します。

```
1> select maxlen
2> from sysindexes
3> where id = object_id("tablename") and (indid = 0 or indid = 1)
```

ロック数の設定の詳細については、『システム管理ガイド』を参照してください。

- それほど大きくないバッチをコピーするには、**-b batchsize** フラグを使用してください。デフォルトのバッチ・サイズは 1000 ローです。bcp が処理できる最小のバッチ・サイズは 1、最大は 2147483647L です。
- 同時に実行するバッチ数を減らします。

パラレル・バルク・コピーの方法

パラレル・バルク・コピーを使ってデータをコピー・インするには、次の方法のいずれかを使用します。

- バックグラウンドで複数の bcp セッションを起動します。この場合、次のことを必ず行ってください。
 - コマンド・ラインでパスワードを指定する。
 - ネイティブ・モード、文字モード、またはフォーマット・ファイルを使用する。

bcp は、テーブルの分割と同じ数だけ起動できます。

- フォーマット・ファイルを作成して使用します。
 - a 対話型モードで `bcp` を起動します。
 - b プロンプト表示に答えます。
 - c そのプロンプトに対する応答を保管するフォーマット・ファイルを作成します。
 - d コピーが始まるときに処理をバックグラウンドに移します。
 - e 次の `bcp` コマンドを発行し、最初の `bcp` コマンドで作成したフォーマット・ファイルを指定します。
- 複数のウィンドウで `bcp` セッションを開始します。
- 単一の `bcp in` コマンドで、ファイル・マッピングするパーティションを指定します。

並列モードでは、クライアントはそれぞれに独立した複数の `bcp in` セッションを実行できます。また、ユーザが `--maxconn` オプションを指定して、`bcp` クライアントがサーバに対してオープンできる並列接続の最大数を制御することもできます。

パラレル・バルク・コピーの構文

パラレル・バルク・コピーの構文は次のとおりです。

```
bcp table_name[:partition_number | partition_name] in file_name  
-Pmypassword
```

各パラメータの意味は次のとおりです。

- *table_name* — データのコピー先となるテーブルの名前です。
- *partition_name* — コピー先となる分割の名前です。
- *file_name* — データを格納するホスト・ファイルです。
- *mypassword* — パスワードです。

ラウンドロビン方式で分割されたテーブルでのパラレル・バルク・コピーの使用

ソートされたデータを特定の分割へ並列にコピーするには、次の手順に従います。

- テーブル名の後にコロン (:) と分割番号を追加し、分割を指定します。次に例を示します。

```
publishers:10
```

注意 ここで指定する分割は、bcp コマンドを発行する前に存在しているものにしてください。

- ソートされたデータを個別のファイルに分割します。または、ホスト・ファイルの最初のロー (-F *first_row*) と最後のロー (-L *last_row*) をそれぞれ指定して、「ファイル」の境界を示します。
- テーブルの分割の数に注意してください。起動できるパラレル・バルク・コピーのセッション数は、このテーブルの分割数によって制限されます。

たとえば、テーブルに 4 つの分割があり、5 つのパラレル・バルク・コピー・ジョブを実行した場合は、最初の 4 つのジョブだけが並列に実行され、5 番目のジョブは最初の 4 つのジョブのいずれかが終了するまで実行されません。

bcp は、それぞれのファイル、または行番号のセットをそれぞれの分割へコピーします。たとえば、パラレル・バルク・コピーを使用して、ソートされたデータを 4 つのファイルから `mydb..bigtable` の 4 つの分割にコピー・インするには、次のように入力します。

```
bcp mydb..bigtable:1 in file1 -Pmypassword -c &
bcp mydb..bigtable:2 in file2 -Pmypassword -c &
bcp mydb..bigtable:3 in file3 -Pmypassword -c &
bcp mydb..bigtable:4 in file4 -Pmypassword -c &
```

パラレル・バルク・コピーと IDENTITY カラム

パラレル・バルク・コピーを使用するとき、IDENTITY カラムがボトルネックの原因になることがあります。bcp は、データの読み込み時に IDENTITY カラムの値を生成すると同時に、各ローの IDENTITY カラムの最大値も更新します。これが、パラレル・バルク・コピーを使用することによるパフォーマンスの向上に悪影響を及ぼすことがあります。

このボトルネックを避けるには、各セッションの IDENTITY 開始ポイントを明示的に指定します。

ソート順の保持

IDENTITY 開始ポイントを明示的に指定しないで、ソートされたデータをテーブルにコピーした場合、bcp は IDENTITY カラムの値をソート順に生成しないことがあります。パラレル・バルク・コピーでは情報をすべてのパーティションに同時に読み込み、そのデータの読み込み時に IDENTITY カラムの値を更新します。

bcp 文に開始ポイントを明示的に指定していない場合、図 4-2 に示すような IDENTITY カラム番号が生成されます。

図 4-2: ソート順による IDENTITY カラムの生成

分割 1			分割 2			分割 3			分割 4		
ID カラム			ID カラム			ID カラム			ID カラム		
100	A		102	C		103	F		101	H	
104	A		106	C		105	F		110	H	
107	B		109	C		111	F		113	I	
108	B		112	D		116	G		115	J	
114	B		117	E		119	G		118	J	

このテーブルの IDENTITY カラム番号の最大値は 119 ですが、この順序はここでは無意味です。

Adaptive Server によって IDENTITY カラムのユニークな値を生成する場合は、-g パラメータまたは -E パラメータを使用して bcp を実行してください。

コマンド・ラインでの開始ポイントの指定

コマンド・ラインでセッションの IDENTITY 開始ポイントを指定するには、-g *id_start_value* フラグを使用します。

-g パラメータを使用すると、Adaptive Server は、ローごとにテーブルの IDENTITY カラムの最大値をチェックしたり更新したりしないで、bcp セッションで使用される IDENTITY カラム値のシーケンスを生成できます。この場合、Adaptive Server は IDENTITY カラムの最大値をチェックする代わりに、各バッチの終わりに最大値を更新します。

警告！ 重複した IDENTITY 値の範囲を指定すると、誤って重複した IDENTITY 値が生成されることに注意してください。

IDENTITY の開始値を指定するには、次のように入力します。

```
bcp [-gid_start_value]
```

たとえば、100 ロールある 4 つのファイルをコピー・インするには、次のように入力します。

```
bcp mydb..bigtable in file1 -g100
bcp mydb..bigtable in file2 -g200
bcp mydb..bigtable in file3 -g300
bcp mydb..bigtable in file4 -g400
```

-g パラメータを使用しても、IDENTITY カラムの値がユニークとなることが保証されるわけではありません。この値が確実にユニークとなるようにするためには、以下のことが必要です。

- 入力ファイル中のローの数と既存の値の最大値を取得します。この情報をもとに **-g** パラメータで開始値を設定し、生成される値の範囲が重複しないようにします。

前述の例では、いずれかのファイルのローが 100 以上あった場合、IDENTITY の値が次の 100 ロールと重なり、重複した IDENTITY 値が生成されてしまいます。

- IDENTITY の値と重複するデータを他の誰かが挿入することのないよう、注意してください。

テーブルの IDENTITY カラムの値を指定しません。

デフォルトでは、IDENTITY カラムがあるテーブルにデータをバルク・コピーするときに、bcp は各ローに IDENTITY カラムのテンポラリの値 0 を割り当てます。これは、テーブルにデータをコピーする場合にだけ有効です。bcp はデータ・ファイルから ID カラムの値を読み込みますが、この値をサーバには送信しません。代わりに、bcp がテーブルに各ローを挿入するときに、サーバが 1 で始まるユニークで連続した IDENTITY カラム値を割り当てます。データをテーブルにコピーするときに **-E** フラグを指定した場合は、bcp はデータ・ファイルからこの値を読み込み、この値をテーブルに挿入するサーバに送信します。挿入されるローの数が IDENTITY カラムの値の最大値を超える場合、Adaptive Server はエラーを返します。

-E パラメータは、バルク・コピー・アウトには影響しません。**-N** パラメータを使用しない場合、Adaptive Server は ID カラムをデータ・ファイルにコピーします。

-E フラグと **-g** フラグを同時に使用することはできません。

暗号化データのバルク・コピー

bcp によって、暗号化データがデータベースの内外でプレーン・テキストまたは暗号テキストの形式で転送されます。デフォルトでは、bcp はプレーン・テキスト・データを次のように処理してコピーします。

- **bcp in** を実行すると、Adaptive Server によって挿入前にデータが自動的に暗号化されます。低速 **bcp** が使用されます。ユーザには、すべてのカラムの **insert** パーミッションと **select** パーミッションが必要です。
- **bcp out** を実行すると、Adaptive Server によってデータが自動的に暗号化されます。すべてのカラムに対して **select** パーミッションが必要です。さらに、暗号化カラムに対しては **decrypt** パーミッションが必要です。

次の例では、**customer** テーブルがプレーン・テキスト・データとしてコンピュータのネイティブ・フォーマットでコピーされます。

```
bcp uksales.dbo.customer out uk_customers -n -Uroy -Proy123
```

プレーン・テキストとしてコピー・アウトされるデータが、明示的なパスワードを使用したキーによって暗号化されている場合は、**--c password** オプションまたは **--colpasswd** オプションを使用して **bcp** にパスワードを指定できます。

たとえば、**employee** テーブルの **salary** カラムが、明示的なパスワードによって保護されているキーで暗号化されている場合、**salary** データをコピー・アウトするには、次のように **bcp** にパスワードを指定する必要があります。

```
bcp hr.dbo.employee out -c -Upjones -PX15tgo1 --  
colpasswd hr.dbo.employee.salary '4mIneIsonly'
```

あるいは、**salary** カラムを暗号化しているキーの名前を知っている場合は、次のようにすることもできます。

```
bcp hr.dbo.employee out -c -Upjones -PX15tgo1 --  
keypasswd keydb.dbo.hr_key '4mIneIsonly'
```

bcp は、データを選択する前に、指定されたパスワードを使用して **set encryption passwd** コマンドを発行します。

データを再びコピー・インするときには、**bcp** コマンド・ラインで、**--keypasswd** オプションと **--colpasswd** オプションを同じように使用します。

`bcp` の `-C` オプションを使用すると、データは暗号テキストとしてコピーされます。暗号テキストをコピーするとき、異なるオペレーティング・システム間でデータをコピーする場合があります。文字データを暗号テキストとしてコピーする場合、両方のプラットフォームで同じ文字セットがサポートされている必要があります。

`bcp` の `-C` オプションでは、管理者がデータの `decrypt` パーミッションを持たない場合でも `bcp` を実行できます。`-C` オプションを使用すると、`bcp` では次のようにデータが処理されます。

- `bcp in` の実行時にデータは暗号テキスト形式であるとみなされ、`Adaptive Server` は暗号化を行いません。`-C` オプションを使用できるのは、`Adaptive Server` にコピーされるファイルが、`bcp out` の `-C` オプションを使用して作成された場合のみです。暗号テキストは、カラム属性がまったく同じカラムからコピーされ、コピー先のカラムと同じキーで暗号化されているはずですが、高速 `bcp` が使用されます。ユーザはテーブルに対する `insert` および `select` パーミッションを持っている必要があります。
 - `bcp in -C` では、暗号化カラムに対するドメイン・ルールおよび検査制約は、どちらが存在する場合もバイパスされます。この状況では、`Adaptive Server` は高速 `bcp` を使用するためです。ドメイン・ルールと検査制約は、`bcp out -C` には影響を与えません。
 - 暗号化カラムにアクセス・ルールが存在する場合、`bcp out -C` を使用するとエラー 2929 が発生します。アクセス・ルールは、`bcp in -C` には影響を与えません。
 - `bcp out` を実行すると、データが復号化されずに `Adaptive Server` からコピーされます。暗号テキスト・データは 16 進数形式です。ユーザには、すべてのカラムの `select` パーミッションが必要です。暗号テキストをコピーするときは、暗号化カラムの `decrypt` パーミッションは必要ありません。
 - 暗号化された `char` データまたは `varchar` データは、暗号化時に `Adaptive Server` によって使用された文字セットを維持します。データが暗号テキスト形式で別のサーバにコピーされる場合、そのターゲット・サーバで使用される文字セットは、ソースからコピーされる暗号化データの文字セットと一致する必要があります。暗号化時にソース・サーバ上のデータに関連付けられた文字セットは、暗号化データとともに格納されず、ターゲット・サーバで認識されないか、または変換されます。
- `-C` オプションを使用せずに `bcp` を実行して、文字セットの問題を回避することもできます。

文字セット変換用の `-J` オプションを `-C` オプションとともに使用することはできません。

次の例では、`customer` テーブルがコピーされます。`cc_card` カラムは、人間が判読できる暗号テキストとしてコピーされます。その他のカラムは文字フォーマットでコピーされます。ユーザ “roy” には `customer cc_card` の `decrypt` パーミッションは必要ありません。

```
bcp uksales.dbo.customer out uk_customers -C -c -Uroy  
-Proy123
```

暗号テキストとしてデータをコピーする場合は、データを再びコピー・インするときにデータベースで同じキーが使用できることを確認してください。必要に応じて、`ddlgen` ユーティリティを使用して、一方のデータベースからもう一方のデータベースにキーを移動してください。

bcp オプションの使用

この項では、`bcp` 構文のより複雑なオプションについて説明します。構文の完全な説明については、「[bcp](#)」(139 ページ)を参照してください。

デフォルト・フォーマットの使用

`bcp` には、頻繁に使用されるデフォルト・フォーマット付きのファイルを作成する2つのコマンド・ライン・オプションがあります。これらのオプションを使用すると、Adaptive Server のデータを簡単にコピー・インおよびコピー・アウトできます。

- `-n` オプションは、「ネイティブ・フォーマット」(オペレーティング・システム・フォーマット)を使用します。
- `-c` オプションは、すべてのカラムに「文字」(`char` データ型)を使用します。このデータ型の場合、ローのフィールド間にはタブが挿入され、各ローの終わりには復帰改行などの改行ターミネータが挿入されます。

ネイティブ・オプションまたは文字オプションを使用した場合、`bcp` は非対話型で動作し、Adaptive Server のパスワード以外の情報の入力には要求しません。

ネイティブ・フォーマット

-n オプションを指定すると、「**native**」フォーマット (オペレーティング・システム固有のフォーマット) でファイルが作成されます。ネイティブ・フォーマットでは通常、凝縮されたオペレーティング・システム・ファイルが作成されます。たとえば、次のコマンドは、ネイティブ・データ・フォーマットを使用して `publishers` テーブルを `pub_out` ファイルにコピーします。

```
bcp pubs2..publishers out pub_out -n
```

`pub_out` の内容を次に示します。

```
0736^MNew Age Books^FBoston^BMA0877^PBinnet & Hardley^J  
Washington^BDC1389^TAlgodata Infosystems^HBerkeley^BCA
```

`bcp` は、`char(4)` データ型の `pub_id` を除く各フィールドの前に、そのフィールド内のデータ長に相当する ASCII 文字を付けます。たとえば、“New Age Books” は 13 文字で、`^M` (Ctrl-m) は ASCII で 13 です。

`pub_out` ファイルに保管されているすべてのテーブル・データは、`char` データまたは `varchar` データであるため、人間が判読できる文字です。数値データが含まれるテーブルの場合、`bcp` はオペレーティング・システムのデータ表現フォーマットでファイルに情報を書き込むため、人間には判読できません。

`bcp` は、そのネイティブ (データベース) データ型、または暗黙の変換が当該データ型に対してサポートされる任意のデータ型のどちらかとして、データをファイルにコピー・アウトできます。`bcp` は、基本のデータ型か、暗黙的な変換がサポートされているデータ型として、ユーザ定義データ型をコピーします。データ型変換の詳細については、『Open Client DB-Library/C リファレンス・マニュアル』または『ASE リファレンス・マニュアル』の「`dbconvert`」の項を参照してください。

注意 `bcp` ユーティリティは、Windows から UNIX へのコピーなど、異なるオペレーティング・システムからのネイティブ・フォーマットのデータのコピーはサポートしていません。`bcp` を使用して、あるオペレーティング・システムから別のオペレーティング・システムへファイルをコピーする場合は、`-c` フラグを使用してください。

警告! ネイティブ・フォーマットの場合、`bcp` でロー・ターミネータ・パラメータ (`-t`) またはフィールド・ターミネータ・パラメータ (`-r`) は使用しないでください。予期しない結果となったり、データが破損する可能性があります。

文字フォーマット

文字フォーマット (-c) では、すべてのカラムに char データ型が使用されます。このフォーマットでは、各ローのフィールド間にタブが挿入され、各ローの終わりに改行ターミネータが挿入されます。

たとえば、次のコマンドは、publishers テーブルから pub_out ファイルに文字フォーマットでデータをコピー・アウトします。

```
bcp pubs2..publishers out pub_out -c
```

このコマンドによる bcp 出力は次のようになります。

0736	New Age Books	Boston	MA
0877	Binnet & Hardley	Washington	DC
1389	Algodata Infosystems	Berkeley	CA

コマンド・ラインでのターミネータの変更

ターミネータは、データ・フィールド間を区切る文字 (フィールド・ターミネータ) です。ロー・ターミネータは、テーブルまたはファイルの最後のフィールドのフィールド・ターミネータです。ターミネータをコマンド・ラインで変更するには、文字フォーマット・オプション (-c) とともに、コマンド・ライン・オプションの `-tfield_terminator` と `-rrow_terminator` を使用します。

次の例では、フィールド・ターミネータとしてカンマ (,) を、ロー・ターミネータとして改行 (¥r) を使用しています。

- UNIX プラットフォームの場合 :

```
bcp pubs2..publishers out pub_out -c -t , -r ¥¥r
```

必要な場合は、使用しているオペレーティング・システムのコマンド・シェルで円記号を「エスケープ」してください。

- Windows の場合 :

```
bcp pubs2..publishers out pub_out -c -t , -r ¥r
```

この bcp コマンド・ラインからは、次のような情報が出力されます。

```
0736,New Age Books,Boston,MA
0877,Binnet & Hardley,Washington,DC
1389,Algodata Infosystems,Berkeley,CA
```

注意 -t オプションと -r オプションを使用すると、文字オプション (-c) を使用しないでデフォルト・ターミネータを変更できます。

対話型 bcp のデフォルトの変更

ネイティブ・フォーマット (-n) または文字フォーマット (-c) を指定しなかった場合、bcp は次のような情報の入力を要求するプロンプトを表示します。

- ファイル記憶タイプ
- プレフィクス長
- コピーするデータの各カラムのターミネータ
- フィールド長 (char または binary として保管されるフィールドの場合)

これらのプロンプトに対してデフォルト値で応答すると、ネイティブ・フォーマットを使用した場合と同じ結果が得られ、データをデータベースからコピー・アウトし、そのあとで Adaptive Server に簡単に再ロードできます。

他のプログラムで使用するためにデータを Adaptive Server に、または Adaptive Server からコピーする場合は、そのソフトウェアに必要なフォーマットに基づいて、プロンプトに応答してください。

これら 4 つのプロンプトによってシステムが非常に柔軟になっており、他のソフトウェアからファイルを読み込むことや、ほとんどまたはまったく編集を行わずに他の多くのデータ・フォーマットに準拠するファイルを作成することができます。

以降の項では、これらのプロンプトについて説明し、プロンプトに対する応答がデータにどのように作用するかを説明します。

bcp プロンプトに対する応答

-n (ネイティブ・フォーマット) パラメータまたは -c (文字フォーマット) パラメータを使用して、データをコピー・インまたはコピー・アウトする場合、-P パラメータでパスワードを指定していないと、bcp はパスワードを入力するよう要求します。-n パラメータ、-c パラメータ、または -f *formatfile* パラメータのどれも指定していない場合、bcp は、テーブルまたはビューの各フィールドに情報を入力するようにプロンプトを表示します。

- 各プロンプトでは、デフォルト値は角カッコで表示されます。
[Return] キーを押すと、この値を選択できます。プロンプトには、次のものがあります。
 - ファイル記憶タイプ。character データ型または Adaptive Server で有効な任意のデータ型。

- プレフィクス長 (後続のデータの長さをバイトで示す整数) 。
- ファイル内の非 NULL フィールドのデータの記憶長。
- フィールド・ターミネータ (任意の文字列)
- **Windows** – 数値データ型と 10 進数データ型の位取りと精度
ロー・ターミネータは、テーブル、ビュー、またはファイルの最終フィールドのフィールド・ターミネータです。
- 角カッコ内のデフォルト値は、該当するフィールドのデータ型として適切な値を表しています。ファイルヘコピー・アウトする場合の空き領域の最適な使用方法は、次のとおりです。
 - デフォルトのプロンプトを使用する
 - テーブルによって定義されたデータ型にすべてのデータをコピーする
 - 指定どおりにプレフィクスを使用する
 - ターミネータを使用しない
 - デフォルトの長さを使用する

表 4-2 は、bcp のプロンプト、デフォルト、可能なユーザの応答を示します。

表 4-2 : bcp プロンプトのデフォルトとユーザ応答

プロンプト	デフォルト設定	可能なユーザの応答
ファイル記憶タイプ	次の 2 つを除くほとんどのフィールドにデータベース記憶タイプを使用する。 <ul style="list-style-type: none"> • varchar では char • varbinary では binary 	char – 人間が判読できるファイル (暗黙の変換がサポートされる Adaptive Server のデータ型) の作成または読み込みを行う
プレフィクス長	<ul style="list-style-type: none"> • 0 – char データ型 (記憶タイプではない) および固定長のデータ型を指定して定義されたフィールド • 1 – その他のデータ型 • 2 – char として保存された binary および varbinary • 4 – text および image 	プレフィクスが必要でない場合は 0、それ以外の場合はデフォルト

プロンプト	デフォルト設定	可能なユーザの応答
記憶長	<ul style="list-style-type: none"> • char と varchar の場合は、定義されている長さを使用する。 • char として保存した binary および varbinary では、定義された長さの 2 倍の長さを使用する。 • これ以外のデータ型では、トランケーションまたはデータのオーバフローを回避するために必要な最大長を使用する。 	デフォルト値またはそれ以上の値
フィールド・ターミネータまたはロー・ターミネータ	なし	最大 30 文字、または次のうちの 1 つ <ul style="list-style-type: none"> • ¥t - タブ • ¥n - 改行 • ¥r - 復帰改行 • ¥0 - null ターミネータ • ¥ - 円記号

ファイル記憶タイプ

ファイル記憶タイプのプロンプトにより、ファイルへのデータの保管方法を選択できます。データはファイルに以下のデータ型としてコピーできます。

- データベース・テーブルのデータ型
- 文字列型
- 暗黙の変換がサポートされているデータ型

注意 bcp は、ユーザ定義データ型をベース・タイプとしてコピーします。

表 4-3 は、各 Adaptive Server データ型のデフォルト記憶タイプと bcp で有効な省略形を示します。

- ファイル・サイズを最もコンパクトにするには、デフォルト値を使用してください。
- 文字ファイルの場合には、char を使用します。
- date 記憶タイプは、ホスト・オペレーティング・システムのフォーマットの日付ではなく、Adaptive Server 内部記憶フォーマットの datetime です。
- timestamp データは、binary(8) として扱われます。

表 4-3 では、角カッコ [] はワードの頭文字またはワードの始めの何文字かを使用できることを示します。たとえば、“bit” の場合、“b”、“bi”、“bit” のいずれも使用できます。

表 4-3 : bcp のファイル記憶データ型

テーブルでのデータ型	記憶タイプ
char、varchar	c[har]
text	T[ext]
int	i[nt]
smallint	s[mallint]
tinyint	t[inyint]
float	f[loat]
money	m[oney]
bit	b[it]
datetime	d[atetime]
binary、varbinary、timestamp	x
image	I[mage]
smalldatetime	D
real	R
smallmoney	M
numeric	n
decimal	e

bcp を対話形式で使用しているときにこのリストを表示するには、「ファイル記憶タイプを入力してください」プロンプトに対して疑問符 (?) を入力してください。

プロンプトに表示される推奨値がデフォルトです。応答によって、出力ファイルにデータがどのように保管されるかが決まります。データベース・テーブルのカラムのデータ型を示す必要はありません。

暗黙に変換可能な型でも char 型でもないデータ型を入力すると、bcp は失敗します。たとえば、int データに smallint を使用できない場合があります (オーバーフロー・エラーになる可能性があります) が、smallint には int を使用できます。

文字データ型以外のデータ型をデータベース型として保管する場合、bcp は人間が判読できる形式ではなく、ホスト・オペレーティング・システムの Adaptive Server 内部データ表示フォーマットでデータをファイルに書き込みます。

ファイルからデータベース・テーブルに文字フォーマットでデータをコピーする前に、『ASE リファレンス・マニュアル』を参照して、データ型の入力規則をチェックしてください。bcp を使用してデータベースにコピーする文字データは、この規則に従っていなければなりません。特に、(yy)yyymmdd のような区切り文字のない日付形式の場合は、年が最初に指定されていないと、オーバフロー・エラーになる可能性があります。

異なる端末を使用するサイトにホスト・データ・ファイルを送信する場合は、ファイルを作成するときに使用した *datafile_charset* を通知してください。

プレフィクス長

デフォルトでは、bcp は、可変記憶長の各フィールドの先頭にフィールドの長さを示す 1 バイト以上の文字列を付加します。このプレフィクスにより、ファイル・サイズを最もコンパクトにすることができます。プロンプトのデフォルト値は、最も効率のよいプレフィクス長を示しています。

- 固定長フィールドの場合、プレフィクス長は **0** にしてください。
- 255 バイト以下のフィールドの場合、デフォルトのプレフィクス長は **1** です。
- **text** データ型または **image** データ型の場合、デフォルトのプレフィクス長は **4** です。
- **binary** データ型と **varbinary** データ型が **char** 記憶タイプに変換される場合、デフォルトのプレフィクス長は **2** です。これは、テーブル・データが 1 バイトごとに 2 バイトのファイル記憶領域を必要とするためです。
- **binary**、**varbinary**、**image** データのプレフィクスと長さは偶数にしてください。Adaptive Server はこれらのデータを偶数の 16 進数として保管するため、データの一貫性が保たれます。
- **null** 値を使用できるデータ・カラムの場合、**0** 以外のプレフィクス長またはターミネータを使用して、各ローのデータ長を示してください。bcp は、このようなカラムを可変長カラムとみなしません。これには通常、固定長とみなされる整数データ型のカラムも含まれます。
- カラムの前にプレフィクスがないデータの場合は、プレフィクス長を **0** にしてください。

プレフィクス長は、バイト単位で各データ値の長さを表す、1 バイト、2 バイト、または 4 バイトの整数です。プレフィクス長は、ホスト・ファイルのデータ値の直前に指定します。

ターミネータを入力しなかった場合、bcp は次のプロンプト「長さ」で指定された長さになるまで、保管される各フィールドにスペースを埋め込みます。

プレフィクス長は「ネイティブ・フォーマット」の整数であるため、結果のホスト・ファイルには表示できない文字も含まれます。そのため、人間が判読できない文字を扱う通信プログラムを使用しなければ、ホスト・ファイルを表示または送信することはできません。

プレフィクス長の詳細については、表 4-7 (84 ページ) を参照してください。

フィールド長

データをコピー・アウトするときは、ほとんどの場合、記憶長には bcp のデフォルト値を使用してください。

注意 この項でいう「長さ」や「記憶長」とは Adaptive Server のフィールド長ではなく、オペレーティング・システム・ファイルのことを指します。

- Adaptive Server に再ロードするファイルを作成する場合、デフォルトのプレフィクスと長さを使用すると、必要な記憶領域は最小限に保たれます。
- 人間が判読できるファイルを作成する場合にも、デフォルト長を使用すれば、データのトランケーションや、bcp の失敗の原因となるオーバーフロー・エラーの発生を防ぐことができます。

別の値を入力すると、デフォルト長を変更できます。そのため、転送するデータについて把握しておく必要があります。他のソフトウェアから文字データをコピー・インする場合、ソース・ファイルを十分に調べてから長さの値を選択してください。

注意 記憶タイプが文字データ型以外の場合、bcp はデータをオペレーティング・システムのネイティブ・データ表示フォーマットで保管し、長さの入力を要求するプロンプトは表示しません。

bcp は、文字データ以外のデータを文字記憶領域に変換するときに、**datetime** データをトランケートしたり、数値データのオーバーフローを発生させたりしないでデータを十分に保管できるデフォルト・フィールド長を使用します。

- デフォルト長は、Adaptive Server データ型の最も長い値を表示するのに必要なバイト数です。表 4-4 は、データを文字記憶領域に変換するときのデフォルトのフィールド長を示します。

表 4-4 : 文字データ型以外のデータ型から文字データ型への変換におけるデフォルトのフィールド長

データ型	デフォルト・サイズ
int	12 バイト
smallint	6 バイト
tinyint	3 バイト
float	25 バイト
money	24 バイト
bit	1 バイト
datetime	26 バイト
smalldatetime	26 バイト
real	25 バイト
smallmoney	24 バイト

- データをコピー・アウトするときに、数値データに短すぎるフィールド長を指定すると、bcp はオーバーフロー・メッセージを表示し、データのコピーは行いません。
- binary** フィールドおよび **varbinary** フィールドのデフォルト長は、そのカラムに定義されている長さの 2 倍です。これは、フィールドのデータが 1 バイトごとに 2 バイトのファイル記憶領域を必要とするためです。
- デフォルトの記憶長を使用する場合、実際に割り付けられる記憶領域の量は、プレフィクス長とターミネータを指定するかどうかによって変わります。
 - プレフィクス長として 1、2、または 4 を指定した場合、bcp はデータの実際の長さにプレフィクスとターミネータの長さを加えた記憶領域を使用します。

- プレフィクス長として 0 を指定し、ターミネータを指定していない場合、**bcp** はプロンプトに表示された最大記憶領域を割り付けます。これは、対象となるデータ型で必要となる最大領域です。つまり、**bcp** はこのフィールドを固定長フィールドとして扱い、あるフィールドがどこで終わり、次のフィールドがどこから始まるかを判断します。

たとえば、フィールドが `varchar(30)` と定義されている場合、**bcp** は、長さが 1 文字分しかない値があったとしても、各値に 30 バイトを使用します。

- `char`、`nchar`、`binary` としてデータベースで定義されたフィールドと `null` 値を使用できないフィールドには、常にデータベースで定義された通りの長さとなるようにスペース (バイナリには `null` バイト) が埋め込まれます。`timestamp` データは、`binary(8)` データとみなされます。
- `varchar` フィールドおよび `varbinary` フィールドのデータが、コピー・アウト用に指定した長さより長い場合は、**bcp** がファイルのデータを指定された長さに暗黙的にトランケートします。
- すべてのデータをコピーし終わるまで、データ値の大きさはわからないため、**bcp** は必ず指定された長さになるまで `char` データ型にスペースを埋め込みます。
- カラムのファイル記憶タイプおよび長さは、データベース・テーブルのカラムのタイプおよび長さと同じである必要はありません。コピー・インされたタイプとフォーマットがデータベース・テーブルの構造と矛盾する場合、コピーは失敗します。
- 一般的に、ファイル記憶領域の長さは、カラムに転送される最大データ量から、ターミネータかプレフィクスまたはその両方を除いたものです。
- データをテーブルにコピーする場合、**bcp** はカラムに定義されたデフォルトとユーザ定義データ型を調べます。しかし、**bcp** は可能な限り迅速にデータをロードするためにルールを無視します。
- **bcp** は、`null` 値を含むデータ・カラムを可変長と認識するので、プレフィクス長またはターミネータのどちらかを使用して各ローのデータの長さを記述してください。
- カラムのファイル記憶タイプおよび長さは、データベース・テーブルのカラムのタイプおよび長さと同じである必要はありません (コピー・インされたタイプとフォーマットがデータベース・テーブルの構造と矛盾する場合、コピーは失敗します)。

フィールド・ターミネータとロー・ターミネータ

ターミネータを使用すると、カラムまたはローの終わりをマーク付けし、次のローまたはカラムとの区切りを示すことができます。デフォルトは「ターミネータなし」です。

- フィールド・ターミネータはテーブル・カラムを区切る文字です。
- ロー・ターミネータは、テーブルまたはファイルの各ローの最終フィールドに使用するフィールド・ターミネータです。

人間が判読できるターミネータを選択できるので、ターミネータは文字データを扱う場合に非常に便利です。各カラムの間にはタブを使用し、各ローの最後には改行ターミネータを使用する **bcp** 文字オプションは、データ・ファイルの読みやすさを向上させるターミネータの使用法の例です。

他のプログラムで使用するデータを作成する場合や **bcp** を使用してデータを表形式にする場合には、ユーザ固有のターミネータを使用してください。ターミネータとして使用可能な文字は次のとおりです。

- `¥t` で示されるタブ
- `¥n` で示される改行文字
- `¥r` で示される復帰改行文字
- `¥` で示される円記号
- `¥0` で示される null ターミネータ (見えないターミネータ)
- 印刷可能な任意の文字 (*、A、t、| など)
- 上記のターミネータを含む最大 10 文字の印刷可能な文字で構成される文字列 (たとえば、**¥t**、end、!!!!!!!!!!、¥t--¥n など)

注意 制御文字 (ASCII の 0 ~ 25) は出力できません。

ターミネータの選択

データのどこにも使用されていないパターンをターミネータとして選択してください。

たとえば、タブを含むデータの文字列にタブ・ターミネータを使用すると、どのタブがデータ文字列の終わりを表すのかがわかりにくくなります。この場合、どのタブが文字列の終わりを表すことになるのでしょうか。 **bcp** は必ず最初のターミネータを探しますが、この場合は最初に検出されるタブはデータ文字列の一部であり、ターミネータではありません。

ネイティブ・フォーマットのデータもターミネータと競合することがあります。ネイティブ・フォーマットの 4 バイト整数が含まれているカラムの場合、これらの整数値が厳密に制限されていないかぎり、データ内部で絶対に使用されていないターミネータを選択することは不可能です。そのため、ネイティブ・フォーマットのデータには **bcp** のネイティブ・フォーマット・オプションを使用してください。

注意 「ターミネータなし」と「null ターミネータ」は異なることに注意してください。null ターミネータは目には見えませんが、実際の文字です。

- フィールド・ターミネータ文字列は、30 文字まで指定できます。最も一般的なターミネータは、タブ (¥t と入力し、最後のカラム以外のすべてのカラムに使用される) と改行 (¥n と入力し、ローの最終フィールドに使用される) です。その他、¥o (null ターミネータ)、¥ (円記号)、¥r (復帰改行) があります。**bcp** は、ターミネータとして指定された文字列が最初に出現したときに、それをターミネータと認識するので、ターミネータを選択する場合は、使用している文字データでそのパターンが使用されていないことを確認してください。

たとえば、タブを含む文字列に対してタブをターミネータとして使った場合、**bcp** はどのタブが文字列の終わりを指しているかを判断できません。**bcp** は常に、ターミネータとして指定された文字列が最初に出現したときにターミネータと判断するため、この例ではターミネータではないものをターミネータとして判断してしまいます。

ターミネータまたはプレフィクスは、転送するデータの実際の長さに影響します。

ターミネータやプレフィクスがある場合、それは転送するデータの長さに影響します。ファイルにコピー・アウトするエントリの長さが記憶領域の長さより短い場合、次のフィールドに対するターミネータまたはプレフィクスがこのすぐあとに続きます。この場合、エントリに記憶領域の長さ分の埋め込みは行われません (**char**、**nchar**、**binary** データは、Adaptive Server から返されるときすでに、いっぱい長さまで埋め込みが行われています)。

ファイルからコピー・インする場合、bcp は、「長さ」プロンプトで示されるバイト数に到達するか、ターミネータを検出するまでデータを転送します。指定された長さ分のデータが転送されると、bcp はターミネータを検出するまで残りのデータをフラッシュします。ターミネータがない場合、テーブルの記憶領域の長さが使用されます。

- フィールドが、データベースのデータ型ではなく char (char、nchar、binary を除く) として保管されている場合は、デフォルト長とプレフィクス、またはターミネータに使用する記憶領域が少なくても済みます。bcp では、最も効率のよい記憶領域の使用を決めるターミネータまたはプレフィクスを使用できます。また、bcp は、各フィールドで必要とされる記憶領域の最大容量をデフォルトで推奨します。char データまたは varchar データの場合は、bcp はどの長さも使用できます。
- 表 4-5 と 表 4-6 に、ファイルに入っている情報のプレフィクス長、ターミネータ、およびフィールド長の関係を示します。“P” は保管されたテーブルのプレフィックスを、“T” はターミネータを、2 つのダッシュ (--) は追加領域をそれぞれ表します。省略記号 (...) は、各フィールドでパターンが繰り返されることを表します。フィールドの長さはカラムごとに 8 バイトです。“string” は、それぞれ 6 文字のフィールドを表します。

表 4-5 : Adaptive Server の char データ

	プレフィクス長 = 0	プレフィクス長 = 1、2、 または 4
ターミネータなし	string--string--...	Pstring--Pstring--...
ターミネータ	string--Tstring--T...	Pstring--TPstring--T...

表 4-6 : char 記憶領域に変換された他のデータ型

	プレフィクス長 = 0	プレフィクス長 = 1、2、 または 4
ターミネータなし	string--string--...	PstringPstring...
ターミネータ	stringTstringT...	PstringTPstringT...

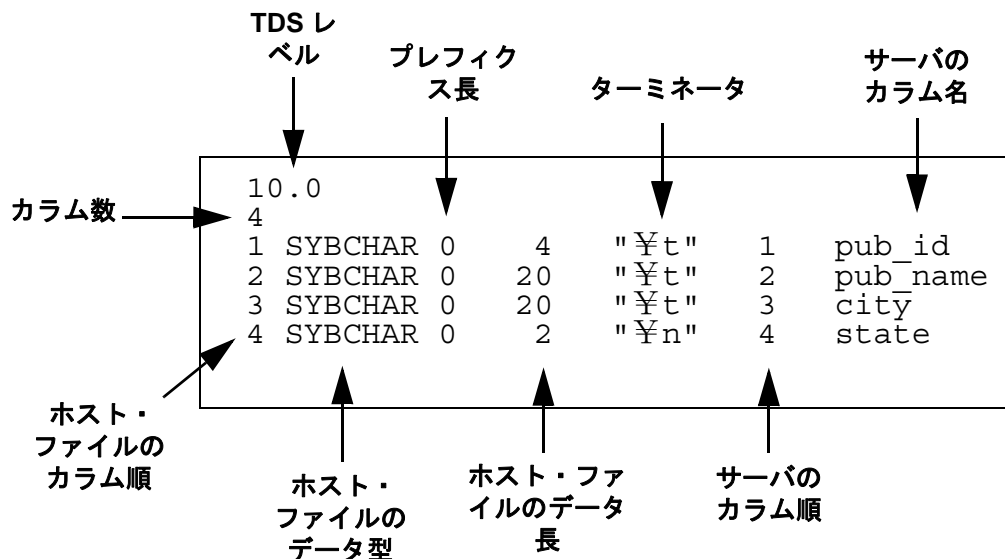
フォーマット・ファイル

bcp は、テーブル内の各フィールドに関する情報を収集してから、その情報を「フォーマット・ファイル」に保存するかどうかを確認し、ファイル名を入力するよう要求します。

bcp ユーティリティを使用してコピーするデータ用に作成されたフォーマット・ファイルを使用すると、非対話型方式でデータをコピー・インまたはコピー・アウトできます。つまり、bcp が情報を入力するよう要求することはありません。bcp は、必要な情報をこのフォーマット・ファイルから取得するからです。この新しく作成したフォーマット・ファイルは、データを Adaptive Server にコピーし戻したり、テーブルからコピー・アウトするとき可以使用します。

図 4-3 は、bcp フォーマット・ファイルのフォーマットを示します。この図は、pubs2 データベースの publishers テーブルを示しています。ホスト・ファイル・カラムはすべて文字フォーマットで、プレフィクスはなく、デフォルトのデータ長を使用しています。ローの最終カラムの末尾には改行ターミネータを使用し、他のすべてのカラムにはターミネータとしてタブを使用しています。

図 4-3 : bcp フォーマット・ファイル



bcp フォーマット・ファイルの要素

次のリストに bcp フォーマット・ファイルの要素を示します。図 4-3 は、フォーマット・ファイルの例としてお使いください。

- ファイルの最初の行は、常に Tabular Data Stream (TDS) のバージョンです。これは、Adaptive Server のバージョンではなく、使用中の TDS のバージョンを表し、引用符なしのリテラル文字列です。図 4-3 の場合、バージョンは 10.0 です。
- bcp フォーマット・ファイルの 2 行目は、カラム数です。このカラム数は 1 行目と 2 行目を除くフォーマット・ファイルのレコードの数を表します。ホスト・テーブルの各カラムはそれぞれ 1 行で表されています。
- 1 行目と 2 行目の下に、データベース・テーブルの各カラムごとに 1 行ずつ続きます。各行は通常、タブで区切られた要素でできています。ただし、ホスト・ファイルのデータ型とプレフィクス長はスペースで区切られます。これらの要素は次のとおりです。
 - ホスト・ファイルのカラム順
 - ホスト・ファイルのデータ型
 - プレフィクス長
 - ホスト・ファイルのデータ長
 - ターミネータ
 - サーバのカラム順
 - サーバのカラム名
 - カラムの精度
 - カラムの位取り

以降の項では、フォーマット・ファイルのカラム要素について説明します。

ホスト・ファイルのカラム順

ホスト・ファイルのカラム順は、ホスト・データ・ファイルのフィールドの 1 から始まる連続した番号です。

ホスト・ファイルの データ型

ホスト・ファイルのデータ型は、データベース・テーブル・カラムのデータ型のことではなく、ホスト・データ・ファイルのフィールドの記憶フォーマットを指します。ホスト・ファイルのデータ型と記憶フォーマットは、次のとおりです。

- SYBCHAR — charE, char, varchar (ASCII), nchar, nvarchar
- SYBTEXT — text
- SYBBINARY — binary, timestamp, unichar, univarchar, varbinary
- SYBIMAGE — image
- SYBINT1 — tinyint
- SYBINT2 — smallint
- SYBINT4 — int
- SYBINT8 — bigint
- SYBFLT8 — float
- SYBREAL — real
- SYBBIT — bit
- SYBNUMERIC — numeric
- SYBDECIMAL — decimal
- SYBMONEY — money
- SYBMONEY4 — smallmoney
- SYBDATETIME — datetime
- SYBDATETIME4 — smalldatetime
- SYBDATE — date
- SYBTIME — time
- SYBUINT8 — unsigned bigint
- SYBUINT4 — unsigned int
- SYBUINT2 — unsigned smallint
- SYBUNITEXT — unitext
- SYBFLT8 — double

ネイティブ・フォーマットでホスト・ファイルに書き込まれたデータは、その精度をすべて保持します。datetime 値および float 値は、文字フォーマットに変換されるときにも精度をすべて保持します。

Adaptive Server は、通貨単位の 1 万分の 1 の精度で money 値を保管します。しかし、money 値が文字フォーマットに変換される場合は、文字フォーマット値は、近似値 2 桁しか記録されません。

Adaptive Server データ型の詳細と適切な使用方法については、『リファレンス・マニュアル：ビルディング・ブロック』の「第 1 章 システム・データ型とユーザ定義データ型」を参照してください。

プレフィクス長

プレフィクス長は、フィールド長プレフィクスのバイト数を指します。このプレフィクス長は、フィールド内の実際のデータ長を表す 0、1、2、または 4 バイトの符号なし整数値で、ホスト・データ・ファイルに埋め込まれています。フィールドには、長さプレフィクスを含むものと含まないものがあります。

表 4-7 に、有効なプレフィクス長の値を示します。

表 4-7：有効なプレフィクス長の値

長さ (バイト数)	範囲
0	プレフィクスなし
1	2^8-1 ; 0-255
2	$2^{16}-1$; 0-65535
4	$2^{32}-1$; 0-4,294,967,295

ホスト・ファイルのデータ長

ホスト・ファイルのデータ長は、フィールドにコピーする最大バイト数を指します。bcp は、どれくらいの量のデータをコピー・インまたはコピー・アウトするかを決定するのに、次の項目を使用します。

- 最大フィールド長
- プレフィクス長 (指定されている場合)
- フィールド・ターミネータ文字列 (指定されている場合)

フィールド長が複数の方法で指定されている場合、bcp はコピーするデータ量が最も少ない方法を選択します。

ターミネータ

ターミネータは、引用符 (" ") で囲まれた最大 30 バイトの文字列です。ターミネータは、ホスト・データ・ファイルにあるフィールドのデータの終了を示します。

サーバのカラム順

サーバのカラム順は、ホスト・データ・ファイルのカラムがロードされる syscolumns カラムの colid (カラム ID) を表します。この要素とホスト・ファイルのカラム順によって、ホスト・データ・ファイルのフィールドがデータベース・テーブル・カラムにマップされます。

サーバのカラム名	サーバのカラム名は、このフィールドがロードされるデータベース・テーブル・カラムの名前です。
カラムの精度	カラムの精度は、このフィールドがロードされるデータベース・テーブル・カラムの精度です。この要素は、記憶フォーマットが <code>numeric</code> または <code>decimal</code> の場合にだけ存在します。
カラムの位取り	カラムの位取りは、このフィールドがロードされるデータベース・テーブル・カラムの位取りです。この要素は、記憶フォーマットが <code>numeric</code> または <code>decimal</code> の場合にだけ存在します。

データを対話形式でコピー・アウトする例

`bcp` プロンプトのデフォルト値を変更すると、他のソフトウェアで使用するデータを作成できます。

人間が判読できるファイルを作成するには、`bcp` プロンプトに対して次のように応答してください。

- ファイル記憶タイプには `0` を入力します。
- プレフィクス長には `0` を入力します。
- フィールド長はデフォルトのままにします。
- ターミネータ - 入力するフィールド・ターミネータは、使用するソフトウェアによって異なります。
- 区切られるフィールドまたは固定長フィールドの間に使用します。最後のフィールドの終わりには、必ず改行ターミネータの `¥n` を使用してください。

固定長フィールドの場合、ターミネータは使用しないでください。各フィールドは固定長であり、指定された長さになるまでフィールドにスペースが埋め込まれます。出力の各行にはフィールド・セパレータがないため、前のフィールドがデータでいっぱいになると、隣り合うフィールドが1つのフィールドのように見えます。後述の例を参照してください。

- カンマ区切りの出力の場合は、カンマ (,) を各フィールドのターミネータとして使用してください。表形式の出力を作成する場合は、タブ文字 (¥t) をターミネータとして使用してください。

フィールド長を指定してデータをコピー・アウトする例

次の例では、固定長フィールドを使用して、SDF (システム・データ・フォーマット) と呼ばれるパーソナル・コンピュータ・フォーマットで出力を作成します。このフォーマットは、他のソフトウェアでも簡単に読み込んだり作成したりできます。

注意 フォーマット・ファイルの詳細については、「[フォーマット・ファイル](#)」(81 ページ) を参照してください。

```
bcp pubs2..sales out sal_out
```

sal_out ファイルに保管される結果は次のとおりです。

```
5023      AB-123-DEF-425-1Z3      Oct 31 1985 12:00AM
5023      AB-872-DEF-732-2Z1      Nov  6 1985 12:00AM
5023      AX-532-FED-452-2Z7      Dec  1 1990 12:00AM
5023      BS-345-DSE-860-1F2      Dec 12 1986 12:00AM
5023      GH-542-NAD-713-9F9      Mar 15 1987 12:00AM
5023      NF-123-ADS-642-9G3      Jul 18 1987 12:00AM
5023      XS-135-DER-432-8J2      Mar 21 1991 12:00AM
5023      ZA-000-ASD-324-4D1      Jul 27 1988 12:00AM
5023      ZD-123-DFG-752-9G8      Mar 21 1991 12:00AM
5023      ZS-645-CAT-415-1B2      Mar 21 1991 12:00AM
5023      ZZ-999-ZZZ-999-0A0      Mar 21 1991 12:00AM
6380      234518                   Sep 30 1987 12:00AM
6380      342157                   Dec 13 1985 12:00AM
6380      356921                   Feb 17 1991 12:00AM
7066      BA27618                   Oct 12 1985 12:00AM
7066      BA52498                   Oct 27 1987 12:00AM
7066      BA71224                   Aug  5 1988 12:00AM
7067      NB-1.142                 Jan  2 1987 12:00AM
7067      NB-3.142                 Jun 13 1990 12:00AM
7131      Asoap132                  Nov 16 1986 12:00AM
7131      Asoap432                  Dec 20 1990 12:00AM
7131      Fsoap867                  Sep  8 1987 12:00AM
7896      124152                   Aug 14 1986 12:00AM
7896      234518                   Feb 14 1991 12:00AM
8042      12-F-9                    Jul 13 1986 12:00AM
8042      13-E-7                    May 23 1989 12:00AM
8042      13-J-9                    Jan 13 1988 12:00AM
8042      55-V-7                   Mar 20 1991 12:00AM
8042      91-A-7                   Mar 20 1991 12:00AM
8042      91-V-7                   Mar 20 1991 12:00AM
```

フォーマット・ファイル *sal_fmt* の内容は次のとおりです。

```
10.0
3
1 SYBCHAR 04 "" 1 stor_id
2 SYBCHAR 020 "" 2 ord_num
3 SYBCHAR 026 "" 3 date
```

フォーマット・ファイルの詳細については、「[フォーマット・ファイル](#)」(81 ページ)を参照してください。

デリミタを指定してデータをコピー・アウトする例

次の例では、bcp は `publishers` テーブルからファイルに対話形式でデータをコピーします。

注意 フォーマット・ファイルの詳細については、「[フォーマット・ファイル](#)」(81 ページ)を参照してください。

フォーマット・ファイルによるカンマ区切りおよび改行区切りの出力の作成

最初の例では、ローのすべてのフィールド間にカンマを挿入し、各ローの終わりには改行ターミネータを挿入して、出力ファイルを作成します。この例では、同じデータまたは類似するデータを Adaptive Server にコピーし戻すときに使用できるフォーマット・ファイル (*pub_fmt*) を作成します。

```
bcp pubs2..publishers out pub_out
```

pub_out ファイルに保管される結果は次のとおりです。

```
0736,New Age Books,Boston,MA
0877,Binnet & Hardley,Washington,DC
1389,Algodata Infosystems,Berkeley,CA
```

フォーマット・ファイル *pub_fmt* の内容は次のとおりです。

```
10.0
4
1 SYBCHAR 0 4 ", " 1 pub_id
2 SYBCHAR 0 40 ", " 2 pub_name
3 SYBCHAR 0 20 ", " 3 city
4 SYBCHAR 0 2 "¥n" 4 state
```

フォーマット・ファイルによるタブ区切り出力の作成

同様に、次の例では `pubs2..publishers` テーブルから `pub_out` ファイルにタブ区切りで出力します。

```
bcp pubs2..publishers out pub_out
```

`pub_out` ファイルに保管される結果は次のとおりです。

0736	New Age Books	Boston	MA
0877	Binnet & Hardley	Washington	DC
1389	Algodata Infosystems	Berkeley	CA

フォーマット・ファイル `pub_fmt` の内容は次のとおりです。

```
10.0
4
1 SYBCHAR 04 "¥t" 1 pub_id
2 SYBCHAR 040 "¥t" 2 pub_name
3 SYBCHAR 020 "¥t" 3 city
4 SYBCHAR 02 "¥n" 4 state
```

データを対話形式でコピー・インする例

ファイルからテーブルに正常にデータをコピーするには、ファイル内で使用されているターミネータまたはフィールド長をあらかじめ確認しておき、`bcp` を使用するときはその値を指定する必要があります。

次の例では、`bcp` を使用してデータを固定長フィールドまたはデリミタ指定のいずれかでコピー・インする方法を、フォーマット・ファイルを使用する方法と、使用しない方法に分けて説明します。

フィールド長を指定してデータをコピー・インする例

この例では、`bcp` は `salesnew` ファイルから `pubs2..sales` テーブルにデータをコピーします。

`salesnew` ファイルには3つのフィールドがあります。最初のフィールドは4文字長、2番目のフィールドは20文字長、3番目のフィールドは26文字長です。次に示すように、各ローは改行ターミネータ (`¥n`) で終了しています。


```

5023ZS-731-AAB-780-2B9      May 24 1993 12:00:00:000AM
5023XC-362-CFB-387-3Z5      May 24 1993 12:00:00:000AM
6380837206                  May 24 1993 12:00:00:000AM
6380838441                  May 24 1993 12:00:00:000AM

```

salesnew から対話形式でデータをコピーするには、次のコマンドを使用してください。

```
bcp pubs2..sales in salesnew
```

この **bcp** コマンドに対するシステムの応答は次のとおりです。

```

Password:
Enter the file storage type of field stor_id [char]:
Enter prefix-length of field stor_id [0]:
Enter length of field stor_id [4]:
Enter field terminator [none]:
Enter the file storage type of field ord_num [char]:
Enter prefix-length of field ord_num [1]:0
Enter length of field ord_num [20]:
Enter field terminator [none]:
Enter the file storage type of field date [datetime]:char
Enter prefix-length of field date [1]:0
Enter length of field date [26]:
Enter field terminator [none]:Yn
Do you want to save this format information in a file?[Y/n] y
Host filename [bcp.fmt]:salesin_fmt
Starting copy...
4 rows copied.
Clock Time (ms.):total = 1 Avg = 0 (116000.00 rows per sec.)

```

Adaptive Server にログインして *sales* にアクセスすると、このテーブルに追加された *salesnew* から次のようなデータを参照できます。

```

select * from sales
stor_id  ord_num      date
-----
5023     AB-123-DEF-425-1Z3  Oct 31 1985 12:00AM
5023     AB-872-DEF-732-2Z1  Nov 6 1985 12:00AM

5023     AX-532-FED-452-2Z7  Dec 1 1990 12:00AM
5023     BS-345-DSE-860-1F2  Dec 12 1986 12:00AM
5023     GH-542-NAD-713-9F9  Mar 15 1987 12:00AM
5023     NF-123-ADS-642-9G3  Jul 18 1987 12:00AM
5023     XS-135-DER-432-8J2  Mar 21 1991 12:00AM
5023     ZA-000-ASD-324-4D1  Jul 27 1988 12:00AM
5023     ZD-123-DFG-752-9G8  Mar 21 1991 12:00AM

```

```

5023      ZS-645-CAT-415-1B2   Mar 21 1991 12:00AM
5023      ZZ-999-ZZZ-999-0A0   Mar 21 1991 12:00AM
6380      234518                Sep 30 1987 12:00AM
6380      342157                Dec 13 1985 12:00AM
6380      356921                Feb 17 1991 12:00AM
7066      BA27618              Oct 12 1985 12:00AM
7066      BA52498              Oct 27 1987 12:00AM
7066      BA71224              Aug  5 1988 12:00AM
7067      NB-1.142             Jan  2 1987 12:00AM
7067      NB-3.142             Jun 13 1990 12:00AM
7131      Asoap132             Nov 16 1986 12:00AM
7131      Asoap432             Dec 20 1990 12:00AM
7131      Fsoap867             Sep  8 1987 12:00AM
7896      124152                Aug 14 1986 12:00AM
7896      234518                Feb 14 1991 12:00AM
8042      12-F-9                 Jul 13 1986 12:00AM
8042      13-E-7                 May 23 1989 12:00AM
8042      13-J-9                 Jan 13 1988 12:00AM
8042      55-V-7                 Mar 20 1991 12:00AM
8042      91-A-7                 Mar 20 1991 12:00AM
8042      91-V-7                 Mar 20 1991 12:00AM
(34 rows affected)

```

`sales` の `stor_id` カラムと `ord_num` カラムにはユニークなクラスタード・インデックスがあるため、新しいローは順序どおりにソートされています。

重複や違反があると、コピー・プロセスは次のような影響を受けます。

- ファイルからコピーされたデータの列にユニーク・インデックス違反があると、`bcp` は違反しているローがあるバッチ全体を破棄します。

バッチ・サイズを 1 にすると、各ローは個別に評価されますが、ロード速度が遅くなり、高速または高速にログを取る `bcp` セッション中にローごとに別々のデータ・ページが作成されます。
- コピー・インするデータ型とデータベースのデータ型に互換性がない場合、コピー全体が失敗します。

デリミタを指定してデータをコピー・インする例

この例では、bcp は *newpubs* ファイルから *pubs2..publishers* テーブルにデータをコピーします。*newpubs* ファイルでは、例に示すように、ローの各フィールドはタブ文字 (¥t) で終了し、各ローは改行ターミネータ (¥n) で終了します。

1111	Stone Age Books	Boston	MA
2222	Harley & Davidson	Washington	DC
3333	Infodata Algosystems	Berkeley	CA

newpubs はすべて文字データなので、次に示すように文字コマンド・ライン・フラグを使用し、コマンド・ライン・オプションでターミネータを指定できます。

- UNIX プラットフォームの場合：

```
bcp pubs2..publishers in newpubs -c -t¥¥t -r¥¥n
```

- Windows の場合：

```
bcp pubs2..publishers in newpubs -c -t¥t -r¥n
```

フォーマット・ファイルを指定してデータをコピー・インする例

保存されている *pub_fmt* フォーマット・ファイルを使用して Adaptive Server にデータをコピーし戻すには、次のコマンドを実行します。

```
bcp pubs2..publishers in pub_out -fpub_fmt
```

pub_fmt ファイルを使用すると、同じフォーマットの任意のデータを Adaptive Server にコピーできます。デリミタが異なる同様のデータ・ファイルがある場合、フォーマット・ファイル内のデリミタを変更できます。

同様に、すべてのフィールドが同じ長さであれば、フォーマット・ファイルを編集し、フィールド長に対して行った変更を反映させることができます。たとえば、*moresales* ファイルに次のようなデータが入っているとします。

```
804213-L-9 Jan 21 1993 12:00AM
804255-N-8 Mar 12 1993 12:00AM
804291-T-4 Mar 23 1993 12:00AM
804291-W-9 Mar 23 1993 12:00AM
```

フォーマット・ファイル *sal_fmt* を次のように編集します。

```
10.0
3
1 SYBCHAR 0 4 "" 1 stor_id
2 SYBCHAR 0 7 "" 2 ord_num
3 SYBCHAR 0 21 "¥n" 3 date
```

次のようにコマンドを入力してください。

- UNIX プラットフォームの場合 :

```
bcp pubs2..sales in moresales -fsal_fmt
```
- Windows の場合 :

```
bcp pubs2..sales in moresale -fsal_fmt
```

システムの応答は次のとおりです。

```
Starting copy...
4 rows copied.
Clock Time (ms.):total = 1 Avg = 0 (116000.00 rows
per sec.)
```

bcp と代替言語

Adaptive Server は、インストール中に設定されたデフォルト文字セットを使用してデータを保管します。使用している端末がこのデフォルト文字セットをサポートしていない場合、文字を入力したりホスト・ファイルのスクリプトを使用してプロンプトに応答すると、文字化けして **bcp** に送信されることがあります。

すべての文字セット・オプションを省略すると、**bcp** はそのプラットフォームでデフォルトとして設定された文字セットを使用します。このデフォルト設定では、次のような理由から通信上の問題が生じる可能性があります。

- デフォルトの文字セットは、必ずしも Adaptive Server に設定された文字セットと同じであるとはかぎりません。
- デフォルトの文字セットは、クライアントが使用する文字セットと同じであるとはかぎりません。

文字セットとそれに関連するフラグの詳細については、『システム管理ガイド』の「第 8 章 クライアント／サーバの文字セット変換の設定」を参照してください。

初期化文字列に対するサポート

bcp ユーティリティでは、データを転送する前に `set replication off` などの Transact-SQL コマンドを Adaptive Server に送信することがサポートされます。

任意の Transact-SQL コマンドを bcp の初期化文字列として使用できませんが、サーバ設定への可能な永続的変更は、bcp の実行後にリセットする必要があります。たとえば、別の isql セッションで変更をリセットできます。

bcp とロー・レベルのアクセス・ルール

ロー・レベルのアクセスが可能であるように Adaptive Server が設定されていて、データをバルク・コピー・アウトする場合、bcp はアクセスしているデータのローのみをコピー・アウトします。テーブル全体をコピー・アウトするには、まずアクセス・ルール、次に `bcp out` を削除します。その後、必要に応じてアクセス・ルールを復旧します。

アクセス・ルールが使用可能なテーブルにデータをバルク・コピー・インすると、「一意性違反」エラーが発行されます。たとえば、テーブルにアクセス・ルールが作成される前に生成された bcp データ・ファイルからデータをロードし、かつその bcp データ・ファイル内に以前テーブルに挿入されたローがある場合に、このタイプのエラーが表示されることがあります。

このエラーが発生すると、一意性違反のために、表示されるテーブルには bcp の挿入に失敗したローは含まれません。アクセス・ルールのため、ユーザはこのような「見えない」ローにアクセスすることはできません。

テーブル全体にコピー・インするには、アクセス・ルールを削除し、データをロードし、エラーを解決した後、アクセス・ルールを復旧します。

コピー・インとバッチ・ファイル

バッチ処理はバルク・コピー・インだけに適用され、コピー・アウトに影響を与えることはありません。デフォルトでは、Adaptive Server は 1000 行のバッチに指定されたすべてのローをコピーします。別のバッチ・サイズを指定するには、コマンド・ライン・オプション (-b) を使用します。

bcp は各バッチを 1 つのトランザクションの中でコピーします。Adaptive Server がバッチ内のローを拒否すると、トランザクション全体がロール・バックされます。デフォルトでは、bcp はすべてのローを 1 つのバッチでコピーします。-b パラメータを使用すると、デフォルトのバッチ・サイズを変更できます。Adaptive Server は各バッチを 1 つの bcp オペレーションとみなし、各バッチを個別のデータ・ページに書き込んで、先行のトランザクションが成功したかどうかに関係なく次のバッチに進みます。

データをコピー・インするときは、Adaptive Server または bcp はそのデータを拒否できます。

- Adaptive Server は、各バッチを個別のトランザクションとして扱います。サーバがバッチ内のローを拒否すると、トランザクション全体がロールバックされます。
- bcp は、バッチを拒否すると次のバッチに進みます。トランザクションがロールバックされるのは、致命的なエラーが発生した場合だけです。
- Adaptive Server はローごとではなくバッチごとにエラー・メッセージを生成し、エラーが検出されたバッチを拒否します。エラー・メッセージは端末に表示され、エラー・ファイルに保存されます。

リカバリ性の向上

リカバリ性を高めるには、次のような方法があります。

- 大規模な入力ファイルを小さい単位に分割する。

たとえば、bcp を使用して 100,000 ローのバッチ・サイズで 300,000 ローをバルク・コピー・インした場合、200,000 ロー以降で致命的エラーが発生しても、最初の 2 つのバッチ (200,000 ロー) は Adaptive Server に正常にコピーされます。この場合、バッチを使用していなければ、bcp は Adaptive Server にローを 1 つもコピーできません。

- `trunc log on chkpt` を `true` (オン) に設定する。

これにより、バッチ終了後に、トランザクション用のログ・エントリをトランケーション用に使用できます。`trunc log on chkpt` データベース・オプションがオン (`true`) に設定されているデータベースにコピー・インする場合、終了したバッチのログ・エントリは次の自動チェックポイントで削除されます。この結果、大規模な `bcp` オペレーションは分割され、ログがいっぱいになるのを防ぎます。

- `-b batch_size` を 10 に設定する。

このバッチ・サイズ・パラメータを 10 に設定すると、`bcp` は不完全なローを含む 10 ローのバッチを拒否します。そのため、エラー・ログを調べると、どのローが失敗したのかを正確に識別できます。

`bcp` プロセスの最小のバッチサイズは 1 ローです。

注意 `bcp` はバッチごとにデータ・ページを 1 つ作成するため、`b batch_size` を 10 に設定すると、1 ページあたり 10 ローを含むデータ・ページが作成されます。`-b batch_size` を 1 に設定すると、1 ページあたり 1 ローを含むデータ・ページが作成されます。この設定では、データのロードが遅くなり、記憶領域の使用量も多くなります。

バッチと分割されたテーブル

分割されたテーブルに分割番号を指定しないでデータをバルク・コピーする場合、Adaptive Server は各バッチを使用可能な分割にランダムに割り当てます。1 つのバッチでローをコピーすると、すべてのローが単一の分割に挿入されるので、分割されたテーブルでロードのバランスが悪くなります。

分割されたテーブルのバランスを保つために、データをバルク・コピーするときには小さなバッチ・サイズを使用するか、`bcp` セッション中にパーティション ID を指定してください。テーブルの分割に関する詳細については、『パフォーマンス&チューニング・シリーズ』を参照してください。

text または image データのコピー・アウト

text データまたは image データをコピー・アウトする場合、デフォルトでは Adaptive Server は text フィールドまたは image フィールド内の最初の 32K のデータだけをコピーします。-T *text_or_image_size* パラメータを使用して、別の値を指定することもできます。たとえば、コピー・アウトする text フィールドに 40K のデータが含まれている場合は、次のコマンドを使用して 40K のデータすべてをコピー・アウトできます。

```
bcp pubs2..publishers out -T40960
```

注意 text フィールドまたは image フィールドが指定の値またはデフォルト値より大きい場合、bcp はオーバフローした残りのデータをコピー・アウトしません。

ネットワーク・パケット・サイズの指定

大量のバルク・コピー・オペレーションのパフォーマンスを向上させるために、デフォルトより大きいネットワーク・パケット・サイズを使用できます。このような場合には、-A *size* オプションで、開始する bcp セッションで使用するネットワーク・パケット・サイズを指定します。

size に指定する値は、次の条件を満たしている必要があります。

- default network packet size 設定パラメータと max network packet size 設定パラメータの間の値である。
- 512 の倍数である。

注意 新しいパケット・サイズは、現在の bcp セッションでのみ有効です。

たとえば、次のコマンドでは、Adaptive Server は bcp セッションで 4096 バイトのパケット・サイズを使用して、40K の text データまたは image データを送信します。

```
bcp pubs2..authors out -A 4096 -T40960
```


コピー・インとエラー・ファイル

コピー・インで `-e error_file` オプションを指定すると、`bcp` は Adaptive Server にコピー・インできないローを指定されたエラー・ファイルに保管します。

- エラー・ファイルには次のような要素が保管されます。
 - 失敗したローと発生したエラーを示す 1 行
 - ホスト・ファイルから正確にローをコピーした 1 行
- `-e` の後に指定したファイルがすでに存在する場合、`bcp` はこのファイルを上書きします。
- エラーが発生しなければ、`bcp` はこのファイルを作成しません。

`bcp in` は、次の 2 種類のエラーを検出します。

- データ変換エラー
- ローの構築エラー。たとえば、NULL 値を受け付けないカラムへの NULL の挿入、3 バイト整数などの無効なデータ・フォーマットの使用などによって発生します。

コピー・イン・プロセスでは、エラー・メッセージはモニタ上に表示されます。

次の例では、`newpubs` ファイルを `publishers` データベースにロードし、エラーがあれば、そのエラー・ローを `pub_err` ファイルに保管します。

```
bcp pubs2..publishers in newpubs -epub_err
```

コピー・イン時に生成されたエラー・ファイルを参照するときには、次の点に注意してください。

- `bcp` は、`bcp` プログラム自体がエラーを検出した場合にだけエラー・ファイルにローを保管します。
- `bcp` はエラー・ローの最大数に達するまでローのコピーを続け、最大数に達するとコピーを中止します。
- `bcp` はローをバッチで Adaptive Server に送信するため、ユニーク・インデックスを持つテーブルの重複ローなど、Adaptive Server が拒否したローのコピーは保存できません。
- Adaptive Server は、エラーを検出すると、ローごとではなくバッチごとエラー・メッセージを生成し、そのバッチ全体を拒否します。

- テーブルのインデックスが作成されたときに、`allow_dup_row` または `ignore_dup_key` が設定されていれば、Adaptive Server が重複ローを拒否してもエラーとはみなされません。この場合、コピーは正常に続行されますが、重複ローは、テーブルにも `bcp` エラー・ファイルにも保管されません。

コピー・アウトとエラー・ファイル

コピー・インの場合と同様にコピー・アウト・プロセスにおいても、`bcp` は同じ名前の任意のファイルを上書きし、エラーが発生しなければ、エラー・ファイルは作成しません。

コピー・アウト中にエラー・ファイルにローのログが記録されるのは、次の2つのエラーが発生した場合です。

- ローのいずれかのカラムで発生したデータ変換エラー
- ホスト・ファイルへの書き込み中に発生した I/O エラー

コピー・アウト時に生成されたエラー・ファイルを参照するときには、次の点に注意してください。

- `bcp` はデフォルトの文字フォーマットでエラー・ファイルにローのログを記録します。
- すべてのデータ値は、カラム間にはタブ、各ローの終わりには改行ターミネータが入った文字フォーマットで出力されます。

デフォルト、ルール、トリガのデータの整合性

データの整合性を保つために、`bcp` はコピーするデータをその要素に応じて処理します。

デフォルトとデータ型

データをテーブルにコピーするとき、**bcp** はカラムおよびデータ型に定義されている任意のデフォルトを参照します。つまり、ファイル内のデータに **null** フィールドがある場合、**bcp** はコピーするときに **null** 値の代わりにデフォルト値をロードします。

たとえば、**authors** にロードされるファイルには、次のような2つのローがあります。

```
409-56-7008,Bennet,David,415 658-9932,622 Pine
St.,Berkeley,CA,USA,94705213-46-8915,Green,Marjorie,,309 63rd St.
#411,Oakland,CA,USA,94618
```

この例では、フィールドはカンマで区切られており、ローとローの間は改行ターミネータで区切られています。**Marjorie Green** には電話番号がありません。**authors** テーブルの **phone** カラムのデフォルトは“**unknown**”なので、このファイルがロードされたテーブルのローは次のようになります。

```
409-56-7008  Bennet  David      415 658-9932  622 Pine St.
Berkeley  CA  USA  94705
213-46-8915  Green  Marjorie  unknown      309 63rd St. #411
Oakland  CA  USA  94618
```

ルールとトリガ

bcp は、データのロードを最高速度で行えるよう、ルールとトリガを起動しません。

ルールおよびトリガに違反するローを見つけるには、データをテーブルにコピーし、ルールまたはトリガの条件をテストするクエリやストアド・プロシージャを実行してください。

bcp と他のユーティリティの相違点

テーブル全体または1つのテーブルの一部をコピーする **bcp** ユーティリティは、データをある場所から別の場所に移動する他のユーティリティとは異なります。

次のリストでは、このような他のユーティリティとそのコマンドの名前を示し、データの移動においてどのように使用するのが最善かについて説明します。

dump database、load database、dump transaction、load transaction

SQL コマンドの `dump database`、`load database`、`dump transaction`、`load transaction` は、バックアップだけを目的として使用します。`dump` コマンドは、`bcp` とは異なり、データベース全体の物理的なイメージを作成します。

`dump database` または `dump transaction` でバックアップされたデータを読み込むには、`load database` または `load transaction` を使用してください。

SQL コマンドの `dump` および `load` の詳細については、『システム管理ガイド』および『ASE リファレンス・マニュアル』を参照してください。

insert、update、delete

データ変更コマンドの `insert`、`update`、`delete` はそれぞれ、テーブルまたはビューにおける新しいローの追加、既存のローの変更、ローの削除に使用します。

- テーブル間でデータを移動するには、`insert` コマンドを `select` 文と一緒に使用します。
- `select` 文の中で `into` 句を使用すると、次の要素に基づいて新しいテーブルが作成されます。
 - `select` 文で指定したカラム
 - `from` 句で指定したテーブル
 - `where` 句で指定したローのデータ

データの追加、変更、削除の詳細については、『ASE リファレンス・マニュアル』の `insert`、`update`、`delete` の項を参照してください。

dscp の使用

トピック	ページ
dscp について	101
サーバ・エントリの操作	103
dscp の終了	108
dscp ユーティリティ・コマンドのクイック・リファレンス	109

dscp は、interfaces ファイルのサーバ・エントリの表示、編集に使用するユーティリティ・プログラムです。

注意 dscp は Windows では使用できません。

dscp の構文の詳細については、「[dscp](#)」(215 ページ)を参照してください。

dscp について

❖ dscp の起動

- 次のように入力します。

```
$SYBASE/$SYBASE_OCS/bin/dscp
```

dscp のプロンプト >> が表示されます。

❖ dscp のヘルプを利用する

- dscp のヘルプ画面を表示するには、次のいずれかのコマンドを入力します。

```
help  
h  
?
```

dscp セッションの使用

サーバ・エントリを表示、追加、変更する場合は、まずセッションをオープンします。これによって、対話形式で `interfaces` ファイルを使用できます。

一度に複数のセッションをオープンできます。

❖ `interfaces` ファイルでセッションをオープンする

- 次のように入力します。

```
open InterfacesDriver
```

セッションをオープンすると、`dscp` はセッション番号を通知します。たとえば、`open InterfacesDriver` コマンドを使ってセッションをオープンすると、`dscp` は次のメッセージを表示します。

```
ok  
Session 1 InterfacesDriver>>
```

❖ オープンされているセッションをすべて表示する

- 次のように入力します。

```
sess
```

❖ オープンされている別のセッションに切り替える

- 次のコマンドを入力します。`sess` はセッション番号です。

```
switch sess
```

たとえば、次のコマンドを入力してセッション3に切り替えま
す。

```
switch 3
```

`switch` キーワードはオプションです。たとえば、“3”と入力して
も、セッション3に切り替わります。

❖ セッションのクローズ

- 次のコマンドを入力します。`sess` はセッション番号です。

```
close sess
```

たとえば、セッション3をクローズするには、次のコマンドを入
力します。

```
close 3
```

セッション番号が指定されていない場合、`dscp` は現在のセッシ
ョンをクローズします。

サーバ・エントリの操作

dscp を使って、サーバ・エントリの追加、変更を行います。

サーバ・エントリの追加と変更

セッションをオープンすると、そのセッションに関連付けられているサーバ・エントリを追加または変更できます。

注意 サーバ・エントリを追加または変更すると、dscp は、master 行と query 行の両方を自動的に作成または変更します。interfaces ファイルのエントリにある master 行と query 行には、まったく同じ情報が含まれています。

各サーバ・エントリは、一連の属性で構成されています。サーバ・エントリを追加または変更すると、dscp は各属性についての情報を要求します。表 5-2 は、各属性を示します。

表 5-1 : サーバの属性

属性	値のタイプ	デフォルト値と有効な値	サーバ・エントリを追加または変更するときの、属性の変更	
			追加	変更
Server Object Version	整数	110	不可	不可
Server Name	文字列	N/A	N/A	不可
Server Service	文字列	SQL SERVER	可	不可
Server Status	整数	4 有効な値 : 1 Active 2 停止 3 Failed 4 不明	不可	不可
Transport Type	文字列	tcp。有効な値 : decnet、spx、tcp、tli、spx、tli tcp 注意 スレッド・モードでは、TLI インタフェースはサポートされない。	可	可

属性	値のタイプ	デフォルト値と有効な値	サーバ・エントリを追加または変更するときの、属性の変更	
			追加	変更
Transport Address	文字列	なし。有効な値は、指定されたトランスポートのタイプが認識する文字列	可	可
Security Mechanism	文字列 注意 サーバ・エントリごとに、セキュリティ・メカニズムの文字列を20個まで追加できる	なし。 有効な値は、ユーザの <i>objectid.dat</i> に定義されているオブジェクト識別子に対応する文字列	可	可

❖ **サーバ・エントリを追加する**

- 1 次のように入力します。

```
add servername
```

追加モードに切り替わります。これで、サーバ・エントリを追加できます。他の **dscp** コマンドを実行する場合は、このモードを終了する必要があります。追加モードでは、**dscp** は *servername* に関する情報を要求します。

- 2 次のいずれかを実行します。

- 各属性の値を入力する
- **[Return]** キーを押して、角カッコ ([]) に示されているデフォルト値を使用する

たとえば **dscp** は、入力のとときに次の情報を要求するプロンプトを表示します。

```
add myserver

Service:[SQL Server]
Transport Type:[tcp] tcp
Transport Address:victory 8001
Security Mechanism []:
```

サーバ・エントリには、関連するトランスポートのタイプとトランスポート・アドレスの組み合わせを20個まで指定できます。

サーバの属性については、[表 5-1 \(103 ページ\)](#) を参照してください。

- 3 追加モードを終了するには、次のコマンドを入力します。

```
#done
```


❖ サーバ・エントリを変更する

dscp を使用して、*interfaces* ファイルにある Version、Service、および Status のエントリを変更することはできません。

- 1 次のように入力します。

```
mod servername
```

変更モードに切り替わります。これで、サーバ・エントリを変更できます。他の **dscp** コマンドを実行する場合は、このモードを終了する必要があります。変更モードでは、**dscp** は *servername* に関する情報を入力するよう要求します。

- 2 次のいずれかを実行します。

- 各属性の値を入力する
- [Return] キーを押して、角カッコ ([]) に示されているデフォルト値を使用する

たとえば **dscp** は、入力の際に次の情報を要求するプロンプトを表示します。

```
mod myserver
Version:[1]
Service:[SQL Server] Open Server
Status:[4]
Address:
Transport Type:[tcp]
Transport Address:[victory 1824] victory 1826
Transport Type:[tcp]
Transport Address:[victory 1828]
Transport Type:[]
Security Mechanism []:
```

サーバの属性については、[表 5-1 \(103 ページ\)](#) を参照してください。

- 3 アドレスを削除するには、次のコマンドを入力します。

```
#del
```

- 4 変更モードを終了するには、次のコマンドを入力します。

```
#done
```

サーバ・エントリのコピー

dscp では、1つのセッション内、または複数のセッション間でサーバ・エントリをコピーできます。サーバ・エントリをコピーする場合、4つのオプションがあります。

オプションは次のとおりです。

- 現在のセッションに、新しい名前ですべてサーバ・エントリをコピーする
- 別のセッションにサーバ・エントリをコピーする
- 別のセッションに、新しい名前ですべてサーバ・エントリをコピーする
- 別のセッションに、現在のセッションのエントリをすべてコピーする

❖ 1つのセッション内で、コピーによって新しいサーバ・エントリを作成する

- 次のように入力します。

```
copy name1 to name2
```

たとえば、次のコマンドを入力すると、dscp は、“myserver”とまったく同じエントリ “my_server” を作成します。

```
copy myservers to my_server
```

このようにして、新しいエントリを変更し、元のエントリをそのままにしておくことができます。

❖ 名前を変更しないでサーバ・エントリをコピーする

- 次のように入力します。

```
copy name1 to sess
```

たとえば、次のように入力すると、dscp は現在のセッションにある “myservers” エントリをセッション 2 にコピーします。

```
copy myservers to 2
```

❖ サーバ・エントリをコピーするときに新しい名前を付ける

- 次のように入力します。

```
copy name1 to sess name2
```

たとえば、次のように入力すると、dscp は現在のセッションにある “myservers” エントリをセッション 2 にコピーし、コピーされたエントリの名前を “my_server” に変更します。

```
copy myservers to 2 my_server
```

❖ 現在のセッションにあるすべてのエントリを別のセッションにコピーする

- 次のように入力します。

```
copyall sess
```

たとえば、次のように入力すると、`dscp` は現在のセッションにあるすべてのエントリをセッション 2 にコピーします。

```
copyall 2
```

サーバ・エントリの表示

セッションに関連付けられている名前と属性を表示できます。

❖ サーバ・エントリの名前を表示する

- 次のように入力します。

```
list
```

❖ サーバ・エントリの属性を表示する

- 次のように入力します。

```
list all
```

サーバの属性については、[表 5-1 \(103 ページ\)](#) を参照してください。

❖ サーバ・エントリの内容を表示する

- 次のように入力します。

```
read servername
```

たとえば、次のように入力すると、次の情報が表示されます。

```
read myserver

DIT base for object:interfaces
Distinguish name:myserver
Server Version:1
Server Name:myserver
Server Service:SQL Server
Server Status:4 (Unknown)
Server Address:
Transport Type:tcp
Transport Addr:victory 1824
Transport Type:tcp
Transport Addr:victory 1828
```

サーバの属性については、[表 5-1 \(103 ページ\)](#) を参照してください。

サーバ・エントリの削除

セッションに関連付けられている 1 つまたはすべてのエントリを削除できます。

❖ **セッションに関連付けられているエントリを削除する**

- 次のように入力します。

```
del servername
```

たとえば、次のように入力すると、dscp は “myserver” のエントリを削除します。

```
del myserver
```

❖ **セッションに関連付けられているすべてのエントリを削除する**

- 次のように入力します。

```
delete-all
```

dscp の終了

dscp を終了するには、次のいずれかのコマンドを入力します。

```
exit
```

```
quit
```

dscp ユーティリティ・コマンドのクイック・リファレンス

dscp では、dscp プロンプトにコマンドを入力して機能を実行できます。表 5-2 は、利用できるコマンドのクイック・リファレンスです。

表 5-2 : dscp コマ不要な文字種

コマンド	説明
add <i>servername</i>	サーバ・エントリ <i>servername</i> を現在のセッションに追加する。dscp は、 <i>servername</i> についての情報を要求する。角カッコ ([]) に示されているデフォルト値を使用する場合は [Return] キーを押す。追加モードを終了するには、“#done”と入力する。
addattr <i>servername</i>	現在のセッションのサーバ・エントリ <i>servername</i> に属性を追加する。
close [<i>sess</i>]	番号 <i>sess</i> に指定されたセッションをクローズする。 <i>sess</i> が指定されていない場合は、現在のセッションをクローズする。
config	使用している Sybase 環境の設定情報を表示する。
copy <i>name1</i> to { <i>name2</i> <i>sess</i> <i>sess name2</i> }	現在のセッションのサーバ・エントリ <i>name1</i> を次の位置にコピーする。 <ul style="list-style-type: none"> 現在のセッションのサーバ・エントリ <i>name2</i> セッション <i>sess</i> セッション <i>sess</i> のサーバ・エントリ <i>name2</i>
copyall to <i>sess</i>	現在のセッションのサーバ・エントリをすべてセッション <i>sess</i> にコピーする。
del <i>servername</i>	現在のセッションのサーバ・エントリ <i>servername</i> を削除する。
delete-all	現在のセッションのサーバ・エントリをすべて削除する。
exit	dscp を終了する。
help、?、h	オンラインヘルプを表示する。
list [all]	現在のセッションのサーバ・エントリを表示する。エントリの名前を表示するには、list コマンドを使用する。各エントリの属性もリストするには、list all コマンドを使用する。
mod <i>servername</i>	現在のセッションでサーバ・エントリ <i>servername</i> を変更する。dscp は、 <i>servername</i> についての情報を要求する。角カッコ ([]) に示されているデフォルト値を使用する場合は [Return] キーを押す。変更モードを終了するには、“#done”と入力する。
open [<i>dsname</i>]	指定されたディレクトリ・サービスのセッションをオープンする。 <i>dsname</i> は、ディレクトリ・サービス名を表す。 <i>dsname</i> に値を指定しない場合、このコマンドはデフォルトのディレクトリ・サービスのセッションをオープンする。セッションをオープンするには、 <i>dsname</i> に “InterfacesDriver” を指定する。
quit	dscp を終了する。
read <i>servername</i>	サーバ・エントリ <i>servername</i> の内容を表示する。
sess	オープンされているすべてのセッションを表示する。
[switch] <i>sess</i>	セッション番号 <i>sess</i> を現在のセッションにする。

この章では、`dsedit` ユーティリティを使用して Adaptive Server の `interfaces` ファイルを編集する方法について説明します。

トピック	ページ
dsedit について	111
サーバ・エントリの追加、表示、編集	116
dsedit に関するトラブルシューティング	123

dsedit について

`dsedit` は、`interfaces` ファイル (Windows では `sql.ini`) のサーバ・エントリの表示、編集に使用するグラフィカル・ユーティリティです。`dsedit` の構文の詳細については、「[dsedit](#)」(216 ページ) を参照してください。

注意 UNIX ユーザの場合：使用しているシステムに X-Windows がインストールされていない場合、`interfaces` ファイルのサーバ・エントリの設定には `dscp` を使用します。詳細については、「[第 5 章 dscp の使用](#)」を参照してください。

dsedit の起動

Windows

`dsedit` は、コマンド・プロンプト、Windows エクスプローラ、Sybase for Windows のプログラム・グループから実行できます。

❖ コマンド・プロンプトから `dsedit` を実行する

- 次のように入力します。

```
dsedit
```

以下のコマンド・ライン引数を指定できます。

- `-ddsname` — 接続するディレクトリ・サービスを指定します。`dsname` は、`libtcl.cfg` ファイルにリストされているディレクトリ・サービスのローカル名です。`-ddsname` 引数を指定していない場合、`dsedit` は最初に表示されるダイアログ・ボックスにディレクトリ・サービス・オプションのリストを表示します。
- `-lpath` — `SYBASE_home` ¥INI 以外にある場合は、`libtcl.cfg` ファイルのパスを指定します。`SYBASE_home` ¥INI 以外にある `libtcl.cfg` ファイルを使用する場合にだけ、この引数を使用します。

❖ Windows エクスプローラから dsedit を実行する

- 1 `%SYBASE%` ¥bin ¥ディレクトリに移動します。
- 2 `DSEDIT.exe` ファイルをダブルクリックします。

❖ Sybase for Windows のプログラム・グループから dsedit を実行する

- 1 [スタート]メニューから [Sybase for Windows] を選択します。
- 2 Sybase for Windows のメニューから [dsedit] を選択します。[Select Directory Service] ダイアログ・ボックスが表示されます。

UNIX プラットフォームの場合

`interfaces` ファイルに対する書き込みパーミッションがあることを確認してから、`dsedit` を実行します。

リモート・マシンから `dsedit` を実行する場合、`DISPLAY` 環境変数が正しく設定され、`dsedit` の画面がリモート・マシンではなく、使用しているマシンに表示されていることを確認します。

❖ DISPLAY 環境変数の設定

- 1 リモート・マシンにログインします。
- 2 次のように入力します。

```
setenv DISPLAY your_machine_name:0.0
```

❖ dsedit の起動

- 次のように入力します。

```
¥SYBASE/bin/dsedit
```

[Select a Directory Service] ウィンドウが表示されます。このウィンドウで、`interfaces` ファイルの編集セッションをオープンできます。`interfaces` ファイルのデフォルトのフル・パス名は、[Interfaces File to Edit] ボックスに表示されます。設定ファイルのフル・パス名がその下に表示されます。

編集セッションのオープン

Windows

[Select Directory Service] ダイアログ・ボックスを使用して、ディレクトリ・サービスのセッションをオープンできます。次のサービスやファイルを使用して、セッションをオープンできます。

- *libtcl.cfg* ファイルにリストされたドライブを持つディレクトリ・サービス
- *sql.ini* ファイル

❖ Windows のセッションをオープンする

- 1 [DS Name] ボックスにリストされたものの中から、接続先のディレクトリ・サービスのローカル名をダブルクリックします。
- 2 [DS Name] ボックスにリストされたものの中から、接続先のディレクトリ・サービスのローカル名をクリックし、[OK] ボタンをクリックします。

注意 dsedit は、SYBASE 環境変数を使用して *libtcl.cfg* ファイルを検索します。SYBASE 環境変数が正しく設定されていないと、dsedit は *libtcl.cfg* ファイルを探すことができません。

セッション番号とディレクトリ・サービスのローカル名が、ヘッダ・バーに表示されます。

❖ 追加セッションをオープンする

dsedit では、一度に複数のセッションをオープンできます。

- 1 [File] メニューから [Open Directory Service] を選択します。
[Select Directory Service] ボックスが表示されます。
- 2 接続先のディレクトリ・サービスのローカル名をダブルクリックします(または、ディレクトリ・サービスのローカル名をクリックし、[OK] をクリックします)。

複数のセッションをオープンすることによって、ディレクトリ・サービス間でエントリをコピーできます。詳細については、「[サーバ・エントリのコピー](#)」(122 ページ)を参照してください。

❖ **セッション間を切り替える**

一度に複数のセッションをオープンしている場合は、作業するセッションをアクティブにしてください。

- セッションをアクティブにするには、次のいずれかの操作を行います。
 - セッション・ウィンドウをクリックします。
 - [Window] メニューからセッションを選択します。

dsedit のタイトル・バーに、アクティブになっているセッションが表示されます。

UNIX プラットフォームの場合

❖ **デフォルトの interfaces ファイルをオープンして編集する**

- 1 [Sybase Interfaces File] を選択します。
- 2 [OK] をクリックします。

❖ **デフォルトの interfaces ファイル以外のファイルを開く**

- 1 [Sybase Interfaces File] を選択します。
- 2 表示されるファイル名を編集します。
- 3 [OK] をクリックします。

[Directory Service Session] ウィンドウが表示されます。

複数の interfaces ファイル・セッションをオープンし、それぞれのセッションで異なるファイルを編集できます。

[Directory Service Session] 画面には、interfaces ファイルのフル・パス名と、ファイル内にあるサーバ・エントリのリストが表示されます。

- [新しいサーバ・エントリを追加する] – [Server Entry Editor] ウィンドウが表示されます。このウィンドウで、新しいサーバ・エントリの名前とネットワーク・アドレスを指定します。
- [サーバ・エントリを修正する] – 選択されているサーバ・エントリのネットワーク・アドレスを表示、変更できます。サーバ・エントリを表示または修正するには、リストでそのサーバを選択してから [サーバ・エントリを修正する] ボタンをクリックし、[Server Entry Editor] ウィンドウにサーバの属性を表示します。
- [サーバ・エントリをコピーする] – 1 つ以上のエントリを別の interfaces ファイルにコピーします。

- [セッションをクローズする] – セッション・ウィンドウをクローズし、変更を `interfaces` ファイルに書き込みます。

これらのボタンを使用する手順については、「[Windows でのサーバ・エントリの変更](#)」(116 ページ)を参照してください。

[Session] 画面の [Add new server entry] ボタンまたは [Modify server entry] ボタンをクリックすると、[Server Entry Editor] ウィンドウが表示されます。

`interfaces` ファイルのサーバ・エントリを表示または編集するには、[Server Entry Editor] ウィンドウを使用します。

- [サーバ名] – サーバ・エントリを追加する場合は、新しいサーバの名前を入力します。サーバ・エントリを編集する場合は、名前フィールドを編集し、サーバ名を変更できます。`interfaces` ファイルにすでに存在する名前を新しい名前として指定することはできません。
- [使用可能なネットワーク・トランスポート] – サーバがクライアントと接続するネットワーク・アドレスのリストです。
 - 新しいアドレスを作成するには、[Add network transport] をクリックします。「[Windows でのサーバ・エントリの変更](#)」(116 ページ)を参照してください。
 - 既存のアドレスを編集するには、[Modify network transport] をクリックします。「[Windows でのサーバ・エントリの変更](#)」(116 ページ)を参照してください。
 - 選択されているネットワーク・アドレスを削除するには、[Delete network transport] をクリックします。
 - リスト内のアドレスを並べ替えるには、[Move network transport up] または [Move network transport down] をクリックします。
- [OK] ボタン – 変更を確認してウィンドウをクローズします。[Directory Service Session] 画面の [Close Session] ボタンを使用してセッションをクローズするまで、`interfaces` ファイルへの変更は適用されません。
- [Cancel] ボタン – ウィンドウをクローズし、すべての編集内容を廃棄します。

サーバ・エントリの追加、表示、編集

セッションをオープンすると、そのセッションに関連するサーバ・エントリの追加、変更、削除、名前の変更を行うことができ、またセッション間でサーバ・エントリをコピーできます。

Windows でのサーバ・エントリの変更

セッションに対応するサーバ・エントリが、[Server] ボックスに表示されます。サーバ・エントリをクリックし、選択してください。

各サーバ・エントリは、一連の属性で構成されます。表 6-1 で各属性について説明します。

表 6-1 : サーバの属性

属性名	値のタイプ	説明	デフォルト値
Server Version	整数	サーバ・オブジェクト定義のバージョン・レベル。オブジェクト定義の将来の変更を識別するために、Sybase がこの属性を提供する。	150
Server Name	文字列	サーバの名前。	N/A
Server Service	文字列	サーバが提供するサービスの説明。意味のある説明ならどのような内容でも有効。	Adaptive Server
Server Status	整数	サーバの実行状態。値は次のとおり。 <ul style="list-style-type: none"> • Active • 停止 • Failed • 不明 	4
Security Mechanism	文字列	サーバがサポートするセキュリティ・メカニズムを指定するための、オブジェクト識別子 (OID) の文字列。この属性はオプション。省略すると、クライアントは Open Server が対応するセキュリティ・ドライバを持つすべてのセキュリティ・メカニズムに接続できる。	N/A

属性名	値のタイプ	説明	デフォルト値
Server Address	文字列	<p>サーバの1つまたは複数のアドレス。</p> <p>アドレスのフォーマットはプロトコルによって異なる。また、複数のフォーマットを使用できるプロトコルもある。オプションは次のとおり。</p> <ul style="list-style-type: none"> • TCP/IP – 以下の2つのフォーマットがある。 <ul style="list-style-type: none"> • <i>computer name.port number</i> • <i>ip-address.portnumber</i> • 名前付きパイプ – <i>pipe name</i> : “<i>¥pipe</i>” プレフィックスがすべてのパイプ名に必要。サーバのパイプは、ローカルに限る。 <ul style="list-style-type: none"> • ローカル – <i>¥pipe ¥sql ¥query</i> • リモート – <i>¥¥computer_name ¥pipe ¥sql ¥query</i> • IPX/SPX – 以下の3つのフォーマットがある。 <ul style="list-style-type: none"> • <i>server name</i> • <i>net number, node number, socket number</i> • <i>server name, socket number</i> • DECnet – 以下の4つのフォーマットがある。 <ul style="list-style-type: none"> • <i>area number.node number,object name</i> • <i>area number.node number,object number</i> • <i>node name,object name</i> • <i>node name,object number</i> 	N/A

❖ サーバ・エントリを追加する

- 1 [Server Object] メニューから [Add] を選択します。
- 2 [Server *ame*] ボックスに、サーバ名を入力します。
- 3 [OK] をクリックします。

[Server] ボックスに、サーバ・エントリが表示されます。サーバのアドレスを指定する場合は、このエントリを変更してください。

❖ サーバの属性を変更する

サーバ・エントリの属性はすべて変更できます。

- 1 [Server] ダイアログ・ボックスのサーバ・エントリをクリックします。
- 2 [Server Object] メニューの [Modify Attribute] を選択します。
- 3 [Attributes] ボックスで、変更する属性をクリックします。
ダイアログ・ボックスが表示され、属性の現在の値が表示されます。

- 4 属性の新しい値を入力するか、ドロップダウン・リストから値を選択します。

各属性の説明は、表 6-1 (116 ページ) を参照してください。

- 5 [OK] をクリックします。

❖ **サーバ・エントリの名前を変更する**

- 1 [Server] ダイアログ・ボックスのサーバ・エントリをクリックします。
- 2 [Server Object] メニューの [Rename] を選択します。
- 3 [Server name] ボックスにサーバ・エントリの新しい名前を入力します。
- 4 [OK] をクリックします。

❖ **サーバ・エントリを削除する**

- 1 [Server] ダイアログ・ボックスのサーバ・エントリをクリックします。
- 2 [Server Object] メニューの [Delete] を選択します。

❖ **現在のセッション内でエントリをコピーする**

- 1 [Server] ダイアログ・ボックスで、1 つまたは複数のサーバ・エントリをクリックします。
複数のエントリを選択する場合は、[Shift] キーを使用します。
- 2 メニュー・バーの下の [Copy] ボタンをクリックするか、[Edit] メニューの [Copy] を選択します。
- 3 メニュー・バーの下の [Paste] ボタンをクリックするか、[Edit] メニューの [Paste] を選択します。

dsedit は、コピーされたサーバ・エントリにバージョン番号を表す *_n* を付加します。[Server Object] メニューの [Rename] オプションをオンにして、コピーされたサーバ・エントリの名前を変更できます。詳細については、「[サーバ・エントリの名前を変更する \(118 ページ\)](#)」を参照してください。

❖ セッション間でサーバ・エントリをコピーする

- 1 コピー先のディレクトリ・サービスまたは *sql.ini* ファイルのセッションをオープンします。
- 2 [File] メニューから [Open Directory Service] を選択してセッションをオープンします。詳細については、「追加セッションをオープンする」(113 ページ) を参照してください。
- 3 コピー元のセッションの [Server] ダイアログ・ボックスから、1 つまたは複数のサーバ・エントリを選択します。
複数のエントリを選択する場合は、[Shift] キーを使用します。
- 4 メニュー・バーの下の [Copy] ボタンをクリックするか、[Edit] メニューの [Copy] を選択して、サーバ・エントリをコピーします。
メニュー・バーの下の [Cut] ボタンをクリックするか、[Edit] メニューの [Cut] を選択して、サーバ・エントリを削除します。
- 5 サーバ・エントリのコピー先のセッションをアクティブにします。
セッションをアクティブにする方法については、「セッション間を切り替える」(114 ページ) を参照してください。
- 6 メニュー・バーの下の [Paste] ボタンをクリックするか、[Edit] メニューの [Paste] を選択します。

[Server Object] メニューの [Rename] コマンドを使用して、コピーされたサーバ・エントリの名前を変更できます。詳細については、「セッション間を切り替える」(114 ページ) を参照してください。

UNIX プラットフォームでのサーバ・エントリの変更

この項の手順を実行するには、「編集セッションのオープン」(113 ページ) の指示に従って *interfaces* セッション・ウィンドウをオープンします。

注意 この項で説明する各手順を実行したあとで、[Close Session] をクリックして、すべての変更を *interfaces* ファイルに適用してください。このボタンをクリックすると、*interfaces* セッション・ウィンドウもクローズされます。

❖ **新しいサーバ・エントリを追加する**

- 1 [新しいサーバ・エントリを追加する]をクリックします。
- 2 新しいサーバ・エントリの名前とネットワーク・アドレスを指定します。

❖ **サーバ・エントリを表示または変更する**

- 1 [サーバ・エントリを修正する]をクリックします。
- 2 必要に応じて属性を変更します。

❖ **サーバ・エントリを別の interfaces ファイルにコピーする**

- 1 次のいずれかの方法で、コピーするエントリを選択します。
 - エントリを1つだけコピーするには、そのエントリを1回だけクリックします。
 - 隣り合う複数のエントリを一度にコピーするには、その範囲の最初のエントリをクリックし、[Shift] キーを押したままの状態最後のエントリをクリックします。逆に、最後のエントリをクリックし、[Shift] キーを押したままの状態最初のエントリをクリックする方法もあります。
 - 互いに離れた位置にある複数のエントリを選択するには、[Ctrl] キーを押しながらそれぞれのエントリをクリックします。
- 2 [サーバ・エントリをコピーする]をクリックします。
- 3 リストから、[Sybase interfaces file]を選択します。
- 4 表示されるファイル名を編集します。
- 5 [OK]をクリックします。

ネットワーク・トランスポート・アドレスの追加または編集

[Network Transport Editor] ウィンドウでは、サーバがクライアントと接続するトランスポート・アドレスを表示、編集、作成できます。このウィンドウには、アドレスに対応するサーバ・エントリの名前が表示され、次の項目を設定できます。

- [Transport type] メニュー – アドレスのプロトコルとインタフェースを指定します。
- アドレス情報 – トランスポートのタイプによって、必要なアドレスのコンポーネントが異なります。アドレスのフォーマットについては、以降の各項で詳しく説明します。

TCP/IP アドレス

TCP/IP エントリのアドレス情報は、ホスト名 (または IP アドレス) とポート番号 (10 進数として入力) で構成されます。tli tcp フォーマットの `interfaces` エントリでは、ホストの IP アドレスとポート番号は、tli tcp フォーマットの `interfaces` エントリに必要な 16 バイトの 16 進表現に変換されます。

注意 スレッド・モードでは、TLI インタフェースはサポートされません。

tli tcp フォーマットの `interfaces` エントリは、次のものに対応します。

- tli フォーマット - の `interfaces` エントリを使用するプラットフォーム上の 10.0 より前のすべてのクライアント
- tli フォーマットの `interfaces` エントリを使用するプラットフォーム上の Adaptive Server、またはバージョン 11.0.x 以前の Replication Server

これ以外のクライアントとサーバでは、tcp を使用します。

TCP/IP アドレスを指定するには、[Transport Type] メニューから、[tcp] または [tli tcp] を選択します。

SPX/IPX アドレス

Adaptive Server は、Novell ネットワーク上で稼働するクライアント・アプリケーションの接続要求を、SPX/IPX アドレスを使って受信します。SPX/IPX アドレスは、次の情報で構成されます。

- ホスト・アドレス — サーバが稼働するコンピュータの IP アドレスを表す 8 桁の 16 進数。IP アドレス形式のドットで区切られた 10 進数のそれぞれを 1 バイトの 16 進数にマップすることで、16 進数のアドレス形式が作成されます。たとえば、ホストの IP アドレスが 128.15.15.14 であれば、SPX/IPX のホスト・アドレス値として “800F0F0E” と入力します。
- ポート番号 — ポート番号は、4 桁の 16 進数として表されます。
- 終了ポイント — SPX デバイス・ドライバを指すデバイス・ファイルのパス。デフォルトは、Soralis では `/dev/mspx`、それ以外のプラットフォームでは `/dev/nspx` です。必要に応じ、サーバが稼働するマシンに合わせてパスを変更してください。デフォルトのパスは、dsedit を実行しているプラットフォームを基にしています。

SPX/IPX アドレスを指定するには、[Transport Type] メニューから、[tli spx] または [spx] を選択します。

サーバ・エントリのコピー

dsedit では、1つのセッション内、またセッション間でサーバ・エントリをコピーできます。*sql.ini* ファイルからディレクトリ・サービスにエントリをコピーすることもできます。

❖ Windows で現在のセッション内でエントリをコピーする

- 1 [Server] ダイアログ・ボックスで、1つまたは複数のサーバ・エントリをクリックします。

複数のエントリを選択する場合は、[Shift] キーを使用します。

- 2 メニュー・バーの下の [Copy] ボタンをクリックするか、[Edit] メニューの [Copy] を選択します。
- 3 メニュー・バーの下の [Paste] ボタンをクリックするか、[Edit] メニューの [Paste] を選択します。

dsedit は、コピーされたサーバ・エントリにバージョン番号を表す *_n* を付加します。[Server Object] メニューの [Rename] コマンドを使用して、コピーされたサーバ・エントリの名前を変更できます。詳細については、「[サーバ・エントリの名前を変更する](#)」(118 ページ) を参照してください。

❖ Windows でセッション間でサーバ・エントリをコピーする

- 1 コピー先のディレクトリ・サービスまたは *sql.ini* ファイルのセッションをオープンします。
- 2 [File] メニューから [Open Directory Service] を選択してセッションをオープンします。詳細については、「[追加セッションをオープンする](#)」(113 ページ) を参照してください。
- 3 コピー元のセッションの [Server] ダイアログ・ボックスから、1つまたは複数のサーバ・エントリを選択します。

複数のエントリを選択する場合は、[Shift] キーを使用します。

- 4 メニュー・バーの下の [Copy] ボタンをクリックするか、[Edit] メニューの [Copy] を選択して、サーバ・エントリをコピーします。
メニュー・バーの下の [Cut] ボタンをクリックするか、[Edit] メニューの [Cut] を選択して、サーバ・エントリを削除します。
- 5 サーバ・エントリのコピー先のセッションをアクティブにします。
セッションをアクティブにする方法については、「[オープンされている別のセッションに切り替える](#)」(102 ページ) を参照してください。

- メニュー・バーの下の [Paste] ボタンをクリックするか、[Edit] メニューの [Paste] を選択します。

[Server Object] メニューの [Rename] コマンドを使用して、コピーされたサーバ・エントリの名前を変更できます。詳細については、「[サーバ・エントリの名前を変更する](#)」(118 ページ) を参照してください。

❖ **UNIX でサーバ・エントリを別の interfaces ファイルにコピーする**

- 次のいずれかの方法で、コピーするエントリを選択します。
 - エントリを 1 つだけコピーする – そのエントリを 1 回だけクリックします。
 - 隣り合う複数のエントリを一度にコピーする – その範囲の最初のエントリをクリックし、[Shift] キーを押したままの状態ですべてのエントリをクリックします。逆に、最後のエントリをクリックし、[Shift] キーを押したままの状態ですべてのエントリをクリックする方法もあります。
 - 互いに離れた位置にある複数のエントリを選択する – [Ctrl] キーを押しながらそれぞれのエントリをクリックします。
- [サーバ・エントリをコピーする] をクリックします。
- リストから、[Sybase interfaces file] を選択します。
- 表示されるファイル名を編集します。
- [OK] をクリックします。

dsedit に関するトラブルシューティング

この項では、dsedit に関する一般的な問題をいくつか挙げ、その対処方法について説明します。

dsedit ユーティリティ
が起動しない

次の各項に該当していないか確認してください。

- SYBASE 環境変数が設定されていないか、誤ったディレクトリが指定されている。
- UNIX プラットフォームの場合** . X-Windows が正しく設定されていない。リモート・ホスト上で dsedit を実行している場合は、リモート・ホスト上の X-Windows クライアントが、使用しているマシン上の X-Windows サーバに接続できることを確認します。トラブルシューティングの詳細については、X-Windows のマニュアルを参照してください。X-Windows を使用できない場合は、dsedit の代わりに dscp を使用します。

エラー・メッセージ：
“Unable to open X
display”

(UNIX プラットフォーム) 表示用のマシンがリモート・ホストからの X-Windows 接続を拒否するように設定されていると、dsedit が機能しないことがあります。この問題が発生している場合には、次のようなメッセージが表示されます。--

```
Unable to open X display.Check the value of your
$DISPLAY variable.If it is set correctly, use the
'xhost +' command on the display machine to authorize
use of the X display.If no X display is available, run
dscp instead of dsedit.
```

このエラーは、次のどちらかの状況で発生します。

- DISPLAY 環境変数の値が正しく入力されていないか、設定されていない。

解決法：DISPLAY 環境変数の値を正しく入力します。

- DISPLAY が参照するマシン上でウィンドウをオープンする権限がない。

解決法：表示用のマシンで、コマンド `Eexhost +Ef` を実行します。

サーバ・エントリを追加、変更、または削除できない

interfaces ファイルのパーミッションに関する問題がないか確認してください。interfaces のエントリを編集するには、interfaces ファイルと Sybase のインストール・ディレクトリの両方に対して、書き込みパーミッションが必要です。

ユーティリティ・コマンド・リファレンス

この章は、Adaptive Server のユーティリティ・プログラム・コマンドについて説明します。

はじめる前に

UNIX

UNIX シェルのシステム・プロンプトで、ユーティリティ・プログラム・コマンドを入力します。

Windows

ユーティリティのアイコンの有無によって、次のようにします。

- Sybase for Windows または Sybase for Windows プログラム・グループにアイコンがある場合は、そのアイコンをダブルクリックしてユーティリティ・プログラムを起動します。
- プログラム・グループにアイコンがない場合は、Windows コマンド・プロンプトでユーティリティ・プログラム・コマンドを入力して、ユーティリティ・プログラムを起動します。

円記号 (¥)、アスタリスク (*)、スラッシュ (/)、スペースなど、シェル (Windows のコマンド・プロンプト) にとって特別な意味を持つ文字は引用符で囲みます。円記号 (¥) を特殊文字の前に付けて、それらの文字を「エスケープ」することができます。これによって、シェル (コマンド・プロンプト) はこの特殊文字を解釈しなくなります。

Adaptive Server で利用できるユーティリティ・プログラムは、次のとおりです。

- **backupserver** – 実行形式の Backup Server™ プログラム。
- **bcp** – ユーザが指定したフォーマットで、データベース・テーブル内のローをオペレーティング・システム・ファイルにコピーしたり、オペレーティング・システム・ファイルからデータベース・テーブル内のローにコピーします。

- [certauth](#) –サーバ証明書の要求を認証機関による署名付きの証明書に変換します。
- [certpk12](#) – PKCS#12 ファイルをエクスポートまたはインポートします。
- [certreq](#) –サーバ証明書要求と対応するプライベート・キーを2つの方法で作成します。
- [charset](#) –文字セット・ファイルとソート順ファイルをロードします。
- [cobpre](#) – COBOL のプリコンパイラ。
- [cpre](#) – C のプリコンパイラ。
- [dataserver](#) –実行形式の Adaptive Server プログラム。
- [ddlgen](#) – ASE でサーバ・レベル・オブジェクトとデータベース・レベル・オブジェクトのデータ定義言語を作成します。
- [defncopy](#) –特定のビュー、ルール、デフォルト、トリガ、プロシージャ、レポートの定義を、データベースからオペレーティング・システム・ファイルに、またはオペレーティング・システム・ファイルからデータベースにコピーします。
- [dscp](#) –コマンド・ライン・モードで [interfaces](#) ファイルのサーバ・エントリを表示したり、編集したりします。
- [dsedit](#) –グラフィカル・ユーザ・インタフェースで [interfaces](#) ファイルのサーバ・エントリを表示したり、編集したりします。
- [extractjava](#) –保持された JAR を Adaptive Server からクライアント・ファイルにコピーします。
- [installjava](#) –クライアント・ファイルから Adaptive Server に JAR をインストールします。
- [isql](#) – Adaptive Server の Interactive SQL パーサです。
- [langinstall](#) – Adaptive Server に新しい言語をインストールします。
- [optdiag](#) –オプティマイザ統計値を表示します。または、更新された統計値をシステム・テーブルにロードします。
- [preupgrade](#) –アップグレードできる状態かどうかを判断するために、インストールまたはデータベースでテストを実行します。
- [pwdcrypt](#) – [libtcl.cfg](#) ファイルに LDAP 暗号化パスワードを作成し、表示します。

- **qmutil** – (Cluster Edition のみ) クォーラム・デバイスをバックアップ、リストア、および再設定できます。
- **qptune** – 欠落統計を修正し、最適なクエリ・プラン、最適化目標、またはその他の構成設定を特定し、それらをクエリまたはサーバのレベルで適用できます。
- **showserver** – ローカルのマシンで現在稼働している Adaptive Server と Backup Server を表示します。
- **sqldbg** – ストアド・プロシージャとトリガをデバッグします。
- **sqlloc** – GUI モードで、Adaptive Server の言語、文字セット、ソート順のデフォルトをインストールしたり、変更したりします。
- **sqllocres** – コマンド・ライン・モードで、Adaptive Server の言語、文字セット、ソート順のデフォルトをインストールしたり、変更したりします。
- **sqlsrvr** – 実行形式の Adaptive Server プログラム。
- **sqlupgrade** – GUI モードで、現在インストールされている Adaptive Server のリリースを最新のリリースへアップグレードします。
- **sqlupgraderes** – コマンド・ライン・モードで、現在インストールされている Adaptive Server のリリースを最新のリリースへアップグレードします。
- **srvbuild** – GUI モードで、キー設定属性のデフォルト値またはユーザ指定の値を使用して、Adaptive Server、Backup Server、XP Server を新しく作成します。
- **srvbuildres** – コマンド・ライン・モードで、キー設定属性のデフォルト値またはユーザ指定の値を使用して、Adaptive Server、Backup Server、XP Server を新しく作成します。
- **startserver** – Adaptive Server または Backup Server を起動します。
- **sybcluster** – 共有ディスク・クラスタを管理します。「[第8章 sybcluster を使用したクラスタの管理](#)」を参照してください。
- **sybdiag** – 包括的な Adaptive Server 設定および環境データを収集します。
- **sybmigrate** – 2K の論理ページを使用しているサーバのデータベースを、4、8、または 16K の論理ページを使用しているサーバにマイグレートします。

- `sybtsmpasswd` – ユーザのパスワードを記録および変更し、TSM (Tivoli Storage Manager) 暗号化パスワード・ファイル `TSM.PWD` を TSM クライアント・マシンに作成します。
- `xpserver` – XP Server を手動で起動します。

注意 これらのユーティリティ・プログラムでは、`-P` パラメータを使用してパスワードを入力できる場合があります。セキュリティを考慮する必要がある場合は、このパラメータを使ってパスワードを指定しないでください。他のユーザにパスワードを見られるおそれがあります。`-P` パラメータを使用しないで、通常のログインを行って、Adaptive Server がパスワードの入力を要求するプロンプトを表示するようにしてください。

*_r ユーティリティ

一部のユーティリティでは、スレッド・ドライバで使用するための `_r` バージョンが提供されています。

このマニュアルに記載されているユーティリティの中で、`_r` バージョンがあるものは次のとおりです。

- `bcp`
- `cobpre`
- `cpre`
- `defncopy`
- `dscp`
- `isql`

ユーティリティ・クイック・リファレンス

この項はユーティリティのクイック・リファレンスです。ユーティリティはカテゴリごとに分類されています。

- 「[インストール・ユーティリティと設定ユーティリティ](#)」 (129 ページ)

- 「言語、文字セット、ソート順の設定ユーティリティ」(130 ページ)
- 「サーバ起動ユーティリティ」(130 ページ)
- 「データベース作成ユーティリティと操作ユーティリティ」(131 ページ)
- 「情報収集ユーティリティ」(131 ページ)
- 「チューニング・ユーティリティ」(132 ページ)
- 「クラスタ管理ユーティリティ」(132 ページ)

インストール・ユーティリティと設定ユーティリティ

データベースのインストールや設定には、以下を使用します。

- **dataserver** — Adaptive Server を新しく構築できます。
- **dscp** — **interfaces** ファイルのサーバ・エントリをコマンド・ラインから表示、編集できます。
- **dsedit** — GUI を使用して **interfaces** ファイルのサーバ・エントリを表示したり、編集したりします。

Windows で、**interfaces** ファイルに保存されたネットワーク接続情報を作成、修正できます。

- **preupgrade** — アップグレードできる状態かどうかを判断するために、インストールまたはデータベースでテストを実行し、検出された問題をレポートします。
- **sqlupgrade** — 現在インストールされている Adaptive Server のリリースを最新のリリースへアップグレードします。
- **sqlupgraderes** — UNIX プラットフォームでリソース・ファイルを使用して、現在インストールされている Adaptive Server のリリースを最新のリリースにアップグレードします。
- **srvbuild** — キー設定属性のデフォルト値またはユーザ指定の値を使用して、Adaptive Server、Backup Server、XP Server を新しく作成します。
- **srvbuildres** — UNIX プラットフォームで、リソース・ファイルを使用してキー設定属性の値を指定し、Adaptive Server、Backup Server、XP Server を新しく作成します。

言語、文字セット、ソート順の設定ユーティリティ

言語、文字セット、ソート順を設定するには、以下のユーティリティを使用します。

- **charset** — Windows で文字セット・ファイルとソート順ファイルをロードします。
- **langinstall** — Adaptive Server に新しい言語をインストールします。
- **sqlloc** — Adaptive Server の言語、文字セット、ソート順のデフォルトをインストールしたり、変更したりします。
- **sqllocres** — UNIX プラットフォームでリソース・ファイルを使用して、Adaptive Server の言語、文字セット、ソート順のデフォルトをインストールしたり、変更したりします。

サーバ起動ユーティリティ

サーバを手動で起動するには、以下のユーティリティを使用します。

- **backupserver** — Backup Server 実行プログラムを起動します。Backup Server を手動で起動する場合は、このユーティリティではなく **startserver** コマンドを使用します。Windows で Backup Server を手動で起動するには、代わりに **srvmgr** ユーティリティを使用します。
- **dataserver** — Adaptive Server 実行プログラムを起動します。Adaptive Server を手動で起動する場合は、このユーティリティではなく **startserver** コマンドを使用します。
- **sqlsvr** — Windows で Adaptive Server 実行プログラムを起動します。Adaptive Server を手動で起動する場合は、このユーティリティではなく **services manager** ユーティリティを使用します。
- **srvmgr** — Adaptive Server、Backup Server、Adaptive Server Monitor™ を Windows サービスとして、起動、一時停止、停止します。
- **startserver** — UNIX プラットフォームで Adaptive Server または Backup Server を起動します。

データベース作成ユーティリティと操作ユーティリティ

データベースの作成や操作には、以下のユーティリティを使用します。

- **bcp** — ユーザが指定したフォーマットで、データベース・テーブルをオペレーティング・システム・ファイルに、または、そのオペレーティング・システム・ファイルからデータベース・テーブルにコピーします。
- **ddlgen** — ASE でサーバ・レベル・オブジェクトとデータベース・レベル・オブジェクトのデータ定義言語を作成します。
- **defncopy** — 特定のビュー、ルール、デフォルト、トリガ、プロシージャの定義を、データベースからオペレーティング・システム・ファイルに、またはオペレーティング・システム・ファイルからデータベースにコピーします。
- **extractjava** — 保持された JAR とそれに含まれるクラスを Adaptive Server からクライアント・ファイルにコピーします。
- **installjava** — クライアント・ファイルから Adaptive Server データベースに JAR をインストールします。
- **isql** — Adaptive Server の Interactive SQL パーサです。
- **optdiag** — オプティマイザ統計値を表示します。または、更新された統計値をシステム・テーブルにロードします。

情報収集ユーティリティ

情報の収集には、以下のユーティリティを使用します。

- **showserver** — UNIX プラットフォームで、ローカル・マシンで現在稼働している Adaptive Server と Backup Server を表示します。
- **sybdiag** — Sybase サポート・センタがサーバの問題を診断するための包括的な Adaptive Server 設定や環境データを収集します。
- **wdllvers** — Windows で、メモリにロードされた Sybase ダイナミック・リンク・ライブラリ (DLL: Dynamic Link Library) に関する情報を提供します。

チューニング・ユーティリティ

Adaptive Server のチューニングには、以下のユーティリティを使用します。

- **qptune** ー欠落統計を修正し、最適なクエリ・プラン、最適化目標、またはその他の構成設定を特定し、それらをクエリまたはサーバのレベルで適用できます。

クラスタ管理ユーティリティ

(Cluster Edition のみ) Adaptive Server 共有ディスク・クラスタの管理には、以下のユーティリティを使用します。

- **sybcluster** ークラスタを作成および管理するための一連の対話型コマンド・ライン・オプションを提供します。[「第 8 章 sybcluster を使用したクラスタの管理」](#)を参照してください。

backupserver

説明 実行形式の Backup Server プログラムです。\$SYBASE/\$SYBASE_ASE/bin にあります。

(Windows の場合) このユーティリティは bcksvr.exe で、%SYBASE%\%SYBASE_ASE%\bin にあります。

構文

```
backupserver
  [-C server_connections]
  [-S b_servername]
  [-I interfaces_file]
  [-e error_log_file]
  [-M sybmultbuf_binary]
  [-N network_connections]
  [-T trace_value]
  [-L Sybase_language_name]
  [-J Sybase_character_set_name]
  [-c tape_config_file]
  [-D n]
  [-A pathname]
  [-P active_service_threads]
  [-V level_number]
  [-p n]
  [-m max_shared_memory]
```

または

```
backupserver -v
```

パラメータ

-C server_connections

Backup Server のサーバ接続数を指定します。Backup Server には、次のものがが必要です。

- 各ダンプ・セッションに 2 接続
- 各ロード・セッションに 1 接続
- ボリューム変更メッセージ用に 1 接続

同時に行うことが予想されるダンプとロードのセッション数の 3 倍を最大数として考慮してください。サーバ接続数のデフォルト値は 30 です。

-S b_servername

起動する Backup Server の名前を指定します。デフォルトは SYB_BACKUP です。このパラメータには、interfaces ファイルにある Backup Server 名を指定してください。

-l interfaces_file

Backup Server に接続するときに検索する interfaces ファイルの名前とロケーションを指定します。**-l** を省略した場合、**backupserver** は、**SYBASE** 環境変数によって指定されたディレクトリにある *interfaces* という名前のファイルを探します。

-v

backupserver ソフトウェアのバージョン番号と著作権メッセージを表示して、終了します。

-e error_log_file

Backup Server のエラー・ログ・ファイルの名前とロケーションを指定します。このエラー・ログ・ファイルは、Open Server の内部エラー、**sybmultbuf** エラー、Backup Server を停止させるエラー、接続解除セッションのエラーをレポートするときに使用されます。それ以外のエラーはすべて、**dump database**、**dump transaction**、**load database**、**load transaction** のコマンド内で **notify** を使って指定された宛先に送られます。

-M sybmultbuf_binary

sybmultbuf 実行プログラムのフル・パス名を指定します。このパラメータを使用するのは、Sybase インストール・ディレクトリの *bin* ディレクトリ以外のディレクトリから Backup Server を起動する場合、または **sybmultbuf** の診断バージョンを使用する場合だけです。

-N network_connections

マスタ Backup Server から開始するネットワーク接続 (DBPROCESS) の総数を指定します。デフォルト値は 25 です。

-T trace_value

trace_value をビットマスク (2 進数) として解釈します。*trace_value* の中で 1 になっているビットは、Open Server のトレース・フラグではオンに対応します。コマンド・ラインで複数の **-T** パラメータを指定した場合は、最後に指定した **-T** の値によって、前の **-T** パラメータの値は無効になります。*trace_value* は、正の整数にします。

-L *Sybase_language_name*

Backup Server のデフォルト言語を指定します。このパラメータが指定されていない場合、Backup Server は LC_ALL 環境変数または LANG 環境変数によって指定されたロケールを使用します。これらの変数が設定されていない場合、Backup Server は、*locales.dat* の “default” エントリを検索します。

注意 -L パラメータによって LANG 環境変数に設定された値が無効になることはありません。

-J *Sybase_character_set_name*

Backup Server のデフォルト文字セットを指定します。

-c *tape_config_file*

dump database または dump transaction を実行する前に、テープ設定ファイルの名前とロケーションを指定して、テープ・デバイスの設定情報を検索します。-c を指定しない場合、テープ設定ファイルのデフォルトのパス名は *\$SYBASE/backup_tape.cfg* です。

-D *n*

Backup Server 内で使用される診断フラグのビットマップ (10 進数) を指定します。

-A *pathname*

アーカイブ API の動的ロード可能ライブラリが存在するディレクトリのパス名を指定します。

-P *active_service_threads*

複数のダンプ／ロード・オペレーション中にストライプの数を増やすことができます (1 オペレーションあたりの最大ストライプ数は 12286)。

-V level_number

Backup Server エラー・ログに出力するメッセージを制限します。
level_number 変数によって、Backup Server のエラーの詳細度 (-V) が決まります。

- -V4 – 接続イベントごとに出力される “Connection from Server” メッセージを除くすべての -V0 メッセージを表示します。
- -V3 – 通常の `dump` コマンドまたは `load` コマンドからの正常終了メッセージと以下のメッセージだけを表示します。
 - Backup Server と `sybmultipbuf` からのエラー・メッセージ
 - その他の `sybmultipbuf` メッセージ
 - ボリューム変更メッセージ
 - Open Server™ メッセージ
 - トレース出力メッセージ
 - System モジュールおよび Tape Auto Config モジュールからの情報メッセージ
- -V2 – 以下のメッセージを表示します。
 - すべての -V3 メッセージ
 - ファイル作成メッセージとファイル・マウント・メッセージ
- -V1 – 以下のメッセージを表示します。
 - すべての -V2 メッセージ
 - フェーズ・メッセージ
- -V0 (デフォルト) – バックアップの進行状況を含むすべてのメッセージを表示します。

この制限は、`dump` コマンドまたは `load` コマンドの `NOTIFY=` パラメータに指定されたクライアントまたはコンソールに送信されるメッセージには適用されません。

このオプションは、以下のメッセージ・タイプのロギングにも影響しません。

- Open Server メッセージ
- `bs_traceprint` からのトレース出力メッセージ
- `sybmultipbuf` メッセージ

-p n

ネットワーク・ダンプ中に「ローカル」の Backup Server が「リモート」の Backup Server から要求する TDS パケット・サイズをバイトで指定します。使用される実際のパケット・サイズはリモート Backup Server の -p パラメータ値に制限されています。-p を指定しない場合、デフォルトは 2048 バイトです。パケット・サイズは、256 以上の整数にします。

-m max_shared_memory

Backup Server がすべての dump セッションや load セッションで使用できる共有メモリの最大メガバイト値を指定します。

使用法

- Adaptive Server 15.5 に付属の使用する Backup Server 15.5 が、クラスターおよびノンクラスターの両方のバージョンであることを確認します。Backup Server 15.0.3 より前のバージョンは、Adaptive Server 15.5 で使用しないでください。
- Backup Server を起動するには、`backupserver` プログラムを直接実行するのではなく、`startserver` コマンドを使用します。
 - UNIX でデフォルト値を変更するには、Sybase インストール・ディレクトリにある `RUN_servername` ファイルを編集します。詳細については、`startserver` のページを参照してください。
 - Windows でデフォルト値を変更するには、Server Config を使用して Backup Server のコマンド・ライン・パラメータを変更します。詳細については、『設定ガイド』を参照してください。
- `dump` コマンドにはデバイス・ドライバ・オプションを正確に指定してください。Backup Server は、指定されたデバイス・ドライバ・オプションを `dump` コマンドの実行時には検証しません。たとえば、テープを使用する前に Backup Server で強制的に巻き戻すオプションを指定した場合、テープは、ダンプ箇所から読み込まれるのではなく、常に先頭まで巻き戻されます。
- UNIX では、-S パラメータで Backup Server 名を指定せず、DSLISTEN 環境変数も設定していない場合、`backupserver` はデフォルトの Backup Server 名である `SYB_BACKUP` を使用します。

(Windows の場合) `bcksvr` はデフォルトの Backup Server 名である `server_name_BS` を使用します。DSLISTEN 環境変数の値によって、このデフォルト値は無効になります。また、-S パラメータを指定すると、デフォルト値と DSLISTEN に指定された値の両方が無効になります。

- Backup Server と、Backup Server を介してダンプやロードを直接実行する Adaptive Server では、できるかぎり同じ interfaces ファイル (UNIX では *sql.ini*) を共有してください。Backup Server で使用する interfaces ファイルには、次のエントリが必要です。
 - Backup Server
 - 上記の Backup Server とやりとりする他の Backup Server
- Backup Server でトレース・フラグを設定すると、実行中のオペレーションについての情報が出力されるので、Backup Server で発生する問題のデバッグに利用できます。トレース・フラグの詳細については、『Open Server Server-Library/C リファレンス・マニュアル』を参照してください。Backup Server は、-T に対応する Open Server 定義の SRV_TR 記号をサポートしていません。
- Backup Server が -L パラメータと -J パラメータによって指定された *locales* ディレクトリと *charsets* ディレクトリを検出できなかった場合、またはこれらのパラメータに誤った組み合わせの言語と文字セットが指定されている場合は、Backup Server はエラー・メッセージを発行してデフォルトの言語と文字セットを使用します。
- Backup Server は、異なる論理ページ・サイズを使用するサーバ間でのロードまたはダンプを実行できません。たとえば、4K の論理ページ・サイズのデータベース・ダンプは、4K の論理ページ・サイズを使用する別のサーバにロードできます。しかし、Backup Server は 4K の論理ページ・サイズのデータベースをダンプし、それを 16K の論理ページ・サイズを使用するデータベースにロードすることはできません。

パーミッション

このバイナリの実行パーミッションを持つユーザと、すべてのファイルに対する読み込み／書き込みアクセス権を持つユーザが使用できます。

参照

ユーティリティ [startserver](#)

bcp

説明

ユーザが指定したフォーマットで、データベース・テーブルをオペレーティング・システム・ファイルに、またはオペレーティング・システム・ファイルからコピーします。bcp は `$$SYBASE/$$SYBASE_OCS/bin` にあります。

(Windows の場合) このユーティリティは `bcp.exe` で、`$$SYBASE$$SYBASE_OCS/bin` にあります。

構文

```
bcp [[database_name.]owner.]table_name [:[ partition_id | slice_number ] |
partition
partition_name] {in | out} [datafile]
[-a display_charset]
[-A packet_size]
[-b batch_size]
[-c]
[-C]
[-d discardfileprefix]
[-e errfile]
[-E]
[-f formatfile]
[-F firstrow]
[-g id_start_value]
[-i input_file]
[-I interfaces_file]
[-J client_character_set]
[-K keytab_file]
[-L lastrow]
[-m maxerrors]
[-MLabelName LabelValue] [-labeled]
[-n]
[-N]
[-o output_file]
[-P password]
[-Q]
[-r row_terminator]
[-R remote_server_principal]
[-S server]
[-t field_terminator]
[-T text_or_image_size]
[-U username]
[-v]
[-V [security_options]]
[-W]
[-x trusted.txt_file]
[-X]
[-y alternate_home_directory]
[-Y ]
[-z language ]
```

```

[-Z security_mechanism]
[--colpasswd [[[db_name.[owner].]table_name.]
                column_name [password]]]
[--keypasswd [[db_name.[owner].]key_name [password]]]
[--hide-vcc]
[--initstring "TSQL_command"]
[--maxconn maximum_connections]
[--show-fi]
[--skiprows nSkipRows]

```

パラメータ

database_name

コピーするテーブルがデフォルト・データベース内または **master** 内にある場合は、このパラメータはオプションとして使用できます。そうでない場合は、データベース名を指定しなければなりません。

owner

ユーザまたはデータベース所有者がコピーするテーブルを所有している場合は、このパラメータはオプションとして使用できます。ユーザが所有者を指定しなかった場合、**bcp** は、まずそのユーザが所有している名前のテーブルを検索し、次にデータベース所有者が所有しているテーブルを検索します。それ以外のユーザがテーブルを所有している場合は、所有者の名前を指定しなければなりません。指定しないと、コマンドは失敗します。

table_name

コピーするデータベース・テーブルの名前です。Transact-SQL の予約語をテーブル名に使用することはできません。

partition_id

データのコピー先となるパーティション番号を指定します。このパラメータをサポートしているのは、**bcp in** だけです。Adaptive Server 12.5.x の **slice_number** に相当します。

slice_number

データのコピー先となるパーティション・スライスを指定します。このパラメータをサポートしているのは、**bcp in** と Adaptive Server 15.0 以降のラウンドロビン方式で分割されたテーブルだけです。

partition partition_name

1 つ以上のパーティションをカンマで区切って指定します。

in | out

コピーの方向を示します。**in** は、ファイルからデータベース・テーブルへのコピーであることを示し、**out** は、データベース・テーブルまたはビューからファイルへのコピーであることを示します。

datafile

1つ以上のユニークなデータ・ファイルをカンマで区切って指定します。このパラメータは、**bcp in** と **bcp out** の両方でサポートされています。パス名は、1～255文字で指定します。

-a display_charset

bcp を実行しているマシンの文字セットと異なる文字セットを使用する端末から、**bcp** を実行できます。**-a** を **-J** とともに使用して、変換に必要な文字セット変換ファイル (*.xlt* ファイル) を指定します。**-J** を使用しないで **-a** を使用するの、クライアントの文字セットがデフォルトの文字セットと同じである場合だけです。

-a パラメータで指定した文字変換ファイルが存在しないか、ファイル名の入力ミスがあると、次のエラー・メッセージが表示されます。

```
Error in attempting to determine the size of a pair of translation tables.: 'stat' utility failed.
```

-A packet_size

この **bcp** セッションで使用するネットワーク・パケット・サイズを指定します。たとえば、この **bcp** セッションのパケット・サイズを4096バイトに指定するときは、次のように入力します。

```
bcp pubs2..titles out table_out -A 4096
```

packet_size は、**default network packet size** 設定変数と **maximum network packet size** 設定変数の間の値であり、512の倍数であることが必要です。

大規模なバルクコピー・オペレーションのパフォーマンスを向上させるには、デフォルトのネットワーク・パケットよりも大きいサイズを使用します。

-b *batchsize*

バッチごとにコピーされるデータのロー数です。デフォルトでは、**bcp in** は1つのバッチ処理で *n* 個のローをコピーします。*n* はバッチ・サイズに相当します。バッチ処理は、データベース・テーブルへのバルク・コピー・インだけに適用されます。ファイルへのバルク・コピー・アウトには適用されません。**bcp** が *batchsize* に受け入れる最小数は1です。**bcp** が *batchsize* に受け入れる最大数は2147483647Lです。

注意 バッチ・サイズを1に設定すると、Adaptive Server はコピー・インする1つのローに1つのデータ・ページを割り当てます。このオプションは、高速 **bcp** だけに適用され、データの破損したローを見つける場合にのみ有用です。**-b1** の使用には注意してください。**-b1** を使用すると、各ローに新しいページが割り当てられるため、領域の利用効率が低下します。

-c

データ・ファイル内の全カラムのデフォルト記憶タイプとして **char** データ型を使用して、コピー・オペレーションを実行します。プラットフォーム間でデータを共有するときはこのフォーマットを使用してください。このパラメータを指定すると、各フィールドに対するプロンプトは表示されません。このパラメータは、デフォルトの記憶タイプとして **char** を使用し、プレフィクスは付きません。デフォルトのフィールド・ターミネータとして **¥t** (タブ) を使用し、デフォルトのロー・ターミネータとして **¥n** (改行) を使用します。

-C

Adaptive Server が暗号化カラムをサポートしている場合は、暗号化カラムのバルク・コピーもサポートします。**-C** を指定すると、バルク・コピー・オペレーションの開始前に **ciphertext** オプションが有効化されます。

-d *discardfileprefix*

拒否されたローを専用の破棄ファイルに記録します。破棄ファイルのフォーマットはホスト・ファイルと同じです。このファイルは、指定された破棄ファイル・プレフィクスの後に入力ファイル名を追加することによって作成されます。このファイル内のローを修正し、それを使用して修正後のローを再ロードできます。

Sybase では、**-d *discardfileprefix*** オプションを **-e *errorfile*** とともに使用して、破棄ファイルに記録された、問題のあるローを特定および診断するのに役立てることをおすすめします。

-e errfile

bcp がファイルからデータベースに転送できなかったローを保管するエラー・ファイルのフル・パス名です。**bcp** からのエラー・メッセージは、使用している端末に表示されます。**bcp** がエラー・ファイルを作成するのは、このパラメータが指定されたときだけです。

-E

テーブルの **IDENTITY** カラムの値を明示的に指定します。

-E パラメータは、バルク・コピー・アウトには影響しません。**-N** パラメータを使用しない場合、**Adaptive Server** は **ID** カラムをデータ・ファイルにコピーします。

-E フラグと **-g** フラグを同時に使用することはできません。

-f formatfile

同一テーブル内で前回の **bcp** 実行時の応答が保管してあるファイルのフル・パス名です。**bcp** に対して使用するフォーマットを入力すると、**bcp** はフォーマット・ファイルとしてその形式を保存するかどうかを尋ねてきます。フォーマット・ファイルの作成はオプションです。デフォルトのファイル名は、*bcp.fmt* です。**bcp** プログラムはデータのコピー時にフォーマット・ファイルを参照できるため、ユーザは以前に指定したフォーマットを繰り返し指定する必要はありません。コピー・インまたはコピー・アウトを行うときに以前作成したフォーマット・ファイルを使用する場合にだけ、この **-f** パラメータを使用してください。このパラメータを指定しない場合、**bcp** はフォーマット情報を対話形式で問い合わせてきます。

-F firstrow

入力ファイルからのコピー開始のロー番号です (デフォルトでは、先頭のロー)。

高負荷、マルチプロセスのコピーを実行する場合は、**-F** オプションの使用を避けてください。このオプションを使用すると、**bcp** は一般に、動作のために必要な作業が多くなり、処理が遅くなります。**-F** は、単一のプロセス、特定のコピーに使用してください。

-g id_start_value

データをコピー・インするときの開始ポイントとして使用する、**IDENTITY** カラムの値を指定します。

-g フラグと **-E** フラグを同時に使用することはできません。

-i input_file

入力ファイル名を指定します。デフォルトとして標準入力を使用されます。

-I *interfaces_file*

Adaptive Server に接続するときに検索する *interfaces* ファイルの名前とロケーションを指定します。**-I** を指定しない場合、**bcp** は SYBASE 環境変数によって指定されたディレクトリ (Windows では *ini* ディレクトリ) にある *interfaces* ファイル (Windows では *sql.ini*) を探します。

-J *client_charset*

クライアントで使用する文字セットを指定します。**bcp** は、フィルタを使用して *client_charset* と Adaptive Server の文字セット間で入力を変換します。

-J *client_charset* は、クライアントで使用する文字セットである *client_charset* とサーバの文字セット間の変換を Adaptive Server に要求します。

-J に引数を指定しない場合、文字セット変換は NULL に設定されます。この場合、変換は行われません。クライアントとサーバが同じ文字セットを使用する場合に、このパラメータを使用してください。

-J を省略すると、文字セットはプラットフォームのデフォルトに設定されます。デフォルトの文字セットは、クライアントが使用している文字セットと同じであるとはかぎりません。

-J パラメータで不正な文字セットまたは認識できない文字セットを指定すると、次のエラー・メッセージが表示されます。

```
Unrecognized localization object.Using default value
'iso_1'.
Starting copy...
=> warning.
```

文字セットおよび対応するフラグの詳細については、『システム管理ガイド』を参照してください。

-K *keytab_file*

DCE での認証に使用する *keytab* ファイルのパスを指定します。

-L *lastrow*

入力ファイルからのコピー終了のロー番号です (デフォルトでは、最後のロー)。複数のファイルを使用している場合、このオプションはそれぞれのファイルに適用されます。

-m maxerrors

bcp がコピーをアボートするまでに許容される致命的でないエラーの最大数です。**bcp** は、(データ変換エラー、または **null** 値を受け付けないカラムに **null** 値を挿入しようとしたために) 挿入できないローを破棄し、拒否した各ローを1つのエラーとしてカウントします。このパラメータを指定しない場合、**bcp** はデフォルト値の 10 を使用します。

-M LabelName LabelValue

(Secure Adaptive Server のみ) マルチレベル・ユーザがバルク・コピーのセッション・ラベルを設定できるようにします。**LabelName** の値は、次のとおりです。

- **current** (現在の読み込みレベル) — このセッション中に読み込むことができるデータの初期レベル。**current** は、**curwrite** よりも高いレベルにしてください。
- **curwrite** (現在の書き込みレベル) — このセッション中に書き込むすべてのデータに適用される初期 **sensitivity** レベル。
- **maxread** (読み込みレベルの最大値) — データを読み込むことができる最大レベル。この値は、マルチレベル・ユーザとしてこのセッション中に **current** に設定できる上限値です。**maxread** は、**maxwrite** よりも高いレベルにしてください。
- **maxwrite** (書き込みレベルの最大値) — データを書き込むことができる最大レベル。この値は、マルチレベル・ユーザとしてこのセッション中に **curwrite** に設定できる上限値です。**maxwrite** は、**minwrite** と **curwrite** よりも高いレベルにしてください。
- **minwrite** (書き込みレベルの最小値) — データを書き込むことができる最小レベル。この値は、マルチレベル・ユーザとしてこのセッション中に **curwrite** に設定できる下限値です。**minwrite** は、**maxwrite** と **curwrite** よりも低いレベルにしてください。

LabelValue は、システム上で使用される、人間の目で判読できるフォーマットで表現された実際のラベル値(たとえば “Company Confidential Personnel”)です。

-labeled

(Secure Adaptive Server のみ) インポートしているデータの、すべてのレコードの最初のフィールドに、すでにラベルがあることを示します。

エクスポートしているデータの場合、このオプションはすべてのローの **sensitivity** ラベルを最初のフィールドとしてコピー・アウトすることを示します。

-n

ネイティブの (オペレーティング・システムの) フォーマットを使用して、コピー・オペレーションを実行します。-n パラメータを指定すると、bcp は各フィールドに対するプロンプトを表示しません。ネイティブ・データ・フォーマットのファイルは人間が判読できるフォーマットにはなっていません。

警告！ 次のような使い方はできません。

- データ・リカバリやサルベージ、または緊急の問題解決のために、ネイティブ・フォーマットを使用して bcp を実行する。
- 異なるハードウェア・プラットフォーム間、異なるオペレーティング・システム間、または異なるメジャー・リリースの Adaptive Server 間で、ネイティブ・フォーマットの bcp を使用してデータを転送する。
- フィールド・ターミネータ (-t) やロー・ターミネータ (-r) を、ネイティブ・フォーマットの bcp とともに使用する。

予期しない結果になったり、データが破損したりする可能性があります。

ネイティブ・フォーマットを使用して bcp を実行した場合、Adaptive Server に再ロードできないフラット・ファイルが作成され、データをリカバリできなくなることがあります。bcp を文字フォーマットで再実行できない場合 (たとえば、テーブルがトランケートされたり削除されたりした場合、ハードウェアが損傷した場合、データベースが削除された場合など) は、データをリカバリできません。

-N

IDENTITY カラムをスキップします。ホスト・データ・ファイルに IDENTITY カラムの値のためのプレースホルダが含まれていない場合にデータをコピー・インするとき、またはデータをコピー・アウトするときにホスト・ファイルに IDENTITY カラム情報を含めたくない場合にこのオプションを使用します。

データをコピー・インするときに、-N パラメータと -E パラメータの両方を使用することはできません。

-o *output_file*

出力ファイル名を指定します。デフォルトとして標準出力が使用されます。

-P password

Adaptive Server のパスワードを指定します。**-Ppassword** を指定しない場合、**bcp** はパスワードを入力するよう要求します。パスワードが NULL の場合は、**-P** フラグを省略できます。

-Q

null 入力可能カラムを含むコピー・オペレーションの **bcp** バージョン 10.0.4 との下位互換性を提供します。

-r row_terminator

ロー・ターミネータを指定します。

警告！ ネイティブ・フォーマットで **bcp** を実行する場合は、**-t** パラメータまたは **-r** パラメータを使用しないでください。予期しない結果になったり、データが破損したりする可能性があります。

コマンド・ラインから **-t** パラメータまたは **-r** パラメータでターミネータを指定する場合、UNIX オペレーティング・システム (または Windows のコマンド・プロンプト・シェル) にとって特別な意味を持つ文字をエスケープしてください。詳細については、**bcp** の例を参照してください。特殊文字の前に円記号を付けるか、特殊文字を引用符で囲んでください。**bcp** がプロンプトを表示する場合 (対話型モード) は、その必要はありません。

-R remote_server_principal

セキュリティ・メカニズムに指定されたリモート・サーバのプリンシパル名を指定します。デフォルトでは、サーバのプリンシパル名はサーバのネットワーク名 (**-S** パラメータまたは **DSQUERY** 環境変数で指定) と一致します。サーバのプリンシパル名とネットワーク名が一致しない場合は、**-R** パラメータを使用してください。

-S server

接続する Adaptive Server の名前を指定します。引数なしで **-S** を指定した場合、**bcp** は **DSQUERY** 環境変数によって指定されたサーバを使用します。

-t field_terminator

デフォルトのフィールド・ターミネータを指定します。

-T text_or_image_size

Adaptive Server が送信する **text** データまたは **image** データの最大長をバイト単位で指定します。デフォルトは、32K です。**text** フィールドまたは **image** フィールドが **-T** の値またはデフォルトより大きい場合は、**bcp** はオーバフローしたデータを送信しません。

-U *username*

Adaptive Server ログイン名を指定します。

-v

bcp のバージョン番号と著作権メッセージを表示して、オペレーティング・システムに戻ります。

-V *security_options*

ネットワーク・ベースのユーザ認証を指定します。このオプションを使用する場合、ユーザはユーティリティを実行する前にネットワークのセキュリティ・システムにログインする必要があります。この場合、ユーザは **-U** オプションにネットワーク・ユーザ名を指定し、**-P** オプションによって指定されたパスワードは無視されます。

-V の後にキー文字オプションの *security_options* 文字列を続けると、他のセキュリティ・サービスを有効にできます。これらのキー文字は、以下のとおりです。

- **c** – データ機密性サービスを有効にする
- **i** – データ整合性サービスを有効にする
- **m** – 接続の確立に相互認証を有効にする
- **o** – データ・オリジン・スタンプング・サービスを有効にする
- **r** – データ・リプレイの検出を有効にする
- **q** – 順序不整合の検出を有効にする

-W

bcp で接続しようとしているサーバが通常のパスワード暗号化と拡張パスワード暗号化のどちらもサポートしていない場合、プレーン・テキスト形式のパスワードを使用した接続再試行を無効にするように指定します。このオプションを使用すると、**CS_SEC_NON_ENCRYPTION_RETRY** 接続プロパティは **CS_FALSE** に設定され、プレーン・テキスト形式の (暗号化されていない) パスワードは接続の再試行時に使用されません。

-x *trusted.txt file*

代替の *trusted.txt* ファイルを指定します。

-X

サーバへの現在の接続で、アプリケーションがクライアント側のパスワード暗号化を使用してログインを開始するように指定します。**bcp** (クライアント) は、パスワードの暗号化が必要であることをサーバに通知します。サーバは、**bcp** がパスワードを暗号化するために使う暗号化キーを返送し、パスワードを受け取ると、そのキーを使用してそのパスワードを確認します。

bcp がクラッシュすると、パスワードを含むコア・ファイルが作成されます。暗号化オプションを使用していない場合、パスワードは、コア・ファイルにプレーン・テキストで表示されます。暗号化オプションを使用した場合、パスワードは表示されません。

-y *sybase_directory*

代替の Sybase ホーム・ディレクトリを設定します。

-Y

bcp in の使用時に、サーバでの文字セット変換を無効にし、代わりにクライアント側で **bcp** によって文字セット変換を実行することを指定します。

注意 クライアント側の Unicode 変換は、Adaptive Server 15.0 以降でのみサポートされています。

bcp out の使用時には、すべての文字セット変換はサーバで行われます。

-z *language*

サーバが **bcp** のプロンプトとメッセージの表示に使用する代替言語の公式名です。**-z** フラグを指定しないと、**bcp** はサーバのデフォルトの言語を使用します。

言語はインストール時に Adaptive Server に追加できます。インストール後でも、**langinstall** ユーティリティ (Windows では **langinst**) または **sp_addlanguage** ストアド・プロシージャを使用して言語を追加できます。

-z パラメータに不正な言語または認識できない言語を指定すると、次のエラー・メッセージが表示されます。

```
Unrecognized localization object.Using default value
'us_english'.
Starting copy...
=> warning.
```

-Z security_mechanism

接続で使用するセキュリティ・メカニズムの名前を指定します。

セキュリティ・メカニズム名は `$SYBASE/install/libtcl.cfg` 設定ファイルに定義されています。`security_mechanism` の名前が指定されていない場合は、デフォルトのメカニズムが使用されます。セキュリティ・メカニズム名の詳細については、『Open Client/Server 設定ガイド』の `libtcl.cfg` ファイルの説明を参照してください。

--colpasswd [[database_name [owner].]table_name.]column_name [password]]

`set encryption passwd password` for column `column_name` を Adaptive Server に送信して、暗号化カラムにパスワードを設定します。これで、他の暗号化カラムが同じキーで暗号化されている場合でも、2 番目のカラムにはパスワードが自動的に適用されません。2 番目のカラムにアクセスするには、パスワードをもう一度指定します。

--hide-vcc

仮想計算カラムをデータ・ファイルにコピーしたり、データ・ファイルからコピーしたりしないよう `bcp` に指示します。`bcp OUT` でこのオプションを使用すると、データ・ファイルに仮想計算カラムのデータは含まれません。`bcp IN` でこのオプションを使用すると、データ・ファイルに仮想計算カラムのデータが含まれない場合があります。

このオプションを使用した場合、Adaptive Server は仮想計算カラムのデータを計算したり、送信したりしません。

--initstring ["Transact-SQL_command"]

任意の Transact-SQL コマンドを現在の `isql` セッションのみの初期化文字列として設定します。初期化文字列によって発行された結果セットは、エラーが発生しないかぎり暗黙的に無視されます。

--keypasswd [[database_name.[owner].]key_name [password]]

`set encryption passwd password` for key `key_name` を Adaptive Server に送信して、キーを使用してアクセスできるすべてのカラムにパスワードを設定します。

--maxconn maximum_connections

`bcp` クライアントがサーバに対してオープンできる並列接続の最大数です。このパラメータを指定しない場合、`bcp` はデフォルト値の 10 を使用します。

--show-fi

`bcp in` または `bcp out` の使用時に、機能インデックスをコピーするよう `bcp` に指示します。

このオプションを使用すると、機能インデックスとサーバ間でデータが送信されます。

--skiprows *nSkipRows*

指定されたロー数をスキップしてから、入力ファイルからのコピーを開始するよう `bcp` に指示します。`--skiprows` の有効範囲は、0 から入力ファイルの実際のロー数までです。無効な値を指定すると、エラー・メッセージが表示されます。

注意 `--skiprows` を `-F` オプションとともに使用することはできません。

例

例 1 データ・ファイル内の全カラムのデフォルト記憶タイプとして `char` データ型を使用し、デフォルトのフィールド・ターミネータとロー・ターミネータを指定して、`publishers` テーブルから `pub_out` ファイルにデータをコピーします。

UNIX では、最後の “r” の前にある 1 つ目の円記号は 2 つ目の円記号をエスケープするため、1 つの円記号だけが出力されます。

```
bcp pubs2..publishers out pub_out -c -t , -r ¥¥r
```

Windows の場合：

```
bcp pubs2..publishers out pub_out -c -t , -r ¥r
```

例 2 `publishers` テーブルから `pub_out` というファイルにデータをコピーします。このデータは、後で `Adaptive Server` に再ロードします。`[Return]` キーを押すと、プロンプトで指定されたデフォルトが使用されます。`publishers` テーブルにデータをコピーするときにも同じプロンプトが表示されます。

```
bcp pubs2..publishers out pub_out
Password:
Enter the file storage type of field pub_id [char]:
Enter prefix length of field pub_id [0]:
Enter length of field pub_id [4]:
Enter field terminator [none]:
Enter the file storage type of field pub_name [char]:
Enter prefix length of field pub_name [1]:
Enter length of field pub_name [40]:
Enter field terminator [none]:
Enter the file storage type of field city [char]:
Enter prefix length of field city [1]:
```

```

Enter length of field city [20]:
Enter field terminator [none]:
Enter the file storage type of field state [char]:
Enter prefix length of field state [1]:
Enter length of field state [2]:
Enter field terminator [none]:

```

UNIX の場合は、次のように問い合わせてきます。

```

Do you want to save this format information in a
file?[Y/n] y
Host filename [bcp.fmt]:pub_form
Starting copy...
3 rows copied.
Clock Time (ms.):total = 1 Avg = 0 (3000.00 rows per
sec.)

```

例 3 publishers テーブルから pub_out というファイルにデータをコピーします。このデータは、後で Adaptive Server に再ロードします。
[Return] キーを押すと、プロンプトで指定されたデフォルトが使用されます。publishers テーブルにデータをコピーするときも、同じプロンプトが表示されます。

```

bcp pubs2..publishers out pub_out

Password

Enter the file storage type of field pub_id [char]:
Enter prefix length of field pub_id [0]:
Enter length of field pub_id [4]:
Enter field terminator [none]:
Enter the file storage type of field pub_name [char]:
Enter prefix length of field pub_name [1]:
Enter length of field pub_name [40]:
Enter field terminator [none]:
Enter the file storage type of field city [char]:
Enter prefix length of field city [1]:
Enter length of field city [20]:

Enter field terminator [none]

Enter the file storage type of field state [char]:
Enter prefix length of field state [1]:
Enter length of field state [2]:
Enter field terminator [none]:

```

次のように問い合わせてきます。


```

Do you want to save this format information in a
file?[Y-n]

Host filename [bcp.fmt]:pub_form

Starting copy...

3 rows copied.

Clock time (ms.):total = 1 Avg = 0 (3000.00 rows per
sec.)

```

例 4 テーブルの `p1` パーティションのデータを、現在のディレクトリの `mypart.dat` ファイルにコピー・アウトします。

```
bcp t1 partition p1 out mypart.dat
```

例 5 この例では、保存された `pub_form` フォーマット・ファイルを使用して、Adaptive Server にデータをコピーして戻します。

```
bcp pubs2..publishers in pub_out -f pub_form
```

例 6 この例では、画面の表示に従って 1 文字を入力します。

To see examples of datatypes, enter "?" at the prompt:

```

フィールド 'pub_id' のファイル記憶タイプを入力してください。
['char']:?

```

Invalid column type.Valid types are:

```

<cr>:same type as Adaptive Server column.
c :char
T :text
i :int
s :smallint
t :tinyint
f :float
m :money
b :bit
d :datetime
x :binary
I :image
D :smalldatetime
r :real
M :smallmoney
n :numeric
e :decimal

```

例 7 VT200 端末で使用している文字セットで作成したデータ・ファイルを `pubs2..publishers` テーブルにコピーします。`-z` フラグは、`bcp` メッセージをフランス語で表示します。

```
bcp pubs2..publishers in vt200_data -J iso_1 -z french
```

例 8 (UNIX の場合) Macintosh を使用して、`roman8` を使用しているワークステーション上で `bcp` を実行していることを指定します。

```
bcp pubs2..publishers in -a mac -J roman8
```

例 9 Adaptive Server が 4096 バイトのパケット・サイズを使用して 40K の `text` データまたは `image` データを送信することを指定します。

```
bcp pubs2..publishers out -T 40960 -A 4096
```

例 10 各オペレーションで許可する並列接続の最大数を 2 に設定します。

```
bcp_r --maxconn 2
```

例 11 現在のディレクトリの `mypart.dat` ファイルを `t1` テーブルの `p1` パーティションにコピーします。

```
bcp t1 partition p1 in mypart.dat
```

例 12 パーティション `p1`、`p2`、`p3` を `¥work2¥data` ディレクトリにあるファイル `a`、`b`、`c` にそれぞれコピーします。

```
bcp t1 partition p1, p2, p3 out ¥work2¥data¥1,
¥work2¥data¥b, ¥work2¥data¥c
```

例 13 これは現在のセッションに限定されるため、`titles.txt` から `pubs2..titles` へのデータ転送中に `bcp` 接続の複写を無効にします。

```
bcp pubs2..titles in titles.txt --initstring 'set replication off'
```

`bcp` の終了後に設定オプションを明示的に再設定する必要はありません。Adaptive Server からエラーが返された場合、`bcp` ではデータ転送が停止し、エラー・メッセージが表示されます。

例 14 `db_1` データベースをコピー・アウトします。このデータベースには、マテリアライズされた計算カラム `c1` のある `t1` テーブルが含まれます。

```
bcp db_1..t1 out db_1.dat -Usa -P -S big_db -I./interfaces -f ./bcp.fmt
```

例 15 例 12 で作成したデータ・ファイル (`db_1.dat`) をコピー・インします。このファイルには、マテリアライズされた計算カラム `c1` のある `t1` テーブルが含まれます。

```
bcp db_1..t1 in db_1.dat -Usa -P -S big_db -I./interfaces -f ./bcp.fmt
```

例 16 `titles.txt` データを `pubs2..titles` テーブルに転送するときに、高速にログを取る `bcp` を有効にします。

```
bcp pubs2..titles in titles.txt --initstring 'set logbulkcopy on'
```

使用法

- スレッド・ドライバを使用する場合は、`bcp` のこの構文を使用します。
- ファイルのコピー・イン、コピー・アウトに名前付きパイプを使用することはできません。
- サーバとの間で暗号化データをコピー・インおよびコピー・アウトするには、`bcp` を使用できます。「暗号化データのバルク・コピー」(65 ページ) を参照してください。
- `bcp` の初期化文字列として `--initstring` を指定して `Transact-SQL` コマンドを使用できますが、サーバ設定に加えられている可能性のある永続的変更を `bcp` の実行後に再設定する必要があります。たとえば、別の `isql` セッションで変更をリセットできます。
- `bcp` バージョン 15.7 以降では、マテリアライズされていないカラムを含むテーブルにデータをコピーできます。
- エラー・メッセージ・フォーマットは、以前のバージョンの `bcp` と異なります。以前のメッセージの値に基づいてルーチンを実行するスクリプトがある場合は、それらの書き換えが必要な場合があります。

転送されたロー数を示すメッセージが変更されています。
セッション中に、このバージョンの `bcp` は、転送されたロー数をレポートします。
このメッセージは、以前のバージョンの `bcp` で出力されていたメッセージ "1000 rows transferred" に置き換わるものです。
- `--hide-vcc` を使用すると、`Adaptive Server` が仮想計算カラムのデータを転送したり、計算したりしないため、パフォーマンスが向上します。
- `slice_number` は、`Adaptive Server 12.5.x` 以前のバージョンとの下位互換性を保つために含まれており、ラウンドロビン方式で分割されたテーブルでのみ使用できます。
- `partition_id` か `partition_name` のいずれかを指定できます。ただし、この両方を指定することはできません。
- 複数のパーティションとデータ・ファイルを指定できます。各パーティション名またはデータ・ファイル名をカンマで区切ります。
- パーティション名を指定しない場合、`bcp` はテーブル全体にコピーします。
- `bcp out` を使用する場合：

- *partition_name* と *datafile* の両方を指定する場合、*datafile* でデータ・ファイルを1つ指定するか、パーティション名とデータ・ファイルの1対1のマッピングを指定してください。
- *datafile* が指定されていない場合、各パーティションのデータは、パーティション名に *.dat* 拡張子が付いたファイル名のファイルにコピーされます。たとえば、パーティション名が *ptn1* の場合、データ・ファイルは *ptn1.dat* になります。
- *initstring* を使用して任意の Transact-SQL コマンドを実行できますが、*initstring* によって実行されたサーバへの永続的変更を *bcp* の完了後にリセットする必要があります。たとえば、上記の例 11 で、Adaptive Server アカウントに適切なパーミッションがない場合、Adaptive Server によって初期化文字列に対するエラー・メッセージが返されます。*bcp* は、データを転送する前にサーバ・エラー・メッセージを表示して停止します。

初期化文字列によって発行された結果セットは、エラーが発生しないかぎり無視されます。

- *bcp in* の使用時に *partition_name* を指定する場合、*datafile* では対応する数のデータ・ファイルを指定してください。

圧縮データでの *bcp* の使用

圧縮テーブル内のページは、ロー圧縮ロー、ページ圧縮ロー、および非圧縮ローの組み合わせになる場合があります。*not compressed* としてリストされたテーブルとパーティションは、テーブルの圧縮レベルが異なるときに作成されたため、圧縮状態が異なるローが混在する場合があります。

- *bcp out*
 - 圧縮ローが圧縮解除され、ネイティブ形式または文字形式でクライアントに返されます。
 - IDENTITY、暗号化カラムなどをサポートしています。
 - テキスト・データを非圧縮として返します。
- *bcp in* では、テーブルやパーティションの圧縮レベルに合わせて、クライアントから受け取った非圧縮データを圧縮します。

データをコピーで取り出して、*bcp* で圧縮に対して設定されたテーブルに戻すために *bcp* を使用すると、データが初期設定で非圧縮になっている場合でも、圧縮データになります。

- パーミッション** bcp を使用するには、Adaptive Server アカウントと、データベースのテーブルまたはビューおよび転送に使用するオペレーティング・システム・ファイルに対する適切なパーミッションが必要です。
- テーブルにデータをコピーするには、そのテーブルに対する insert パーミッションが必要です。
 - テーブルをオペレーティング・システム・ファイルにコピーするには、次のテーブルに対する select パーミッションが必要です。
 - コピーするテーブル
 - sysobjects
 - syscolumns
 - sysindexes

監査 event カラムと extrainfo カラムの値は次のとおりです。

イベント	監査オプション	監査対象のコマンドまたはアクセス	extrainfo の情報
4	bcp	bcp in	<ul style="list-style-type: none"> • <i>Roles</i> – 現在のアクティブな役割 • <i>Keywords</i> または <i>options</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – set proxy が有効な場合は元のログイン名

使用するテーブル sysaudits_01 ~ sysaudits_08

参照 **コマンド** insert

マニュアル bcp の詳細については、「[第4章 bcp を使用した Adaptive Server とのデータの転送](#)」を参照してください。大きなバッチの場合に、パラメータの変更によって bcp の動作がどのように変わるかについては、『パフォーマンス&チューニング・ガイド』を参照してください。

システム・プロシージャ sp_audit、sp_dboption、sp_displayaudit

buildmaster

説明	Adaptive Server バージョン 12.5 以降では、マスタ・デバイスの構築に buildmaster バイナリを使用しません。代わりに、dataserver バイナリに buildmaster の機能が組み込まれました。詳細については、「 第 1 章 dataserver を使用したサーバの構築 」を参照してください。構文の詳細については、「 dataserver 」(173 ページ)を参照してください。
構文	なし

certauth

説明 サーバ証明書要求を CA- (認証機関) の署名付き証明書に変換します。
 \$SYBASE/\$SYBASE_OCS/bin にあります。

(Windows の場合) このユーティリティは certauth.exe で、 %SYBASE%\\$SYBASE_OCS%\\$bin にあります。

構文

```
certauth
    [-r]
    [-C caCert_file]
    [-Q request_filename]
    [-K caKey_filename]
    [-N serial_number]
    [-O SignedCert_filename]
    [-P caPassword]
    [-s start_time]
    [-T valid_time]
```

または

```
certauth -v
```

パラメータ

- r
 テスト環境用の自己署名付きルート証明書を作成します。
- C *caCert_file*
 -r を指定したときに CA の証明書要求の名前を指定します。または、CA のルート証明書の名前を指定します。
- Q *request_filename*
 証明書要求ファイルの名前を指定します。
- K *caKey_filename*
 CA のプライベート・キーの名前を指定します。
- N *serial_number*
 署名付き証明書のシリアル番号を指定します。-N が指定されていない場合、certauth は疑似ランダム・シリアル番号を生成します。
- O *SignedCert_filename*
 署名付き証明書ファイルを作成する場合に出力用を使用する名前を指定します。-r を指定する場合は、*SignedCert_filename* は自己署名付きルート証明書です。-r オプションを使用しない場合、*SignedCert_filename* は *caCert_file* によって署名された証明書です。
- P *caPassword*
 プライベート・キーの復号化に使用する CA のパスワードを指定します。

-s start_time

有効期間の開始時刻を指定します。有効期間は、現在の時刻を基点として日単位で計算されます。デフォルトは現在の時刻です。

-T valid_time

署名付き証明書の有効期間を指定します。有効期間は日単位です。

-v

certauth のバージョン番号と著作権メッセージを表示して、終了します。

例

例 1 プライベート・キー (*ca_pkey.txt*) を使用して、CA の証明書要求 (*ca_req.txt*) を証明書に変換します。プライベート・キーは *password* で保護されています。この例では、有効期間を 365 日に設定し、証明書に自己署名してルート証明書 (*trusted.txt*) として出力します。

```
certauth -r -C ca_req.txt -Q ca_req.txt
        -K ca_pkey.txt -P password -T 365 -O trusted.txt
```

ユーティリティは、次のメッセージを返します。

```
-- Sybase Test Certificate Authority--
Certificate Validity:
    startDate = Tue Sep 5 10:34:43 2000
    endDate = Wed Sep 5 10:34:43 2001
    CA sign certificate SUCCEED (0)
```

注意 テスト CA 用に信頼されたルート証明書を 1 回だけ作成する必要があります。信頼されたルート証明書を作成すると、テスト環境で多数のサーバ証明書の署名に使用できます。

例 2 サーバ証明書要求 (*srv5_req.txt*) を証明書に変換して、有効期間を 180 日に設定します。CA の証明書とプライベート・キー (*trusted.txt* と *ca_pkey.txt*) を持つ証明書に署名し、パスワード保護を使用し、署名付き証明書を *sybase_srv5.crt* として出力します。

```
certauth -C trusted.txt -Q srv5_req.txt
        -K ca_pkey.txt -P password -T 180 -O sybase_srv5.crt
```

注意 有効期間を設定しない場合は、デフォルトの 365 日が使用されます。

ユーティリティは、次のメッセージを返します。

```
-- Sybase Test Certificate Authority--
Certificate Validity:
```



```

startDate = Tue Sep  5 10:38:32  2000
endDate = Sun Mar  4 09:38:32  2001
CA sign certificate SUCCEEDED (0)

```

次に、証明書の例を示します。サーバが使用できるサーバ証明書の作成手順については、`certauth` の「使用法」の項を参照してください。

```

-----BEGIN CERTIFICATE-----
MIICSTCCAgUCAVAwCwYHKOZiZjgEAWUAMG8xCzAJBgNVBAYTA1VTMRMwEQYDVQQLI
EwpDYWxpZm9ybmlhMRRMwEQYDVQQHEWpFbWVyeXZpbGx1MQ8wDQYDVQQKFAZTeWh
c2UxDDAKBgNVBAsUA0RTVDEXMBUGA1UEAxQOc3liYXNlX3R1c3RfY2EwHhcNMDAw
ODE4MTkxMzM0WhcNMDAwODE4MTkxMzM0WjBvMQswCQYDVQQGEWJVUzETMBEGAUE
CBMKQ2FsaWZvcn5pYTETMBEGA1UEBxMKRW1lcml2aWxsZTEPMA0GA1UEChQGUGU3li
YXNlMQwwCgYDVQQQLFANEU1QxYFzAVBgNVBAMUDnN5YmFzZV90ZXN0X2NhMIHwMIo
BgcqhkhjOOAQBMIGcAkEA+6xG7XCxiklxbP96nHBnQrTLTCjHlcy8QhIekwv90lqG
EMG9AjJLxj6VckPOD75vqVMEkaPPj0IbXEJEe/aYXQIVAPyY1+B9phC2e2YFcf7
cReCcSNxAkBBHt7rnOJZ1Dnd8iLQGtOwd1w4lo/Xx2OeZS4CJW0KVkGIid1hNGz8r
GrQTspWcwTh2rNGbXxlNXhAV5g4OCgrYA0MAAkA70uNEl90KmhdT3RISiceCMGOf
1J8dgtWF15mcHeS8OmF9s/vqPAR5NkaVk7LJK6k7QvXUBY+8LM0ugpJf/TYMASG
ByqGSM44BAMFAAMxADAuAhUAhM2Icn1pSavQtXFzXJUCoOmNlPkCFQDtE8RUGuo8

```

```

ZdxnQtPu9uJDmoBiUQ==

```

```

-----END CERTIFICATE-----

```

使用法

- `Adaptive Server` では `openssl` オープン・ソース・ユーティリティが `$$SYBASE/$$SYBASE_OCS/bin` (Windows では `%SYBASE%\$SYBASE_OCS%\bin`) に含まれています。`certreq`、`certauth`、`certpk12` で実装されたすべての証明書管理タスクを実行するには `openssl` を使用します。`Sybase` では便宜上このバイナリを組み込んでいますが、バイナリを使用して発生した問題についてはいっさい責任を負いません。詳細については、[OpenSSL Web site \(http://www.openssl.org\)](http://www.openssl.org) を参照してください。
- `-N` オプションで指定するシリアル番号の最大長は、16 進文字で 20 文字です。指定したシリアル番号がこれよりも長い場合、`certauth` はシリアル番号を最大長にトランケートします。
- `certauth` を実行するには、ダイナミック・ライブラリ検索パス内で `$$SYBASE/$$SYBASE_OCS/libp364` のエントリの前に `$$SYBASE/$$SYBASE_OCS/lib3p` のエントリを配置する必要があります。

- Adaptive Server が認識するサーバ証明書ファイルを作成するには、署名付き証明書ファイルの最後に証明書リクエストのプライベート・キーを追加します。上記の例 2 を使用して、署名付き証明書ファイル *sybase_srv5.crt* の最後に *srv5_pkey.txt* をカット・アンド・ペーストします。
- サーバが起動時にロードできる信頼されたルート・ファイルを作成するには、ファイル名 *trusted.txt* を *sybase_srv5.txt* に変更します。*sybase_srv5.txt* はサーバの共通名です。
- 次に、*sybase_srv5.txt* ファイルを Adaptive Server インストール・ディレクトリにコピーします。たとえば、*\$SYBASE/\$SYBASE_ASE/certificates* にコピーします。
- `-s` オプションと `-T` オプションをともに使用して、証明書の時間範囲を指定します。

このファイルは、SSL ベースのセッションに必要であり、SSL が有効な Adaptive Server を起動するときに使用されます。

CA のルート証明書を作成すると、複数のサーバ証明書の署名に使用できます。

参照

ユーティリティ [certpk12](#), [certreq](#)

certpk12

説明	<p>PKCS #12 ファイルを証明書ファイルとプライベート・キーにエクスポートまたはインポートします。\$SYBASE/\$SYBASE_OCS/bin にあります。</p> <p>(Windows の場合) このユーティリティは certpk12.exe で、%SYBASE%\\$SYBASE_OCS%\bin にあります。</p>
構文	<pre>certpk12 {-O Pkcs12_file -I Pkcs12_file} [-C Cert_file] [-K Key_file] [-P key_password] [-E Pkcs12_password]</pre> <p>または</p> <pre>certpk12 -v</pre>
パラメータ	<p>-O Pkcs12_file エクスポートする PKCS#12 ファイルの名前を指定します。ファイルの内容は、証明書とプライベート・キー、証明書だけ、プライベート・キーだけの3つの場合があります。-O または -I をオンにしてください。</p> <p>-I Pkcs12_file インポートする PKCS#12 ファイルの名前を指定します。ファイルの内容は、証明書とプライベート・キー、証明書だけ、プライベート・キーだけの3つの場合があります。-I または -O をオンにしてください。</p> <p>-C Cert_file -O をオンにしている場合は、PKCS #12 ファイルにエクスポートする証明書ファイルの名前を指定します。-I をオンにしている場合は、PKCS #12 ファイルからインポートする証明書ファイルの名前を指定します。</p> <p>-K Key_file -O をオンにしている場合は、PKCS #12 ファイルにエクスポートするプライベート・キー・ファイルの名前を指定します。-I をオンにしている場合は、PKCS #12 ファイルからインポートするプライベート・キー・ファイルの名前を指定します。</p>

-P Key_password

-K で指定したプライベート・キーを保護するために使用するパスワードを指定します。-O をオンにしている場合は、プライベート・キーを PKCS #12 ファイルにエクスポートするときにパスワードが要求されます。-I をオンにしている場合は、PKCS #12 ファイルからインポートしたプライベート・キーをテキスト・ファイルに出力するときにパスワードが要求されます。

-E Pkcs12_password

PKCS #12 ファイルを保護するために使用するパスワードを指定します。-O をオンにしている場合は、エクスポートする PKCS #12 ファイルを暗号化するときにパスワードを使用します。-I をオンにしている場合は、インポートする PKCS #12 ファイルを復号化するときにパスワードを使用します。パスワードは「トランスポート・パスワード」とも呼ばれます。

-v

certpk12 ツールのバージョン番号と著作権メッセージを表示して、終了します。

例

例 1 証明書ファイル (*caRSA.crt*) とプライベート・キー・ファイル (*caRSAPkey.txt*) を PKCS#12 ファイル (*caRSA.p12*) にエクスポートします。*password* は *caRSAPkey.txt* の復号化に使用されるパスワードです。*pk12password* は最後の *caRSA.p12* の暗号化に使用されるパスワードです。

```
certpk12 -O caRSA.p12 -C caRSA.crt -K caRSAPkey.txt
-P password -E pk12password
```

```
-- Sybase PKCS#12 Conversion Utility certpk12 Thu Nov 9 16:55:51 2000--
```

例 2 証明書とプライベート・キーが収められている PKCS #12 ファイル (*caRSA.p12*) をインポートします。埋め込み証明書をテキスト・ファイル (*caRSA_new.crt*) に出力し、埋め込みプライベート・キーをテキスト・ファイル (*caRSAPkey_new.txt*) に出力します。

```
certpk12 -I caRSA.p12 -C caRSA_new.crt -K caRSAPkey_new.txt
-P new_password -E pk12password
```

```
-- Sybase PKCS#12 Conversion Utility certpk12 Thu Nov 9 16:55:51 2000--
```

new_password は、*caRSAPkey_new.txt* を保護するために使用されます。*pk12password* は、*caRSA.p12* ファイルを複合化するために必要です。

注意 例 1 と例 2 を実行すると、*caRSA.crt* と *caRSA_new.crt* は同じ内容になりますが、*caRSAPkey.txt* と *caRSAPkey_new.txt* はランダムに暗号化されるので、同じにはなりません。

例 3 証明書ファイル (*caRSA.crt*) を PKCS#12 ファイル (*caRSACert.p12*) にエクスポートします。*pkcs12password* を使用して *caRSACert.p12* を暗号化します。

```
certpk12 -O caRSACert.p12 -C caRSA.crt -E pk12password
-- Sybase PKCS#12 Conversion Utility certpk12 Thu Nov 9 16:55:51 2000--
```

例 4 証明書が収められた PKCS#12 ファイル (*caRSACert.p12*) をインポートします。埋め込み証明書をテキスト・ファイル (*caRSACert.txt*) に出力します。

```
certpk12 -I caRSACert.p12 -C caRSACert.txt -E pk12password
-- Sybase PKCS#12 Conversion Utility certpk12 Thu Nov 9 16:55:51 2000--
```

pk12password は、*caRSACert.p12* ファイルを復号化するために必要です。

注意 例 3 と例 4 を実行すると、*caRSA.crt* と *caRSACert.txt* は同じ内容になります。

使用法

- Adaptive Server では openssl オープン・ソース・ユーティリティが `$$SYBASE/$$SYBASE_OCS/bin` (Windows では `%SYBASE%\$SYBASE_OCS%\bin`) に含まれています。certreq、certauth、certpk12 で実装されたすべての証明書管理タスクを実行するには openssl を使用します。Sybase では便宜上このバイナリを組み込んでいますが、バイナリを使用して発生した問題についてはいっさい責任を負いません。詳細については、OpenSSL Web site (<http://www.openssl.org>) を参照してください。
- certpk12 がサポートしているのは、トリプル DES 暗号化方式により暗号化された PKCS #12 ファイルだけです。
- certpk12 を実行するには、ダイナミック・ライブラリ検索パス内で `$$SYBASE/$$SYBASE_OCS/libp364` のエントリの前に `$$SYBASE/$$SYBASE_OCS/lib3p` のエントリを配置する必要があります。
- 証明書要求者のプライベート・キーを署名付き証明書ファイルの最後に付加します。
- *servername.crt* ファイルに名前を付けます。*servername* はサーバの名前です。このファイルを、`$$SYBASE/$$SYBASE_ASE` (Windows では `%SYBASE%\$SYBASE_ASE%`) にある証明書ディレクトリに置きます。

このファイルは、SSL が有効な Adaptive Server を起動するときに必要です。

参照

ユーティリティ `certauth`、`certreq`

certreq

説明

サーバ証明書要求と対応するプライベート・キーを作成します。このユーティリティは対話型モードで使用できます。また、コマンド・ラインにオプションのパラメータをすべて提供できます。
`$$SYBASE/$$SYBASE_OCS/bin` にあります。

(Windows の場合) このユーティリティは `certreq.exe` で、`%SYBASE%\$SYBASE_OCS%\bin` にあります。

構文

```
certreq
    [-F input_file]
    [-R request_filename]
    [-K PK_filename]
    [-P password]
```

または

```
certreq -v
```

パラメータ

-F *input_file*

属性情報が含まれたファイル名を指定して、証明書要求を作成します。*input_file* 名を指定しない場合は、必要な情報をユーザが対話形式で入力します。

input_file には、次のエントリが必要です。

```
req_certtype={Server,Client}
req_keytype={RSA,DSA}
req_keylength={for RSA:512-4096;
               for DSA:512,768,1024}
req_country={string}
req_state={string}
req_locality={string}
req_organization={string}
req_orgunit={string}
req_commonname={string}
```

注意 共通名はサーバ名と同じにしてください。

サンプル・ファイルの *input_file* については、「例」の項を参照してください。

-R *request_filename*

証明書要求ファイルの名前を指定します。

-K *PK_filename*

プライベート・キー・ファイルの名前を指定します。

-P password

プライベート・キーを保護するために使用されるパスワードを指定します。

-v

バージョン番号と著作権メッセージを表示して、終了します。

例

例 1 この例では、**-F input_file** パラメータを使用していません。したがって、対話型モードになります。サーバ証明書要求 (*server_req.txt*) とプライベート・キー (*server_pkey.txt*) を作成するには、次のように入力します。

```
certreq
Choose certificate request type:
  S - Server certificate request
  C - Client certificate request (not supported)
  Q - Quit
Enter your request [Q] :s

Choose key type:

  R - RSA key pair
  D - DSA/DHE key pair
  Q - Quit

Enter your request [Q] :R

Enter key length (512, 768, 1024 for DSA; 512-2048 for
RSA) :512

Country:US

State:california

Locality:dublin

Organization:sybase

Organizational Unit:dst

Common Name:server
```

ユーティリティから次のメッセージが返されます。

```
Generating key pair (please wait) . . .
```

キーのペアが生成された後、さらに情報を入力するためのプロンプトが **certreq** ユーティリティから表示されます。

```
Enter password for private key :password
```

```
Enter file path to save request:server_req.txt
```

```
Enter file path to save private key :server_pkey.txt
```

例 2 このテキスト・ファイルの例では、`tag=value` フォーマットが非対話型モードでの証明書要求の入力に使用されます。非対話型モードには `-F` オプションを使用できます。`-F` オプションを使用する場合は、有効な値と上記のフォーマットを使用してください。これらに誤りがある場合、証明書は正しく作成されません。

```
certreq -F input_file
req_certtype=server
req_keytype=RSA
req_keylength=512
req_country=us
req_state=california
req_locality=dublin
req_organization=sybase
req_orgunit=dst
req_commonname=server
```

このファイルを作成し、保存したら、次のようにコマンド・ラインに入力します。`path_and_file` は、テキスト・ファイルのロケーションです。

```
certreq -F path_and_file -R server_req.txt -K server_pkey.txt -P password
```

このファイルによって、サーバ証明書要求 (`server_req.txt`) とそのプライベート・キー (`server_pkey.txt`) が作成されます。プライベート・キーは、`password` によって保護されます。

サーバ証明書ファイルは、標準的な ASCII テキスト・エディタを使用して編集できます。

使用法

- Adaptive Server では `openssl` オープン・ソース・ユーティリティが `$SYBASE/$SYBASE_OCS/bin` (Windows では `%SYBASE%\$SYBASE_OCS%\bin`) に含まれています。`certreq`、`certauth`、`certpk12` で実装されたすべての証明書管理タスクを実行するには `openssl` を使用します。Sybase では便宜上このバイナリを組み込んでいますが、バイナリを使用して発生した問題についてはいっさい責任を負いません。詳細については、[OpenSSL Web site \(http://www.openssl.org\)](http://www.openssl.org) を参照してください。
- 入力ファイルでは、`tag=value` のフォーマットを使用します。`tag` では大文字と小文字が区別されるので、上記のとおりに入力してください。

- `certreq` を実行するには、ダイナミック・ライブラリ検索パス内で `$$SYBASE/$$SYBASE_OCS/libp364` のエントリの前に `$$SYBASE/$$SYBASE_OCS/lib3p` のエントリを配置する必要があります。
- "=" は必須です。有効な *value* は、文字または数字で開始し、単一のワードにしてください。 *value* の中には、スペースを入れないでください。
- *value* は、`req_certtype`、`req_keytype`、`req_keylength`、`req_commonname` では必須です。
- `<tag>`、`=`、*value* の前後のスペースまたはタブは許容されます。ブランク行も許容されます。
- 各コメント行は、`#` で始めてください。
- 証明書要求ファイルは、PKCS #10 フォーマットになっています。この証明書要求ファイルは、`certauth` ツールが要求を CA 署名付き証明書に変換するときに受け入れ可能な入力として使用されます。

参照

ユーティリティ [certauth](#)、[certpk12](#)

charset

説明

(UNIX のみ) 文字セット・ファイルとソート順ファイルを Adaptive Server にロードします。\$SYBASE/\$SYBASE_ASE/bin にあります。

構文

```
charset  
    [-Ppassword]  
    [-Sserver]  
    [-linterface]  
    sort_order  
    [ charset ]
```

または

```
charset -v
```

パラメータ

-P password

パスワードを指定します。**-P** を指定しない場合、**charset** はパスワードの入力を要求するプロンプトを表示します。

-S server

文字セットおよびソート順を変更するサーバの名前を指定します。

-l interface

サーバが使用するネットワーク・インタフェースを指定します。

sort_order

Adaptive Server で使用するソート順ファイルの名前を指定します。

charset

Adaptive Server で使用する文字セットを指定します。

-v

charset. のバージョン番号と著作権メッセージを表示します。

使用法

- 現在のリリースのディレクトリを指すように SYBASE 環境変数を設定してから、**charset** を使用してください。

パーミッション

charset を使用できるのは、システム管理者のみです。

参照

コマンド set

ユーティリティ [langinstall](#)

cobpre

説明

COBOL のプリコンパイラです。\$SYBASE/\$SYBASE_OCS/bin (Windows では %SYBASE%\%SYBASE_OCS%\bin) にあります。cpre の詳細については、『Open Client/Server プログラマーズ・ガイド補足』の付録 A を参照してください。

構文

上記を参照してください。

cpre

説明

C のプリコンパイラです。\$SYBASE/\$SYBASE_OCS/bin (Windows では %SYBASE%\%SYBASE_OCS%\bin) にあります。cpre の詳細については、『Open Client/Server プログラマーズ・ガイド補足』の付録 A を参照してください。

構文

上記を参照してください。

dataserver

説明 (UNIX のみ) 実行形式の Adaptive Server プログラムです。
\$SYBASE/\$SYBASE_ASE/bin にあります。

構文

```
dataserver [-f] [-g] [-G] [-h] [-H] [-m] [-q] [-v] [-X]
           [-A system_role]
           [-a path_to_CAPs_directive_file]
           [-b master_device_size [k | K | m | M | g | G | t | T]]
           [-c config_file_for_server]
           [-d device_name]
           [-e path_to_error_log]
           [-i interfaces_file_directory]
           [-K keytab_file]
           [-L config_file_name_for_connectivity]
           [-M shared_memory_repository_directory]
           [-N licinstan]
           [-n system_privileges]
           [-p sa_login_name]
           [-r mirror_disk_name]
           [-s server_name]
           [-T trace_flag]
           [-u sa/sso_name]
           [-w master | model database]
           [-y [password] ]
           [-z page_size [ k | K ] ]
```

Cluster Edition の構文 :

```
dataserver
-u, --admin-name=sa/sso_name
   --buildquorum=[force]
-a, --caps-file=filename
-F, --cluster-input=filename
   --cluster-takeover
-L, --conn-config-file=[filename]
   --create-cluster-id [=quorum]
-D, --default-db-size=size_spec
-e, --error-log=[filename]
-G, --event-log-server=logserv_name
-f, --forcebuild
-H, --ha-server
-h, --help=[{0|1|2|3}[, display_width]]
   --instance=instance_name
-y, --key-password=[key_password]
-K, --keytab-file=filename
-N, --license-prop-file=filename
-z, --logical-page-size=page_size
-Z, --master-db-size=size_spec
-d, --master-dev=master_device_name
-b, --master-dev-size=[size_spec]
```

```

--master_key_password [=password]
-r, --master-mirror=filename
-m, --masterrecover
-g, --no-event-logging
-n, --permission-login=system_privilege
-Q, --quorum-dev=quorum_dev
-q, --recover-quiet
-w, --rewrite-db=database_name
-A, --role-logins=system_role
-p, --sa-name={SSO_login_account | sso_role | sa_role}
-k, --server-principal=s_principal
-M, --shared-mem-dir=directory_name
-X, --sybmon
-T, --trace=trace_flag
-v, --version

```

または

```
dataserver -v
```

パラメータ

-A system_role

enable granular permissions が 0 に設定されており、すべてのユーザが Adaptive Server にログインできない場合、sso_role が付いたログイン・アカウントをサーバ管理者に提供します。

-a path_to_CAPs_directive_file

CAP のディレクティブ・ファイルへのパスを指定します。

-b master_device_size [k|K|m|M|g|G|t|T]

構築するマスタ・デバイスまたはデータベースのサイズを指定します。サイズの計算はサーバによって行われるため、バイト数を 1 の位まで指定することなく “K”、“M”、“G”、“T” を使用できます。

-c config_file_for_server

Adaptive Server 設定ファイルのフル・パス名を指定します。このパラメータを使用すると、指定した設定ファイルの設定値で Adaptive Server が起動します。dataserver -c パラメータで設定ファイルを指定する場合、この設定ファイル内のすべてのパラメータに互換性があることを確認してからサーバを起動してください。設定パラメータの中に互換性のないものがあると、サーバが起動しないことがあります。これを回避するには、マスタ・デバイスを構築するときに設定ファイルを指定しないでください。設定ファイルを指定しない場合は、構築フェーズですべてのデフォルト設定が使用されます。

詳細については、『システム管理ガイド』を参照してください。

-d device_name

master データベースのデバイスのフル・パス名です。**master** データベースのデバイスは、**Adaptive Server** を起動するユーザが書き込み可能である必要があります。**-d** パラメータを使用しない場合、デフォルトの **master** データベース・デバイス名は **d_master** になります。

-e errorlogfile

Adaptive Server のシステム・レベルのエラー・メッセージに使用するエラー・ログ・ファイルのフル・パス名です。

-f

デバイスまたはデータベースを強制的に初期化します。**-f** は、**-b** と **-w** のいずれか、またはその両方とともに使用した場合にのみ有効です。**-b** または **-w** を使用しないで **-f** だけ使用しても、サーバは起動しません。**-w** の有無によって、**-f** によるサーバの動作が異なります。詳細については、「[-f オプションと -ww オプションをともに使用する場合の潜在的な問題](#)」(181 ページ)と「[-b オプションと -w オプションの依存性と条件](#)」(181 ページ)を参照してください。

-G logserv_name

イベント・ログ・サーバの名前を指定します。

-g

イベント・ロギングを無効にします。

-H

Adaptive Server に HA (高可用性) 機能をインストールしてある場合に、HA サーバを起動します。

-h

ヘルプ・メッセージを表示します。

-i interfaces_file_directory

Adaptive Server に接続するときに検索する **interfaces** ファイルがあるディレクトリのロケーションを指定します。**-i** を省略すると、**dataserver** は **SYBASE** 環境変数によって指定されたディレクトリにある **interfaces** という名前のファイルを探します。

-K keytab_file

DCE での認証に使用する **keytab** ファイルのパスを指定します。

-k, --server-principal=s_principal

サーバ・プリンシパル名を指定します。

-L config_file_name_for_connectivity

コネクティビティ用の設定ファイルの名前を指定します。

-M *sharedmem_directory*

共有メモリのファイルを、デフォルト・ロケーションの *\$SYBASE* ではなく、指定したディレクトリ内に置きます。

sharedmem_directory が “*¥*” で始まる場合、そのディレクトリ名は絶対パス名として解釈されます。それ以外の場合は、ディレクトリ名は *\$SYBASE* の相対パスとして解釈されます。

--master_key_password [=password]

コマンド・ラインで *password* を指定する場合や、Adaptive Server の起動時にマスタ・キーのパスワードの入力を要求するプロンプトが表示される場合は、マスタ・キーのパスワードを指定します。パスワード文字は表示されず、パスワードは Adaptive Server の起動順序の後半まで検証されません。

コマンド・ラインでパスワードを指定すると、メモリが読み込まれて使用されるまでパスワードが表示されます。

-m

Adaptive Server をシングル・ユーザ・モードで起動します。

-N *licinstant*

ライセンス・キャッシュ・ファイルのディレクトリのデフォルト以外のロケーションを指定します。デフォルトのロケーションは、*\$SYBASE/\$SYBASE_ASE/sysam/server_name.properties* です。

-n *system_privileges*

enable granular permissions が 1 に設定されており、すべてのユーザが Adaptive Server にログインできない場合、*change password* 権限が付いたログイン・アカウントをサーバ管理者に提供します。

-p *sso_login_name*

アカウントの新しいパスワードを取得するために、Adaptive Server を起動するときのシステム・セキュリティ担当者のログイン名を指定します。Adaptive Server は、ランダムなパスワードを生成して表示し、暗号化してから、そのアカウントの新しいパスワードとして *master.syslogins* に保存します。

-q

クワイス状態のデータベースを「リカバリ中」として扱います。

-r *mastermirror*

マスタ・デバイスのミラーリングを開始します。マスタ・デバイスが破損している場合には、このパラメータを使用して Adaptive Server を起動します。

-s *servername*

起動する Adaptive Server の名前を指定します。**-s** を省略すると、SYBASE という名前のサーバが起動します。

-T *trace_flag***-u *sa/sso_name***

ロックを解除するシステム管理者またはシステム・セキュリティ担当者の名前を指定します。

-v

dataserver のバージョン番号と著作権メッセージを表示して、終了します。

-w [*master* | *model*]

master データベースと **model** データベースのどちらを書き込むかを指定します。

-X

このサーバを、**dataserver** ではなく **sybmon** として起動します。

-y [*password*]

ユーザは、暗号化されたプライベート・キーのパスワードを割り当てられます。サーバは、パスワードを入力するためのプロンプトを表示します。このパスワードはプライベート・キーの作成時にプライベート・キーを暗号化するために使用したパスワードと一致する必要があります。バックグラウンドでサーバを実行している場合、このパラメータは使用できません。

注意 セキュリティ上の理由から、**-y** を使用してパスワードを設定することはおおすすめしません。

プライベート・キーはサーバのデジタル証明書に含まれています。デフォルトでは、証明書ファイルは `/usr/local/sybase/certificates/<servername>.crt` にあります。

sp_ssladmin addcert コマンドを呼び出すと、証明書ファイルのセッションが変更されます。

-Z [*size_spec*]

master データベースの初期サイズを指定します。

-z *page_size* [*k* | *K*]

サーバのページ・サイズを指定します。このフラグを使用する場合、**-b** と **-w** も使用し、2K ~ 16K の間の 2 の偶数乗を指定してください。それ以外の場合は、サーバが起動しません。

例 **例 1** 100MB のマスタ・デバイスと 4K ページを持つ新規インストールを作成します。

```
dataserver -d my_master_device -z 4k -b 100.02M
```

オプションとそれに続く引数の間にスペースを入れてもかまいません。この例では、サーバの設定領域に 16K のオーバーヘッドが必要であるため、100MB のマスタ・デバイスに “100.02M” を指定しています。

例 2 この例では、破壊された model データベースの再書き込みを行います。

```
dataserver -d d_master -w model -s server_name
```

例 3 “aseprincipal@MYREALM.COM” プリンシパル名を指定します。

```
$$SYBASE/$$SYBASE_ASE/bin/dataserver -dmaster.dat  
-s secure_ase -k aseprincipal@MYREALM.COM
```

例 4 この例では、破壊された master データベースの再書き込みを行い、デバイス・サイズを指定します。

```
dataserver -d my_master_device -w master -z 4k
```

例 5 この例では、破壊された master データベースの再書き込みを行い、デバイス・サイズとページ・サイズを指定し、サーバが設定ブロックの値よりもこの値を優先して受け入れるように指定します。

```
dataserver -d my_master_device -w master -z 4k -b  
100.02M -f
```

例 6 この例では、破壊された master データベースの再書き込みを行い、サーバが設定ブロックで検出したサイズとは異なるページ・サイズを指定します。これによってエラーが発生します。

```
dataserver -d my_master_device -w master -z 8k  
  
00:00000:00000:2001/01/19 12:01:26.94 server The  
configured server page size does not match that  
specified on the command line.To use the configured  
size, omit the command line size; to use the command  
line size, specify 'force' (-f).
```

例 7 この例では、破壊された master データベースの再書き込みを行い、通常の再起動でも、正しくないページ・サイズを指定します。これによってエラーが発生します。

```
dataserver -d my_master_device -z4000  
  
dataserver:the 'z' flag may not be used without 'b' or  
'w'. dataserver:server will ignore the 'z' flag.
```

```

dataserver:the 'z' flag contained an invalid page size.
dataserver:the page size must be an even power of two
between 2048 and 16384 bytes, inclusive.

```

例 8 マスタ・キーのパスワードの入力を要求するプロンプトを表示します。

```
dataserver --master_key_passwd -dd_master -eerrorlog
```

例 9 sso_role 役割を持つアカウント名をリストします。

```

$SYBASE/$SYBASE_ASE/bin/dataserver
-d master.dat
-s server_name
-A sso_role

```

例 10 sso_role 役割を持つアカウント名をリストします。

```

$SYBASE/$SYBASE_ASE/bin/dataserver
-d master.dat
-s server_name
--role-logins sso_role

```

例 11 change password 権限を持つアカウント名をリストします。

```

$SYBASE/$SYBASE_ASE/bin/dataserver
-d master.dat
-s server_name
-n "change password"

```

例 12 change password 権限を持つアカウント名をリストします。

```

$SYBASE/$SYBASE_ASE/bin/dataserver
-d master.dat
-s server_name
--permission-logins "change password"

```

使用法

- `dataserver` を使用すると、最大 32GB のサイズのデバイスとデータベースを作成できます。サイズは、使用しているオペレーティング・システムの制限によって異なります。サイズ制限の詳細については、使用しているプラットフォームの『インストール・ガイド』を参照してください。

- Adaptive Server を起動するには、**dataserver** プログラムを直接実行するのではなく、**startserver** コマンドを使用します。デフォルト値を変更する必要がある場合は、Sybase インストール・ディレクトリにある **RUN_servername** ファイルを編集します。詳細については、**startserver** のページを参照してください。
- Adaptive Server のパスワードは暗号化されるため、パスワードを忘れた場合には回復することができません。システム・セキュリティ担当者全員が各自のパスワードを忘れてしまった場合は、**-p** パラメータを使用してシステム・セキュリティ担当者のアカウント用に新しいパスワードを生成できます。**-p** を指定して Adaptive Server を起動し、ランダムに生成された新しいパスワードで Adaptive Server にすぐにログインしてから、**sp_password** を実行してパスワードをより安全なものに設定し直してください。
- **-w** パラメータを使用する場合、**dataserver** は **iso_1** 文字セットではなく **ascii-8** 文字セットを使用します。**master** に **iso_8** 文字セットが必要な場合は、**master** データベースのダンプをロードするか、**sqlloc** を使用して文字セットを変更できます (**sqlloc** には **sysystemprocs** データベースが必要です)。
- Adaptive Server インストール・プログラムを実行した後、**dataserver** 実行プログラムにファイル・パーミッションを設定して、実行できるユーザを制限してください。
- Adaptive Server 名に **-s** パラメータを指定せず、**DSLISTEN** 環境変数も設定していない場合には、**dataserver** は、デフォルトの Adaptive Server 名である **SYBASE** を使用します。**DSLISTEN** 環境変数の値によって、このデフォルト値は無効になります。**-s** パラメータを指定すると、デフォルト値と **DSLISTEN** 環境変数の両方が無効になります。
- 自動ログイン・ロックアウトを使用すると、サイトではログインをロック解除できるすべてのアカウント (システム管理者とシステム・セキュリティ担当者) がロックされた状態になることがあります。この状況が発生した場合は、**-u** パラメータを指定した **dataserver** ユーティリティを使用して、システム管理者権限またはシステム・セキュリティ担当者権限に指定したログインを確認し、アカウントのロックを解除して、失敗したログイン・カウンタの現在の値を 0 にリセットします。

-f オプションと -ww オプションをともに使用する場合の潜在的な問題

-f オプションと -w オプションをともに使用する場合は、特に注意が必要です。-w オプションを使用して **master** データベースの再書き込みを行う場合、サーバは、設定ブロックのページ・サイズとデバイス・サイズが正確であることを要求します。これらをコマンド・ラインで指定しない場合には、設定ブロックのサイズに一致している必要があります。サーバはマスタ・デバイスを再調整し、**master** データベースと含まれているその他すべてのデータベースを適切な場所に戻します。

-f オプションの強制初期化を使用する場合、指定するページ・サイズとマスタ・デバイス・サイズは、設定ブロック内の値を無効にします。また、-f はその他の不明なすべての領域 (未使用または破損した割り付けブロック) を **master** データベースに割り当てます。

-b オプションと -w オプションの依存性と条件

-b の影響は、-w の有無によって次のように異なります。

- -w なしで -b を使用すると、作成される新しいマスタ・デバイスは、-d で指定した名前 (デフォルトは `d_master`) と、-z で指定したページ・サイズ (デフォルトは `2048`) になります。
- 指定したデバイスが OS ファイルとしてすでに存在する場合、新しいデバイスの作成は失敗し、次のようなメッセージが表示されます。

```
File already exists.You must remove the existing
file before attempting to create a new one using
the server's -b option.
Unable to create master device.
```

- 指定したデバイスが既存のロー・パーティションである場合、-f フラグが指定されていないと失敗します。このフラグにより、サーバのマスタ・デバイスとしてそのロー・パーティションが再初期化されます。
 - -b を使用して -w **master** も使用すると、**dataserver** に、マスタ・デバイスの再作成時には -z で指定したサイズを使用するよう指示されます。新しいデバイスの作成については何も示されません。
- w では、追加フラグが必要な場合と必要でない場合があります。
- -w **model** を使用する場合、-z フラグと -b フラグは許容されますが無視されます。
 - -w **master** を「新しい」インストールに使用する場合、デバイス・サイズ情報が `config_block` に格納されているため、-z と -b は必要ありません。

- `-w master` を使用して以前のインストールを「アップグレード」する場合、次のようになります。
- 関連するサイズの有効なエントリが `config_block` がない場合、サーバは `-b` か `-z`、またはその両方を要求します。ページ・サイズまたはデバイス・サイズの有効なデータを得られない場合、コマンドは失敗します。
- ユーザが `-b` か `-z` またはその両方を指定できるのは、`config_block` にそれらのパラメータが示すサイズの有効なエントリがある場合です。ただし、サイズが `config_block` にあるサイズと一致しない場合は、`-f` を追加して新しい優先サイズを強制的に適用してください。

パーミッション

このバイナリの実行パーミッションを持つユーザと、すべてのファイルに対する読み込み／書き込みアクセス権を持つユーザが使用できます。

参照

コマンド `disk mirror`、`disk remirror`、`disk unmirror`

システム・プロシージャ `sp_ssladmin addcert`

ユーティリティ [startserver](#)

ddlgen

説明

Adaptive Server でサーバ・レベルおよびデータベース・レベルのオブジェクトの定義を生成する Java ベースのツール。

以下の DDL を生成するためには、`sa_role` または `sso_role` を持っている必要があります。

- 暗号化キー
- ログイン
- 役割

その他のすべてのオブジェクトについては、DDL を生成するための特定のパーミッションまたは役割は必要ありません。

ddlgen のコマンド・ライン版は `ASE-15_0/bin` (UNIX)、`ASE-15_0\bin` (Windows) にあります。

構文

```
ddlgen
    -Ulogin
    -Ppassword
    -S[[ssl:]server | host_name :port_number]
    [-I interfaces_file]
    [-Tobject_type]
    [-Nobject_name]
    [-Ddbname]
    [-Xextended_object_type]
    [-Ooutput_file]
    [-Error_file]
    [-Lprogress_log_file]
    [-Jclient_charset]
    [-LC -N logical_cluster_name]
    -F [% | SGM | GRP | USR | R | D | UDD | U | V |
        P | XP | I | RI | KC | TR | PC ]
```

または

```
ddlgen -v
```

パラメータ

`-U login`

ログイン名を指定します。大文字と小文字を区別します。

`-P password`

パスワードを指定します。

ddlgen 文に `-P` パラメータを含めないと、ddlgen はパスワードを指定するように要求します。

パスワードを隠す方法については、「[ddlgen でのパスワードを隠す](#)」(203 ページ) を参照してください。

-S `[[ssl:]server | host_name :port_number]`

Adaptive Server の名前を指定します。ddlgen は、interfaces ファイルまたは LDAP 設定ファイルでこの名前を探します。次のオプションがあります。

- `[ssl:]` – SSL 対応サーバのオブジェクトの DDL を生成できます。このパラメータはオプションです。
- `-S [host_name:port_number]` – ddlgen は指定された `host_name` と `port_number` を使用します。interfaces ファイルと LDAP 設定ファイルの読み込みや解析は行いません。
- `-S[server]-l` – ddlgen はユーザ・ロケーションにある指定の interfaces ファイルを解析してサーバ名を探します (詳細については、`-l` パラメータの説明を参照してください)。
- `-S[server]` – interfaces ファイルを指定しない場合、ddlgen は次のように動作します。
 - a ddlgen は、まず標準ロケーションから LDAP 設定ファイルの読み取りを試行します。
 - b LDAP ファイルが存在しないか、存在していても Adaptive Server エントリがない場合は、標準ロケーションにある interfaces ファイルを解析してサーバ名を探します。
 - c LDAP ファイルが存在する場合は、ddlgen はそのファイルを使用してサーバ名を検索します。interfaces ファイルは解析されず、LDAP 設定ファイルが解析されます。

注意 `-S` オプションを使用するのは、ddlgen ではデフォルト・サーバに接続されないためです。

-l

interfaces ファイル名を指定します。UNIX では `$$SYBASE/interfaces`、Windows では `%SYBASE%\$ini %sql.ini` です。このオプションのパラメータは、`-S` とともに使用します。

-Object_type

作成するオブジェクトのタイプを指定します。-T を使用しない場合、**ddlgen** はログインのデフォルト・データベースの DDL を生成します。-T のオブジェクト・タイプは次のとおりです。

オブジェクト・タイプ	説明
C	キャッシュ
D	デフォルト
DB	データベース
DBD	データベース・デバイス
DPD	ダンプ・デバイス
EC	実行クラス
EG	エンジン・グループ
EK	暗号化キー
GRP	グループ
I	インデックス
KC	キー制約
L	ログイン
LK	論理キー
P	ストアド・プロシージャ
R	ルール
RI	参照整合性
RO	役割
RS	リモート・サーバ
SGM	セグメント
TR	トリガ
U	テーブル
UDD	ユーザ定義データ型
USR	ユーザ
V	ビュー
WS	ユーザ定義 Web サービス
WSC	Web サービス・コンシューマ
XOD	ローカル・キャッシュ
XOU	グローバル・キャッシュ
XP	拡張ストアド・プロシージャ

-Nobject_name

-Ndb_name.owner_name.table_name.object_name など、作成するオブジェクトの完全に修飾された名前を指定します。-N オプションには、次のような特徴があります。

- -T パラメータで DB (データベース) 以外の *object_type* を指定する場合に必須。
- % を使用するワイルドカードを受け入れる。
- -Ndb_name.table_owner.table_name.trigger_name の形式を使用して、テーブルのトリガの DDL を生成する。
テーブルのすべてのトリガを生成するには、*trigger_name* を % に置き換えて、-Ndb_name.table_owner.table_name.% の形式を使用します。
- -Ndb_name.owner.key_name の形式で、暗号化キーの DDL を生成する。
- サーバ上にある特定のオブジェクト・タイプの全項目の DDL を生成する。
- -Ndb_name.owner_name.table_name.object_name の形式で、名前を解析する順序を厳密に指定する。引数を 3 つだけ指定すると、ddlgen は、最初の引数から順に *owner_name*、*table_name*、*object_name* であると判断します。または、-Nowner_name.table_name -Ddb_name を使用することもできます。*object_name* がインデックス (I) の場合、ddlgen はこの制限を適用しません。

-Ddbname

-N オプションで指定するオブジェクトのデータベース名を指定します。デフォルトはユーザのデフォルト・データベースです。

テーブルのすべてのトリガの DDL を生成するときは、-D パラメータを使用できません。

-Xextended_object_type

次の区別を行います。

- テーブルをオブジェクト・タイプ (-TU) として指定した場合に、ユーザ・テーブル (OU) をプロキシ・テーブル (OD) から区別します。
- データベースをオブジェクト・タイプ (-TDB) として指定した場合に、テンポラリ・データベース (OD) を通常のデータベース (OU) またはアーカイブ・データベース (OA) から区別します。
- プロシージャをオブジェクト・タイプ (-TP) として指定した場合に、SQLJ プロシージャ (OD) をストアド・プロシージャ (OU) から区別します。

object_type (-T) が U (テーブル) で、-X が指定されていない場合、*ddlgen* はユーザ・テーブルとプロキシ・テーブルの両方の DDL を生成します。一方の DDL だけを生成するには、次のようにします。

- ユーザ・テーブル — -X オプションで OU 拡張オブジェクト・タイプを使用します。
- プロキシ・テーブル — -X オプションで OD 拡張オブジェクト・タイプを指定します。
- インメモリ・データベース、キャッシュ、デバイス — -X オプションで OI 拡張オブジェクト・タイプを使用します。
- テンポラリ・インメモリ・データベース — -X オプションで OIT 拡張オブジェクト・タイプを使用します。

注意 *ddlgen* は、システム・テーブルのスキーマの生成をサポートしていません。

-Ooutput_file

生成された DDL の出力ファイルを指定します。-O を指定しない場合、作成した DDL はコンソール・ウィンドウに表示されます。

-Error_file

エラーを記録するログ・ファイルを指定します。-E を指定しない場合、生成されたエラーはコンソール・ウィンドウに表示されます。

-Lprogress_log_file

ddlgen の進行状況を記録するログ・ファイルを指定します。-L を指定しない場合、進行状況は記録されません。

-Jclient_charset

クライアントで使用する文字セットを指定します。**-Jclient_charset** は、クライアントで使用する文字セットである *client_charset* とサーバの文字セット間の変換を Adaptive Server に要求します。フィルタによって、*client_charset* と Adaptive Server の文字セット間で入力に変換されます。

-J の指定を省略すると、文字セットはプラットフォームのデフォルトに設定されます。デフォルトの文字セットは、クライアントが使用する文字セットと同じであるとはかぎりません。

注意 HP プラットフォームの場合 — **-Jiso_1** を使用して適切な文字セットを指定してください。

-LC

サーバ上の 1 つまたはすべての論理クラスタの DDL を生成します。

-F

テーブル・レベル・オブジェクトの DDL のテーブル定義と、データベース・レベル・オブジェクトの DDL のデータベース定義から、インデックス、トリガ、制約をフィルタします。有効なフィルタは次のとおりです。

- テーブルの場合 - [% | I | RI | KC | TR | PC]
- データベースの場合 - [% | SGM | GRP | USR | R | D | UDD | U | V | P | XP | I | RI | KC | TR]

フィルタのオプションは次のとおりです。

オプション	フィルタ対象
%	すべて。データベースまたはテーブルのスキーマのみの定義を取り出す。
SGM	セグメント
GRP	グループ
USR	ユーザ
R	ルール
D	デフォルト
UDD	ユーザ定義データ型
U	ユーザ・テーブル
V	ビュー
P	ストアド・プロシージャ
PC	パーティション条件
XP	拡張ストアド・プロシージャ
I	インデックス
RI	参照整合性制約
KC	主キー制約とユニーク・キー制約
TR	トリガ

無効なフィルタ・パラメータを使用した場合、`ddlgen` は警告を生成してそのパラメータを無視し、指定したパラメータのうち有効なものを使用して処理を続行します。

他のフィルタ・パラメータとともに % を指定すると、`ddlgen` はフィルタ可能な他のすべてのパラメータを無視し、スキーマのみの定義を表示します。その後、`ddlgen` はデータベースにフィルタ可能として適用されたパラメータのサブセット内の従属性を引き続き評価します。

-V

`ddlgen` のバージョン番号と著作権メッセージを表示して、オペレーティング・システムに戻ります。

例 **例 1** (キャッシュ) ポート 1955 を使用する HARBOR というマシン上の *default data cache* というキャッシュの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TC -N"default data cache"
```

すべてのキャッシュの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TC -N%
```

例 2 (デフォルト) ポート 1955 を使用する HARBOR という名前のマシン上で稼働している *pubs2* データベースで *jones* が所有する “*phondflt*” という DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TD -Njones.phondflt -Dpubs2
```

または、*ddlgen* では完全に修飾された名前を *-N* フラグで使用できるため、次のように *-Ddbname* を省略してデータベース名を *-N* オプションに含めることもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TD -Ndbname.owner.defaultname
```

“*owner*” が所有するデータベース内のすべてのデフォルトの DDL を生成する場合は、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TD -Nowner.% -Ddbname
```

例 3 (データベース) ポート 1955 を使用する HARBOR というマシン上で稼働している *pubs2* というデータベースの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TDB -Npubs2
```

dbname を指定しない場合、*ddlgen* は *login* のデフォルト・データベースの DDL を生成します。次のようになります。

```
ddlgen -Ulogin -Ppassword -Sserver:port
```

-T パラメータを使用しない場合、*ddlgen* はデフォルト・タイプのデータベースの DDL を生成します。次のようになります。

```
ddlgen -Ulogin -Ppassword -Sserver:port -Ndbname
```

すべてのデータベースの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -N%
```

例 4 (データベース・デバイス) ポート 1955 を使用する HARBOR というマシン上で稼働している *master* というデータベース・デバイスの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TDBD -Nmaster
```

すべてのデータベース・デバイスの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDBD -N%
```

例 5 (テンポラリ・データベース) tempdb を含め、すべてのデータベースの DDL を生成します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -N%
```

すべてのテンポラリ・データベースの DDL を生成するには、次のように OD 拡張データベース・タイプを使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -XOD -N%
```

Adaptive Server バージョン 12.5.0.3 以降では OD 拡張タイプを使用できますが、12.5.0.3 より前のバージョンでは警告メッセージが表示されます。このメッセージは無視してもかまいません。ddlgen はコマンドの処理を続行します。

テンポラリ・データベースを除くすべてのデータベースの DDL を生成するには、次のように OU 拡張タイプを使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -XOU -N%
```

次の例は、tempdb1 というテンポラリ・データベースの DDL を生成します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -Ntempdb1
```

出力には次のものが含まれます。

- create temporary database 文

```
create temporary database tempdb1 on master = 4,
      asdas = 2
go
```

- isql アプリケーションが tempdb1 にバインドされる sp_tempdb bind 文

```
sp_tempdb 'bind','ap', 'isql', 'DB', 'tempdb1'
go
```

注意 ビュー、ストアド・プロシージャ、テーブルなどのオブジェクトの DDL はテンポラリ・データベースの DDL とともに生成されることはありません。これらのオブジェクトは一時的なものであり、サーバの再起動時に再作成されるためです。

データベース・オブジェクトの DDL の生成時に -F パラメータを使用してテーブルをフィルタすると、テーブル・オブジェクトのサブセットであるインデックス、参照整合性、キー制約、トリガが自動的にフィルタされます。

例 6 (インメモリ・データベース) インメモリ・データベースの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TDB -Nimdb_1
```

例 7 テンポラリ・インメモリ・データベースの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TDB -Nimdb_temp1
```

例 8 (アーカイブ・データベース) すべてのアーカイブ・データベースの DDL を生成するには、拡張フィルタ・オプション “OA” を使用します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TDB -N% -XOA
```

1 つのアーカイブ・データベースの DDL を生成するには、標準的なデータベースの構文を使用します。次の例では、アーカイブ・データベース adb1 の DDL を作成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TDB -Nadb1
```

例 9 (ダンプ・デバイス) ポート 1955 を使用する HARBOR というマシン上で稼働している `tapedump1` というダンプ・デバイスの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TDPD -Ntapedump1
```

すべてのダンプ・デバイスの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDPD -N%
```

例 10 (暗号化キー) 拡張オプション -XOD を指定した場合、暗号化キーの DDL とともにシステム暗号化パスワードを生成します。`sp_encryption` 文の後にすべての暗号化キーの DDL 文が続く出力を生成します。この例では、ポート 1955 を使用する HARBOR というマシン上のログイン “george” の DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TEK -Ngeorge -XOD
```

ポート 1955 を使用するマシン HARBOR の `authors` データベースのすべての暗号化キーに対する DDL を生成するには、次の構文を使用します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TEK -Nauthors.dbo.%
```

例 11 ポート 1955 を使用するマシン “HARBOR” の `accounts` データベースのすべての暗号化キーに対する DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TEK  
-Naccounts.dbo.%
```

または、-D オプションを使用してデータベース名を指定します。

例 12 `SampleKeysDB` データベースで暗号化キー “`ssn_key`” の DDL を生成します。

```
ddlgen -Usa -P -Sserver -TEK -NSampleKeysDB.dbo.ssn_key
```


例 13 この例では、“ssn_key” 暗号化キーの DDL の生成時に、-FEKC を使用してキー・コピーの DDL が生成されないようにします。

```
ddlgen -Usa -P -Sserver -TEK -NSampleKeysDB.dbo.ssn_key
-FEKC
```

例 14 “ssn_key” とそのキー・コピーの DDL を生成します。

```
ddlgen -Usa -P -Sserver -TEK -NSampleKeysDB.dbo.ssn_key
```

例 15 この例では、暗号化キー “ssn_key” で暗号化され、復号化のデフォルト値が 100 に設定された “ssn” カラムを持つ employee というテーブルで ddlgen コマンドを発行します。

```
ddlgen -Usa -P -Sserver -TU -N db1.dbo.employee
```

例 16 (エンジン・グループ) ポート 1955 を使用する HARBOR という名前のマシン上で稼働している LASTONLINE というエンジン・グループの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TEG -NLASTONLINE
```

すべてのエンジン・グループの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TEG -N%
```

例 17 (実行クラス) ポート 1955 を使用する HARBOR というマシン上で稼働している EC2 という実行クラスの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TEC -NEC2
```

すべての実行クラスの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TEC -N%
```

例 18 (拡張ストアド・プロシージャ) -N オプションに、完全に修飾された *dbname.owner.extendedstoredprocedure* 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働し、Jones が所有する pubs2 データベースの xp_cmdshell 拡張ストアド・プロシージャの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TXP -Npubs2.jones.xp_cmdshell
```

または、完全に修飾された名前を使用する代わりに、次のように -D オプションを使用することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TXP
-Nowner.extendedstoredprocedure -Ddbname
```

すべての拡張ストアド・プロシージャの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TXP -Ndbname.owner.%
```

例 19 (フィルタ) テーブルの DDL から、すべてのインデックス (I)、参照整合性制約 (RI)、プライマリ・キー/ユニーク・キー制約 (KC)、トリガ (TR)、パーティション条件 (PC) をフィルタして、pubs2 データベース内の authors テーブルの DDL を生成します。

```
ddlgen -Uroy -Proy123 -TU -Nauthors -Dpubs2 -F%
```

または、次のようにフィルタを個別に指定することもできます。

```
ddlgen -Ulogin -Ppassword -TU -Ndbname.owner.table
-FI,RI,KC,TR
```

次の例は、*table_name* の定義を生成し、外部キーとプライマリ/ユニーク・キーをフィルタします。

```
ddlgen -Ulogin -Ppassword -TU -Ntable_name -Ddbname
-FRI,KC
```

次の例はどちらも、データベース全体の指定ユーザの外部キーを生成します。

```
ddlgen -Ulogin -Ppassword -TRI -N%.%.% -Ddbname
```

または

```
ddlgen -Ulogin -Ppassword -TRI -Ndbname%.%.%
```

次の例はどちらも、データベース内の “PK” で始まるすべてのテーブルのプライマリ/ユニーク・キーの DDL を生成します。

```
ddlgen -Ulogin -Ppassword -TKC -Ndbname%.%.PK%
```

または

```
ddlgen -Ulogin -Ppassword -TKC -N%.%.PK% -Ddbname
```

次の例は、データベースのスキーマのみの定義を生成します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TF -Ndbname -F%
```

または、次のようにフィルタを個別に指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -Ndbname
-FSGM,GRP,USR,R,D,UDD,V,P,XP,I,RI,KC,TR
```

次の例は、コンパイル済みオブジェクトをスキップして、データベース DDL を生成します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -Ndbname
-FTR,D,XP,V,R
```

次の例は、テーブル定義のないデータベース定義を生成します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TDB -Ndbname
-FU
```

例 20 (グループ) `-N` オプションに、完全に修飾された `dbname.groupname` 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働している `pubs2` データベース内の “public” というグループの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TGRP -Npubs2.public
```

または、`-D` オプションを使用して `dbname` を指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TGRP -Ngroupname -Ddbname
```

すべてのグループの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TGRP -Ndbname.%
```

例 21 (インデックス) `pubs2` データベースで `dbo` が所有する `authors` テーブルの `au_lname` というインデックスの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TI -Ndbo.authors.au_lname -Dpubs2
```

または、`ddlgen` では完全に修飾された名前を `-N` フラグで使用できるため、次のように `-Ddbname` を省略してデータベース名を `-N` オプションに含めることもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port  
-TI -Ndbname.owner.tablename.indexname
```

完全に修飾された名前を使用する場合は、`-D` オプションを省略できません。

1 つのテーブルのすべてのインデックスの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TI  
-Ndbname.owner.tablename.%
```

データベース内のすべてのテーブルのすべてのインデックスの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TI  
-Ndbname.%.%.%
```

たとえば、次の例は `pubs2` データベース内のすべてのテーブルのすべてのインデックスの DDL を生成します。

```
ddlgen -Usa -P -SHARBOR:1955 -TI -Npubs2.%.%.%
```

例 22 (キー) 次のどちらのコマンドでも、データベースのすべてのテーブルに “PK” で始まるプライマリ/ユニーク・キーの DDL を生成します。

```
ddlgen -Usa -P -TKC -Ndbname.%.%.PK%
```

または

```
ddlgen -Usa -P -TKC -N%.%.PK% -Ddbname
```

例 23 (論理キー) LK は、`sp_primarykey` 文、`sp_commonkey` 文、`sp_foreignkey` 文で定義されたテーブルの論理キーを生成します。これらのキーには名前がないため、この場合のオブジェクトの名前はテーブルの名前になります。この例では、ポート 1955 を使用する HARBOR というマシン上で稼働している `pubs2` データベース内の `authors` テーブルの論理キーの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TLK -Npubs2.dbo.authors
```

`pub2` データベース内のすべての論理キーの DDL を生成するには、次の構文を使用します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TLK -Npubs2.%.%
```

-F 引数に LK を使用して、`authors` テーブルの DDL から論理キーの定義をフィルタするには、次の構文を使用します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TLK -Npubs2.dbo.authors -FLK
```

例 24 (ログイン) TL は、1 つまたはすべてのログインに対して DDL を生成します。この例では、ポート 1955 を使用する HARBOR というマシン上のすべてのログインの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TL -N%
```

注意 すべてのログインに対して生成された DDL のパスワードは “password” です。

または、-N% の代わりに `-Nusername` を使用して個々のログインを指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TL -Nusername
```

サーバ全体の複雑なパスワード・オプションが 1 つまたは複数のログインに指定されている場合、すべての `sp_addlogin` および `sp_loglogin` DDL 文が最初に生成され、次に、そのパスワード・オプションに対する DDL 文が生成されます。この例では、ポート 1955 を使用する HARBOR というマシン上のログイン “george” の DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TL -Ngeorge
```

例 25 (リモート・サーバ) ポート 1955 を使用する HARBOR というマシン上の ORANGE というリモート・サーバの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TRS -NORANGE
```

すべてのリモート・サーバの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TRS -N%
```

例 26 (役割) ポート 1955 を使用する HARBOR というマシン上の sa_role の DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TRO -Nsa_role
```

すべての役割の DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TRO -N%
```

注意 生成されたすべての役割の DDL のパスワードは “password” です。

例 27 (ルール) ポート 1955 を使用する HARBOR というマシン上の authors に関連するすべてのルールの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TR -Nauthors.dbo.%
```

% 記号は、サーバ上に存在するすべてのルールの DDL を作成するように ddlgen に指示します。

次のように、ルールの完全に修飾された名前を指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TR -Ndbname.owner.rulename
```

または、次のように -D パラメータを使用することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TR -Nowner.rulename -Ddbname
```

例 28 (セグメント) -N オプションに、完全に修飾された *dbname.segmentname* 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働している pubs2 データベースの logsegment というセグメントの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TSGM -Npubs2.logsegment
```

または、-D オプションを使用して、*dbname* を指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TSGM -Nsegmentname -Ddbname
```

すべてのセグメントの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TSGM -Ndbname.%
```

例 29 (SQLJ 関数) master データベースで dbo が所有する region_of という SQLJ 関数の DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TF -Nmaster.dbo.region_of
```

または、次のように -D パラメータを使用することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TF -Ndbo.region_of -Dmaster
```

データベース内のすべての SQLJ 関数の DDL を生成するには、オブジェクト・タイプ F を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TF -Ndbname.owner.%
```

例 30 (SQLJ プロシージャ) ストアド・プロシージャの一種です。ストアド・プロシージャの DDL とともに SQL プロシージャの DDL を生成します。次の例は、master データベースで dbo が所有する SQLJ プロシージャを含む、すべてのストアド・プロシージャの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TP -Nmaster.dbo.%
```

master データベースで dbo だけが所有するすべての SQLJ プロシージャの DDL を生成するには、次の構文を使用します。拡張タイプ OD は、SQLJ プロシージャを示します。

```
ddlgen -Ulogin -Ppassword-Sserver:port -TP -Nmaster.dbo.% -XOD
```

master データベースで dbo が所有する SQLJ プロシージャを除くすべてのプロシージャの DDL を生成するには、次の構文を使用します。拡張タイプ OU は、SQLJ プロシージャ以外のすべてのストアド・プロシージャを示します。

```
ddlgen -Ulogin -Ppassword-Sserver:port -TP -Nmaster.dbo.% -XOU
```

例 31 (ストアド・プロシージャ) -N オプションに、完全に修飾された *dbname.owner.procedure_name* 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働している pubs2 データベースの sp_monitor ストアド・プロシージャの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TP -Npubs2.dbo.sp_monitor
```

または、-D オプションを使用して、*dbname* を指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TP -Nowner.procedurename -Ddbname
```

すべてのストアド・プロシージャの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TP -Ndbname.owner.%
```

例 32 (SSL 対応サーバ) ポート 1955 を使用する HARBOR というマシン上で稼働する SSL 対応 Adaptive Server の pubs2 データベース内のオブジェクトに対する DDL を生成します。

```
ddlgen -Uroy -Proy123 -Sssl:HARBOR:1955 -TDB -Npubs2
```

例 33 (テーブル) ポート 1955 を使用する HARBOR というマシン上で稼働し、“dbo” が所有する pubs2 データベース内のすべてのユーザ・テーブルの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TU -Ndbo.% -Dpubs2
```

-N パラメータを使用して、テーブルの完全に修飾された名前を指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TU
-Ndbname.tableowner.tablename
```

または、次のように -D パラメータを使用してデータベースを指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TU
-Ntableowner.tablename -Ddbname
```

すべてのプロキシ・テーブルについて、値 OD を使用する DDL を生成するには、次のように -XOD を使用します。X は拡張タイプで、OD はプロキシ・テーブルを表します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TU
-Ntableowner.% -Ddbname -XOD
```

すべてのユーザ・テーブルについて、値 OU を使用する DDL を生成するには、次のように -XOU を使用します。X は拡張タイプで、OU はユーザ・テーブルを表します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TU
-Ntableowner.% -Ddbname -XOU
```

ユーザ・テーブルとプロキシ・テーブルを含むすべてのテーブルの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TU -Ndbname.tableowner.%
```

例 34 (トリガ) -N オプションに、完全に修飾された *dbname.owner.trigger_name* 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働している pubs2 データベースの checksum というトリガの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TTR -Npubs2.dbo.checksum
```

または、-D オプションを使用して、*database_name* を指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TTR
      -Nowner.triggername -Ddbname
```

また、次の構文を使用して、テーブルのトリガの DDL を生成できます。

```
-Ndb_name.table_owner.table_name.trigger_name
```

データベースのすべてのトリガの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TTR -Ndbname.owner.%
```

次の構文を使用して、テーブルのすべてのトリガの DDL を生成することもできます。

```
-Ndb_name.table_owner.table_name.%
```

注意 テーブルのすべてのトリガの DDL を生成するときは、**-D** パラメータを使用できません。

例 35 (ユーザ定義データ型) **-N** オプションに、完全に修飾された *dbname.userdefined_datatype* 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働している **pubs2** データベースの “Identype” というユーザ定義データ型の DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TUDD -Npubs2.Identype
```

または、**-D** オプションを使用して、*dbname* を指定することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TUDD
      -Nuserdefined_datatype -Ddbname
```

すべてのユーザ定義データ型の DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TUDD -Ndbname.%
```

例 36 (ビュー) **-N** オプションに、完全に修飾された *dbname.owner.viewname* 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働している **pubs2** データベースで Miller が所有する **retail** というビューの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TV -Npubs2.miller.retail
```

または、完全に修飾された名前を使用する代わりに、次のように **-D** オプションを使用することもできます。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TV -Nowner.viewname -Ddbname
```

すべてのビューの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TV -Ndbname.owner.%
```


例 37 (ユーザ) `-N` オプションに、完全に修飾された `dbname.username` 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働している `pubs2` データベース内の Smith というユーザの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TUSR -Npubs2.smith
```

または、完全に修飾された名前を `-N` オプションで使用する代わりに、`-N` オプションと `-D` オプションの両方を使用することもできます。

```
ddlgen -Ulogin -Ppassword -Shost_name:port -TUSR -Nusername -Ddbname
```

すべてのユーザの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Sserver:port -TUSR -Ndbname.%
```

例 38 (論理クラスタ) サーバ “ase1” 上の “my_lcluster” の DDL を生成します。次のように入力します。

```
ddlgen -Usa -P -Sase1 -TLC -Nmy_lcluster
```

例 39 (論理クラスタ) サーバ “ase1” 上のすべての論理クラスタの DDL を生成します。次のように入力します。

```
ddlgen -Usa -P -Sase1 -TLC -N%
```

例 40 (ユーザ定義 Web サービス) `-N` オプションと `-T` オプションに、完全に修飾された `dbname.username.webservice_name` 形式を使用して、ポート 1955 を使用する HARBOR というマシン上で稼働している `pubs2` データベース内の `sp_who_service` というユーザ定義 Web サービスの DDL を生成します。

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TWS  
-Npubs2.dbo.sp_who_service
```

指定されたユーザ定義 Web サービスの DDL を生成する構文は次のとおりです。

```
ddlgen -Ulogin -Ppassword -Shost_name:port -TWS  
-Ndbname.owner.webservice_name
```

`dbname` データベース内のすべてのユーザが所有するすべてのユーザ定義 Web サービスの DDL を生成するには、次の構文を使用します。

```
ddlgen -Ulogin -Ppassword -Shost_name:port -TWS -Ndbname.%.%
```

注意 `sp_webservices 'addalias'` 文を生成するのは、すべてのユーザ定義 Web サービスまたは 1 つのデータベースの DDL を生成する場合だけです。

使用法

- **ddlgen** は、ビュー内の既存の順序、ストアド・プロシージャ、またはトリガを識別しません。このため、データベースの DDL を生成するときは、独立したビュー、ストアド・プロシージャ、トリガで **ddlgen** を実行してから、それに従属しているビュー、ストアド・プロシージャ、トリガで **ddlgen** を実行します。たとえば、ビュー B がビュー A に従属している場合は、**ddlgen** を、ビュー A で実行してから、ビュー B で実行します。
- **ddlgen** のデフォルト情報は、次のとおりです。

オプション	パラメータ	必須	デフォルト
-U	<i>username</i>	はい	なし
-P	<i>password</i>	はい	なし
-S	<i>host_name:port_number</i>	はい	なし
-T	<i>object_type</i> 有効なオブジェクト・タイプのリストについては、-T パラメータの説明を参照	いいえ	Database
-N	<i>object_name</i>	はい (-T の <i>object_type</i> が DB (データベース) でない場合)	-T <i>object_type</i> が db の場合、または -T が指定されていない場合は、 <i>username</i> のデフォルト・データベース名
-D	<i>database_name</i>	いいえ	<i>username</i> のデフォルト・データベース
-X	<i>extended_object_type</i> オプションは次のとおり <ul style="list-style-type: none"> • OU — ユーザ・テーブル、ユーザ・データベース (テンポラリ・データベースを除く)、ストアド・プロシージャ (SQLJ プロシージャを除く) の場合 • OD — プロキシ・テーブル、テンポラリ・データベース、SQLJ プロシージャの場合 	いいえ。-T の <i>object_type</i> が以下の場合にのみ使用 <ul style="list-style-type: none"> • U (ユーザ・テーブル) • P (プロシージャ) • DB (データベース) 	なし
-O	<i>output_file_name</i>	いいえ	標準出力
-E	<i>error_file_name</i>	いいえ	標準出力
-L	<i>log_file_name</i>	いいえ	なし
-V	<i>ddlgen</i> の <i>version_number</i>	いいえ	なし

- コマンド・ラインで、Adaptive Server インストール環境に含まれている *ddlgen* シェル・スクリプト・ファイル (Windows では *ddlgen.bat*) を使用して、*ddlgen* を呼び出します。*DDLGen.jar* のメイン・クラスは *com.sybase.ddlgen.DDLGenerator* です。
- Adaptive Server の Sybase Central プラグインで *ddlgen* を起動するには、次の手順に従います。
 - a DDL を生成するオブジェクトを右クリックします。
 - b [DDL の生成] を選択します。
- *create table* の出力 DDL では、バインド文は従属 DDL ではなく独立 DDL として生成されます。

以降の項では、次の内容について詳しく説明します。

- [「ddlgen でのパスワードを隠す」 \(203 ページ\)](#)
- [「暗号化カラムでの ddlgen の使用」 \(204 ページ\)](#)

ddlgen でのパスワードを隠す

UNIX コマンド・ライン環境で *ddlgen* ユーティリティを発行するとき、その *ddlgen* コマンドは、パスワードを含め、同じ UNIX マシンの他のユーザが *ps* プロセス管理コマンドを発行することで、それらのユーザに表示されます。このプロセス管理コマンドはそのマシンで実行されているプロセスのステータスを表示します。

ddlgen -P パスワード・パラメータ・オプションを使用すると、パスワードが他のユーザに表示されないように、スクリプトから *ddlgen* を呼び出すことができます。

これを行うには、*\$PSWD* 環境変数を Adaptive Server のログイン・パスワードに明示的に設定し、*-P* パラメータ内の文字列 “*ext*” を指定します。*ext* は疑似パスワードとして機能するため、実際のパスワードを次の行に指定できます。C シェルからではなく、コマンド・ラインまたは Bourne シェル・スクリプトで *\$PSWD* を設定します。

\$PSWD を使用して *ddlgen* を実行するには、次のようにします。

- 1 *\$PSWD* 環境変数を設定します。

```
setenv PSWD pass_word
```

- 2 *ddlgen* を実行します。

```
ddlgen -Ulogin -Pext -Sserver:port -Ttype -Nname << END
$PSWD
END
```

パスワードを特定のファイルに保持する場合は、*\$PSWD* を 'cat *filename*' に置き換えます。*filename* はパスワード・ファイルのロケーションです。次に例を示します。

```
ddlgen -Ulogin -Pext -Sserver:port -Ttype -Nname << END
'cat filename'
END
```

暗号化カラムでの *ddlgen* の使用

- *ddlgen* ユーティリティは、15.0.2 より前の暗号化をサポートしています。15.0.2 より前の *ddlgen* のサポートには、データベースでの暗号化キーの DDL の生成と、サーバ間で暗号化キーを同期するための DDL の生成が含まれます。

ddlgen を使用して Adaptive Server バージョン 15.0.2 以降で暗号化キーの DDL を生成すると、この DDL によって 15.0.2 より前のバージョンの Adaptive Server でエラーが発生することがあります。特に、暗号化キーがユーザ指定のパスワードで暗号化された場合、または暗号化キーにキー・コピーがある場合に発生します。

- 暗号化キーに使用される EK タイプは、暗号化キーを作成し、そのキーにパーミッションを付与する DDL を生成します。*ddlgen* は、暗号化カラム情報と `grant decrypt` 文をテーブル定義とともに生成します。
- `-XOD` オプションを指定しない場合、およびマイグレートするキーが `with passwd` 句を使用してソース・データベース内に作成されている場合、*ddlgen* は `password` をその明示的なパスワードとして使用して `create encryption key` コマンドを生成します。これは、役割とログインのパスワードに対する *ddlgen* の動作に似ています。
- `-XOD` は、`create encryption key` を生成します。この文では、`sysencryptkeys` として暗号化されたキーの値が指定されます。`-XOD` を使用して、データ移動のためにサーバ間で暗号化キーを同期します。

ddlgen -XOD は、システム暗号化パスワード (*ddlgen* が設定され、システム暗号化パスワードによって暗号化されたキーに対して DDL が生成される場合) を含む DDL、およびキーの DDL を生成します。

❖ XOD フラグ指定なしの DDL の生成

- ユーザが *ddlgen* で `-XOD` フラグを指定しないと、次の2つのことが発生する可能性があります。

- 暗号化キーの作成時にパスワードを指定しないと、`ddlgen`によりパスワードなしの DDL が生成されます。
- 暗号化キーの最初の作成時にパスワードを指定すると、`ddlgen`によってデフォルトのパスワード ‘password’ が生成されます。これは、役割およびログインのパスワードに対する `ddlgen` の動作と同様で、その出力は以下と似たものになります。

```
-----
-- DDL for EncryptedKey 'ssn_key'
-----
print 'ssn_key'

--The DDL is generated with a default password - 'password' as
--a password was specified when this key was created.

create encryption key SampleKeysDB.dbo.ssn_key for AES
with keylength 128
passwd 'password'
init_vector random
go
```

❖ XOD フラグを設定した DDL の生成

- ユーザが `ddlgen` コマンドで “-XOD” を指定すると、`ddlgen`によって、システム暗号化パスワード (`ddlgen` が設定され、システム暗号化パスワードによって暗号化されたキーに対して DDL が生成される場合) を含む DDL、およびキーの DDL が生成されます。

次の構文では、システム暗号化パスワードが生成されます。

```
ddlgen -Usa -P -Sserver -TEK -NsampleKeysdb.dbo.ek1 -XOD
```

このコマンドの出力は次のとおりです。

```
-- System Encryption Password

use SampleKeysDB
go

sp_encryption 'system_encr_passwd',
'0x8e050e3bb607225c60c7cb9f59124e99866ca22e677b2cdc9a4d09775850f4721',
NULL, 2, 0
go

-----
-- DDL for EncryptedKey 'ek1'
-----
```

```

print '<<<< CREATING EncryptedKey - "ek1" >>>>'
go

create encryption key SampleKeysDB.dbo.ek1 for AES
with keylength 128
passwd 0x0000C7BC28C3020AC21401
init_vector NULL
keyvalue
0xCE74DB1E028FF15D908CD066D380AB4AD3AA88284D6F7742DFFCADCAABE4100D01
keystatus 32
go

```

注意 ddlgen を使用してソースからターゲット・サーバにキーをマイグレートする場合、“-XOD”パラメータを使用して生成された暗号化キーに対して ddlgen 出力 (ソース・サーバから) を実行するには、ターゲット・サーバでシステム暗号化パスワードを NULL (存在する場合) に設定します。この設定を行わないと、ターゲット・サーバに対して ddlgen 出力を実行しようとするときにエラーが発生します。

キー・コピーのサポート ddlgen ユーティリティではベース・キーの DDL とともにキー・コピーの DDL も生成されます。たとえば、次の構文では “ssn_key” とそのキー・コピーの DDL が生成されます。

```
ddlgen -Usa -P -Sserver -TEK -NSampleKeysDB.dbo.ssn_key
```

ddlgen からの出力は次のようになります。

```

-----
-- DDL for EncryptedKey 'ssn_key'
-----
print 'ssn_key'

--The DDL is generated with a default password - 'password' as
--a password was specified when this key was created.

create encryption key SampleKeysDB.dbo.ssn_key for AES
with keylength 128
passwd 'password'
init_vector random
go

print 'Key Copies for ssn_key'

-- Generating DDL for Key Copies for 'ssn_key'

```

```
alter encryption key 'ssn_key'
with passwd 'password'
add encryption with passwd 'passwd'
for user 'dbo'.
```

-XOD フラグを指定すると、キー・コピーの DDL は次のようになります。

```
alter encryption key SampleKeysDB.dbo.ssn_key add encryption
with keyvalue
0x84A7360AA0B28801D6D4CBF2F8219F634EE641E1082F221A2C58C9BBEC9F49B501
passwd 0x000062DF4B8DA5709E5E01
keystatus 257
for user 'user1'
go
```

EKC 暗号化キー・コピー・フィルタ

Adaptive Server 15.0.2 では、`ddlgen` によって `-F` フィルタ引数の EKC (暗号化キー・コピー) 拡張タイプがサポートされ、暗号化キーのキー・コピーの生成が抑制されます。

この例では、“`ssn_key`” 暗号化キーの DDL の生成時に、`-FEKC` を使用してキー・コピーの DDL が生成されないようにします。

```
ddlgen -Usa -P -Sserver -TEK -NSampleKeysDB.dbo.ssn_key -FEKC
```

`ddlgen` からの出力は次のようになります。

```
-----
-- DDL for EncryptedKey 'ssn_key'
-----
```

```
print 'ssn_key'
```

```
--The DDL is generated with a default password - 'password' as
--a password was specified when this key was created.
```

```
create encryption key SampleKeysDB.dbo.ssn_key for AES
with keylength 128
passwd 'password'
init_vector random
go
```

テーブル DDL の作成

`ddlgen` は DDL のテーブルとともに、`decrypt_default` 文 (暗号化カラムに設定されている場合) を生成できます。

次の例では、暗号化キー “`ssn_key`” で暗号化され、復号化のデフォルト値が “100” に設定された “`ssn`” カラムを持つ `employee` というテーブルで `ddlgen` コマンドを発行します。

```
ddlgen -Usa -P -Sserver -TU -Nemployee
```

このコマンドからの DDL 出力は次のとおりです。

```
create table employee (  
  ssn          int          not null  encrypt with ssn_key decrypt_default  
100 ,  
  last_name    int          not null ,  
  first_name   int          not null  
)  
lock allpages  
  on 'default'  
go
```

パーミッション

以下の DDL を生成するためには、sa_role または sso_role を持っている必要があります。

- 暗号化キー
- ログイン
- 役割

その他のすべてのオブジェクトについては、DDL を生成するための特定のパーミッションまたは役割は必要ありません。

defncopy

説明

特定のビュー、ルール、デフォルト、トリガ、プロシージャの定義を、データベースからオペレーティング・システム・ファイルに、またはオペレーティング・システム・ファイルからデータベースにコピーします。\$SYBASE/\$SYBASE_OCS/bin にあります。

(Windows の場合) このユーティリティは defncopy.exe で、%SYBASE%\\$SYBASE_OCS%\bin にあります。

構文

```
defncopy
    [-X]
    [-a display_charsef]
    [-I interfaces_file]
    [-J [client_charsef]]
    [-K keytab_file]
    [-P password]
    [-R remote_server_principal]
    [-S [server_name]]
    [-U username]
    [-V security_options]
    [-Z security_mechanism]
    [-z language]
    { in file_name database_name |
      out file_name database_name [owner.]object_name
      [[owner.]object_name...]}

```

または

```
defncopy -v
```

パラメータ

-X

サーバへの現在の接続で、クライアント側のパスワード暗号化を使用してログインを開始します。defncopy (クライアント) は、パスワードの暗号化が必要であることをサーバに通知します。サーバは、defncopy がパスワードの暗号化に使用する暗号化キーを返送し、パスワードを受け取ると、そのキーを使用してパスワードを認証します。

defncopy がクラッシュすると、パスワードを含むコア・ファイルが作成されます。暗号化オプションを使用していない場合、パスワードは、コア・ファイルにプレーン・テキストで表示されます。暗号化オプションを使用した場合、パスワードは表示されません。

-a *display_charset*

defncopy を実行しているマシンの文字セットとは異なる文字セットを使用する端末から defncopy を実行します。-a を -J とともに使用して、変換に必要な文字セット変換ファイル (.xlt ファイル) を指定します。-J を使用しないで -a を使用するの、クライアントの文字セットがデフォルトの文字セットと同じである場合だけです。

注意 `ascii_7` 文字セットは、すべての文字セットと互換性があります。Adaptive Server の文字セットとクライアントの文字セットのどちらかが `ascii_7` に設定されていれば、すべての 7 ビット ASCII 文字が変更されることなくクライアントとサーバの間で受け渡されます。その他の文字セットを使用している場合は、変換エラーが発生します。文字セット変換の詳細については、『システム管理ガイド』を参照してください。

-l *interfaces_file*

Adaptive Server に接続するときには検索する `interfaces` ファイルの名前とロケーションを指定します。-l を指定しない場合、defncopy は UNIX プラットフォームでは SYBASE 環境変数によって指定されたディレクトリの `interfaces` という名前のファイルを探します。Windows では Sybase リリース・ディレクトリの `ini` サブディレクトリにある `sql.ini` という名前のファイルを探します。

-J *client_charset*

クライアントで使用する文字セットを指定します。フィルタによって、`client_charset` と Adaptive Server の文字セット間で入力に変換されます。

-J `client_charset` は、クライアントで使用する文字セットである `client_charset` とサーバの文字セット間の変換を Adaptive Server に要求します。

-J に引数を指定しない場合、文字セット変換は NULL に設定されません。この場合、変換は行われません。クライアントとサーバが同じ文字セットを使用する場合に、このパラメータを使用してください。

-J を省略すると、文字セットはプラットフォームのデフォルトに設定されます。デフォルトの文字セットは、クライアントが使用する文字セットと同じであるとはかぎりません。文字セットと関連するフラグの詳細については、使用しているプラットフォームの『システム管理ガイド』と『設定ガイド』を参照してください。

-K *keytab_file*

DCE での認証に使用する `keytab` ファイルのパスを指定します。

-P password

パスワードを指定します。**-P** を指定しない場合、**defncopy** はパスワードの入力を要求するプロンプトを表示します。

-R remote_server_principal

リモート・サーバのプリンシパル名を指定します。デフォルトでは、サーバのプリンシパル名はサーバのネットワーク名 (**-S** パラメータまたは **DSQUERY** 環境変数で指定) と一致します。サーバのプリンシパル名とネットワーク名が一致しない場合は、**-R** パラメータを使用してください。

-S server_name

接続する Adaptive Server の名前を指定します。引数なしで **-S** を指定した場合、**defncopy** は **SYBASE** という名前のサーバを探します。**-S** を指定しない場合、**defncopy** は **DSQUERY** 環境変数によって指定されたサーバを使用します。

-U username

ログイン名を指定します。ログイン名では大文字と小文字が区別されます。**username** を指定しない場合、**defncopy** は現在のユーザのオペレーティング・システム・ログイン名を使用します。

-V security_options

ネットワーク・ベースのユーザ認証を指定します。このオプションを使用する場合、ユーザはユーティリティを実行する前にネットワークのセキュリティ・システムにログインする必要があります。この場合、ユーザは **-U** オプションにネットワーク・ユーザ名を指定し、**-P** オプションによって指定されたパスワードは無視されます。

-V の後にキー文字オプションの **security_options** 文字列を続けると、他のセキュリティ・サービスを有効にできます。これらのキー文字は、以下のとおりです。

- **c** - データ機密性サービスを有効にする
- **i** - データ整合性サービスを有効にする
- **m** - 接続の確立に相互認証を有効にする
- **o** - データ・オリジン・スタンプング・サービスを有効にする
- **r** - データ・リプレイの検出を有効にする
- **q** - 順序不整合の検出を有効にする

-Z security_mechanism

接続で使用するセキュリティ・メカニズムの名前を指定します。

セキュリティ・メカニズム名は `$SYBASE/install/libtcl.cfg` 設定ファイルに定義されています。`security_mechanism` の名前が指定されていない場合は、デフォルトのメカニズムが使用されます。セキュリティ・メカニズム名の詳細については、『Open Client/Server 設定ガイド』の `libtcl.cfg` ファイルの説明を参照してください。

-z language

サーバが `defncopy` のプロンプトとメッセージの表示に使用する代替言語の公式名です。`-z` フラグを指定しないと、`defncopy` はサーバのデフォルト言語を使用します。

インストール時に `Adaptive Server` に言語を追加します。インストール後に追加する場合は、`langinstall` ユーティリティ (Windows では `langinst`) または `sp_addlanguage` ストアド・プロシージャを使用します。

in | out

定義コピーの方向 (宛先) を指定します。

file_name

定義コピーの送信元または送信先であるオペレーティング・システム・ファイルの名前を指定します。コピー・アウトを行うと、既存のファイルはすべて上書きされます。

database_name

定義コピーの送信元または送信先であるデータベースの名前を指定します。

owner

ユーザかデータベース所有者がコピーするテーブルを所有している場合はオプションです。所有者を指定しない場合、`defncopy` はまずユーザが所有する該当の名前のテーブルを探します。次に、データベース所有者が所有するテーブルを探します。それ以外のユーザがテーブルを所有している場合は、所有者の名前を指定しなければなりません。指定しないと、コマンドは失敗します。

object_name

`defncopy` がコピー・アウトするデータベース・オブジェクトの名前を指定します。定義をコピー・インするときは `objectname` を使用しないでください。

-v

`defncopy` のバージョン番号と著作権メッセージを表示して、オペレーティング・システムに戻ります。

例

例 1 `new_proc` ファイルから、MERCURY サーバ上の `stagedb` データベースに定義をコピーします。MERCURY との接続は、ユーザ名 “sa”、パスワード NULL で確立されます。

```
defncopy -Usa -P -SMERCURY in new_proc stagedb
```

例 2 `sp_calccomp` オブジェクトと `sp_vacation` オブジェクトの定義を、SYBASE サーバ上の `employees` データベースから `dc.out` ファイルにコピーします。メッセージとプロンプトはフランス語で表示されます。ユーザはパスワードの入力を要求されます。

```
defncopy -S -z french out dc.out employees sp_calccomp sp_vacation
```

使用法

- スレッド・ドライバを使用する場合は `defncopy_r` の構文を使用します。
- IBM プラットフォームでスレッド・ドライバを使用する場合は、`defncopy` のこの構文を使用します。
- SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、`defncopy` を使用してください。
- `defncopy` プログラムは、オペレーティング・システムから直接起動します。`defncopy` を使用すると、ビュー、ルール、デフォルト、トリガ、プロシージャ用の定義 (`create` 文) をデータベースからオペレーティング・システム・ファイルにコピー・アウトするときに、非対話型操作でコピーを実行できます。または、指定したファイルからすべての定義をコピー・インします。
- `in filename` または `out filename` とデータベース名を指定する必要があります。不明確な名前は避けるようにしてください。コピー・アウトする場合は、オブジェクト名とその所有者の両方を表すファイル名を使用してください。
- `defncopy` は、コピー・アウトした各定義の終わりに次のようなコメントを付けます。

```
/* ### DEFNCOPY:END OF DEFINITION */
```

テキストとして作成する定義は、`defncopy` が正しくコピーを行うことができるように、このコメントで終了していなければなりません。

- シェルにとって特別な意味のある文字が `defncopy` に含まれている場合は、その値を引用符で囲んでください。

警告！ 100 文字を超える長いコメントが `create` 文の前にあると、`defncopy` が失敗することがあります。

パーミッション

- 定義をコピー・アウトするには、**sysobjects** テーブルおよび **syscomments** テーブルに対する **select** パーミッションが必要です。オブジェクト自体のパーミッションは必要ありません。
- システム・セキュリティ担当者が **sp_configure** システム・プロシージャを使用して **allow select on syscomments.text column** パラメータをリセットした場合、ユーザは **syscomments** テーブルの **text** カラムに対する **select** パーミッションを持つことはできません。このリセットにより、**select** パーミッションはオブジェクトの所有者とシステム管理者に制限されます。この制限は、**Adaptive Server** を「評価済み設定」で実行するために必要です。これについては、使用しているプラットフォームの『インストール・ガイド』および『設定ガイド』を参照してください。この場合、オブジェクト所有者またはシステム管理者は、**defncopy** を実行して定義をコピー・アウトします。

注意 テキストが暗号化されている場合、必要なパーミッションをすべて持っても、表示されないことがあります。詳細については、『*Transact-SQL ユーザーズ・ガイド*』の「ソース・テキストの検証および暗号化」を参照してください。

- コピー・インするには、コピー・インするオブジェクトのタイプの適切な **create** パーミッションが必要です。コピー・インされたオブジェクトは、コピーを実行した人のものになります。ユーザの代わりに定義をコピー・インするシステム管理者は、そのユーザとしてログインして、再構築したデータベース・オブジェクトへの適切なアクセス権をそのユーザに提供してください。

使用するテーブル

syscomments、**sysobjects**

参照

コマンド **create**、**select**

システム・プロシージャ **sp_addlanguage**、**sp_checkreswords**、**sp_configure**、**sp_procqmode**、**sp_remap**

ユーティリティ **langinstall**

dscp

説明	(UNIX のみ) UNIX プラットフォームのコマンド・ラインから、 <code>interfaces</code> ファイルのサーバ・エントリを表示、編集できます。 <code>\$SYBASE/\$SYBASE_OCS/bin</code> にあります。
構文	<code>dscp [-p]</code> または <code>dscp -v</code> <code>dscp</code> を終了するには、以下を入力します。 <code>quit</code> または <code>exit</code>
パラメータ	<code>-p</code> コマンド・ライン・プロンプトを抑制します。 <code>-v</code> <code>dscp</code> のバージョン番号と著作権メッセージを表示して、オペレーティング・システムに戻ります。
例	編集用にデフォルトの <code>interfaces</code> ファイルをオープンして、コマンド・ライン・プロンプトを抑制します。 <code>dscp -p</code>
使用法	<ul style="list-style-type: none">• SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、<code>dscp</code> を使用してください。• <code>dscp</code> ユーティリティ・プログラムはテキスト・ベースのユーティリティです。• <code>dscp</code> ユーティリティ・プログラムの詳細については、「第5章 dscp の使用」を参照してください。
参照	ユーティリティ <code>dsedit</code>

dsedit

説明	<p>dsedit ユーティリティでは、GUI を使用して <code>interfaces</code> ファイルのサーバ・エントリを表示したり、編集したりできます。このユーティリティは <code>\$SYBASE/\$SYBASE_OCS/bin</code> にあります。</p> <p>(Windows の場合) <code>dsedit.exe</code> ユーティリティを使用して <code>interfaces</code> ファイルに保存されたネットワーク接続情報を作成、修正できます。このユーティリティは <code>%SYBASE%\¥%SYBASE_OCS%\¥bin</code> にあります。</p>
構文	<p>dsedit</p> <p>または</p> <p>dsedit -v</p>
パラメータ	<p>-v</p> <p>dsedit のバージョン番号と著作権メッセージを表示します。</p>
使用法	<ul style="list-style-type: none">• SYBASE 環境変数を Adaptive Server の現在のバージョンのロケーションに設定してから、dsedit を使用してください。• バージョン番号を表示するために -v パラメータしか使用しない場合を除き、DISPLAY 環境変数を設定してから dsedit を呼び出してください。• dsedit ユーティリティ・プログラムの詳細については、「第 6 章 dsedit の使用」を参照してください。また、使用しているプラットフォームの『インストール・ガイド』と『設定ガイド』も参照してください。
参照	<p>ユーティリティ dscp</p>

extractjava

説明	<p>保持された JAR とそれに含まれるクラスを Adaptive Server からクライアント・ファイルにコピーします。\$SYBASE/\$SYBASE_OCS/bin にあります。</p> <p>(Windows の場合) このユーティリティは extrjava.exe で、%SYBASE%\\$SYBASE_OCS%¥bin にあります。</p>
構文	<pre>extractjava (Windows では extrjava) -j jar_name -f file_name [-S server_name] [-U user_name] [-P password] [-D database_name] [-I interfaces_file] [-a display_charsef] [-J client_charsef] [-z language] [-t timeout] [-v]</pre> <p>または</p> <pre>extractjava -v</pre>
パラメータ	<p>-j jar_name 転送元データベースの保持された JAR に割り当てられた名前を指定します。</p> <p>-f file_name 転送先のクライアント・ファイルの名前を指定します。</p> <p>-S server_name サーバの名前を指定します。</p> <p>-U user_name Adaptive Server ログイン名を指定します。-U フラグとパラメータを省略した場合、またはパラメータなしで -U フラグを指定した場合は、Adaptive Server は現在のユーザのオペレーティング・システムへのログイン名を使用します。</p> <p>-P password Adaptive Server のパスワードを指定します。-P フラグとパラメータを省略すると、extractjava はパスワードを入力するよう要求します。パスワードなしで -P フラグを指定すると、null パスワードが使用されます。</p>

-D *database_name*

JAR をインストールするデータベース名を指定します。**-D** フラグを省略した場合、またはパラメータなしで **-D** フラグを指定した場合は、ユーザのデフォルト・データベースが使用されます。

-I *interfaces_file*

Adaptive Server に接続するときに検索する *interfaces* ファイルの名前とロケーションを指定します。**-I** フラグとパラメータを省略した場合、またはパラメータなしで **-I** フラグを指定した場合は、SYBASE 環境変数によって指定されたディレクトリにある *interfaces* ファイルが使用されます。

-a *display_charset*

文字セットがサーバとは異なるマシンで *extractjava* を使用できます。**-a** を **-J** とともに使用して、変換に必要な文字セット変換ファイル (*.xlt* ファイル) を指定します。**-J** を使用しないで **-a** を使用するの、クライアントの文字セットがデフォルトの文字セットと同じである場合だけです。

-J *client_charset*

クライアントで使用する文字セットを指定します。*extractjava* は、フィルタを使用して *client_charset* と Adaptive Server の文字セット間で入力を変換します。

-J*client_charset* は、クライアントで使用する文字セットである *client_charset* とサーバの文字セット間の変換を Adaptive Server に要求します。

-J に引数を指定しないと、文字セット変換が無効になります。クライアントとサーバが同じ文字セットを使用する場合に、このパラメータを使用してください。

-J を省略すると、文字セットはプラットフォームのデフォルトに設定されます。デフォルトの文字セットは、クライアントが使用している文字セットと必ず同じであるとはかぎりません。文字セットとそれに関連するフラグの詳細については、『システム管理ガイド』を参照してください。

-z *language*

extractjava のプロンプトとメッセージを表示するときの代替言語の名前を指定します。**-z** フラグを指定しないと、*extractjava* はサーバのデフォルト言語を使用します。言語はインストール時に Adaptive Server に追加できます。インストール後でも、[langinstall](#) ユーティリティまたは [sp_addlanguage](#) ストアド・プロシージャを使用して言語を追加できます。

-t timeout

SQL コマンドがタイムアウトするまでの秒数を指定します。タイムアウトを指定しないと、コマンドは無期限に実行されます。これは、`extractjava` 内から発行されたコマンドに影響するもので、接続時間には影響しません。`extractjava` にログインするためのデフォルトのタイムアウトは、60 秒です。

-v

`extractjava` のバージョン番号と著作権メッセージを表示して、終了します。

例 `employees` という名前の JAR に関連付けられたクラスを、クライアント・ファイル `newaddr.jar` にダウンロードします。

- UNIX の場合：

```
extractjava -j employees -f '/home/usera/jars/addr.jar' -new
```

- Windows の場合：

```
extrjava -j employees -f '¥home¥usera¥jars¥addr.jar' -new
```

使用法

- SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、`extractjava` を使用してください。
- ターゲット・クライアント・ファイルがすでに存在する場合、`extractjava` によってファイルの内容は上書きされます。
- パラメータ・フラグ `-f`、`-j`、`-S`、`-U`、`-P`、`-D`、`-I` を指定する場合、フラグとその後のパラメータの間にはスペースを入れても入れなくてもかまいません。
- `extractjava` を実行すると、`sysxtypes` に排他ロックが設定されます。
- `-jar` を指定すると、`sysjars` に排他テーブル・ロックが設定されます。

パーミッション

`extractjava` を使用できるのは、システム管理者とデータベース所有者のみです。

使用するテーブル

`sysjars`, `sysxtypes`

参照

コマンド `remove java`

マニュアル データベースで Java を使用可能にした場合に、このユーティリティをどのように使用するかについては、『Adaptive Server Enterprise における Java』を参照してください。

システム・プロシージャ `sp_helpjava`

ユーティリティ [installjava](#)

installjava

説明

クライアント・ファイルから Adaptive Server に JAR をインストールする。このユーティリティは `$$SYBASE/$SYBASE_OCS/bin` にあります。

(Windows の場合) このユーティリティは `instjava.exe` で、`%SYBASE%\$SYBASE_OCS%\bin` にあります。

構文

```
installjava
  -f file_name
  [-new | -update ]
  [-j jar_name ]
  [-S server_name ]
  [-U user_name ]
  [-P password ]
  [-D database_name ]
  [-I interfaces_file ]
  [-a display_charset ]
  [-J client_charset ]
  [-z language ]
  [-t timeout ]
  [-v]
```

または

```
installjava -v
```

パラメータ

-f *file_name*

データベースにインストールするクラスが入っているソース・ファイルの名前を指定します。

-new | -update

ソース・ファイル内のクラスがデータベースにすでに存在するかどうかを指定します。次のオプションがあります。

- **-new** — 既存のクラスと同じ名前のクラスをインストールすることはできません。
- **-update** — 既存のクラスと同じ名前のクラスをインストールし、新しくインストールしたクラスに置き換えることができます。

-j *jar_name*

データベースにインストールするクラスが入っている JAR の名前を指定します。このパラメータを指定すると、JAR がデータベースに保存され、その中のクラスに関連付けられます。

-S *server_name*

サーバの名前を指定します。

-U *user_name*

Adaptive Server ログイン名を指定します。**-U** フラグとパラメータを省略した場合、またはパラメータなしで **-U** フラグを指定した場合は、Adaptive Server は現在のユーザのオペレーティング・システムへのログイン名を使用します。

-P *password*

Adaptive Server のパスワードを指定します。**-P** フラグとパラメータを省略すると、`installjava` はパスワードを入力するよう要求します。パスワードなしで **-P** フラグを指定すると、`null` パスワードが使用されます。

-D *database_name*

JAR をインストールするデータベースの名前を指定します。**-D** フラグを省略した場合、またはパラメータなしで **-D** フラグを指定した場合は、ユーザのデフォルト・データベースが使用されます。

-I *interfaces_file*

Adaptive Server に接続するときに検索する `interfaces` ファイルの名前とロケーションを指定します。**-I** フラグとパラメータを省略した場合、またはパラメータなしで **-I** フラグを指定した場合は、`SYBASE` 環境変数によって指定されたディレクトリにある `interfaces` ファイルが使用されます。

-a *display_charset*

文字セットがサーバとは異なるマシンで `installjava` を使用できません。**-a** を **-J** とともに使用して、変換に必要な文字セット変換ファイル (`.xlt` ファイル) を指定します。**-J** を使用しないで **-a** を使用するのには、クライアントの文字セットがデフォルトの文字セットと同じである場合だけです。

-J *client_charset*

クライアントで使用する文字セットを指定します。installjava は、フィルタを使用して *client_charset* と Adaptive Server の文字セット間で入力を変換します。

-J*client_charset* は、クライアントで使用する文字セットである *client_charset* とサーバの文字セット間の変換を Adaptive Server に要求します。

-J に引数を指定しないと、文字セット変換が無効になります。クライアントとサーバが同じ文字セットを使用する場合に、このパラメータを使用してください。

-J を省略すると、文字セットはプラットフォームのデフォルトに設定されます。デフォルトの文字セットは、クライアントが使用している文字セットと必ず同じであるとはかぎりません。文字セットとそれに関連するフラグの詳細については、『システム管理ガイド』を参照してください。

-z *language*

installjava のプロンプトとメッセージを表示するときの代替言語の名前を指定します。**-z** フラグを指定しないと、installjava はサーバのデフォルト言語を使用します。言語はインストール時に Adaptive Server に追加できます。インストール後でも、[langinstall](#) ユーティリティまたは **sp_addlanguage** ストアド・プロシージャを使用して言語を追加できます。

-t *timeout*

SQL コマンドがタイムアウトするまでの秒数を指定します。タイムアウトを指定しないと、コマンドは無期限に実行されます。これは、installjava 内から発行されたコマンドに影響するもので、接続時間には影響しません。installjava にログインするためのデフォルトのタイムアウトは、60 秒です。

-v

installjava のバージョン番号と著作権メッセージを表示して、終了します。

例 **例 1** *addr.jar* とそのクラスをインストールしますが、JAR とクラスの関連付けは保持しません。

```
installjava -f '/home/usera/jars/addr.jar' -new
```

Windows の場合：

```
instjava -f '¥home¥usera¥jars¥addr.jar' -new
```

例 2 *addr.jar* を再インストールし、そのクラスを *employees* という JAR 名に関連付けます。

```
installjava -f '/home/usera/jars/addr.jar' -update -j employees
```

Windows の場合 :

```
instjava -f '%home%usera%jars%addr.jar' -update -j employees
```

使用法

- SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、*installjava* を使用してください。
- インストール済みのクラスは、すべてのユーザが参照できます。
- パラメータ・フラグ *-f*、*-j*、*-S*、*-U*、*-P*、*-D*、*-I* を指定する場合、フラグとその後のパラメータの間にはスペースを入れても入れなくてもかまいません。

新規 JAR の追加

- *-jar* オプションで *new* を使用したとき、その名前の JAR がデータベースにすでに存在する場合は、例外が発生します。
- ソース JAR 内のクラスと同じ名前のクラスがデータベースにすでに存在する場合は、例外が発生します。

JAR とクラスの更新

警告! クラスの修正バージョンを再インストールすることによって、カラム・データ型として使用しているクラスを変更する場合は、変更したクラスが読み出せて、そのクラスをデータ型として使用しているテーブル内の既存のオブジェクト(ロー)を使用できるかどうかを確認してください。できなかった場合は、クラスを再インストールしないと現在のオブジェクトにアクセスできません。

- *-jar* オプションを指定して *-update* を使用すると、次のようになります。
 - データベース内のクラスのうちターゲット JAR に関連付けられたすべてのクラスがデータベースから削除され、代わりにソース JAR ファイル内のクラスがインストールされます。
 - ソース JAR ファイル内に、データベースにはインストールされていても JAR には付加されていないクラスがある場合、そのクラスがソース JAR ファイルからデータベースにインストールされ、付加されていないクラスは削除されます。

- `-jar` オプションを指定せずに、`-update` を使用すると、次のようになります。
 - ソース JAR ファイル内のクラスが、同じ名前の付加されていないクラスと置き換わります。
 - ソース JAR ファイル内のインストールされていないクラスは、付加されていないクラスとしてデータベースにインストールされます。
- SQLJ プロシージャまたは関数によって参照されているインストール済みのクラスに代わる新しい JAR をインストールする場合は、新しくインストールされたクラスに SQLJ ルーチンの有効なシグニチャがあることを確認してください。シグニチャが無効の場合、SQLJ ルーチンが呼び出されると例外が発生します。

ロック

- `installjava` を実行すると、`sysstypes` に排他ロックが設定されます。
- `-jar` を指定すると、`sysjars` に排他テーブル・ロックが設定されます。

パーミッション

`installjava` を使用できるのは、システム管理者とデータベース所有者のみです。

使用するテーブル

`sysjars`, `sysstypes`

参照

コマンド `remove java`

マニュアル データベースで Java を使用可能にした場合に、このユーティリティをどのように使用するかについては、『Adaptive Server Enterprise における Java』を参照してください。

システム・プロシージャ `sp_helpjava`

ユーティリティ [extractjava](#)

isql

説明

Adaptive Server の Interactive SQL パーサです。
 \$SYBASE/\$SYBASE_OCS/bin にあります。

(Windows の場合) このユーティリティは isql.exe で、%SYBASE%\%SYBASE_OCS%\bin にあります。

構文

```
isql [-b] [-e] [-F] [-p] [-n] [-v] [-W] [-X] [-Y] [-Q]
      [-a display_charsef]
      [-A packet_size]
      [-c cmdend]
      [-D database]
      [-E editor]
      [-h header]
      [-H hostname]
      [-i inputfile]
      [-I interfaces_file]
      [-J client_charsef]
      [-K keytab_file]
      [-l login_timeout]
      [-m errorlevel]
      [-o outputfile]
      [-P password]
      [-R remote_server_principal]
      [-s colseparator]
      [-S server_name]
      [-t timeout]
      -U username
      [-V [security_options]]
      [-w columnwidth]
      [-z locale_name]
      [-Z security_mechanism]
      [--conceal]
```

注意 isql 内から実行できるコマンドのリストと説明については、[表 7-1](#) を参照してください。

パラメータ

- b
 テーブル・ヘッダの出力表示を無効にします。
- e
 入力内容をエコーします。

- F**
FIPS フラガを使用可能にします。**-F** パラメータを指定した場合、サーバは SQL 規格外のコマンドを検出するとメッセージを返します。このオプションは SQL 拡張機能を無効にするではありません。ANSI SQL 以外のコマンドを発行すると、処理は完了します。
- p**
パフォーマンスの統計値を出力します。
- n**
-e とともに使用した場合、出力ファイルにエコーされた入力行から、行番号とプロンプト記号 (>) を削除します。
- v**
isql のバージョン番号と著作権メッセージを表示して、終了します。
- X**
クライアント側のパスワード暗号化を使用して、サーバへのログイン接続を開始します。**-X** は、拡張パスワード暗号化による接続とパスワード暗号化による接続の両方を有効にします。プレーン・テキスト形式のパスワードを使用して再接続することはありません。isql (クライアント) は、パスワードの暗号化が必要であることをサーバに通知します。サーバは、isql がパスワードを暗号化するために使う暗号化キーを返送し、パスワードを受け取ると、そのキーを使用してそのパスワードを確認します。
- isql がクラッシュすると、パスワードを含むコア・ファイルが作成されます。暗号化オプションを使用していない場合、パスワードは、コア・ファイルにプレーン・テキストで表示されます。暗号化オプションを使用した場合、パスワードは表示されません。
- W**
拡張パスワード暗号化とパスワード暗号化の両方のネゴシエーションを無効にします。
- Y**
Adaptive Server に、連鎖トランザクションを使用するよう伝えます。
- Q**
クライアントにフェールオーバー・プロパティを提供します。詳細については『高可用性システムにおける Sybase フェールオーバーの使用』を参照してください。

-a display_charset

isql を実行しているマシンの文字セットとは異なる文字セットを使用する端末から、isql を実行します。**-a** を **-J** とともに使用して、変換に必要な文字セット変換ファイル (.xlt ファイル) を指定します。**-J** を使用しないで **-a** を使用するの、クライアントの文字セットがデフォルトの文字セットと同じである場合だけです。

注意 `ascii_7` 文字セットは、すべての文字セットと互換性があります。Adaptive Server の文字セットとクライアントの文字セットのどちらかが `ascii_7` に設定されていれば、すべての7ビット ASCII 文字が変更されることなくクライアントとサーバの間で受け渡されます。その他の文字セットを使用している場合は、変換エラーが発生します。文字セット変換の詳細については、『システム管理ガイド』を参照してください。

-A packet_size

この isql セッションに使用するネットワーク・パケットのサイズを指定します。たとえば、この isql セッションのパケット・サイズを 4096 バイトに設定するには、次のように入力します。

```
isql -A 4096
```

- ネットワーク・パケット・サイズをチェックするには、次のように入力します。

```
select * from sysprocesses
```
- 値は `network_pktsz` という見出しの下に表示されます。
- `size` は、`default network packet size` 設定変数と `maximum network packet size` 設定変数の間の値で、512 の倍数でなければなりません。
- `readtext` や `writetext` のような大量の I/O 処理を行うオペレーションを実行する場合は、パケット・サイズをデフォルトより大きな値に設定してください。
- Adaptive Server のパケット・サイズの設定や変更は、リモート・プロシージャ・コールのパケット・サイズに影響しません。

-c cmdend

コマンド・ターミネータを変更します。デフォルトでは、行に“go”と入力するだけで、コマンドが終了し、それらのコマンドが送信されます。コマンド・ターミネータを変更する場合、SQL の予約語および制御文字は使用できません。

-D database

isql セッションを開始するデータベースを選択します。

-E editor

デフォルトのエディタである vi 以外のエディタを指定します。エディタを呼び出すには、isql で行の最初の単語としてエディタ名を入力します。

-h headers

カラム見出しから次のカラム見出しまでの間に出力されるローの数を指定します。デフォルトでは、クエリ結果のセットごとに 1 回だけ見出しが出力されます。

-H hostname

クライアント・ホスト名を設定します。

-i inputfile

isql への入力に使用するオペレーティング・システム・ファイルの名前を指定します。このファイルには、コマンド・ターミネータを含むようにしてください(デフォルトでは “go”)。

- パラメータを次のように指定すると、< *inputfile* を指定した場合と同じになります。

`-i inputfile`

- -i を使用し、コマンド・ラインにパスワードを指定しない場合、isql はパスワードを入力するよう要求します。
- < *inputfile* を使用して、コマンド・ラインにパスワードを指定しない場合は、入力ファイルの最初の行にパスワードを指定してください。

-I interfaces_file

Adaptive Server に接続するときに検索する *interfaces* ファイルの名前とロケーションを指定します。-I を指定しない場合、isql は SYBASE 環境変数によって指定されたディレクトリにある *interfaces* という名前のファイルを探します。

-J *client_charset*

クライアントで使用する文字セットを指定します。**-J *client_charset*** は、クライアントで使用する文字セットである *client_charset* とサーバの文字セット間の変換を Adaptive Server に要求します。フィルタによって、*client_charset* と Adaptive Server の文字セット間で入力に変換されます。

-J に引数を指定しない場合、文字セット変換は NULL に設定されます。この場合、変換は行われません。クライアントとサーバが同じ文字セットを使用する場合に、このパラメータを使用してください。

-J を省略すると、文字セットはプラットフォームのデフォルトに設定されます。デフォルトの文字セットは、クライアントが使用する文字セットと同じであるとはかぎりません。文字セットとそれに関連するフラグの詳細については、『システム管理ガイド 第1巻』の「クライアント/サーバの文字セット変換の設定」を参照してください。

-K *keytab_file*

DCE での認証に使用する keytab ファイルのパスを指定します。

-l *login_timeout*

Adaptive Server に接続する場合の最大タイムアウト値を指定します。デフォルトは 60 秒。この値は、サーバがログインの要求に回答するのを *isql* が待つ時間に対してだけ影響します。実行中のコマンドのタイムアウト時間を指定するには、**-t *timeout*** パラメータを使用します。

-m *errorlevel*

エラー・メッセージの表示をカスタマイズします。指定した重大度以上のレベルのエラーでは、メッセージ番号、ステータス、およびエラー・レベルが表示されますが、エラー・テキストは表示されません。指定した重大度より低いレベルのエラーでは、何も表示されません。

-o *outputfile*

isql からの出力を保管するオペレーティング・システム・ファイルの名前を指定します。**-o *outputfile*** のような形式での指定の仕方は、**> *outputfile*** とする場合と同じです。

-P *password*

Adaptive Server のパスワードを指定します。**-P** フラグを指定しない場合、*isql* はパスワードを入力するよう要求します。パスワードが NULL の場合は、パスワードを指定せずに **-P** フラグを使用してください。

-R remote_server_principal

セキュリティ・メカニズムに指定されたりモート・サーバのプリンシパル名を指定します。デフォルトでは、サーバのプリンシパル名はサーバのネットワーク名 (-S パラメータまたは DSQUERY 環境変数で指定) と一致します。サーバのプリンシパル名とネットワーク名が一致しない場合は、-R パラメータを使用してください。

-s colseparator

カラム・セパレータ文字をリセットします。デフォルト・カラム・セパレータ文字はブランクです。オペレーティング・システムに対して特別な意味を持つ文字 (“|”、“;”、“&”、“<”、“>” など) を使用するには、それらを引用符で囲むか、前に円記号を付けます。

カラム・セパレータは、各ローの各カラムの先頭と末尾に表示されます。

-S server_name

接続する Adaptive Server の名前を指定します。isql は、interfaces ファイル内でこの名前のエントリを探します。引数なしで -S を指定した場合、isql は SYBASE という名前のサーバを探します。-S を指定しない場合、isql は DSQUERY 環境変数によって指定されたサーバを探します。

-t timeout

SQL コマンドがタイムアウトするまでの秒数を指定します。タイムアウトを指定しないと、コマンドは無期限に実行されます。これは、isql 内から発行されたコマンドに影響するもので、接続時間には影響しません。isql にログインするためのデフォルトのタイムアウトは 60 秒です。

-U username

ログイン名を指定します。ログイン名では大文字と小文字が区別されます。

-V security_options

ネットワーク・ベースのユーザ認証を指定します。このオプションを使用する場合、ユーザはユーティリティを実行する前にネットワークのセキュリティ・システムにログインする必要があります。この場合、ユーザは **-U** オプションにネットワーク・ユーザ名を指定し、**-P** オプションによって指定されたパスワードは無視されます。

-V の後にキー文字オプションの *security_options* 文字列を続けると、他のセキュリティ・サービスを有効にできます。これらのキー文字は、以下のとおりです。

- **c** – データ機密性サービスを有効にする
- **i** – データ整合性サービスを有効にする
- **m** – 接続の確立に相互認証を有効にする
- **o** – データ・オリジン・スタンプング・サービスを有効にする
- **q** – 順序不整合の検出を有効にする
- **r** – データ・リプレイの検出を有効にする

-w columnwidth

出力画面の幅を設定します。デフォルトでは、80 文字です。出力行が画面幅いっぱいになった場合は、複数の行に分割されます。

-z locale_name

isql のプロンプトとメッセージの表示に使用する代替言語の公式名です。**-z** フラグを指定しないと、*isql* はサーバのデフォルト言語を使用します。言語はインストール時に *Adaptive Server* に追加できます。インストール後でも、*langinstall* ユーティリティ (Windows では *langinst*) または *sp_addlanguage* ストアド・プロシージャを使用して言語を追加できます。

-Z security_mechanism

接続で使用するセキュリティ・メカニズムの名前を指定します。

セキュリティ・メカニズム名は *Sybase* インストール・ディレクトリの中の *ini* サブディレクトリにある *libtcl.cfg* 設定ファイルで定義されます。*security_mechanism* の名前が指定されていない場合は、デフォルトのメカニズムが使用されます。セキュリティ・メカニズム名の詳細については、『*Open Client/Server 設定ガイド*』の *libtcl.cfg* ファイルの説明を参照してください。

--conceal

isql セッション中の入力内容を隠します。--conceal オプションは、パスワードなどの機密情報を入力するときに役立ちます。--conceal オプションの構文は次のとおりです。

```
isql --conceal [':?' | 'wildcard']
```

wildcard は 32 バイトの変数で、最大長は 80 文字です。

注意 --conceal はバッチ・モードでは無視されます。

例 **例 1** これで、クエリを編集できるテキスト・ファイルの状態になります。ファイルに書き込みを行って保存すると、isql に戻ります。クエリが表示されるので、行に “go” とだけ入力して実行してください。

```
isql -Ujoe -Pabracadabra
1> select *
2> from authors
3> where city = "Oakland"
4> vi
```

例 2 **reset** はクエリ・バッファをクリアします。quit でオペレーティング・システムに戻ります。

```
isql -Ualma
Password:
1> select *
2> from authors
3> where city = "Oakland"
4> reset
1> quit
```

例 3 **roman8** 文字セットを使用するサーバに対して、Macintosh から isql を実行していることを伝えます。

```
isql -a mac -J roman8
```

例 4 ストア ID 7896 の pubs2 データベースの出力に “#” を使用して、カラム・セパレータを作成します。

```
isql -Uasa -P -s#
1> use pubs2
2> go
1> select * from sales where stor_id = "7896"
#stor_id#ord_num          #date          #
#-----#-----#-----#
#7896   #124152          #      Aug 14 1986 12:00AM#
#7896   #234518          #      Feb 14 1991 12:00AM#
```


(2 rows affected)

例 5 入力したパスワードを表示させないで、パスワードを変更します。次の例では、プロンプトのラベルとして “old” と “new” を使用します。

```
$ isql -Uguest -Pguest -Smyase --conceal
sp_password
:?: old
/
:?:? new
-----
old
new
Confirm new
Password correctly set.
```

(Return status 0)

例 6 現在のユーザに対して役割をアクティブ化します。次の例では、カスタム・ワイルドカード、プロンプトのラベルとして “role”、 “password” を使用します。

```
$ isql -UmyAccount --conceal '*'
Password:
set role
* role
with passwd
** password
on
go

role
password
Confirm password
```

使用法

- スレッド・ドライバを使用する場合は、`isql_r` の構文を使用します。
- IBM プラットフォームでスレッド・ドライバを使用する場合は、`isql` のこの構文を使用します。
- SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、`isql` を使用してください。
- [表 7-1](#) は、対話型 `isql` 内から実行できるコマンドを示します。

表 7-1 : isql 内で使用できるコマンド

コマンド	説明
:r filename	オペレーティング・システム・ファイルをコマンド・バッファに読み込みます。コマンド・ターミネータをファイルに含めないでください。編集が終了したら、コマンド・ターミネータを対話的に入力してください。
:R filename	オペレーティング・システム・ファイルをコマンド・バッファに読み込んでから表示します。 コマンド・ターミネータをファイルに含めないでください。編集が終了したら、コマンド・ターミネータを対話的に入力してください。
use database_name	現在のデータベースを変更します。
!! os_command	オペレーティング・システム・コマンドを実行します。行の先頭に配置します。
> file_name	T-SQL コマンドの出力を file_name にリダイレクトします。 次の例では、サーバのバージョンを file_name に挿入します。 select @@version go > file_name
>> file_name	T-SQL コマンドの出力を file_name に追加します。 次の例では、サーバのバージョンを file_name に追加します。 select @@version go >> file_name
command	T-SQL コマンドの出力を外部の command にパイプ接続します。 次の例では、sp_who によって作成されたリストにあるすべての “sa” インスタンスを検索します。 sp_who go grep sa
vi (UNIX) edit (Windows)	デフォルト・エディタを呼び出します。
reset	クエリ・バッファをクリアします。
quit または exit	isql を終了します。

- 5701 サーバ・メッセージ(「データベースが変更されました」)は、ログイン後または use database コマンドの発行後には表示されません。
- エラー・メッセージのフォーマットは、isql の前のバージョンと異なっています。これらのメッセージの内容に基づいたルーチンを実行するスクリプトは、修正が必要な場合があります。
- -X パラメータを指定した場合は、サーバの機能に応じてパスワードの有効な接続が進行します。

- サーバで拡張パスワード暗号化とパスワード暗号化をどちらも処理できる場合は、拡張パスワード暗号化によるネゴシエーションが使用されます。
- サーバでパスワード暗号化だけを処理できる場合は、パスワード暗号化によるネゴシエーションが使用されます。
- サーバでパスワード暗号化も拡張パスワード暗号化も処理できない場合は最初の接続試行が失敗し、クライアントはプレーン・テキスト形式のパスワードを使用して再接続を試みます。
- `isql` を対話的に使用するには、オペレーティング・システムのプロンプト画面で `isql` コマンド (およびオプションのパラメータ) を入力します。 `isql` プログラムは、SQL コマンドを受け取り、Adaptive Server に送信します。結果は、フォーマットされ、標準出力に出力されます。 `isql` を終了するには、`quit` または `exit` を使用します。
- `-c` パラメータを使用している場合は、行の先頭にデフォルト・コマンド・ターミネータである `go` か他のコマンド・ターミネータを入力し、コマンドを終了します。コマンド・ターミネータに続けて、コマンドを実行する回数を整数で指定できます。たとえば、あるコマンドを 100 回実行する場合は、次のように入力します。

```
select x = 1  
go 100
```

結果は、実行の終了時に 1 回表示されます。

- コマンド・ラインにオプションを複数回入力した場合、`isql` は最後の値を使用します。たとえば、次のコマンドを入力した場合、`-c` の 2 番目の値 “send” によって最初の値 “.” は無効になります。

```
isql -c"." -csend
```

これによって、設定したすべてのエイリアスを無効にすることができます。

- 現在のクエリ・バッファに対してエディタを呼び出すには、行の先頭にエディタ名を入力します。EDITOR 環境変数でエディタを指定して、優先する呼び出し可能なエディタを定義します。EDITOR が定義されていない場合、デフォルトは `vi` (UNIX の場合)、`edit` (Windows の場合) です。

たとえば、EDITOR 環境変数を “`emacs`” に設定している場合は、行の最初の単語に “`emacs`” を使用して、`isql` から `emacs` を呼び出します。

- 既存のクエリ・バッファをクリアするには、行に **reset** とだけ入力します。isql は、未処理の入力内容をすべて破棄します。また、行の任意の位置で [Ctrl + c] を押して現在のクエリをキャンセルし、isql プロンプトに戻ることもできます。
- 次のような isql で、実行するクエリを含むオペレーティング・システム・ファイルを読み込んでください。

```
isql -U alma -Ppassword < input_file
```

ファイルには、コマンド・ターミネータが必要です。結果は端末に表示されます。次のようにして、クエリを含むオペレーティング・システム・ファイルを読み込み、結果を別のファイルに書き込むことができます。

```
isql -U alma -Ppassword < input_file > output_file
```

- isql のフラグを使用する場合には、大文字と小文字を区別してください。
- isql は float または real データを丸めて、小数点以下 6 桁までを表示します。
- isql によって Adaptive Server に渡される Transact-SQL 文には、コメントを入れることができます。コメントは、次の例に示すように “/*” と “*/” で囲みます。

```
select au_lname, au_fname
/*retrieve authors' last and first names*/
from authors, titles, titleauthor
where authors.au_id = titleauthor.au_id
and titles.title_id = titleauthor.title_id
/*this is a three-way join that links authors
**to the books they have written.*/
```

go コマンドをコメントにする場合は、コマンドが行の先頭にならないようにします。たとえば、go コマンドをコメントにする場合は次のようにしてください。

```
/*
**go
*/
```

次のようにはしないでください。

```
/*
go
*/
```

- isql では、ロケール環境に関係なく、日付フォーマットの順序が月、日、年の順で定義されています (mm dd yyyy hh:mmAM (または PM))。このデフォルトの順序を変更するには、convert 関数を使用します。
- isql セッションでは、デフォルトのプロンプト・ラベルはデフォルトのワイルドカード :? または *wildcard* の値のいずれかになります。プロンプトのラベルをカスタマイズするには、1 単語の文字列をワイルドカードの後に指定します。最大長は 80 文字です。プロンプトのラベルを 1 単語より多く指定した場合、最初の単語より後にある文字は無視されます。

注意 isql セッションで、isql が :? または *wildcard* の値をワイルドカードとして認識するのは、これらの文字が isql 行の先頭に配置されている場合だけです。

参照

コマンド create schema、set

データ型 真数値データ型

マニュアル isql の詳細については、「[第2章 コマンド・ラインからの Interactive isql の使用](#)」を参照してください。設定パラメータ default network packet size および maximum network packet size の詳細については、『ASE リファレンス・マニュアル』を参照してください。

関数 convert

システム ESP xp_sendmail

システム・プロシージャ sp_addlanguage、sp_addlogin、sp_addrmotellogin、sp_add_resource_limit、sp_bindexclass、sp_configure、sp_defaultlanguage、sp_droplanguage、sp_helplanguage、sp_processmail、sp_remoteoption、sp_serveroption、sp_showcontrolinfo、sp_unbindexclass、sp_volchanged

langinstall

説明

Adaptive Server に新しい言語をインストールします。
\$SYBASE/\$SYBASE_ASE/bin にあります。

(Windows の場合) このユーティリティは langinst.exe で、%SYBASE%\\$SYBASE_ASE%\bin にあります。

構文

```
langinstall  
    [-S server]  
    [-U user]  
    [-P password]  
    [-R release_number]  
    [-I path]  
    language  
    character_set
```

または

```
langinstall -v
```

パラメータ

-S server

接続する Adaptive Server の名前を指定します。**-S** を指定しない場合、**langinstall** は DSQUERY 環境変数によって指定されたサーバを使用します。DSQUERY が設定されていない場合、**langinstall** は SYBASE という名前のサーバに接続しようとします。

-U user

ログイン名を指定します。ログイン名では大文字と小文字が区別されます。

-P password

システム管理者 (“sa” アカウント) のパスワードを指定します。**-P** を省略すると、**langinstall** は “sa” アカウントのパスワードを入力するよう要求します。

-R *release_number*

`master.sysmessages` のメッセージをアップグレードするために使用するリリース番号を *n.n.n* の形式で指定します。`langinstall` (Windows では `langinst`) が失敗した場合、ユーザ・エラーが発生した場合、または `sysmessages` 内のメッセージが古くなっていると考えられるようなときにだけ、**-R** を使用します。

-R パラメータを指定すると、`langinstall` は現在のリリースより前のリリースからメッセージを収集します。`langinstall` は、収集した既存のメッセージとこれからインストールされるメッセージを比較し、それらが異なっている場合は、メッセージを置き換えます。

たとえば、現在のバージョンが 15.0 で、前のバージョンが 12.5 であるとき、`sysmessages` が適切でないと思われる場合は、**-R 12.5** `langinstall` と指定して、前のバージョン (この場合は 12.5) のメッセージを `syslanguages.upgrade` カラムに追加します。これにより、Adaptive Server 12.5 のすべてのメッセージがインストールされます。

-I *path*

Adaptive Server に接続するときに `langinstall` が検索する `interfaces` ファイル (Windows では `sql.ini` ファイル) の名前とロケーションを指定します。**-I** を指定しない場合、`langinstall` は SYBASE 環境変数によって指定されたディレクトリにある `interfaces` ファイルを使用します。SYBASE 環境変数が設定されていない場合、`langinstall` は、デフォルトの SYBASE ディレクトリを探します。

language

インストールする言語の公式名です。言語は必ず指定してください。

character_set

Adaptive Server のデフォルトの文字セット名です。*character_set* は、その言語用のローライゼーション・ファイルがあるディレクトリ名を示します。公式言語用のローライゼーション・ファイルである `common.loc` と `server.loc` は、文字セットのディレクトリ (UNIX プラットフォームでは `$SYBASE/locales/language/character_set`、Windows では `%SYBASE%\locales %language %character_set`) にあります。文字セットは必ず指定してください。

-v

`langinstall` のバージョン番号と著作権メッセージを表示して、終了します。

使用法

- Adaptive Server インストール・プログラムは、新規インストールとアップグレードのいずれにおいても langinstall を自動的に実行します。
- langinstall は、次のことを実行します。
 - sp_addlanguage を使用して、指定された言語に対応する情報を master.syslanguages に追加します。言語がすでに存在する場合、langinstall は、syslanguages 内の該当するローを更新します。
 - master.sysmessages に対するエラー・メッセージの追加、更新、および削除を必要に応じて行います。
 - 新しいリリース番号を挿入して syslanguages.update を更新します。
- langinstall は、ローカライゼーション・ファイルの中で使用するセクション内のエントリを有効にします。エントリが1つでも足りない場合には、langinstall はエラー・メッセージを表示し、syslanguages に言語を追加しません。
- langinstall は、使用する common.loc および server.loc の各ローカライゼーション・ファイルのバージョン番号を比較します。この2つのバージョン番号が異なる場合、警告メッセージを表示します。syslanguages.upgrade は、常に server.loc のバージョン番号に従って設定されます。

パーミッション

langinstall を実行できるのは、“sa” アカウントを使用するシステム管理者だけです。

使用するテーブル

master.dbo.syslanguages、master.dbo.sysmessages

参照

システム・プロシージャ sp_addlanguage、sp_addlogin、sp_configure、sp_defaultlanguage、sp_droplanguage、sp_helplanguage

ユーティリティ defncopy, srvbuild

optdiag

説明	<p>オブティマイザ統計値を表示します。または、更新された統計値をシステム・テーブルにロードします。optdiag は <code>\$\$SYBASE/\$SYBASE_ASE/bin</code> にあります。</p> <p>(Windows の場合) このユーティリティは <code>optdiag.exe</code> で、<code>%SYBASE%\\$SYBASE_ASE%\bin</code> にあります。</p>
構文	<pre>optdiag [binary] [simulate] statistics { -i <i>input_file</i> <i>database</i>.[<i>owner</i>].[<i>table</i>.[<i>column</i>]] } [-o <i>output_file</i>] [-U <i>user_name</i>] [-P <i>password</i>] [-T <i>trace_value</i>] [-l <i>interfaces_file</i>] [-S <i>server</i>] [-v] [-h] [-s] [-z <i>language</i>] [-J <i>client_character_set</i>] [-a <i>display_charset</i>]</pre>
パラメータ	<p>binary</p> <p>統計値を人間が判読できる形式とバイナリ形式で抽出します。入力ファイル (<code>-i <i>input_file</i></code>) で使用すると、バイナリ形式の統計値がシステム・テーブルにロードされます。</p> <p>simulate</p> <p>optdiag がシミュレートされた統計値を表示またはロードするように指定します。『パフォーマンス&チューニング・シリーズ』を参照してください。</p> <p>-i <i>input_file</i></p> <p>optdiag への入力に使用するオペレーティング・システム・ファイルの名前を指定します。入力ファイルを指定すると、optdiag は指定したファイル内の値を使用してテーブルまたはカラムのオブティマイザ統計値を更新します(「入力モード」ともいいます)。</p> <p>database</p> <p>表示する統計値が保存されているデータベースの名前を指定します。入力モードでは、optdiag は入力ファイルに指定されたデータベース名を使用し、コマンド・ラインに指定されたデータベース名は無視します。</p>

owner

テーブル所有者の名前を指定します。

- 表示モードでは、所有者を指定しないでテーブル名を指定した場合、**optdiag** はテーブルのすべての所有者について出力を表示します。
- 入力モードでは、**optdiag** はコマンド・ラインに指定されたテーブル所有者を無視し、入力ファイル内の値を使用します。

table

統計値を調べるテーブルの名前を指定します。

- 所有者名またはテーブル名を指定しない場合、**optdiag** はデータベース内のすべてのテーブルに関する統計値を表示します。
- 所有者名を指定し、テーブル名を指定しない場合、**optdiag** は指定した所有者に属するすべてのテーブルを表示します。
- 入力モードでは、**optdiag** はコマンド・ラインに指定されたテーブル名を無視し、入力ファイル内の値を使用します。

column

調査するカラムの名前を指定します。

- カラム名を指定しない場合、**optdiag** はテーブルに関するすべての統計値を表示します。
- 入力モードでは、**optdiag** はコマンド・ラインに指定されたカラム名を無視し、入力ファイル内の値を使用します。

-o output_file

optdiag からの出力を保管するオペレーティング・システム・ファイルの名前を指定します。同じ名前のファイルがすでに存在する場合、**optdiag** は警告を表示しないで既存のファイルを上書きします。

-U user_name

Adaptive Server ログイン名を指定します。

-P password

Adaptive Server のパスワードを指定します。**-P** フラグを指定しない場合、**optdiag** はパスワードの入力を要求するプロンプトを表示します。

-T trace_value

optdiag セッションのトレース・フラグをセットします。**optdiag** のトレース・フラグは次のとおりです。

フラグ値	意味
1	使用している Adaptive Server の optdiag バージョンが入力ファイル内の Adaptive Server のバージョンと一致しない場合でも、警告を表示して停止しません。
2	入力モードにおいて、「次のテーブルは <i>table_name</i> 」というステータス・メッセージを表示します。
4	入力モードにおいて、ヒストグラムのロード中にステップ番号の一貫性の検査を行いません。
6	入力モードにおいて、入力ファイルの各行を表示します。このフラグは表示モードでは無効です。
7	optdiag 入力ファイルにサンプリング率情報が含まれない場合でも、警告を表示して停止しません。

-I interfaces_file

Adaptive Server に接続するとき使用する **interfaces** ファイルの名前とロケーションを指定します。

-I で **interfaces** ファイル名を指定しない場合、**optdiag** は SYBASE 環境変数で指定されたディレクトリにある **interfaces** ファイル (UNIX では *interfaces*) を探します。Windows では、**optdiag** は Sybase インストール・ディレクトリ (*d: ¥sybase*) の *ini* サブディレクトリにある *sql.ini* という名前のファイルを探します。SYBASE 環境変数が設定されていない場合、**optdiag** はデフォルトの **\$\$SYBASE** ディレクトリ (Windows では **%SYBASE%**) にあるファイルを探します。

-S server

接続する Adaptive Server の名前を指定します。**optdiag** は、**interfaces** ファイル (Windows では *sql.ini*) 内でこの名前を探します。

- サーバ名を指定しないで **-S** を使用した場合、**optdiag** は SYBASE という名前のサーバを探します。
- **-S** を使用しない場合、**optdiag** は DSQUERY 環境変数によって指定されたサーバを探します。

-v

optdiag のバージョン番号と著作権メッセージを表示して、終了します。

-h

optdiag の構文ヘルプを表示します。

-s

optdiag の出力にシステム・テーブルを含めます。デフォルトでは、ユーザ・テーブルだけが含まれます。

-z language

サーバが日付フォーマットと optdiag のプロンプトおよびメッセージの表示に使用する代替言語の公式名です。-z フラグを指定しないと、optdiag はサーバのデフォルト言語を使用します。

言語は、インストール時とインストール後のどちらでも Adaptive Server に追加できます。Adaptive Server のインストール後に言語を追加する場合は、langinstall ユーティリティまたは sp_addlanguage ストアド・プロシージャを使用します。

-J client_charset

クライアントで使用する文字セットを指定します。フィルタによって、client_charset と Adaptive Server の文字セット間で入力に変換されます。

-J client_charset は、クライアントで使用する文字セットである client_charset とサーバの文字セット間でのデータの変換を Adaptive Server に要求します。

文字セット名を指定しないで -J を使用すると、文字セット変換が NULL に設定されます。この場合、変換は行われません。クライアントとサーバが同じ文字セットを使用する場合は、-J だけを指定してください。

-J を省略すると、文字セットはプラットフォームのデフォルト文字セットに設定されます。フィルタによって、デフォルト文字セットと Adaptive Server 文字セットとの間で入力に変換されます。デフォルトの文字セットは、クライアントが使用する文字セットと同じであるとはかぎりません。

文字セットと関連するフラグの詳細については、『システム管理ガイド』を参照してください。

-a display_charset

optdiag を実行しているマシンの文字セットとは異なる文字セットを使用する端末から、optdiag を実行できます。

- -a を -J とともに使用すると、変換に必要な文字セット変換ファイル (.xlt ファイル) が指定できます。
- クライアントの文字セットがデフォルトの文字セットと同じ場合にかぎり、-J を指定せずに -a を指定してください。

注意 ascii_7 文字セットは、すべての文字セットと互換性があります。Adaptive Server の文字セットとクライアントの文字セットのどちらかが ascii_7 に設定されていれば、すべての7ビット ASCII 文字が変更されることなくクライアントとサーバの間で受け渡されます。その他の文字セットを使用している場合は、変換エラーが発生します。文字セット変換の詳細については、『システム管理ガイド』を参照してください。

一部の Linux プラットフォームでは、LANG 環境変数はデフォルトで“en_US.UTF-8”に設定されている場合があります。この場合、クライアントとサーバ間で不要な LONGCHAR 変換が発生する可能性があります。サーバとクライアントの文字セットが異なる場合、Sybase では、以下の方法のいずれかを使用して変換を回避することをおすすめします。

- unsetenv LANG
- setenv LANG C
- optdiag -J
- optdiag -Jiso-1 (サーバが iso-1 を使用している場合)

例

例 1 pubs2 データベースにある全ユーザ・テーブルの統計値を表示し、その出力を *pubs2.opt* ファイルに保存します。

```
optdiag statistics pubs2 -Usa -Ppasswd -o pubs2.opt
```

例 2 titles テーブルの統計値を表示します。

```
optdiag statistics pubs2..titles -Usa -Ppasswd -o titles.opt
```

例 3 統計値を roman8 文字セットで表示し、ロー・ラベルとエラー・メッセージをフランス語で表示します。

```
optdiag statistics pubs2..titles -Usa -Ppasswd -o titles.opt -J roman8
-z french
```

例 4 titles テーブルの price カラムの統計値をバイナリ形式で表示します。

```
optdiag binary statistics pubs2..titles.price -Usa -Ppasswd -o price.opt
```

例 5 編集された統計値を price.opt ファイルからロードします。

```
optdiag statistics -i price.opt -Usa -Ppasswd
```

使用方法

- SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、optdiag を使用してください。
- デフォルトでは、データベースの統計値を表示するときに optdiag はシステム・テーブルを統計の対象としません。このコマンドの出力にシステム・テーブルを含めるには、-s フラグを使用します。
- optdiag のコマンド・ラインで特定のパーティションを指定することはできません。optdiag は、指定したテーブルのすべてのパーティションの統計値を表示します。
- binary モードを使用すると、optdiag は、次の例のように、行頭にコメント記号 (#) の付いた人間が判読できる値を表示します。

```
Statistics for column:          "price"
Last update of column statistics:Jan 20
1998    7:16PM
Statistics loaded from Optdiag.
      Range cell density:      0x3f8b9cfefefce26bf
#   Range cell density:      0.0134830400000000
      Total density:          0x3f8b9cfefefce26bf
#   Total density:          0.0134830400000000
      Range selectivity:      default used (0.33)
#   Range selectivity:      default used (0.33)
      In between selectivity: default used (0.25)
#   In between selectivity: default used (0.25)
```

- 入力ファイルを指定して optdiag を実行し、統計値を変更する場合、“#” より後にある文字は無視されます。
- float 値を変換する場合、入力ファイルを使用すると、丸めエラーが発生することがあります。

同じハードウェア・プラットフォームで統計値をロードする場合は、精度を高めるためにバイナリ値を使って統計値を編集してください。

- optdiag は、ユーザが指定したサンプリング率で統計値が収集されることを示す `sampling percent last used` という統計値を表示します。

- `optdiag` は、複数に分割されたテーブルまたはインデックスの各パーティションの統計値を表示します。
- `optdiag` は、複数のパーティションを持つテーブルの各カラムについて、グローバルレベルとパーティションレベルの統計値を表示します。

バイト順序と `optdiag` のバイナリ・ファイル

- バイト順序が異なるマシンの `Adaptive Server` 間で統計値を移動する場合は、`binary` モード・オプションを使用しないでください。
 - 互換性のないアーキテクチャのサーバでは、バイナリ形式の統計値をコメント化し、人間が判読できる形式の統計値をロードしてください。
 - 互換性のあるアーキテクチャのサーバでは、バイナリ形式または人間が判読できる形式のどちらの統計値もロードできます。

入力モード

- `-i input_file` 構文を使用すると、`optdiag` は指定されたファイルを読み込み、`sysstatistics` 内の統計値を更新します。
- 入力モードでは、`optdiag` は設定パラメータ `allow update to system tables` を変更します。つまり、セッションの開始時にこのパラメータを 1 に設定し、終了時に 0 に設定します。
- ヒストグラム入力時には、プロセスは次に示すルールをチェックし、違反がある場合はエラー・メッセージを表示します。
 - `-T4` トレース・フラグを指定していないかぎり、ステップ番号は単調に増加しなければなりません。
 - ステップのカラム値は単調に増加しなければなりません。
 - 各セルのウェイトは 0.0 ~ 1.0 の範囲になければなりません。
 - 1カラムの総ウェイトは 1.0 に近い値でなければなりません。
 - 先頭のセルは `null` 値を表し、`null` 入力可でないカラムにも必要です。`null` 値を表すセルは 1 つしか存在できません。
 - 2つの隣り合うセルの両方が `<` (小なり) 演算子を同時に使用していないようにします。

参照

コマンド create index、delete statistics、set、update statistics

マニュアル optdiag コマンド、および optdiag 出力の詳細については、『パフォーマンス&チューニング・ガイド』を参照してください。
optdiag を使用して統計値を変更する方法の詳細については、『パフォーマンス&チューニング・ガイド』を参照してください。

システム・プロシージャ sp_addlogin、sp_configure、sp_defaultlanguage、sp_droplanguage、sp_flushstats、sp_helplanguage

preupgrade

説明	<p>アップグレードできる状態かどうかを判断するために、インストールまたはデータベースでテストを実行し、検出された問題をレポートします。preupgrade は <code>\$\$SYBASE/\$\$SYBASE_ASE/upgrade</code> にあります。</p> <p>(Windows の場合) このユーティリティは <code>preupgrd.exe</code> で、<code>\$\$SYBASE%¥¥SYBASE_ASE%¥¥upgrade</code> にあります。</p>
構文	<pre>preupgrade [-v] [-h] [-N] [-p [skip_sybprocs] [-D database_name] [-I interfaces_file] [-P password] [-S server_name] [-U user_name] [-X option[,option]...]</pre>
パラメータ	<p>-D database_name 検査を指定されたデータベースと可能な検査のサブセットに限定します。このパラメータを使用して新しくロードされたデータベースを検査してから、データベースをオンラインにしてください。</p> <p>-h ヘルプ・テキストを表示して、終了します。</p> <p>-I interfaces_file サーバの <code>interfaces</code> ファイルを指定します。デフォルトは、<code>\$\$SYBASE/interfaces</code> です。</p> <p>-N <code>preupgrade</code> を非対話型モードで実行するように指定します。したがって、データベースのサイズが小さすぎると <code>preupgrade</code> が判断した場合、ユーティリティは直ちに終了します。</p>

-p [*skip_sybprocs*]

ストアド・プロシージャに対する解析の有効性テストを省略するかどうかを指定します。

アップグレード処理中に、ストアド・プロシージャは自動的にテキスト・ソースから内部で再作成され、正しく解析されることを要求します。preupgrade -p の有効なオプションは次のとおりです。

- **-p** — すべてのデータベースでの解析を省略する。
- **-p skip_procs** — **sybssystemprocs** での解析を省略するが、他のデータベースは解析する。
- **Not using -p** — すべての場所にあるテキストを解析する。これがデフォルト値。

-P *password*

サーバに接続するためのパスワードを指定します。Sybase ではコマンド・ラインでこのオプションを使用しないことをおすすめします。このオプションは、他のユーザがシステム情報画面を表示したときにパスワードを表示するからです。このオプションを使用するのではなく、Adaptive Server がパスワードを入力するよう要求するまで待ってから、情報を入力してください。

-S *server_name*

接続先のサーバの名前を指定します。このサーバは、-I パラメータによって指定された **interfaces** ファイルにリストされている必要があります。デフォルトは **\$DSQUERY** です。

-U *user_name*

サーバへの接続時に使用するユーザ名を指定します。デフォルトは“sa”です。*user_name* には、検査対象のサーバに対する“sa_role”権限が必要です。

注意 -D オプションを使用して、指定したデータベースに検査を限定したときに、そのデータベースがオフラインの場合は、ユーザ名として“sa”を入力するか、デフォルトを受け入れてください。

-v

バージョン情報を表示して、終了します。

-X option[, option...]

実行する検査のリストを指定します。-D オプションを使用して利用可能な検査のサブセットだけを使用するように指定している場合を除き、デフォルトはすべての検査です。コマンド・ラインで -X オプションを何度か指定した場合、preupgrade が実行するのは、最後に入力した -X パラメータに指定された検査だけです。

オプション・リストで -X パラメータを使用するときは、次のいずれかを行います。

- カンマと次のオプションの間にスペースを入れずにオプションをリストする。
- オプション・リストを引用符で囲む。

表 7-2 は、有効な検査オプションを示します。Sybase では、-X パラメータの有効なオプションを変更する場合があります。有効なオプションの最新のセットを表示するには、-h パラメータを使用します。

表 7-2 : -X パラメータの検査オプション

検査オプション	-D パラメータで使用するオプション	説明
all		許可されたすべての検査を実行します。-D オプションとともに使用した場合は、オプションのサブセットだけを検査します。それ以外の場合は、すべてのオプションを検査します。
cache		デフォルト・キャッシュ・サイズの定義を検査します。定義が DEFAULT の場合は、実際の値として設定ファイル内の現在の値を入力してください。これにより、新しいサーバのデフォルト値が現在のサーバのデフォルト値と異なっても、サイズが変更されることはなくなります。
config		サーバの設定パラメータが新しい要件に一致しているかどうかを検査し、不一致がある場合はレポートします。 一部のパラメータでは、不一致によって次のようにエラーまたは警告が発生する場合があります。 <ul style="list-style-type: none"> • エラー - パラメータの現在の値が新しいサーバの範囲外である場合、またはアップグレードするには値が小さすぎる場合に発生する。 • 警告 - パラメータの現在の値が最大値と最小値の間であっても、デフォルト値より小さい場合に発生する。
data_mods		ゼロではないシステム・テーブル・カラムのクリアも含め、システム・テーブルに対する更新を実行します。主に以前の Adaptive Server に適用され、新しいシステムに変更されることはありません。

検査オプション	-D パラメータ で使用するオ プション	説明
datatype	X	systypes テーブルを検査して、既存のデータ型がシステム定義の名前、型、またはユーザ型を使用している場合に、それらが新しいサーバに必要なものと一致していることを確認します。不一致をレポートし、解決策を提示します。
db_size		特定のシステム・データベースがインストールの最小サイズ要件を満たしているかどうかを検査します。
free_space	X	指定したデータベースまたはすべてのデータベースに空き領域があるかどうかを検査します。アップグレードに必要な手順を実行するためのデータ領域とログ領域が十分にあることを確認します。
object_id	X	ユーザ定義オブジェクトのオブジェクト ID が、システム・オブジェクト用に予約されていないかどうかを検査します。 <ul style="list-style-type: none"> Adaptive Server 15.0 以降では、オブジェクト ID 1 ~ 255 が予約されている。 Adaptive Server 12.5x 以前では、オブジェクト ID 1 ~ 99 が予約されている。 エラーを発行しないでください。ユーザ・オブジェクト ID が予約されていることを示す警告が表示された場合は、Sybase サポート・センタに連絡して、アップグレード後にユーザ・オブジェクト ID を変更する方法を確認してください。
required_dbs		必要なシステム・データベースが終了しているかどうかを検査します。Adaptive Server の一部のバージョンでは、 sysystemdb などの特別なデータベースが必要な場合があります。
sproc_text	X	指定したデータベースまたはすべてのデータベース内のストアード・プロシージャ・テキストの存在を確認します。アップグレード後、Adaptive Server はソース・テキストからストアード・プロシージャを再コンパイルする必要があります。この検査によって、すべてのソース・テキストが使用可能であり、なおかつ有効であることを確認します。
srvclass		master.dbo.syssservers で “generic” として分類されているサーバを検査します。Adaptive Server 12.0 以降では、このクラスは推奨されていません。
statistics		sysstatistics 内の重複するローを検査します。重複するローは、Adaptive Server 12.0 から Adaptive Server 15.0 以降へのアップグレード時に sysstatistics テーブルのスキーマの変更によって発生することがあります。

例 **例 1** preupgrade を実行し、インストールのデフォルト・キャッシュ・サイズ、データベースの最小サイズ、**sysstatistics** 内の重複するローの検査を指定します。

```
preupgrade -X cache,db_size,statistics
```

例 2 `preupgrade` を実行し、新しくロードされたオフライン・データベースを検査します。`preupgrade` はデータ型、空き領域、オブジェクト ID、ストアド・プロシージャ・テキストを検査します。

```
preupgrade -Dmy_db -Usa
```

例 3 `sqlupgrade` ユーティリティで `preupgrade` 使用方法を示します。この方法を使用すると、すべてのデータベースが検査されます。`preupgrade` は非対話型で実行され、サイズが小さすぎるデータベースがある場合は、障害ステータスで終了します。

```
preupgrade -N
```

使用法

- `preupgrade` がエラーを検出しなかった場合は、ステータス 0 (ゼロ) で終了します。
- `preupgrade` は、プロシージャを円滑に実行できるように、主にインストールのアップグレード前に使用します。以前のバージョンの Adaptive Server で作成されたデータベースをロードした後に `preupgrade` を実行して、そのデータベースをオンラインにしたときに、アップグレードの妨げとなる問題がないかどうかを検査することもできます。
- `sqlupgrade` は、通常のプロシージャの一部として `preupgrade` を呼び出します。
- `-D` パラメータが指定されていない場合、`preupgrade` はシステム内のすべてのデータベースを検査し、サーバの以前のバージョンに対してのみ実行されます。
- `-D` パラメータが指定されている場合、`preupgrade` は次のようになります。
 - 指定されたデータベースだけを検査する。
 - `preupgrade` ユーティリティと同じバージョン番号のサーバに対して実行される。
 - 検査を可能な検査のサブセットに限定する。このパラメータとともに使用したときに有効な検査のリストについては、表 7-2 を参照してください。
- `-D` パラメータは、主に新しくロードされたデータベースをオンラインにする前に検査するために使用します。この場合、`preupgrade` はオフライン・データベースにアクセスすることになります。これには、`preupgrade` がユーザ “sa” としてログインする必要があります。“sa_role” 権限だけでは不十分です。デフォルトは `-Usa` です。

通常のアップグレードの一部として実行する場合、**-D** パラメータはオプションです。“sa” 以外にシステム管理者ログインを選択できます。

- アップグレード後に **preupgrade** を実行して新しくロードされたオフライン・データベースを検査する場合は、**-D** パラメータを使用し、“sa” ログインを指定するか、**-U** パラメータを省略してください。
- **-D** オプションを指定するか、**-X** パラメータでオプションの部分的なリストを指定すると、**preupgrade** は指定された検査を実行し、エラーがない場合は実行した検査のリストを終了メッセージに表示します。
- **preupgrade** が特定のデータベースまたは領域に関する警告またはエラーをレポートした場合、通常のアップグレードの一部として **-D** パラメータと **-X** パラメータを使用することをおすすめします。これらのパラメータを使用すると、不要な検査を繰り返さないで問題領域に焦点を合わせることができます。

sqlupgrade は、常に **preupgrade** のすべての検査を実行します。問題を解決し、**preupgrade** を実行し、問題が本当に解決しているかどうかを確認したら、**sqlupgrade** で **preupgrade** のすべてのプロセスを繰り返します。

パーミッション

-U パラメータで指定されたユーザ・ログインには、**-S** パラメータで指定されたサーバ上で **preupgrade** を実行するためのシステム管理者権限が必要です。

-D パラメータを使用してオフライン・データベースを検査する場合、**preupgrade** はユーザ “sa” としてログインする必要があります。“sa_role” 権限では不十分です。

参照

ユーティリティ [sqlupgrade](#), [sqlupgraderes](#)

pwdcrypt

説明	<p><i>libtcl.cfg</i> ファイルに LDAP 暗号化パスワードを作成し、表示します。 pwdcrypt は <code>\$SYBASE/\$SYBASE_OCS/bin</code> にあります。</p> <p>(Windows の場合) このユーティリティは <code>%SYBASE%\\$SYBASE_OCS%\bin</code> にあります。</p>
構文	pwdcrypt
パラメータ	なし
例	<p>プロンプト画面で pwdcrypt を入力すると、パスワードを2回入力するよう要求されます。pwdcrypt が、その後 LDAP パスワードを返します。</p> <pre>pwdcrypt Enter password please:password Enter password again :password The encrypted password: 0x01312a775ab9d5c71f99f05f7712d2cded288d0ae1ce79268d0e8669313d1bc4c706</pre> <p><i>libtcl.cfg</i> 内の LDAP URL の末尾部分を、この暗号化パスワードに置き換えます。</p> <pre>ldap=libsybdldap.so ldap://dolly:389/dc=sybase,dc=com????bindname=cn=Manager,dc=sybase,dc=com? 0x01312a775ab9d5c71f99f05f7712d2cded288d0ae1ce79268d0e8669313d1bc4c706</pre> <p>暗号化されていないパスワードは次のようになります。</p> <pre>ldap=libsybdldap.so ldap://dolly:389/dc=sybase,dc=com????bindname=cn=Manager,dc=sybase,dc=com? secret</pre>
使用法	SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、 pwdcrypt を使用してください。
パーミッション	ファイル・システムのパーミッションを使用して、 <i>libtcl.cfg</i> ファイルのこの暗号化パスワードへの不正なアクセスを防止してください。

qrmutil

説明

(Cluster Edition のみ) qrmutil はクォーラム・デバイスのバックアップ、リストア、再設定を可能にするコマンド・ライン・ユーティリティです。qrmutil は `$SYBASE/$SYBASE_ASE/bin` にあります。

構文

```
--additional-run-parameters=parameter_list
--ase-config-extract=file_name
--ase-config-info
--ase-config-store=file_name
--ase-config-version=version_number
--buildquorum=[force]--cluster-take-over
--config-file=file_name
--diag={all | boot | toc | nodes | locks | config | cms}
--display={boot | nodes | heartbeat | master | cluster |
           instance | config | state}
--drop-cluster=[force]
--drop-instance=instance_name
--errorlog=file_name
--extract-config=file_name
-h, --help
-F, --cluster-input=file_name
--fence-capable=device_path
--installation=installation_mode
-s, --instance=instance_name
--instance-node=node_name
--interfaces-dir=path_to_interfaces_file
--max-instances=number_of_instances
--master-dev=master_device
--membership-mode=membership_mode
--primary-address=interconnect_address
--primary-port=port_number
--primary-protocol=protocol
-Q, --quorum-dev=quorum_device
--register-node=node_name
--secondary-address=interconnect_address
--secondary-port=port_number
--secondary-protocol=protocol
--traceflags=traceflag_list
--unregister-node=node_name
--verify-node=node_name
-v, --version]
```


パラメータ

`--additional-run-parameters=parameter_list`

Unified Agent がデータ・サーバを起動するために使用するパラメータです。他の設定とは異なり、`dataserver` は追加の実行パラメータを読み込みません。Unified Agent によって読み込まれ、`dataserver` コマンド・ラインに渡されます。`--instance` パラメータを指定した場合、追加の実行パラメータは指定されたインスタンスに適用されます。パラメータを指定しない場合、追加の実行パラメータはクラスタ内のすべてのインスタンスに適用されます。

`--ase-config-extract=file_name`

クォーラム・デバイスに格納された Adaptive Server 設定ファイルを指定したファイルに抽出します。

`--ase-config-info`

クォーラム・デバイスに格納された Adaptive Server 設定ファイルについての情報を表示します。

`--ase-config-store=file_name`

指定したファイルを Adaptive Server 設定ファイルとしてクォーラム・デバイスに格納します。

`--ase-config-version=[version_number]`

クォーラム・デバイスに格納された Adaptive Server マスタ設定ファイルのバージョンを表示または設定します。

`--buildquorum[=force]`

新しいクォーラム・デバイスを構築します。ロー・パーティション上の既存のファイルまたは既存のクォーラム・デバイスを上書きするには、`=force` を使用します。`--buildquorum` では `--cluster-input` パラメータを指定してください。

`--config-file=config_file_name`

`-instance` とともに使用すると、このパスを指定されたインスタンスの Adaptive Server 設定ファイルに設定します。`-instance` を指定しない場合、このパスをクラスタ全体の設定ファイルに設定します。

`--diag={all | boot | toc | nodes | locks | config | cms}`

内部使用のみ。

`--display={boot | nodes | heartbeat | master | cluster | instance | config | state}`
クラスタまたはインスタンスの現在の状態を表示します。

- `boot` — クォーラム・デバイスのバージョン、起動時に発行されたすべてのトレース・フラグ、クラスタのブート ID、起動時に表示されたすべてのメッセージを含むクラスタの起動情報を表示します。
- `nodes` — 登録された管理ノードを表示します。
- `heartbeat` — クラスタ内のすべてのノードのハートビート情報を表示します。
- `master` — マスタ・デバイス情報を表示します。
- `cluster` — クラスタ設定を表示します。
- `instance` — インスタンス設定を表示します。
`--instance=instance_name` をこのパラメータに含める必要があります。
- `config` — クラスタと、クラスタ内のすべてのインスタンスの設定を表示します。
- `state` — クラスタと、クラスタ内のすべてのインスタンスの現在の状態を表示します。

`--drop-cluster=[force]`

クラスタを削除し、クォーラム・デバイスを削除します。クラスタが実行中であることをクォーラムが示す場合に削除を強制するには、`=force` を使用します。

警告！ `--drop-cluster` はクラスタを削除します。

`--drop-instance=instance_name`

Sybase の内部使用のみ。

警告！ クラスタからインスタンスを削除するには、`sybcluster` ユーティリティを使用してください。

`--errorlog=log_file_name`

指定されたインスタンスのエラー・ログのフル・パス。`-instance-name` パラメータを指定してください。次回インスタンスを再起動したときに有効になります。

- extract-config=*file_name***
クォーラム・デバイスの設定領域を指定されたファイルに抽出します。
- h | --help**
qrmutil の完全な構文を表示します。
- F | cluster-input=*file_name***
指定されたクラスタ入力ファイルからクラスタ設定をロードします。
- fence-capable=*device_path***
指定されたデバイスで I/O フェンシングが可能かどうかをテストします。“Device is fence capable”または“Device is not fence capable”を返します。
- installation=*installation_mode***
クラスタのインストール・モードを変更します。値は次のとおりです。
- shared (デフォルト)
 - private
- instance=*instance_name***
qrmutil パラメータを指定されたインスタンスに適用します。
- interfaces-dir=*interfaces_path***
interfaces ファイルを含むディレクトリのパス。このパラメータを --instance とともに使用すると、パスを指定されたインスタンスの interfaces ファイルに設定します。--instance を指定しない場合、このパスをクラスタ全体の interfaces ファイルに設定します。
- max-instances=*number_of_instances***
クラスタ設定の最大インスタンス数を設定します。
- master-dev=*master_device_name***
クラスタが使用するマスタ・デバイスを変更します。
- membership-mode=*membership_mode***
メンバシップ・モードを設定します。値は次のとおりです。
- native (デフォルト)
 - vcs
- primary-address=*interconnect_address***
指定されたインスタンスのプライマリ相互接続アドレスを変更します。

- primary-port=port_number**
指定されたインスタンスのプライマリ相互接続用の開始ポート番号を変更します。
- primary-protocol=protocol**
プライマリ・クラスタ相互接続に使用するプロトコルを変更します。
- Q | --quorum-dev=quorum_path**
クォーラム・デバイスのフル・パスを指定します。
- register-node=node_name**
クォーラム管理用のノードを登録します。
- secondary-address=inteconnect_address**
指定されたインスタンスのセカンダリ相互接続アドレスを変更します。
- secondary-port=port_number**
指定されたインスタンスのセカンダリ相互接続用の開始ポート番号を変更します。
- secondary-protocol=protocol**
セカンダリ・クラスタ相互接続に使用するプロトコルを変更します。
- traceflags=trace_flag, trace_flag**
クラスタ全体またはインスタンス固有の起動用トレース・フラグを変更します。トレース・フラグのリストを指定しない場合、qrmutilはクラスタ・インスタンスのトレース・フラグをクリアします。
- unregister-node=node_name**
クォーラム管理用のノードを登録解除します。
- verify-node=node_name**
指定されたノードがクォーラム・デバイスで登録されていることを示します。
- v | --version**
qrmutil ユーティリティのバージョン情報を表示します。

例 **例 1** エラー・ログのパスを `/sybase/opt/cluster/ASE-15_0/ase1.log` に変更します。

```
qrmutil --quorum_dev=/dev/raw/raw101 --instance=ase1  
--errorlog=/sybase/cluster/ASE-15_0/ASE-15_0/ase1.log
```

例 2 mycluster のノード “blade5” を登録します。

```
qrmutil --quorum_dev=/dev/raw/raw101 --register-node=blade5
```

例 3 “mycluster” クラスタの新規クォーラム・デバイスを作成します。

```
qrmutil --quorum-dev=/dev/raw/raw101 --cluster-  
input=/sybase/cluster/ase1.inp -buildquorum
```

例 4 クォーラム・デバイスを `/sybase/cluster_bak/quorum.bak` にバックアップします。

```
qrmutil --quorum-dev=/dev/raw/raw101  
--extract-config=/sybase/cluster_bak/quorum.bak
```

例 5 `/sybase/cluster_bak/quorum.bak` に作成されたバックアップからクォーラム・デバイスをリストアします。

```
qrmutil --quorum-dev=/dev/raw/raw101 --cluster-  
input=/sybase/cluster_bak/quorum.bak --buildquorum=force
```

例 6 クォーラム・デバイスに保管されたクラスタ設定を表示します。

```
qrmutil --quorum-dev=/dev/raw/raw101 --display=config
```

例 7 指定されたデバイスでフェンシングが可能かどうかをテストします。

```
qrmutil --quorum-dev=/dev/raw/raw101 --fence-capable=/dev/raw/raw106
```

使用法

- `qrmutil` は主に診断ユーティリティとして使用します。クラスタの設定を変更する場合は `sybcluster` を使用することをおすすめします。
- 最大で 20 のコマンドを `qrmutil` に渡すことができます。ただし、`--instance=` パラメータを指定できるのは 1 回だけです。
- `--buildquorum` を指定すると、クォーラムが構築されて、`qrmutil` は `--cluster-input` 以外のコマンドを実行することなく終了します。
- `qrmutil` は `--drop-cluster` パラメータを実行した後で終了します。
- 次に、複数のコマンドを使用する例を示します。

```
qrmutil --quorum-dev=/dev/raw/raw101 --display=cluster  
--register-node=blade1 --unregister-node=blade2 --verify-node=blade3
```

パーミッション

`qrmutil` を実行するためには、インスタンスを起動したユーザと同じ `sybase` ユーザであること、`qrmutil` バイナリに対する実行特権があること、クォーラム・デバイスに直接アクセスできること、最低でもクォーラム・ファイルに対する `read` パーミッションを持っていることが必要です。

qptune

説明

QPTune は Java および XML で書かれた Adaptive Server ユーティリティです。欠落統計を修正し、最適なクエリ・プラン、最適化目標、またはその他の構成設定を特定し、それらをクエリまたはサーバのレベルで適用できます。これにより、その後のクエリの実行で最適なパフォーマンスが得られます。

構文

```
qptune
  [-U username]
  [-P password]
  [-S hostname:port/database]
  [-A action]
  [-M mode]
  [-T appTime]
  [-i inputFile]
  [-o outputFile]
  [-f fileList(,)]
  [-c configFile]
  [-l limit]
  [-e evalField]
  [-d <diff%(,diff_abs)>]
  [-m missingCount]
  [-n login]
  [-J charset>]
  [-N (noexec)]
  [-g (applyOptgoal)]
  [-v (verbose)]
  [-s (sort)]
  [-h (help)]
```

パラメータ

-U *username*
データベースのユーザ名を指定します。

-P *password*
データベースのパスワードを指定します。

-S *server*
データベース・サーバを指定します。データベース・サーバは *host:port/database* で示されます。

注意 QPTune アクションを使用するときは、**-S** オプションを指定してください。

-A action

実行するアクションを指定します。有効なアクションは次のとおりです。

- start
- collect
- collect_full
- compare
- fix
- start_stats
- collect_stats
- fix_stats
- undo_fix_stats

-J charset

Adaptive Server への接続に使用する文字セットを指定します。このオプションを指定しない場合、Adaptive Server ではサーバのデフォルト文字セットが使用されます。

注意 インストールされた JRE がサーバのデフォルト文字セット・コードをサポートしていない場合は、ログイン・プロセス時にエラー・メッセージが表示されます。-J オプションを使用して、-J utf8 などの汎用文字セットを指定してください。

-M mode

アプリケーションの最適化目標またはカスタム・モードを指定します。以下のいずれかです。allrows_oltp、allrows_dss、allrows_mix のいずれかを指定します。カスタム・モードも指定でき、_basic_ がシステム予約カスタム・モードです。

-T appTime

アプリケーションの実行時間を分単位で指定します。

-o outputFile

出力ファイルを指定します。

-i inputFile

fix、fix_stats、undo_fix_stats アクションの入力ファイルを指定します。また、-i を使用して、カスタム・モードの start で特別なルールを指定のクエリに適用できます。

-f *fileList*

ファイルのリストを比較して最適なプランを取得します。複数のファイル名を区切るには、カンマを使用してください。

-c *configFile*

設定ファイルを指定します。

-l *limit*

特別なルールを使用して解析および適用する必要があるクエリの数の制限を指定します。

-e *evalField*

パフォーマンスの比較に使用される評価フィールドを指定します。

-d *difference*

パフォーマンスの改善が未解決であると判断するためのパーセンテージと絶対値の差分を指定します。

-N

fix_stats および **undo_fix_stats** とともに使用し、**-N** は、**update statistics** 文または **delete statistics** 文を使用する SQL スクリプトを生成します。**update** 文または **delete** 文は QPTune からは実行されません。文は **-o** オプションで指定した SQL スクリプトに記述されます。

-n *login*

クエリ実行を収集および解析するユーザのログインを指定します。

-m *missingCount*

欠落統計のスレッシュホールド値を指定します。デフォルト値は5です。

-v

冗長モードを指定します。

-g

fix アクションとともに使用した場合に、デフォルトの目標を適用します。デフォルトの目標は、QPTune の **fix** アクションを使用する最適なプランとして、ほとんどのクエリが使用した最良の **optgoal** 設定です。このオプションは、サーバのデフォルトの最適化目標を現在使用していないクエリのプランのみを生成します。

例 **例 1** 欠落統計を修正し、**start_stats** アクションを指定してユーティリティを起動します。

```
QPTune -A start_stats -S my_host:4816/my_database -v
```

```
実行中: QPTune -U sa -P [unshown]
-S jdbc:sybase:Tds:my_host:4816/my_database
-A start_stats -M allrows_dss -T 0 -i null
-o metrics.xml -f null -c config.xml -l 5
```



```
-e elap_avg -d 5,5 -m 5 -n null -v
データベースに接続しました:my_database
[INFO] 設定:sp_configure 'capture missing statistics', 1
[INFO] 設定:sp_configure 'system table', 1
[INFO] 設定:delete sysstatistics where formatid =110
```

例 2 collect_stats を使用して、欠落統計数に指定されたスレッショルドを超えている統計を求めするため、欠落統計情報を **sysstatistics** テーブルから取得します。

```
QPTune -A collect_stats -m 1 -o missingstats.xml -v
-S my_host:4816/my_database
```

```
実行中:QPTune -U sa -P [unshown] -S jdbc:sybase:Tds:my_host:4816/my_database
-A collect_stats -M allrows_dss -T 0 -i null -o missingstats.xml -f null -c
config.xml -l 5 -e elap_avg -d 5,5 -m 1 -n null -v
```

データベースに接続しました:my_database

```
Now collecting missing statistics information from sysstatistics on "Fri Sep
26 10:08:06 PDT 2008".
```

```
<?xml version="1.0" encoding="UTF-8"?>
<server url="jdbc:sybase:Tds:my_host:4816/my_database"
file="missingstats.xml"
type="missing stats" datetime="Fri Sep 26 10:08:06 PDT 2008">
  <missingStat id="1">
    <id>1068527809</id>
    <stats>Y(y4,y2)</stats>
    <count>2</count>
  </missingStat>
  <missingStat id="2">
    <id>1068527809</id>
    <stats>Y(y3)</stats>
    <count>1</count>
  </missingStat>
  <missingStat id="3">
    <id>1068527809</id>
    <stats>Y(y2,y1)</stats>
    <count>1</count>
  </missingStat>
  <missingStat id="4">
    <id>1068527809</id>
    <stats>Y(y1)</stats>
    <count>1</count>
  </missingStat>
</server>
```

The missing statistics information is written into XML file:missingstats.xml

```
[INFO] End config:sp_configure 'enable metrics capture', 0
[INFO] End config:sp_configure 'abstract plan dump', 0
```

```
[INFO] End config:sp_configure 'system table', 0
[INFO] End config:sp_configure 'capture missing statistics', 0
Program has restored the data source for metrics collection.
----- QPTune finished executing.-----
```

例 3 欠落統計情報を *missingstats.xml* という XML ファイルに収集した後で、**fix_stats** アクションを使用して統計を更新します。

```
QPTune -A fix_stats -m 1 -i missingstats.xml
      -v -S my_host:4816/my_database
```

```
実行中:QPTune -U sa -P [unshown] -S jdbc:sybase:Tds:my_host:4816/my_database
-A fix_stats -M allrows_dss -T 0 -i missingstats.xml -o metrics.xml -f null
-c config.xml -l 5 -e elap_avg -d 5,5 -m 1 -n null -v
データベースに接続しました: my_database
Fix statistics on "Fri Sep 26 10:14:59 PDT 2008"
```

```
-----
Details of statements(s) fixed:
```

```
-----
Fixed statistics:[Update] Y(y4,y2)
[INFO] Fix Statement = update statistics Y(y4,y2)
Fixed statistics:[Update] Y(y3)
[INFO] Fix Statement = update statistics Y(y3)
Fixed statistics:[Update] Y(y2,y1)
[INFO] Fix Statement = update statistics Y(y2,y1)
Fixed statistics:[Update] Y(y1)
[INFO] Fix Statement = update statistics Y(y1)
----- QPTune finished executing.-----
```

“noexec”を示す **-N** オプションと、出力スクリプト・ファイルを示す **-o** オプションを使用することで、実際の更新を実行しないで統計を更新するための SQL スクリプトを生成します。

```
QPTune -U sa -P -S my_host:5000/my_database
      -A fix_stats -m 5 -i missingstats.xml
      -N -o missingstats.sql
```

例 4 標準の最適化目標設定をクエリに適用するために QPTune を起動します。

```
QPTune -S host:port/database -A start
      [-M {allrows_oltp, allrows_dss, allrows_mix}]
```

カスタム・ルールを指定のクエリに適用するために QPTune を起動します。

```
QPTune -S host:port/database -A start -M custom_1
      -i input.xml -l 3 [-v]
```

例5 アプリケーションを実行して測定基準を *a2.xml* という XML ファイルに収集します。

```
QPTune -S host:port/database -A collect -T 0
        -o a2.xml -v
```

```
Program has configured the data source for metrics collection.
Now collecting information from sysquerymetrics on "Tue Feb 19 22:16:04 PST
2008".
```

```
<?xml version="1.0" encoding="UTF-8"?>
  <server url="jdbc:sybase:Tds:SHANGHI:5000" type="ASE" mode="custom_1"
datetime="Tue Feb 19 22:16:04 PST 2008">
  <query id="1">
  <qtext> select count(T.title_id) from authors A, titleauthor T
where A.au_id = T.au_id </qtext>
  <elap_avg>300</elap_avg>
  <bestmode> custom_1
  </bestmode>
  </query>
</server>
```

例6 測定基準を収集したら、さまざまな XML ファイルを比較して、クエリごとに最適なクエリ最適化目標または基準を取得します。

```
QPTune -A compare -f a1.xml,a2.xml -d 51,10
        -o best.xml -S my_host:5000/my_database
```

この結果は、2つの XML 測定基準ファイル間の比較を示しています。*a1.xml* には6つのクエリがあり、*a2.xml* には7つのクエリがあります。両方のファイルに共通するクエリ間でのみ比較できます。*a2.xml* で高速に実行されたクエリは3つあります。

```
Compare all the files:| a1.xml, a2.xml|
Report generated on "Tue Aug 19 21:13:04 PST 2008"
```

```
-----
File #1:[name= a1.xml   :mode=allrows_mix]
File #2:[name= a2.xml   :mode=custom_1]
Query count in File #1 :[mode=allrows_mix]           6
Query count in File #2 :[mode=custom_1]              7
=====
Query count improved in File #2:[mode=allrows_mix] 3
```

```
Total performance improved [from 422 to 129]:69 %
```

```
Following queries run better in File #2:
[mode=allrows_mix]
```

```
-----
Group 1:improved by no more than 25% [0 queries]
```

```
Group 2:improved by 25% to 50% [1 queries]
Query:select count(T.title_id) from authors A, titleauthors T where A.au_id
= T.au_id
Average elapsed time (ms):File #1=100 File #2=50 Improvement=50.0%
Outstanding=No
Group 3:improved by 50% to 75% [0 queries]
Group 4:improved by 75% to 100% [2 queries]
Query:select count(*) from titlles T, titleauthors TA where T.title_id =
TA.title_id
Average elapsed time (ms):File #1=34 File #2=7 Improvement=79.0%
Outstanding=Yes
Query:select au_lname, au_fname from authors where state in ("CA", "AZ")
Average elapsed time (ms):File #1=9 File #2=0 Improvement=100.0%
Outstanding=No
```

使用法 パラメータに特定の値を指定しない場合は、以下のデフォルトが使用されます。

- -A : collect
- -M : allrows_dss
- -T : 0
- -o : metrics.xml
- -c : config.xml
- -e : elap_avg
- -d : 5,5. 絶対値を指定しないでパーセンテージのみを指定した場合、絶対値はデフォルトで 0 に設定されます。
- -l *limit*
- -m 5

パーミッション QPtune の compare アクションはすべてのユーザが実行できます。他のすべての QPtune のアクションは、sa_role および sso_role を持つユーザのみが実行できます。

参照 QPtune ユーティリティまたは QPtune GUI の詳細については、『マイグレーション技術ガイド』を参照してください。

showserver

説明 (UNIX のみ) ローカル・マシンで現在稼働している Adaptive Server と Backup Server を表示します。UNIX プラットフォームでのみ使用できます。showserver は `$SYBASE/$SYBASE_ASE/install` にあります。

構文 showserver

パラメータ なし

例 ローカル・マシンで現在稼働している Adaptive Server と Backup Server を表示します。

```
showserver
```

```
USER          PID %CPU %MEM  SZ  RSS TT  STAT  START  TIME  COMMAND
user114276    0.0  1.7  712 1000 ?S   Apr  5514:05  dataserver
-d greensrv.dat -sgreensrv -einstall/greensrv+_errorlog
sybase       1071  0.0  1.4  408  820 ?S   Mar 28895:38
/usr/local/sybase/bin/dataserver -d/dev/rsd1f -e/install/errorlog
user128493    0.0  0.0 3692   0 ?IW  Apr  1  0:10  backupserver -SSYB_BACKUP
-e/install/backup.log -Iinterfaces -Mbin/sybmultbuf -Lus_english -Jiso_1
```

使用法 showserver は、Adaptive Server または Backup Server に関する処理情報を表示します。実行中のサーバがない場合は、ヘッダのみを表示します。

参照 **コマンド** [dataserver](#), [startserver](#)

関数 `host_name`

ユーティリティ [langinstall](#)

sqldbgr

説明

sqldbgr は、ストアド・プロシージャとトリガをデバッグするコマンド・ライン・ユーティリティです。多くのソース・レベル・デバッガとともに使用しながら、次のことができます。

- sqldbgr をタスクに付加する。
- ブレークポイントを設定、有効化、無効化する。
- タスクを一度に一行ずつステップ実行する。
- プロシージャにステップ・イン、ステップ・アウトする。
- デバッグが完了したら、ストアド・プロシージャまたはトリガから sqldbgr を分離する。
- (UNIX の場合) sqldbgr は \$SYBASE/\$SYBASE_ASE/bin にあります。
- (Windows の場合) sqldbgr は %SYBASE%\\$SYBASE_ASE%\\$bin にあります。

注意 sqldbgr のバージョン文字列を表示できなくてもかまいません。

構文

```
sqldbgr
      -U username
      -P password
      -S host.port
```

パラメータ

```
-U username
  ユーザ名を指定します。-U と username の間にはスペースを 1 つ挿入します。

-P password
  ユーザのパスワードを指定します。-P と password の間にはスペースを 1 つ挿入します。

-S host.port
  マシン名とポート番号を指定します。-S と host.port の間にはスペースを 1 つ挿入します。
```

例

例 1 次の例では、sqldbgr によってホスト MERCURY 上のストアド・プロシージャとトリガをデバッグしています。

```
$SYBASE/$SYBASE_ASE/bin/sqldbgr -U sa -P -S MERCURY:16896

(sqldbgr) stop in sp_who
Breakpoint moved to line 20
(sqldbgr) run sp_who
```

```
(sp_who::20)if @@trancount = 0
(sqldbg) next
(sp_who::22) set chained off
(sqldbg) cont
fid spid status loginame origname hostname blk_spid dbname cmd block_xloid
0 2 sleeping NULL NULL 0 master NETWORK HANDLER 0
0 3 sleeping NULL NULL 0 master NETWORK HANDLER 0
0 4 sleeping NULL NULL 0 master DEADLOCK TUNE 0
0 5 sleeping NULL NULL 0 master MIRROR HANDLER 0
0 6 sleeping NULL NULL 0 master ASTC HANDLER 0
0 7 sleeping NULL NULL 0 master ASTC HANDLER 0
0 8 sleeping NULL NULL 0 master CHECKPOINT SLEEP 0
0 9 sleeping NULL NULL 0 master HOUSEKEEPER 0
0 10 running sa sa 0 master SELECT 0
0 11 sleeping sa sa
(sqldbg) show breakpoints
1 stop in sp_who
(sqldbg)
```

例 2 次の例では、システム管理者がまず、`isql` を使用して Adaptive Server にログインしています。次に、別のタスクで実行中のストアド・プロシージャをデバッグするために、コマンド・ラインから `sqldbg` を実行しています。

```
$SYBASE/$SYBASE_OCS/bin/isql -U sa -P
1> select @@spid
2> go
-----
12
1>

$SYBASE/$SYBASE_ASE/bin/sqldbg -U sa -P -S MERCURY:16896

(sqldbg) attach 13
The spid is invalid
(sqldbg) attach 12
(sqldbg) show breakpoints
(sqldbg) stop in sp_who
Breakpoint moved to line 20
(sqldbg) /* at this point run the sp_who procedure from spid 12 */
(sqldbg) where
(sp_who::20::@loginname = <NULL>)
(ADHOC::1::null)
(sqldbg) next
(sp_who::22) set chained off
(sqldbg) next
(sp_who::25) set transaction isolation level 1
(sqldbg) cont
```

```
(sqldbgr) /* at this point the sp_who result will show up in the isql screen */
(sqldbgr) detach 12
(sqldbgr)
```

使用法

- **sql** コマンドはデバッグ対象のタスクのコンテキストで実行され、**mysql** コマンドはデバッガ・タスクのコンテキストで実行されま
す。**set quoted_identifier on** などのセッション固有の情報を **sql** で設
定しても機能しません。
- デフォルトでは、Sybase jConnect JDBC ドライバは **set
quoted_identifier on** を使用します。**sqldbgr** ユーティリティは、引用
符を必要とする jConnect 引数を使用して構築されるため、オプ
ションを入力するときは二重引用符ではなく一重引用符を使用し
ます。たとえば、**sp_configure "allow update"** ではなく、
sp_configure 'allow update' を使用します。
- **SYBASE_JRE** または **JAVA_HOME** 環境変数を Java 実行環境があ
る場所に設定してから、**sqldbgr** を実行します。
- コマンド・プロンプトで **sqldbgr** 呼び出すと、ユーティリティが起
動し、プロンプトが **sqldbgr** プロンプトに変わります。

(sqldbgr)

(sqldbgr) プロンプトが表示されたら、以下の **sqldbgr** コマンドを
入力してタスクを実行できます。

表 7-3 : sqldbgr コマンドとその説明

コマンド	説明
attach <i>spid</i>	<p>すでに Adaptive Server にログインしているときに、タスクを sqldbgr に付加 します。</p> <hr/> <p>注意 実行中でないプロシージャに付加するために attach spid を使用すること はできません。</p> <hr/> <p>sqldbgr では、同じセッションで複数のタスクをデバッグすることはできま せん。複数のタスクにこのユーティリティを付加しようとしても、最初の <i>spid</i> が付加とマーク付けされたままです。すでに付加されている <i>spid</i> には 付加できないため、detach コマンドを使用してから、別の <i>spid</i> に付加して ください。</p>
run <i>procname</i>	<p>既存のタスクに sqldbgr を付加しないでストアド・プロシージャとトリガを デバッグします。</p> <p>既存のタスクを attach spid ですでにデバッグ中に run procname を実行しよう とすると、run procname は失敗し、以下のように表示されます。</p> <pre>Cannot run a procedure while debugging another task</pre>

コマンド	説明
stop in <i>procname</i> [at line #]	<p>デバッグ中のストアド・プロシージャまたはトリガを、指定されたプロシージャ名の最初で停止させるブレイクポイントを設定します。</p> <p>stop in <i>procname</i> at line # は、デバッグされているストアド・プロシージャまたはトリガを、指定のプロシージャ内の指定の行で停止させるブレイクポイントを設定します。</p> <p>無効な行番号を入力すると、<code>sqldbgr</code> によってブレイクポイントは有効な次の行番号に移動し、次のように表示されます。</p> <p style="padding-left: 40px;">Invalid line number</p> <p>このコマンドを使用して複数のブレイクポイントの設定もできます。</p>
show breakpoints	<p><code>sqldbgr</code> セッション中にユーザによって指定されたブレイクポイント文とともに、ユニークな番号形式でブレイクポイント・ハンドルを表示します。</p> <p>有効な SQL 文が含まれていないブレイクポイント行番号を指定すると、Adaptive Server は有効な次の行番号にブレイクポイントを移動します。ただし、Adaptive Server は入力されたコマンドを変更するわけではありません。このため、<code>show breakpoints</code> が返すブレイクポイント・ハンドルと、<code>sqldbgr</code> セッション中に指定されたブレイクポイント文が異なる場合があります。</p> <p>ブレイクポイント行のアスタリスク (*) はブレイクポイントが設定されているが、現在無効であることを示しています。</p>
use <i>dbname</i>	データベースのストアド・プロシージャまたはトリガをデバッグするために使用するデータベースを <code>sqldbgr</code> に指示します。
show variables [at level #] show @ <i>varname</i> [at level #]	<p><code>show variables</code> は、現在の SQL ストアド・プロシージャまたはトリガのすべての変数とその値を表示します。</p> <p><code>show variables at level #</code> は、現在の SQL ストアド・プロシージャまたはトリガの変数とその値を指定されたレベルで表示します。</p> <p><code>show @<i>varname</i></code> は、現在の SQL ストアド・プロシージャまたはトリガの指定された変数とその値を表示します。</p> <p><code>show @<i>varname</i> at level #</code> は、現在の SQL ストアド・プロシージャまたはトリガの指定された変数とその値を指定されたレベルで表示します。</p> <hr/> <p>注意 <code>sqldbgr</code> は Java 変数をサポートしません。</p>
show where	デバッグされているタスクに存在するストアド・プロシージャとトリガの呼び出しスタックを表示します。
step または next	<code>step</code> または <code>next</code> は、 <code>sqldbgr</code> に現在のストアド・プロシージャまたはトリガで次の文に移動するように指示します。
step into	現在の文が <code>execute</code> 文である場合、 <code>sqldbgr</code> にプロシージャに移動するように指示します。現在の文が <code>update</code> 文、 <code>delete</code> 文、 <code>insert</code> 文のいずれかである場合、またはそれらにトリガがある場合は、 <code>step into</code> は <code>update</code> トリガ、 <code>delete</code> トリガ、または <code>insert</code> トリガに移動するように <code>sqldbgr</code> に指示します。

コマンド	説明
step out	sqlldbgr に現在のスタッド・プロシージャ外またはトリガ外に移動し、呼び出し元プロシージャの次の行で停止するように指示します。
set @varname = VALUE	指定された変数の値を現在のスタッド・プロシージャまたはトリガのコマンドで宣言されている変数の値に設定します。set @varname = VALUE を使用して設定される変数の値は、現在の sqlldbgr セッションでのみ有効です。
cont[inue]	デバッグを続行し、次のブレークポイント (存在する場合) で停止するように sqlldbgr に指示します。
delete #	sqlldbgr の現在のインスタンスに設定されている指定ブレークポイントを削除します。
enable # と disable #	指定ブレークポイントを有効化します。disable # はそれと逆のを行います。
sql any_sql_statement	特定の SQL 文を実行します。このコマンドを使用して、デバッグされているタスクによって作成されたテンポラリ・テーブルからデータを選択して分析できます。 sql any_sql_statement は、結果セットと発生したエラーを返します。
detach spid	指定 spid から sqlldbgr を分離し、デバッグ中のタスクを解放します。 現在の sqlldbgr セッションにおいてデバッグ中のタスクに設定されたブレークポイントが削除されます。
help [all]	sqlldbgr コマンドを表示します。

表 7-4 は、sqlldbgr のすべてのエラー・メッセージを示します。

表 7-4 : sqlldbgr エラー・メッセージとその意味

エラー・メッセージ	説明
Cannot allocate resource in ASE	Adaptive Server に、sqlldbgr を実行するための十分なメモリ・リソースがないことを示します。procedure cache size の値を大きくして、sqlldbgr を再度実行してください。
Cannot create Debugger handle in ASE	Adaptive Server に、デバッガ・ハンドルを作成するための十分なメモリ・リソースがないことを示します。procedure cache size の値を大きくして、sqlldbgr を再度実行してください。
The spid is invalid	sqlldbgr を無効な spid に付加しようとしたときに表示されます。spid を確認し直し、サイド実行してください。
You cannot debug a task that is not owned by you	所有していないタスクをデバッグしようとする则表示されます。デバッグされるタスクの所有者としてサーバにログインする必要があります。
Spid is already being debugged	attach spid を実行するとき、すでにデバッグ中の spid に付加しようとしたときに表示されます。
Spid is not debugged currently	detach spid を実行し、sqlldbgr に付加されていない spid から分離しようとしたときに表示されます。
Invalid command	無効なコマンドを入力したときに表示されます。

エラー・メッセージ	説明
Invalid procedure name	stop in <i>procname</i> に無効なプロシージャ名を入力したときに表示されます。
Invalid line number	stop in <i>procname</i> at line # に無効な行番号を入力したときに表示されます。
Variable not found	show @ <i>varname</i> 、show @ <i>varname</i> at level #、または set @ <i>varname</i> = VALUE に無効な変数を入力したときに表示されます。
Illegal conversion attempted	set @ <i>varname</i> = VALUE を実行するとき、変数を無効な値に変換しようとしたときに表示されます。
Conversion from text to datatype failed	set @ <i>varname</i> = VALUE が失敗したときに表示されます。
Cannot run a procedure while debugging another task	既存のタスクを attach <i>spid</i> ですでにデバッグしながら run <i>procname</i> を使用したときに表示されます。

sqlloc

説明	(UNIX のみ) GUI を使用して、Adaptive Server の言語、文字セット、ソート順のデフォルトをインストールしたり、変更したりします。 sqlloc は \$SYBASE/\$SYBASE_ASE/bin にあります。
構文	<pre>sqlloc [-S <i>server</i>] [-U <i>user</i>] [-P <i>password</i>] [-s <i>sybase dir</i>] [-l <i>interfaces file</i>] [-r <i>resource file</i>]</pre> <p>または</p> <pre>sqlloc -v</pre>
パラメータ	<p>-S <i>server</i> 接続する Adaptive Server の名前を指定します。</p> <p>-U <i>user</i> ログイン名を指定します。ログインでは、大文字と小文字が区別されます。</p> <p>-P <i>password</i> “sa” アカウントのパスワードを指定します。</p> <p>-s <i>sybase dir</i> SYBASE 環境変数に使用する値を指定します。</p> <p>-l <i>interfaces file</i> Adaptive Server に接続するときに検索する <i>interfaces</i> ファイルの名前とロケーションを指定します。</p> <p>-r <i>resource file</i> 指定されたリソース・ファイルを実行します。</p> <p>-v sqlloc のバージョン番号と著作権メッセージを表示して、終了します。</p>
使用法	<ul style="list-style-type: none">• SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、sqlloc を使用してください。• バージョン番号を表示するために -v パラメータしか使用しない場合を除き、DISPLAY 環境変数を設定してから sqlloc を呼び出してください。
パーミッション	sqlloc を使用できるのは、Sybase システム管理者のみです。

参照

マニュアル `sqlloc` の詳細については、各 *UNIX* プラットフォームに対応した『インストール・ガイド』を参照してください。

ユーティリティ [langinstall](#), [sqllocres](#)

sqllocres

説明	(UNIX のみ) リソース・ファイルを使用して、Adaptive Server の言語、文字セット、ソート順のデフォルトをインストールしたり、変更したりします。sqllocres は <code>SYBASE/SYBASE_OCS/bin</code> にあります。
構文	<pre>sqllocres [-S server] [-U user] [-P password] [-s sybase dir] [-l interfaces file] [-r resource file]</pre> <p>または</p> <pre>sqllocres -v</pre>
パラメータ	<p>-S server 接続する Adaptive Server の名前を指定します。</p> <p>-U user ログイン名を指定します。</p> <p>-P password “sa” アカウントのパスワードを指定します。</p> <p>-s sybase dir SYBASE 環境変数に使用する値を指定します。</p> <p>-l interfaces file Adaptive Server に接続するときを検索する interfaces ファイルの名前とロケーションを指定します。</p> <p>-r resource file 指定されたリソース・ファイルを実行します。</p> <p>-v sqllocres のバージョン番号と著作権メッセージを表示して、終了します。</p>
使用法	SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、sqllocres を使用してください。
パーミッション	sqllocres ユーティリティを使用できるのは、Sybase システム管理者のみです。
参照	sqllocres ユーティリティ・プログラムの詳細については、各 UNIX プラットフォームに対応した『インストール・ガイド』を参照してください。
	ユーティリティ langinstall , sqlloc

sqlsrvr

説明

(Windows のみ) 実行形式の Adaptive Server プログラムであるこのユーティリティは、`%SYBASE%\$%SYBASE_ASE%\$bin` にあります。

構文

```
sqlserver [-f] [-g] [-G] [-h] [-H] [-m] [-P] [-q] [-v] [-X]
          [-a path_to_CAPs_directive_file]
          [-b master_device_size] [k | K | m | M | g | G | t | T]
          [-c config_file_for_server]
          [-d device_name]
          [-e path_to_error_log]
          [-i interfaces_file_directory]
          [-K keytab_file]
          [-L config_file_name_for_connectivity]
          [--master_key_password [=password]]
          [-M shared_memory_repository_directory]
          [-p sa_login_name]
          [-r mirror_disk_name]
          [-s server_name]
          [-T trace_flag]
          [-u sa/sso_name]
          [-w master | model database]
          [-y [password]]
          [-z page_size [k | K]]
```

パラメータ

- f
デバイスまたはデータベースを強制的に初期化します。-f を使用する場合は、-b と -w も使用してください。
- g
イベント・ロギングを無効にします。
- G
イベント・ログ・サーバの名前を指定します。
- h
ヘルプ・メッセージを表示します。
- H
Adaptive Server に HA (高可用性) 機能をインストールしてある場合に、HA サーバを起動します。
- m
Adaptive Server をシングル・ユーザ・モードで起動します。
- q
クワイス状態のデータベースを「リカバリ中」として扱います。
- v
sqlsrvr のバージョン番号と著作権メッセージを表示して、終了します。

-X

このサーバを、`dataserver` ではなく `sybmon` として起動します。

-a *path_to_CAPs_directive_file*

CAP のディレクティブ・ファイルへのパスを指定します。

-b *master_device_size [k|K|m|M|g|G|t|T]*

マスタ・デバイスのサイズを指定します。

-c *config_file_for_server*

Adaptive Server 設定ファイルのフル・パス名を指定します。このパラメータを使用すると、指定した設定ファイルの設定値で Adaptive Server が起動します。sqlsrv -c パラメータを使用して設定ファイルを指定した場合、この設定ファイル内のすべてのパラメータに互換性があることを確認してからサーバを起動してください。設定パラメータの中に互換性のないものがあると、サーバが起動しないことがあります。これを回避するには、マスタ・デバイスを構築するときに設定ファイルを指定しないでください。設定ファイルを指定しない場合は、構築フェーズですべてのデフォルト設定が使用されます。

詳細については、『システム管理ガイド：第 1 巻

-d *device_name*

master データベースのデバイスのフル・パス名です。master データベースのデバイスは、Adaptive Server を起動するユーザが書き込み可能である必要があります。デフォルトの master データベース・デバイス名は、`d_master` です。

-e *errorlogfile*

Adaptive Server のシステム・レベルのエラー・メッセージに使用するエラー・ログ・ファイルのフル・パス名です。

-i *interfaces_file_directory*

Adaptive Server に接続するときに検索する `interfaces` ファイルがあるディレクトリのロケーションを指定します。-i を省略した場合、sqlsrv は SYBASE 環境変数によって指定されたディレクトリにある `interfaces` という名前のファイルを探します。

-K *keytab_file*

DCE での認証に使用する keytab ファイルのパスを指定します。

-L *config_file_name_for_connectivity*

コネクティビティ用の設定ファイルの名前を指定します。

- master_key_password [=password]**
コマンド・ラインで *password* を指定する場合や、Adaptive Server の起動時にマスタ・キーのパスワードの入力を要求するプロンプトが表示される場合は、マスタ・キーのパスワードを指定します。パスワード文字は表示されず、パスワードは Adaptive Server の起動順序の後半まで検証されません。
- コマンド・ラインでパスワードを指定すると、メモリが読み込まれて使用されるまでパスワードが表示されます。
- M sharedmem_directory**
共有メモリのファイルを、デフォルト・ロケーションの *%SYBASE%* ではなく、指定したディレクトリ内に置きます。*sharedmem_directory* が “*¥*” で始まる場合、そのディレクトリ名は絶対パス名として解釈されます。それ以外の場合は、ディレクトリ名は *%SYBASE%* の相対パスとして解釈されます。
- p sso_login_name**
アカウントの新しいパスワードを取得するために、Adaptive Server を起動するときのシステム・セキュリティ担当者のログイン名を指定します。Adaptive Server は、ランダムなパスワードを生成して表示し、暗号化してから、そのアカウントの新しいパスワードとして *master.syslogins* に保存します。
- r mastermirror**
マスタ・デバイスのミラーリングを開始します。マスタ・デバイスが破損している場合には、このパラメータを使用して Adaptive Server を起動します。
- s servername**
起動する Adaptive Server の名前を指定します。**-s** を省略すると、SYBASE という名前のサーバが起動します。
- T trace_flag**
- u sa/sso_name**
ロックを解除するシステム管理者またはシステム・セキュリティ担当者の名前を指定します。
- w master | model_database**
master データベースと *model* データベースのどちらを書き込むかを指定します。

-y [password]

ユーザは、暗号化されたプライベート・キーのパスワードを割り当てられます。サーバは、パスワードを入力するためのプロンプトを表示します。このパスワードはプライベート・キーの作成時にプライベート・キーを暗号化するために使用したパスワードと一致する必要があります。バックグラウンドでサーバを実行している場合、このパラメータは使用できません。

注意 セキュリティ上の理由から、**-y** を使用してパスワードを設定することはおすすめしません。

プライベート・キーはサーバのデジタル証明書に含まれています。デフォルトでは、証明書ファイルは次の場所にあります。

```
%SYBASE%\¥%SYBASE_ASE%\¥certificates¥servername.crt
```

`sp_ssladmin addcert` コマンドを呼び出すと、証明書ファイルのロケーションが変更されます。

-z page_size

サーバのページ・サイズを指定します。このフラグを使用する場合、**-b** と **-w** も使用し、**2K** ~ **16K** の間の 2 の偶数乗を指定してください。それ以外の場合は、サーバが起動しません。

例 **例 1** この例では、**100MB** のマスタ・デバイスと **4K** ページを持つ新規インストールを作成します。

```
sqlsrvr -d d_master -z 4k -b 100.02M
```

オプションとそれに続く引数の間にスペースを入れてもかまいません。この例では、サーバの設定領域に **16KB** のオーバーヘッドが必要であるため、**100MB** のマスタ・デバイスに “**100.02M**” を指定しています。

例 2 この例では、破壊された **model** データベースの再書き込みを行います。

```
sqlsrvr -d d_master -w model
```

例 3 この例では、破壊された **master** データベースの再書き込みを行い、デバイス・サイズを指定します。

```
sqlsrvr -d d_master -w master -z 4k
```

例 4 この例では、破壊された **master** データベースの再書き込みを行い、デバイス・サイズとページ・サイズを指定し、サーバが設定ブロックの値よりもこの値を優先して受け入れるように指定します。

```
sqlsrvr -d d_master -w master -z 4k -b 100.02M -f
```

例 5 この例では、破壊された **master** データベースの再書き込みを行い、サーバが設定ブロックで検出したサイズとは異なるページ・サイズを指定します。これによってエラーが発生します。

```
sqlsrvr -d d_master -w master -z 8k

00:00000:00000:2001/01/19 12:01:26.94 server The
configured server page size does not match that
specified on the command line.To use the configured
size, omit the command line size; to use the command
line size, specify 'force' (-f).
```

例 6 この例では、破壊された **master** データベースの再書き込みを行い、通常の起動でも、正しくないページ・サイズを指定します。これによってエラーが発生します。

```
sqlsrvr -d d_master -z4000

sqlsrvr:the 'z' flag may not be used without 'b' or
'w'. sqlsrvr:server will ignore the 'z' flag. sqlsrvr:
the 'z' flag contained an invalid page size. sqlsrvr:
the page size must be an even power of two between 2048
and 16384 bytes, inclusive.
```

使用法

- **sqlsrvr** ユーティリティは、Sybase の他のマニュアルでは **dataserver** に該当します。
- **Adaptive Server** を起動するには、**sqlsrvr** プログラムを直接実行するのではなく、**services manager** ユーティリティを使用します。デフォルト・パラメータを変更する場合は、**Adaptive Server** のレジストリ・キーを編集してください。詳細については、使用しているプラットフォームの『設定ガイド』を参照してください。
- **Adaptive Server** の実行環境は、**sysconfigures** システム・テーブルの値によって決定されます。設定値を表示するには、**sp_configure** を実行します。設定を変更するには、**sp_configure** と **reconfigure** を使用します。
- **Adaptive Server** のパスワードは暗号化されるため、パスワードを忘れた場合には回復することができません。システム・セキュリティ担当者全員が各自のパスワードを忘れてしまった場合は、**-p** パラメータを使用してシステム・セキュリティ担当者のアカウント用に新しいパスワードを生成できます。**-p** を指定して **Adaptive Server** を起動し、ランダムに生成された新しいパスワードで **Adaptive Server** にすぐにログインしてから、**sp_password** を実行してパスワードをより安全なものに設定し直してください。

- デフォルトでは、Adaptive Server は、ローカルのエラー・ログ・ファイルとローカルの Windows イベント・ログの両方にエラー・メッセージを記録します。`-g` パラメータを指定し、`-G machine_name` で別のイベント・ロギング・マシンを指定することにより、Windows のイベント・ロギングを無効にできます。`machine_name` は、Windows の標準の規則に従って入力してください。たとえば、“LOGSITE” という名前の PC を指定する場合、`machine_name` は “`¥¥LOGSITE`” とします。エラー・メッセージのロギングの詳細については、使用しているプラットフォームの『設定ガイド』を参照してください。
- インストーラを実行したあと、sqlsrv 実行プログラムにファイル・パーミッションを設定して、実行できるユーザを制限してください。
- `-s` パラメータで Adaptive Server 名を指定せず、DSLISNEN 環境変数も設定していない場合は、sqlsrv はデフォルトの Adaptive Server 名である SYBASE を使用します。DSLISNEN 環境変数の値によって、このデフォルト値は無効になります。`-s` パラメータを指定すると、デフォルト値と DSLISNEN 環境変数の両方が無効になります。
- 自動ログイン・ロックアウトを使用すると、サイトではログインをロック解除できるすべてのアカウント (システム管理者とシステム・セキュリティ担当者) がロックされた状態になることがあります。この状況が発生した場合は、`-u` パラメータを指定した sqlsrv ユーティリティを使用して、システム管理者権限またはシステム・セキュリティ担当者権限に指定したログインを確認し、アカウントのロックを解除して、失敗したログイン・カウンタの現在の値を 0 にリセットします。
- `-f` は、`-b` か `-w` またはその両方とともに使用する場合のみ有効です。`-b` または `-w` を使用しないで `-f` だけ使用しても、サーバは起動しません。`-w` の有無によって、`-f` によるサーバの動作が異なります。下記の `-b` と `-w` を参照してください。

Adaptive Server の起動

指定した設定ファイルを使用して Adaptive Server を起動するには、次の方法のいずれかを使用してください。

- Server Config を使用して、サーバが `-c` パラメータを持つように設定します。[Adaptive Server の設定] 画面で [コマンド行オプション] を選択し、[コマンド行パラメータ] 画面で、以下の情報を入力します。

`-Cconfiguration_file_pathname`

たとえば、“`-chaze.cfg`” と入力すると、`haze.cfg` 設定ファイルを使用してサーバを起動します。

- コマンド・ラインから Adaptive Server を起動し、`-c` パラメータを指定します。

`-b` と `-w` の依存性と条件

`-b` の影響は、`-w` の有無によって次のように異なります。

- `-w` を使用しないで `-b` を使用すると、作成される新しいマスタ・デバイスは、`-d` で指定した名前 (デフォルトは `d_master`) と `-z` で指定したページ・サイズ (デフォルトは `2048`) になります。
 - 指定したデバイスが、OS ファイルとしてすでに存在する場合、新しいデバイスの作成は失敗します。既存のファイルを削除してから再試行してください。
 - 指定したデバイスが既存のロー・パーティションである場合、`-f` フラグが指定されていないと失敗します。このフラグにより、サーバのマスタ・デバイスとしてそのロー・パーティションが再初期化されます。
- `-b` を使用して `-w master` も使用すると、`dataserver` に、マスタ・デバイスの再作成時には `-z` で指定したサイズを使用するよう指示されます。新しいデバイスの作成については何も示されません。

`-w` では、追加フラグが必要な場合と必要でない場合があります。

- `-w model` を使用する場合、`-z` フラグと `-b` フラグは許容されますが無視されます。
- `-w master` を「新しい」インストールに使用する場合、デバイス・サイズ情報が `config_block` に格納されているため、`-z` と `-b` は必要ありません。
- `-w master` を使用して以前のインストールを「アップグレード」する場合、次のようになります。
 - 関連するサイズの有効なエントリが `config_block` にない場合、サーバは `-b` か `-z`、またはその両方を要求します。ページ・サイズまたはデバイス・サイズの有効なデータを得られない場合、コマンドは失敗します。
 - ユーザが `-b` か `-z` またはその両方を指定できるのは、`config_block` にそれらのパラメータが示すサイズの有効なエントリがある場合です。ただし、サイズが `config_block` にあるサイズと一致しない場合は、`-f` を追加して新しい優先サイズを強制的に適用してください。

- `-f` は、`-b` と `-z` のどちらも指定していない場合に使用されることがあります。これは、`-f` によって、サーバに、損傷したアロケーション・ページを `master` データベースに属するものとして受け入れるよう指示されるためです。これは、ひどく破壊されたデータベースをリストアするのに役立ちます。`-w master -f` を指定した場合、サーバは `master` 以外のデータベースに属していない、指定されたマスタ・デバイス上のすべてのアロケーション・ページを `master` データベースに割り当てます。

パーミッション

このバイナリの実行パーミッションを持つユーザと、すべてのファイルに対する読み込み／書き込みアクセス権を持つユーザが使用できます。

使用するテーブル

`sysconfigures`

参照

コマンド `disk mirror`、`disk remirror`、`reconfigure`

システム・プロシージャ `sp_configure`、`sp_password`

ユーティリティ [startserver](#)

sqlupgrade

説明	(UNIX のみ) GUI を使用して、Adaptive Server の現在インストールされているバージョンを最新のリリースにアップグレードします。 sqlupgrade は \$SYBASE/\$SYBASE_ASE/bin にあります。
構文	sqlupgrade [-s sybase dir] [-r resource file] または sqlupgrade -v
パラメータ	-s sybase dir SYBASE 環境変数に使用する値を指定します。 -r resource file 指定されたリソース・ファイルを実行します。 -v sqlupgrade のバージョン番号と著作権メッセージを表示して、終了します。
使用法	<ul style="list-style-type: none">• SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、sqlupgrade を使用してください。• バージョン番号を表示するために -v パラメータしか使用しない場合を除き、DISPLAY 環境変数を設定してから sqlupgrade を呼び出してください。
パーミッション	sqlupgrade を使用できるのは、Sybase システム管理者のみです。
参照	マニュアル sqlupgrade の詳細については、各 UNIX プラットフォームに対応した『インストール・ガイド』を参照してください。 ユーティリティ preupgrade 、 sqlupgraderes

sqlupgraderes

説明	(UNIX のみ) リソース・ファイルを使用して、現在インストールされている Adaptive Server のリリースを最新のリリースにアップグレードします。sqlupgraderes は <code>\$SYBASE/\$SYBASE_OCS/bin</code> にあります。
構文	<pre>sqlupgraderes [-s sybase dir] [-r resource file] または sqlupgraderes -v</pre>
パラメータ	<p>-s Sybase Dir SYBASE 環境変数に使用する値を指定します。</p> <p>-r Resource File 指定されたリソース・ファイルを実行します。</p> <p>-v sqlupgraderes のバージョン番号と著作権メッセージを表示して、終了します。</p>
使用法	SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、sqlupgraderes を使用してください。
パーミッション	sqlupgraderes を使用できるのは、Sybase システム管理者のみです。
参照	マニュアル sqlupgraderes の詳細については、各 UNIX プラットフォームに対応した『インストール・ガイド』を参照してください。 ユーティリティ sqlupgrade

srvbuild

説明	(UNIX のみ) キー設定属性のデフォルト値またはユーザ指定の値を使用して、Adaptive Server、Backup Server、XP Server を新しく作成します。srvbuild は、GUI モードでも、非 GUI モードでも使用できます。srvbuild は <code>\$\$SYBASE/\$\$SYBASE_ASE/bin</code> にあります。
構文	<pre> srvbuild [-s sybase_dir] [-l interfaces_file] [-r resource_file] </pre> <p>または</p> <pre> srvbuild -v </pre>
パラメータ	<p><code>-s sybase_dir</code> SYBASE 環境変数に使用する値を指定します。</p> <p><code>-l interfaces_file</code> Adaptive Server に接続するときに検索する interfaces ファイルの名前とロケーションを指定します。</p> <p><code>-r resource_file</code> 指定されたリソース・ファイルを実行します。</p> <p><code>-v</code> srvbuild のバージョン番号と著作権メッセージを表示して、終了します。</p>
使用法	<p>SYBASE 環境変数を次のように設定してください。</p> <ul style="list-style-type: none"> SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、srvbuild を使用してください。 バージョン番号を表示するために、<code>-v</code> パラメータしか使用しない場合以外は、DISPLAY 環境変数を設定してから srvbuild を呼び出してください。 <p>64 ビット環境における srvbuild での LDAP の使用</p> <p>64 ビット環境で srvbuild を使用して、LDAP (Lightweight Directory Access Protocol) サービスを使用する新しいサーバを構築する場合は、LDAP サーバ・エントリを編集してください。</p> <p>srvbuild は 32 ビット・アプリケーションであり、<code>\$\$SYBASE/\$\$SYBASE_OCS/config/libtcl.cfg</code> ファイルの LDAP サーバ・エントリを使用します。Adaptive Server は 64 ビット・アプリケーションであり、<code>\$\$SYBASE/\$\$SYBASE_OCS/config/libtcl64.cfg</code> ファイルの LDAP サーバ情報を使用します。</p>

libtcl.cfg ファイルまたは *libtcl64.cfg* ファイルの LDAP サーバ・エントリの後にはスペースを含めないでください。スペースがあると、`srvbuild` は LDAP サーバに接続できなくなります。

パーミッション

`srvbuild` を使用できるのは、Sybase システム管理者のみです。

参照

マニュアル `srvbuild` の詳細については、各 *UNIX* プラットフォームに対応した『インストール・ガイド』を参照してください。LDAP を使用するサーバの設定方法の詳細については、『システム管理ガイド』の「第 11 章 Adaptive Server のログイン、データベース・ユーザ、クライアント接続の管理」を参照してください。

ユーティリティ [srvbuildres](#)

srvbuildres

説明	(UNIX のみ) リソース・ファイルを使用して、キー設定属性のデフォルト値またはユーザ指定の値を使って Adaptive Server、Backup Server、または XP Server を新しく作成します。srvbuildres は <code>\$\$SYBASE/\$\$SYBASE_ASE/bin</code> にあります。
構文	<pre> srvbuildres [-s sybase_dir] [-l interfaces_file] [-r resource_file] </pre> <p>または</p> <pre> srvbuildres -v </pre>
パラメータ	<p><code>-s sybase_dir</code> SYBASE 環境変数に使用する値を指定します。</p> <p><code>-l interfaces_file</code> Adaptive Server に接続するときには検索する interfaces ファイルの名前とロケーションを指定します。</p> <p><code>-r resource_file</code> 指定されたリソース・ファイルを実行します。 <code>\$\$SYBASE/\$\$SYBASE_ASE/init/sample_resource_files</code> ディレクトリに、次のサンプル・リソース・ファイルがあります。</p> <ul style="list-style-type: none"> • <code>sqlloc.rs*</code> • <code>sqlupgrade.adaptive_server.rs*</code> • <code>sqlupgrade.backup_server.rs*</code> • <code>sqlupgrade.monitor_server.rs*</code> • <code>srvbuild.adaptive_server.rs*</code> • <code>srvbuild.backup_server.rs*</code> • <code>srvbuild.job_scheduler.rs*</code> • <code>srvbuild.monitor_server.rs*</code> • <code>srvbuild.text_server.rs*</code> • <code>srvbuild.xp_server.rs*</code> <p><code>-v</code> srvbuildres のバージョン番号と著作権メッセージを表示して、終了します。</p>

使用法	SYBASE 環境変数を現在のバージョンの Adaptive Server のロケーションに設定してから、 srvbuildres を使用してください。
パーミッション	srvbuildres を使用できるのは、Sybase システム管理者のみです。
参照	マニュアル srvbuildres の詳細については、各 <i>UNIX</i> プラットフォームに対応した『インストール・ガイド』を参照してください。 ユーティリティ srvbuild

startserver

- 説明** (UNIX のみ) Adaptive Server または Backup Server を起動します。
startserver は \$SYBASE/\$SYBASE_ASE/bin にあります。
- 構文** startserver [[-f runserverfile] [-m]] ...
- パラメータ**
- f runserverfile
Adaptive Server または Backup Server を起動するたびに参照として使用される runserver ファイルの相対パス名を指定します。デフォルトでは、runserver ファイルは現在のディレクトリにあり、ファイル名は *RUN_servername* です。同じマシンで別の Adaptive Server を起動する場合、startserver は *RUN_servername* という名前の runserver ファイルを新たに作成します。
 - m
シングルユーザ・モードで Adaptive Server を起動し、1 人のシステム管理者だけがログインできるようにして、allow updates to system tables 設定パラメータを on にします。このモードは、master データベースをリストアする場合に使用してください。システム管理者は、sp_dboption の dbo use only パラメータを使用することによって、バルク・コピーやデータ・ディクショナリの使用など、複数のプロセスを必要とするシステム管理アクティビティを実行できます。startserver は、通常、1 つのノードにサーバを 1 つしか起動しません。
 - m パラメータは、*m_RUNSERVER* ファイルを作成し、既存の *m_RUNSERVER* ファイルを上書きします。
- 例**
- 例 1** この例では、現在のディレクトリにある *RUN_servername* という名前の runserver ファイルから SYBASE という名前の Adaptive Server を起動します。
- ```
startserver
```
- 例 2** MYSERVER という名前の Adaptive Server と SYB\_BACKUP という名前の Backup Server を起動します。
- ```
startserver -f RUN_MYSERVER -f RUN_SYB_BACKUP
```
- 例 3** SYB_BACKUP という名前の Backup Server だけを起動します。
- ```
startserver -f RUN_SYB_BACKUP
```
- 使用法**
- startserver は、runserver ファイルの情報を使用して Adaptive Server または Backup Server を起動します。マスタ・デバイスは、Adaptive Server を起動するユーザが書き込みできるようになっている必要があります。

startserver コマンドは、サーバを起動したディレクトリに Adaptive Server エラー・ログ・ファイル (*errorlog*) を作成し、この情報を runserver ファイルの Adaptive Server 実行ラインの *-e* パラメータの一部として追加します。同じマシンで Adaptive Server をもう 1 つ起動すると、*errorlog\_servername* という名前の新しいエラー・ログ・ファイルが作成され、この情報はそのサーバの runserver ファイルに追加されます。ユーザは指定した runserver ファイルに対する実行パーミッションを持っていなければなりません。

- 2 番目の例のように、複数の runserver ファイルを指定して、複数のサーバを起動できます。その場合は、各 *-f runserverfile* の後に *-m* を指定します。
- Adaptive Server の実行環境は、*config* ファイルの値によって決定されます。設定パラメータを表示するには、*sp\_configure* を実行し、変更するには *config* ファイルを編集します。
- Adaptive Server の整合性を維持するには、startserver 実行ファイルと runserver ファイルに適切なオペレーティング・システム保護を設定することが重要です。

#### runserver ファイル

- インストール時に *srvbuild* によって作成される runserver ファイルには、Adaptive Server を起動する *dataserver* コマンドまたは Backup Server を起動する *backupserver* コマンドがあります。デフォルトでは、runserver ファイルは現在のディレクトリにあり、ファイル名は *RUN\_servername* です。この runserver ファイルを開き、これらのコマンドのオプションやパラメータを変更することができます。次に、2 つの runserver ファイル例を示します。

サーバ MYSERVER の runserver ファイルは、次のようになります。

```
#!/bin/sh
#
Adaptive Server Information:
name: /MYSERVER
master device: /remote/Masters/myserver_dat
master device size: 10752
errorlog: /remote/serverdev/install/errorlog
interfaces: /remote/serverdev/interfaces
#
#
/$SYBASE/$SYBASE_ASE/bin/dataserver -d/remote/Masters/myserver_dat ¥
-sMYSERVER -e/remote/serverdev/install/MYSERVER_errorlog ¥
-i/remote/serverdev &
```

SYB\_BACKUP Backup Server の runserver ファイルは、次のようになります。

```
#!/bin/sh
#
Backup Server Information:
name: SYB_BACKUP
errorlog: /remote/serverdev/install/backup.log
interfaces: /remote/serverdev/interfaces
location of multibuf: /remote/serverdev/bin/sybmultbuf
language: us_english
character set: iso_1
tape configuration file: /remote/serverdev/backup_tape.cfg
#
/remote/serverdev/bin/backupserver -SSYB_BACKUP ¥
-e/remote/serverdev/install/backup.log ¥
-I/remote/serverdev/interfaces ¥
-M/remote/serverdev/bin/sybmultbuf -Lus_english -Jiso_1 ¥
-c/remote/serverdev/backup_tape.cfg
```

参照

**コマンド** disk mirror、disk remirror、disk unmirror

**ユーティリティ** [backupserver](#)、[dataserver](#)

## sybdiag

### 説明

sybdiag は、包括的な Adaptive Server 設定および環境データを収集する Java ベースのツールです。Sybase サポート・センタはこの情報を使用してサーバの問題を診断することにより、迅速なトラブルシューティングを実現します。

---

**注意** sybdiag は、モニタしている Adaptive Server と同じマシン上で実行してください。

---

### 構文

```
sybdiag -U username [-P password] -S [server_name | host.port]
 [-I interfaces_file]
 [-L log_file]
 [-N num_threads]
 [-O output_directory]
 [-R resource_file]
 [-T feature_list]
 [-h]
 [-m message_level]
 [-v]
```

### パラメータ

**-I interfaces\_file**

(オプション) interfaces ファイルの名前を指定する。-I を指定すると、sybdiag は、指定された *interfaces\_file* を使用して、ファイルが見つからない場合はエラーを表示します。-I を指定しない場合、sybdiag は最初にサーバ・エントリの LDAP サーバをチェックし、エントリが見つからない場合、sybdiag は、SYBASE 環境変数で指定されたディレクトリ内のデフォルトの *interfaces* ファイルを使用します。

**-L log\_file**

(オプション) sybdiag を作成するログ・ファイルの名前を指定します。log\_file が絶対パスでない場合、ログ・ファイルは sybdiag が実行されるディレクトリに作成されます。

このパラメータを指定するかどうかで、sybdiag は、.zip 出力ファイル内の *sybdiag.log* というデフォルトのログ・ファイルを作成します。

**-N num\_threads**

(オプション) sybdiag を並列で実行する最大並列スレッド数を指定します。デフォルト値は 5 です。並列の収集を実行できない限り、デフォルト値を変更しないでください。



**-O *output\_directory***

(オプション) **sybdiag** 出力を保管するローカル・ディレクトリの名前を指定します。名前を指定しない場合、**sybdiag** は、コマンドを実行したディレクトリ内に *.zip* 出力ファイルを作成します。出力ファイルの名前は「**sybdiag-*<server name>-<datetime stamp>.zip***」となります。

**-R *resource\_file***

(オプション) 起動時に **sybdiag** が使用するリソース・ファイルを指定します。このオプションは、Sybase 製品の保守契約を結んでいるサポート・センタの指示があった場合のみ使用してください。

**-P *password***

Adaptive Server のパスワードを指定します。**-P** フラグを指定しない場合、**sybdiag** はパスワードを入力するよう要求します。パスワードが NULL の場合は、パスワードを指定せずに **-P** フラグを使用してください。

**-S *server\_name* | *host:port***

*server\_name* は **sybdiag** が接続する Adaptive Server の名前を指定します。**sybdiag** は、*interfaces* ファイルまたは LDAP ディレクトリ内でこの名前を探します。

**-S** を引数なしで指定した場合、**sybdiag** は SYBASE という名前のサーバを探します。**-S** を指定しなかった場合、**sybdiag** は DSQUERY 環境変数によって指定されたサーバを探します。

*host:port* はマシン名とポート番号を指定します。

**-T *feature\_list***

(オプション) 次の *feature\_list* 値に基づいて、**sybdiag** が収集する診断データのタイプを指定します。

- **osdata** — オペレーティング・システム・データ。
- **asecore** — Adaptive Server の設定データ。
- **aseadd** — Adaptive Server 監視データ。
- **keyfile** — Adaptive Server に関する情報とオペレーティング・システム・ファイル。

**-T** を指定しないと、すべての診断データが収集されます。「[sybdiag 出力の表示](#)」(300 ページ)を参照してください。

**-U *username***

ログイン名を指定します。ログイン名では大文字と小文字が区別されます。

- h  
(オプション)すべてのヘルプ・オプションを表示します。
- m *message\_level*  
(オプション) *message\_level* の値によって異なるレベルのエラー・メッセージを表示します。
- 0 - メッセージなし。
  - 1 - 致命的なエラーのみ。
  - 2 - すべてのエラー。
  - 3 - 警告とすべてのエラー。
  - 4 - 情報メッセージ、警告、すべてのエラー。
  - 5 - デバッグと情報メッセージ、警告、すべてのエラー。
- デフォルトでは、レベルが 4 に設定されているエラー・メッセージを表示します。
- v  
(オプション)バージョン情報を表示します。

**例** **例 1** すべての診断を `/work/ASEInstall/ASE-15_0/bin` から収集し、`sybdiag-testserver-20110312024652.zip` という出力ファイルを同じディレクトリ内に作成します。

```
sybdiag -Usa -P -Stestserver
```

---

**注意** 読みやすくするために、この出力例では何行か省略しています。

---

```
Collecting data for "Adaptive Server Version" (ase_version) ...
Completed data collection for "Adaptive Server Version" (ase_version).
Collecting data for "Server License" (ase_license) ...
Completed data collection for "Server License" (ase_license).
Collecting data for "Adaptive Server Configuration" (ase_cfg) ...
Completed data collection for "Adaptive Server Configuration" (ase_cfg).
Collecting data for "Adaptive Server Non-default Configuration"
(ase_nondefault_cfg) ...
Completed data collection for "Adaptive Server Non-default
Configuration" (ase_nondefault_cfg).
...Lines deleted...
Collecting data for "Adaptive Server LDAP Configuration File" (ase_libtcl) ...
Completed data collection for "Adaptive Server LDAP Configuration File"
(ase_libtcl).
Collecting data for "Adaptive Server LDAP Configuration File (64bit)"
(ase_libtcl64) ...
Completed data collection for "Adaptive Server LDAP Configuration File
```

```
(64bit)" (ase_libtcl64).
Collecting data for "Virtual Memory Statistics" (os_vmstat) ...
Collecting data for "Adaptive Server General Performance Information"
(ase_sysmon) ...
Collecting data for "I/O Statistics" (os_iostat) ...
Collecting data for "CPU Statistics" (os_mpstat) ...
Completed data collection for "Virtual Memory Statistics" (os_vmstat).
Completed data collection for "I/O Statistics" (os_iostat).
Completed data collection for "CPU Statistics" (os_mpstat).
Completed data collection for "Adaptive Server General Performance
Information" (ase_sysmon).
Data collection statistics:43 task(s) succeeded, 0 task(s) skipped, and
0 task(s) failed.
The collected data is stored as
/work/ASEInstall/ASE-15_0/bin/sybdiag-testserver-20110312024652.zip
Data collection completed.
```

**例2** 基本の Adaptive Server 設定データを `/work/ASEInstall/ASE-15_0/bin` から収集し、`sybdiag-smmdi_9966-20110502202909.zip` という出力ファイルを同じディレクトリ内に作成します。

```
sybdiag -Usa -P -Ssmmdi:9966 -Tascore
Collecting data for "Adaptive Server Version" (ase_version) ...
Completed data collection for "Adaptive Server Version" (ase_version).
Collecting data for "Server License" (ase_license) ...
Completed data collection for "Server License" (ase_license).
Collecting data for "Adaptive Server Configuration" (ase_cfg) ...
Completed data collection for "Adaptive Server Configuration" (ase_cfg).
Collecting data for "Adaptive Server Non-default Configuration"
(ase_nondefault_cfg) ...
Completed data collection for "Adaptive Server Non-default Configuration"
(ase_nondefault_cfg).
Collecting data for "Remote Server Configuration" (ase_remote_server)...
Completed data collection for "Remote Server Configuration" (ase_remote_server).
Collecting data for "Adaptive Server Script Version" (ase_script_version) ...
Completed data collection for "Adaptive Server Script Version" (ase_script_version).
Collecting data for "Adaptive Server Configuration Monitor" (ase_mon_cfg) ...
Completed data collection for "Adaptive Server Configuration Monitor" (ase_mon_cfg).
Collecting data for "Adaptive Server Cache Configuration" (ase_cache_cfg) ...
Completed data collection for "Adaptive Server Cache Configuration" (ase_cache_cfg).
Collecting data for "Adaptive Server Pool Configuration" (ase_pool_cfg) ...
Completed data collection for "Adaptive Server Pool Configuration" (ase_pool_cfg).
Collecting data for "Adaptive Server Shared Memory Dump Configuration"
(ase_shmdumpconfig) ...
Completed data collection for "Adaptive Server Shared Memory Dump
Configuration" (ase_shmdumpconfig).
Collecting data for "Adaptive Server Traceflags and Switches" (ase_switches) ...
Completed data collection for "Adaptive Server Traceflags and Switches"
(ase_switches).
Data collection statistics:11 task(s) succeeded, 0 task(s) skipped, and
0 task(s) failed.
```

The collected data is stored as /work/ASEInstall/ASE-15\_0/bin/sybdiag  
 smmdi\_9966-20110502202909.zip  
 Data collection completed.

## 使用法

**sybdiag** は Adaptive Server に接続して **sp\_configure** のようなストアド・プロシージャを実行し、**monLicense** のようなテーブルにクエリを送ります。**ps**、**vmstat**、**netstat** のようなコマンドを実行することにより、オペレーティング・システムとプラットフォームの診断情報を収集します。

**sybdiag** の出力は、Web ブラウザで圧縮解除して表示できる HTML ファイルとデータ・ファイルを含む圧縮 ZIP ファイルです。収集される情報には、オペレーティング・システムと環境データ、Adaptive Server の設定とモニタリング・データ、Adaptive Server ファイルとスクリプトが含まれます。

**sybdiag** では、ログイン、パスワード、ユーザ・リストに関する Adaptive Server データやオペレーティング・システム・データに加え、アプリケーション・データベース・テーブルからの情報を収集しません。

### **sybdiag** の設定オプション

特定のレポートを生成するには、**sp\_configure** を使用するかサーバ設定ファイルを編集して、対応する設定オプションを有効にする必要があります。設定オプションが正しく設定されていないと、**sybdiag** は関連するレポートをスキップします。

| オプション                      | 値                                           | <b>sybdiag</b> のレポート                                                        |
|----------------------------|---------------------------------------------|-----------------------------------------------------------------------------|
| enable monitoring          | 1                                           | Adaptive Server の待機イベント<br>Adaptive Server ロック<br>Adaptive Server のデッドロック履歴 |
| wait event timing          | 1                                           | Adaptive Server の待機イベント                                                     |
| deadlock pipe active       | 1                                           | Adaptive Server のデッドロック履歴                                                   |
| deadlock pipe max messages | 0 以外の値。すべての関係するデッドロック・ローを取得できる大きさである必要があります | Adaptive Server のデッドロック履歴                                                   |

### **sybdiag** 出力の表示

**sybdiag** 出力は、**sybdiag-servername-datetime\_stamp.zip** のフォーマットの圧縮ファイルにあります。個々の出力ファイルを生成するには、zip ファイルを圧縮解除します。

ZIP ファイルには、次の HTML ファイル、データ・ファイル、ログ・ファイルがあります。

- *sybdiag\_start.html* – 出力ディレクトリ内の診断データ・ファイルへのリンクを持つ HTML ファイル。*sybdiag* 出力を表示するには、このファイルを Web ブラウザで開きます。

*sybdiag* には、次のカテゴリで情報が表示されます。

- プロセス・ステータス、物理メモリ、仮想メモリなどのオペレーティング・システム情報、プロセス間通信、ディスク使用状況、I/O、ネットワーク情報。
- サーバ・バージョン、プラットフォームおよびライセンス情報、設定値、リモート・サーバ設定データなどの Adaptive Server 設定データ。Cluster Edition の場合は、クラスタの概要、クラスタ・インスタンス、論理クラスタ情報も含まれます。
- プロセス、データベース、デバイス、ロックなどに関する Adaptive Server モニタリング・データ。Cluster Edition の場合は、クラスタ間通信のプロトコル情報、クラスタ・ロックの使用状況、クラスタ・クォラム・デバイスのダンプ情報も含まれます。
- *errorlog*、*interfaces*、設定ファイル、SySAM のプロパティ・ファイル、環境設定スクリプトなどの Adaptive Server ファイル。Cluster Edition の場合、1 つのレポートに、いくつかのクラスタ・インスタンスの多数の外部ファイルからの情報が含まれている場合があります。
- 診断データ・ファイル – 収集した Adaptive Server 情報および異なるディレクトリで構成される環境情報。これらのファイルは *sybdiag\_start.html* がアクセスするファイルです。
- ログ・ファイル – デフォルトでは、ZIP ファイルには、*sybdiag* によって実行されたアクティビティのログを提供する *sybdiag.log* というログ・ファイルが含まれます。
- すべての Adaptive Server データを収集するには、*sybdiag* で使用されたすべてのデータソースにアクセスする必要があります。デフォルトの Adaptive Server 設定では、Adaptive Server の設定およびモニタリング・データを収集するには、*sa\_role* と *mon\_role* が必要です。*sybdiag* ユーザに *sa\_role* と *mon\_role* を付与するには、*sp\_role* を使用します。

```
sp_role "grant", sa_role, sybdiag_user
go
sp_role "grant", mon_role, sybdiag_user
go
```

## パーミッション

システム管理者がデフォルトのアクセス制限を変更した場合は、データベース・オブジェクトにアクセスする必要がある場合もあります。特定のデータベース・オブジェクトにアクセスするための十分なパーミッションを持っていない場合は、アクセスできないデータベース・オブジェクトをリストするエラー・メッセージが表示されます。

- すべての環境データを収集するには、オペレーティング・システム・ファイルとデバイス・ファイルにアクセスする権限を持つ必要があります。

---

**注意** Linux では、`/var/log/messages` のオペレーティング・システム・メッセージにアクセスするための読み込みパーミッションが必要です。

---

- 次のファイルでは、読み込みパーミッションが必要です。
  - Adaptive Server のエラー・ログ
  - Adaptive Server 設定ファイル
  - `SYBASE.csh`、`SYBASE.sh`、または `SYBASE.bat` などの環境スクリプト。
- `-O` パラメータで指定された出力ディレクトリの書き込みパーミッションが必要です。

---

**注意** ファイルに必要なパーミッションがない場合、`sybdiag` には、そのファイルを処理しないというエラー・メッセージが表示されます。

---

## sybmigrate

### 説明

sybmigrate では、Adaptive Server のページ・サイズを異なるページ・サイズに変換したり、プラットフォーム間でマイグレートしたりできます。実行ファイルは次のディレクトリにあります。

- (UNIX プラットフォームの場合)  
\$SYBASE/\$SYBASE\_ASE/bin/sybmigrate
- (Windows の場合) %SYBASE%\bin\sybmigrate.bat

### 構文

```
sybmigrate [-v] [-h] [-f]
 [-D 1 | 2 | 3 | 4]
 [-l interfaces_file]
 [-r input_resource_file]
 [-m setup | migrate | validate | report]
 [-rm status | space_est | repl | diff | password]
 [-l log_file]
 [-t output_template_resource_file]
 [-J client_charset]
 [-z language]
 [-T trace_flags]
 [-Tase trace_flags]
 [-f]
```

### パラメータ

- v  
バージョンの文字列を印刷して、終了します。
- h  
ヘルプ情報と構文の使用方法を印刷して、終了します。
- f  
ロック・セッションを無効にします。
- D  
sybmigrate のデバッグ・レベルを設定します。デフォルトのデバッグ・レベルは 2 です。
- l  
サーバ名を検索する *interfaces* ファイルを指定します。*interfaces* ファイルのロケーションが指定されていない場合、sybmigrate では次のロケーションを使用します。
  - UNIX の場合 - \$SYBASE/*interfaces*
  - Windows の場合 - %SYBASE%\bin\sql.ini

-r

マイグレーション・プロセスでリソース・ファイル・モードを使用することを指定します。-r パラメータを使用して入力リソース・ファイルを指定しない場合、sybmigrate は GUI モードで動作します。

-m

実行するオペレーションのタイプを指定します。

- **setup** — レポジトリとマイグレーション作業データベースを設定し、サーバ全体のデータをマイグレートします。
- **migrate** — データとオブジェクトのマイグレーションを実行します。
- **validate** — マイグレートしたオブジェクトを検証します。
- **report** — 5 つのレポートのいずれかを実行します。レポートは GUI モードとリソース・ファイル・モードで実行できます。以下のレポートを実行できます。
  - **status** — マイグレート・オブジェクト・ステータス・レポートは、マイグレートされたオブジェクトに関する情報を提供します。
  - **space\_est** — ターゲット・データベース領域見積もりレポートを使用して、ターゲット・データベースに十分なリソースが割り当てられているかどうか確認します。
  - **repl** — 複写レポートを使用して、マイグレートされた明示的な複写オブジェクトをチェックし、複写システムのタイプを確認し、ユーザがターゲット Adaptive Server と Replication Server 上で実行する SQL コマンドを生成します。
  - **diff** — 送信元データベースとターゲット・データベース間のオブジェクトを確認します。diff レポートでは、次のオブジェクト・タイプに関する情報が出力されます。
    - サーバ情報
    - データベース情報
    - DDL オブジェクト
    - ユーザ・テーブル・データ
- **password** — 変更されたパスワード用のファイルを作成します。

-m

生成するレポートのタイプを指定します。-m を指定しない場合、5 つのレポートがすべて実行されます。



- l  
マイグレーション・プロセスの出力が格納されるユーザ定義のログ・ファイルを指定します。`-l`を使用しない場合、ログは `$$SYBASE/$SYBASE_ASE/init/logs` または作業ディレクトリに格納されます。
- t  
`sybmigrate` のリソース・ファイル・モードでこれ以降のマイグレーションに使用する出力テンプレート・リソース・ファイルの生成を指定します。
- J  
Adaptive Server 接続に使用する文字セットを指定します。
- z  
Adaptive Server 接続に使用する言語を指定します。
- T  
コマンド・ライン・トレース・フラグを設定します。
- Tase  
`sybmigrate` で開いたすべての Adaptive Server 接続の Adaptive Server トレース・フラグ (`dbcc traceon` を使用してオンにしたフラグ) を実行します。カンマで区切ったリストでトレース・フラグを指定してください。

## 例

**例 1** `status` レポートを実行します。

```
sybmigrate -r resource file -m report -rn status
```

**例 2** リソース・ファイル・モードで `space_est` レポートを実行します。

```
sybmigrate -r resource file -m report -rn space_est
```

**例 3** `repl` レポートを実行するには、次のコマンドを発行します。

```
sybmigrate -r resource file -m report -rn repl
```

## 使用法

- `allow resource limits` 設定パラメータが 0 に設定されていることを確認してから、`sybmigrate` を実行してください。
- ターゲット Adaptive Server にメタデータがすでに存在する場合、サーバ・データをマイグレートすることはできません。
- `sybmigrate` でセッションが適切に終了しなかった場合は、`-f` を使用して、作成された送信元データベースとターゲット・データベースのバインドを無効にし、`sybmigrate` の 1 セッションだけが送信元／ターゲット・データベース・パス上で実行されるようにします。

- `-r` パラメータを使用する場合は、`-m` 引数を使用して、実行するオペレーションのタイプとして `setup`、`migrate`、`validate`、または `report` を指定する必要があります。リソース・ファイル・モードで、マイグレーション・プロセス全体を実行するか、プロセスの一部だけを実行することができます。
- `-t` はリソース・ファイル・モードでのみ使用できます。  
`-t` を指定するには、ログイン情報を指定する `-r` 引数を使用して、`sybmigrate` を起動する必要があります。この引数を使用するには、`-m` を使用して、生成するリソース・ファイルのタイプも指定する必要があります。
- Adaptive Server バージョン 12.5.3 以降を使用すると、ターゲット・サーバ上のワーク・データベースのサイズとロケーションを指定できます。
- Adaptive Server バージョン 12.5.1 は、今後はサポートされません。12.5.1 から 12.5.4 にアップグレードした後で 12.5.4 から 15.5 以降のバージョンにアップグレードすることをおすすめします。12.5.1 のサーバを 12.5.4 以降のサーバが認識できる状態にするには、以下を使用します。

```
sp_addserver 'servername', local
```

12.5.1 サーバを認識するには、Adaptive Server を再起動します。

- LDAP エントリが次のロケーションに定義されている場合は、`sybmigrate` を無効にし、`-l` 引数を指定して `interfaces` ファイルを使用できます。
  - UNIX の場合 — `$SYBASE/$SYBASE_OCS/config/libtcl.cfg`
  - Windows の場合 — `%SYBASE%\¥%SYBASE_OCS%\¥ini ¥libtcl.cfg`
- `sybmigrate` は、次の場合に述部付きの権限を自動的にマイグレートします。
  - `ddlgen` が呼び出されて、付与、拒否、取り消しなどのスキーム定義を生成する。
  - CIS とプロキシ・テーブルを使用してデータをマイグレートする。

#### レポート

- `status` — マイグレート・オブジェクト・ステータス・レポートは、マイグレートされたオブジェクトに関する情報を提供します。

- **space\_est** – ターゲット・データベース領域見積もりレポートを使用して、ターゲット・データベースに十分なリソースが割り当てられているかどうか確認します。
- **repl** – 複写レポートを使用して、マイグレートされた明示的な複写オブジェクトをチェックし、複写システムのタイプを確認し、ユーザがターゲット **Adaptive Server** と **Replication Server** 上で実行する SQL コマンドを生成します。
- **diff** – 送信元データベースとターゲット・データベース間のオブジェクトを確認します。ユーザは、サーバとデータベースの情報またはメタデータを除き、個々のオブジェクトまたはデータベース全体に対してこのレポートを実行できます。**diff** レポートはいつでも実行できます。**diff** レポートを実行するために、設定セッションを実行する必要はありません。**diff** レポートの実行時、送信元データベース名とターゲット・データベース名が同じである必要はありません。

**diff** レポートでは、次のオブジェクト・タイプに関する情報が出力されます。

- **サーバ情報** – 送信元 **Adaptive Server** とターゲット **Adaptive Server** 間で **master** データベースのシステム・カタログのロー・カウントを比較します。このタスクは検証セッションと似ています。
- **データベース情報** – 送信元 **Adaptive Server** とターゲット **Adaptive Server** 間でユーザ・データベースのシステム・カタログのロー・カウントを比較します。このタスクは検証セッションと似ています。
- **DDL オブジェクト** – オブジェクトが送信元 **Adaptive Server** とターゲット **Adaptive Server** のどちらに存在するかをレポートに表示します。オブジェクトが両方のデータベースに存在する場合、そのオブジェクトはレポートに表示されません。
- **ユーザ・テーブル・データ** – 送信元 **Adaptive Server** とターゲット **Adaptive Server** にあるユーザ・テーブルのロー・カウントを比較します。テーブルが送信元データベースとターゲット・データベースのどちらかにしか存在しない場合、そのテーブルはレポートに表示されません。

#### パーミッション

**sybmigrate** を使用できるのは、Sybase システム管理者と **sa\_role** でログインしたユーザのみです。

#### 参照

**マニュアル** **sybmigrate** の詳細については、「[第9章 マイグレーション・ユーティリティ](#)」を参照してください。

## sybtsspasswd

### 説明

ユーザのパスワードを記録および変更し、TSM (Tivoli Storage Manager) 暗号化パスワード・ファイル *TSM.PWD* を TSM クライアント・マシンに作成します。このファイルのロケーションは、TSM 設定ファイルの *PASSWORDDIR* 設定パラメータによって指定されたディレクトリです。

---

**注意** sybtsspasswd がサポートされるのは、そのサイトで IBM Tivoli Storage Manager がライセンスされている場合です。

---

### 構文

sybtsspasswd

### 例

sybtsspasswd がパスワード情報を入力するよう要求します。

```
sybtsspasswd
```

```
Enter your current password:
```

```
Enter your new password:
```

```
Enter your new password again:
```

```
Your new password has been accepted and updated.
```

### 使用法

- sybtsspasswd は、ユーザのパスワードを記録または変更するときに実行します。同じユーザ名とパスワードが TSM クライアント・ノードの *TSM.PWD* ファイルに存在することを確認してください。

### パーミッション

sybtsspasswd を実行できるのはオペレーティング・システムの “root” ユーザだけです。

## xpserver

説明 XP Server を手動で起動します。

構文 `xpserver -S XP_Server`

```
xpserver
-S XP_Server
 [-i interfaces_file]
 [-p priority]
 [-s stack_size]
 [-u]
 [-v]
 [-x]
```

### パラメータ

`-S XP_Server`

起動する XP Server の名前を指定します。XP Server 名のフォーマットは `SQLSERVERNAME_XP` です。`SQLSERVERNAME` には、XP Server が割り当てられている Adaptive Server の名前が入ります。たとえば、SMOKE という名前の Adaptive Server の XP であれば、SMOKE\_XP となります。XP Server 名は大文字にしてください。

`-i interfaces_file`

XP Server に接続するときに Adaptive Server が検索する interfaces ファイル (`sql.ini`) のあるディレクトリの名前とロケーションを指定します。`-i` を指定しない場合は、xpserver は `%SYBASE%` リリース・ディレクトリの `ini` サブディレクトリを使用します。

`-p priority`

Open Server プロセスの優先度を指定します。0 (最低) から 15 (最高) までの値が有効です。esp execution priority 設定パラメータを上書きします。デフォルトは 8 です。

`-s stack_size`

拡張スタアド・プロシージャ (ESP) の実行に使用するプロセスのスタック・サイズをバイト単位で指定します。このパラメータを指定すると、esp execution stacksize 設定パラメータは無効になります。デフォルトは 34816 バイトです。

`-u`

ESP 要求の終了後に、XP Server のメモリから関数が自動的にアンロードされるように指定します。このパラメータを指定すると、esp unload dll 設定パラメータは無効になります。デフォルトでは関数をアンロードしません。

`-v`

XP Server のバージョン番号と著作権メッセージを表示して、終了します。

-x

システム ESP (xp\_cmdshell) を使ってオペレーティング・システム・コマンドを実行するためにクライアント・セキュリティ・コンテキストが使用されるように指定します。ここでの指定は、xp\_cmdshell context 設定パラメータよりも優先されます。デフォルトでは Adaptive Server セッションのオペレーティング・システム・アカウントのセキュリティ・コンテキストが使用されます。

## 使用法

- XP erVer は、通常は Adaptive Server によって自動的に起動されます。手動コマンドを使用して XP Server を起動するのは、「XP Server Failed to Start」エラー・メッセージで手動による起動を指示された場合だけです。
- XP erVer は、Adaptive Server サーバごとに 1 つしか存在できません。複数の ESP を実行している Adaptive Server は 1 つの XP Server のみと通信し、それらの ESP は同期的に実行されます。
- -p パラメータは、Open Server スケジューラが使用する優先度に影響します。-p の設定が多い場合は、スケジューラは XP Server を実行してから実行キューにあるその他のスレッド in を実行します。-p に設定されている数値が小さい場合は、スケジューラは実行キューに他の Open Server スレッドがないときにだけ XP Server を実行できます。このパラメータは、sp\_bindexclass によって設定された Adaptive Server 内のアプリケーション・キュー優先度には関係ありません。

Open Server スレッドのスケジュールについては、『Open Server Server-Library/C リファレンス・マニュアル』のマルチスレッド・プログラミングの説明を参照してください。

- -u パラメータまたは esp unload dll 設定パラメータによって ESP 関数の自動アンロードが設定されていない場合は、sp\_freeldll を使用して実行時に関数をアンロードできます。
- Adaptive Server や Backup Server とは異なり、XP Server には runserver ファイルはありません。
- XP erVer を設定するときは、“abcdef\_XP” や “ABCDEF\_XP” のように、ディレクトリ・サービス・エントリ名の最後に大文字で “\_XP” を付けます。

## パーミッション

xpserver の実行には、特別なパーミッションは必要ありません。

## 参照

**システム ESP** xp\_cmdshell

**システム・プロシージャ** sp\_configure、sp\_freeldll

## ***sybcluster*** を使用したクラスタの 管理

この章では、共有ディスク・クラスタの作成、起動、管理のほか、クラスタ内のインスタンスの管理に使用できるコマンド・ライン・ユーティリティである ***sybcluster*** について説明します。また、***sybcluster*** で使用できるパラメータとオプションについて説明します。

***sybcluster*** は、共有ディスク・クラスタ環境でのみ使用できます。***sybcluster*** の使用方法については、『Cluster ユーザーズ・ガイド』を参照してください。

# sybcluster

**説明** Sybase 共有ディスク・クラスタを管理します。sybcluster では、クラスタ、またはクラスタ内の任意のインスタンスを作成、起動、停止、管理できます。

**構文** sybcluster

```
[-C cluster_name]
[-d discovery_list]
[-F agent_connection]
[-h]
[-l instance_name]
[-i input_file_path]
[-L]
[-m message_level]
[-P [password]]
[-U user_name] (デフォルト値は “uafadmin”)
[-v]
```

**パラメータ**

**-C *cluster\_name***

管理対象の Sybase 共有ディスク・クラスタのユニークな名前です。sybcluster は、クラスタ・ディレクトリで名前を検索するか、またはエージェントの検出サービスを使用します。

**-d *discovery\_list***

共有ディスク・クラスタ・エージェントの検出に使用する検出サービスと検出順序を指定します。表 8-1 は、Cluster Edition でサポートされる検出サービスを示します。フォーマットは次のとおりです。

```
"method[(method_specification)[, ...]]"
```

次に例を示します。

```
-d "udp(),jini(jinihost1;jinihost2)"
```

**表 8-1 : 検出方法**

| 検出方法                     | 説明                                                                                                                                                                                                                                                                                                        |
|--------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UDP()                    | UDP ブロードキャストを実行し、受信中の Unified Agent からの応答を待機します。UDP 検出は、サブネットの境界を越えることはありません。                                                                                                                                                                                                                             |
| JINI( <i>JINI_spec</i> ) | <p>クラスタ内のノードのロケーションの検索に使用する JINI サーバを指定します。フォーマットは次のとおりです。<i>host-name[:port_num]</i></p> <p>複数の JINI サーバを指定するには、それぞれの指定の間をセミコロンで区切ります。デフォルトでは、sybcluster はポート番号 4160 を使用して JINI サーバに付加します。</p> <p>JINI サーバが実行中であり、管理エージェント (UAF) が JINI サーバに登録されている必要があります。ノードのロケーションとインスタンスのステータスは JINI サーバに格納されます。</p> |



| 検出方法            | 説明                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| LDAP(LDAP_spec) | <p>クラスタ内のノードのロケーションの検索に使用する LDAP サーバを指定します。フォーマットは次のとおりです。 <code>host_name[:port_num][?registry]</code></p> <p>複数の LDAP サーバを指定するには、それぞれの指定の間をセミコロンで区切ります。デフォルトでは、<code>sybcluster</code> はポート番号 389 を使用して LDAP サーバに付加し、LDAP ディレクトリは “<code>cn=ua-registry,ou=ua,dc=sybase,dc=com</code>” になります。</p> <p><b>-F agent_connection</b><br/>           クラスタへのアクセスに使用するエージェントを指定します。<br/>           フォーマットは次のとおりです。</p> <pre>host_name[:port_num] [, host_name[:port_num ]]</pre> <p>次に例を示します。</p> <pre>-F "node1,node2,node3,node4:9999"</pre> <p>デフォルトのポート番号は 9999 です。</p> <p><b>-h</b><br/> <code>sybcluster</code> の構文を表示し、サポートしている対話型コマンドを示します。</p> <p><b>-l instance_name</b><br/>           アクセスするインスタンスを指定します。<code>sybcluster</code> の実行時に <b>-l</b> オプションを指定しないと、特定の対話型コマンドを入力するとき、その指定が必要になる場合があります。<code>sybcluster</code> は、この名前を使用してリモート・ホストを検出し、対話型コマンドを実行するときのデフォルトとして使用します。対話型コマンドが複数のインスタンスに影響を及ぼす場合、<b>-l</b> オプションで指定されたインスタンスがあると、そのインスタンスが優先的な接続先として使用されます。</p> <p><b>-l</b> で指定されたインスタンスを上書きするには、対話型モードで <code>use</code> コマンドを実行します。</p> <p><b>-i</b><br/> <code>sybcluster</code> へ入力するオペレーティング・システム・ファイルを指定します。このファイルは、<code>sybcluster</code> コマンドを 1 行に 1 コマンドずつ記述したものです。ファイルの最後のコマンドは、<code>quit</code> である必要があります。</p> <p><b>-L</b><br/> <code>sybcluster.log</code> ファイルを作成します。<code>sybcluster</code> は、<b>-m</b> オプションで設定されたメッセージ・レベルに関係なく、すべてのメッセージをこのファイルに書き込みます。</p> |

**-m message\_level**

クライアントのコンソールに表示される **sybcluster** メッセージおよび **Unified Agent** メッセージを指定します。メッセージ・レベルは次のとおりです。

- 0 – オフ (ログ・ファイルにもコンソールにもメッセージなし)
- 1 – 致命的
- 2 – エラー
- 3 – 警告
- 4 – 情報
- 5 – デバッグ

**sybcluster** は、選択したレベルのすべてのメッセージと、選択したレベルよりも重大度の大きい (番号が小さい) すべてのメッセージを表示します。つまり、メッセージ・レベルとして 3 を選択した場合、**sybcluster** はメッセージ・レベル 3、2、1 のメッセージを表示します。デフォルトのレベルは 4 です。

**-P [password]**

**Unified Agent** フレームワークにある **Sybase Common Security Infrastructure** の管理エージェントのパスワードです。インストール後のデフォルトのユーザ名は “**uafadmin**” で、パスワードは設定されていません。これは、**Agent** 設定にある簡易ログイン・モジュールです。実行中のインスタンスの使用やオペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。

**-P** オプションを指定しないと、**sybcluster** はパスワードの入力を求めます。パスワードがブランクまたは **null** の場合は、**-P** オプションで値を指定しないか、中身の無い空の引用符の組を入力します。

**Sybase passencrypt** ユーティリティを使用すると、パスワードを暗号化できます。『**Cluster ユーザーズ・ガイド**』を参照してください。

**-U user\_name**

管理エージェントのログイン名です。インストール後のデフォルトのログインは “**uafadmin**” です。詳細については、**-P** の説明を参照してください。

**-v**

**sybcluster** のバージョン番号などの情報を表示します。

例

**例 1** 直接接続を使用し、デフォルトのポート番号で sybcluster を起動します。

```
sybcluster -U uafadmin -P -C mycluster
-F "blade1,blade2,blade3"
```

**例 2** 直接接続を使用し、ポート番号を指定して sybcluster を起動します。

```
sybcluster -U uafadmin -P -C mycluster
-F "blade1:9100,blade2:9292,blade3:9393"
```

**例 3** 検出を使用して sybcluster を起動することもできます。「[検出方法](#)」(312 ページ)を参照してください。

```
sybcluster -U uafadmin -P -C mycluster
-d "JINI(myjiniserver:4564)"
```

使用法

**sybcluster の起動**

sybcluster を起動し、クラスタに接続する方法として、次を推奨します。

```
sybcluster -U login_name -P password -C cluster_name
-F agent_spec
```

`-C cluster_name`、`-P password`、`-I instance_name`、`-F agent_connection`、`-d discovery_list` の各パラメータはデフォルト値を指定します。このデフォルト値は、後述の sybcluster の対話型コマンドを使用して変更できます。これらの値を sybcluster コマンド・ラインで指定しないと、必要に応じて sybcluster から値を入力するよう要求されます。

sybcluster を起動してから、対話型の connect コマンドを使用してクラスタに接続することもできます。次に例を示します。

```
sybcluster
> connect to mycluster login uafadmin password " "
agent "blade1,blade2,blade3"
```

---

**注意** sybcluster の対話型コマンドの構文と使用方法については、「[sybcluster の対話型コマンド](#)」を参照してください。

---

## sybcluster の対話型コマンド

対話型コマンドには、クラスタに接続する前にアクティブであるもの(「[クラスタに接続する前にアクティブなコマンド](#)」(316 ページ)を参照)とクラスタに接続した後でのみアクティブになるもの(「[クラスタに接続した後でアクティブなコマンド](#)」(317 ページ)を参照)があります。

現在のクラスタとデフォルトのインスタンスの値が設定されていると、`sybcluster` のコマンド・プロンプトには、これらの値が表示されません。プロンプトは次のようになります。

- `> — sybcluster` がクラスタに接続されていない場合
- `cluster_name>` — `sybcluster` がクラスタに接続されている場合
- `cluster_name instance_name>` — `sybcluster` がクラスタに接続されており、デフォルトのインスタンスが設定されている場合

クラスタに接続する前にアクティブなコマンド

次のコマンドは、クラスタに接続する前にアクティブになっていません。クラスタに接続した後は使用できません。

Connect

```
[to cluster_name]
[Login login_name]
[Password [password]]
[Agent agent_spec]
[Discovery discovery_spec]
```

Create Cluster [cluster\_name]

```
[Login login_name]
[Password password]
[Agent agent_spec]
[Discovery discovery_spec]
[File input_file]
```

Deploy Plugin

```
[Login login_name]
[Password password]
[Agent agent_spec]
[Discovery discovery_spec]
```

Exit

Help

Quit

Show agents

```
[login login_name]
[password password]
[agent “agent_spec[, agent_spec[,...]]”]
[discovery “discovery_spec[, discovery_spec[,...]]”]
```

Upgrade server server\_name

```
[login login_name]
[password password]
[agent agent_spec]
[discovery discovery_spec]
[file input_file_name]
[checkonly]
```

クラスタに接続した  
後  
でアクティブなコマ  
ンド

次のコマンドは、クラスタに接続した後でのみアクティブになります。

```

Add
 Instance instance_name
 file file_name
Backupserver

Create
 Backupserver
 Monitorserver
 Xpserver

Disconnect

Diagnose
 Cluster
 Instance instance_name

Drop
 Backupserver
 Cluster
 Instance instance_name
 Monitorserver
 Xpserver

Localize

Set
 Cluster
 MaxInst max_num_instances
 Login [login_name] [Password password]
 TraceFlags trace_flag
 Primary Protocol protocol
 Secondary Protocol protocol
 Instance [instance_name]
 LogPath log_file_path
 Primary
 Address ip_address
 Port port_range_start port_range_end
 Secondary
 Address ip_address
 Port port_range_start port_range_end
 StartArgs startup_arguments
 BackupServer
 MonitorServer
 XPServer
 Port

Show
 Cluster
 Config
 Template
 Log

```

```
 [Errors]
 [MinSeverity severity_level]
 [StartDate [date_string]]
 [EndDate [date_string]]
 [Last num_of_lines]
Status

Instance [instance_name]
 Config
 Log
 [Errors]
 [MinSeverity severity_level]
 [StartDate [date_string]]
 [EndDate [date_string]]
 [Last num_of_lines]
 Status

BackupServer
 Config

Membership mode

MonitorServer
 Config
 Session

XPServer
 Config

Shutdown
 Cluster
 Instance [instance_name]

Start
 Cluster
 Instance instance_name
 [Unlock]

Use [instance_name]

Help
 Add
 Diagnose
 Disconnect
 Drop
 Set
 Show
 Shutdown
 Start
```

## add backupserver

- 説明** Backup Server が設定されていないノードに対して Backup Server を設定します。
- 構文** `add backupserver`
- 例** Backup Server をノード “blade3” および “blade4” の “mycluster” に追加します。

```
add backupserver
```

```
Finding nodes for which Backup Server is not configured...
Do you want to configure Backup Server for node "blade3"?[Y]
Please enter the Backup Server port number for node "blade3":5001
Do you want to configure Backup Server for node "blade4"?[Y]
Please enter the Backup Server port number for node "blade4":50011
```

- 使用法**
- Backup Server はクラスタの 1 つまたは複数のノードに設定できます。
  - `add backupserver` は、単一の Backup Server を対象として追加のノードを設定するためのものです。このコマンドを使用して複数の Backup Server を追加することはできません。

## add instance

### 説明

クラスタに新しいインスタンスを1つ追加します。インスタンスを追加するには、`sybcluster` の指示に従って必要な設定情報を対話形式で入力するか、または入力ファイルを使用します。`add instance` は、新しいインスタンス用のローカル・システム・テンポラリ・データベースも作成します。`add instance` から求められる情報は、クラスタの設定が共有であるのか、プライベートであるのかに応じて変わります。

### 構文

```
add instance instance_name [file "input_file"]
```

### パラメータ

*instance\_name*

インスタンスの名前です。

file "*input\_file*"

インスタンスの追加に関する設定情報を含んだファイルの名前を指定します。

### 使用法

- `add instance` は、新しいインスタンス用のローカル・システム・テンポラリ・データベースを作成します。`add instance` を実行する前に、ローカル・システム・データベース用の十分な空き領域を備えたデバイスが存在することを確認してください。
- `add instance` の入力ファイルのフォーマットは、クラスタの入力ファイルと同じです。ただし、`add instance` の入力ファイルでは、ノード・セクション内のインスタンス定義が新しいインスタンスに制限される場合があります。
- `add instance` は、次の情報について入力を求めることがあります。
  - インスタンス名 ( コマンド文でインスタンス名を入力していない場合 )
  - インスタンスをホストしているノード
  - ノード上の UAF エージェントのポート番号
  - クエリ・ポート番号
  - ノードのプライマリ・アドレスとセカンダリ・アドレス
  - プライマリ・ポートとセカンダリ・ポートの指定
- クラスタに単一の Backup Server を設定している場合、`add instance` は Backup Server が新しいインスタンス・ノード用にすでに設定されているかどうかの確認を要求します。「いいえ」の場合、`add instance` は Backup Server を設定するかどうかの確認を要求します。「はい」の場合、ノードの Backup Server ポートの入力を要求します。



クラスタに複数の Backup Server を設定している場合、`add instance` は次の情報の入力を要求します。

- Backup Server の名前
- Backup Server のログ・ファイルのパス
- 新しいインスタンス用の Backup Server ポート

`add instance` は、新しいインスタンス用の XP Server のポート番号情報の入力も要求します。

- インストール・モードがプライベートの場合、`add instance` は新しいインスタンス用の次の追加情報の入力を要求します。
  - `$SYBASE` ホーム・ディレクトリ
  - 環境シェル・スクリプトのパス
  - Adaptive Server ホーム・ディレクトリ
  - サーバ設定ファイルのパス
  - `interfaces` ファイルのパス (LDAP が設定されていない場合)

## connect

**説明** 既存のクラスタに接続します。

**構文**

```
connect [to cluster_name]
 [login login_name]
 [password [password]]
 [agent “agent_spec [, agent_spec[...]]”]
 [discovery “discovery_spec [, discovery_spec [, ...]]”]
```

**パラメータ**

*cluster\_name*

接続先のクラスタの名前です。

login *login\_name*

Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのログインです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。

Sybase Common Security の詳細については、『Unified Agent および Agent Management Console バージョン 2.0 Windows および UNIX 版』を参照してください。

password *password*

Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのパスワードです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。

agent *agent\_spec*

Unified Agent を実行しているクラスタのノードと、sybcluster が Unified Agent に接続するために使用するポート番号を識別するためのエージェント指定です。フォーマットは、*node\_name:port\_number* [, *node\_name:port\_number* [, ...]] です。デフォルトのポート番号は 9999 です。

クラスタに接続するときには、この方法をおすすめします。

**discovery** *discovery\_spec*

要求されたクラスタに関するエージェントを識別するために使用される検出方法です。フォーマットは、

`method[(method_specification)][, (method_specification)[,...]]` です。

[表 8-1 \(312 ページ\)](#) を参照してください。

**例**

**例 1** `sybcluster` コマンド文で “mycluster” がデフォルト・クラスタとして指定されているときに “mycluster” に接続します。

```
connect
```

**例 2** この例では、エージェント指定とデフォルトのポート番号を使用して “mycluster” に接続します。

```
connect to mycluster agent "blade1,blade2,blade3"
```

**使用法**

- 直接接続とは、ユーザがクラスタ・ノードを指定し、オプションとして UAF エージェントのポート番号を指定するものです。エージェント指定の例を次に示します。
  - `myhost` – ホストのノードを指定し、デフォルトの受信ポートである 9999 を仮定しています。
  - `myhost.mydomain.com` – ホストのドメイン名を使用しています。
  - `myhost:9999` – ホストのノードと受信ポート番号を指定しています。

## create backupserver

**説明** クラスタに Backup Server を作成します。クラスタが複数の Backup Server 用に設定されている場合は、クラスタ内の各インスタンスに対して Backup Server を作成します。

**構文** create backupserver

**例** 例 1 “mycluster” に Backup Server “mycluster\_BS” を作成します。

```
create backupserver

Do you want to create multiple Backup Servers?[Y] N
Enter the Backup Server name:[mycluster_BS]
Enter the Backup Server log file path:[$SYBASE/ASE-15_0/
install/mycluster_BS.log]
Do you want to create a Backup Server for node "blade1"?[Y]
Enter the Backup Server port number for node "blade1":
The Backup Server "mycluster_BS" was successfully defined.
```

**例 2** “blade1” で “ase1” および “blade2” で “ase2” が実行されている “mycluster” に複数の Backup Server を作成します。

```
create backupserver

Do you want to create multiple Backup Servers?[Y] Y
The "dump/load" commands would be routed to appropriate Backup Server based
on following policies:
1. Dedicated - Each instance associated with exactly one Backup Server.
2. Round Robin - Choose the Backup Server with least number of requests in
round robin fashion starting from global cluster level counter.
Enter the number corresponding to the policy to be used:[1] 1

Enter the Backup Server name for instance ase1:[ase1_BS]
Enter Backup Server log file path:[/remote/var/sybase/install/ase1_BS_log]
Enter the Backup Server port for node "blade1":23001

Enter the Backup Server name for instance ase2:[ase2_BS]
Enter Backup Server log file path:[/remote/var/sybase/install/ase2_BS_log]
Enter the Backup Server port for node "blade2":23002

Backup Servers successfully defined.
```

## 使用法

- `create backupserver` は、各ノードについて Backup Server の受信ポートの入力を要求します。その他の必要な設定情報はクラスタの設定ファイルからコピーされます。`create backupserver` は次のように動作します。
  - 各ノードに Backup Server のディレクトリ・サービス・エントリを作成します。
  - Backup Server の設定ファイルとログ・ファイルを作成し、`RUN_<backup_server>` スクリプトを作成します。
  - Backup Server 名をクラスタの `sysServers` テーブルに追加します。
  - Backup Server HA を有効にします。

## create cluster

**説明** Adaptive Server 共有ディスク・クラスタを作成します。必要な設定情報を一連のプロンプトへの応答として対話形式で入力するか、または入力ファイルを使用します。

**構文**

```
create cluster [cluster_name]
 [login login_name]
 [password password]
 [agent "agent_spec[, agent_spec,...]"]
 [discovery "discovery_spec[, discovery_spec,...]"]
 [file "input_file"]
```

**パラメータ** *cluster\_name*  
クラスタの名前です。

*login login\_name*  
Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのログインです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。

*password password*  
Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのパスワードです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。

*agent agent\_spec*  
Unified Agent を実行しているクラスタのノードと、sybcluster が Unified Agent に接続するために使用するポート番号を識別するためのエージェント指定です。フォーマットは、*node\_name:port\_number* [, *node\_name:port\_number* ] [,...] です。デフォルトのポート番号は “9999” です。

*discovery discovery\_spec*  
要求されたクラスタに関するエージェントを識別するために使用される検出方法です。フォーマットは、*method[(method\_specification)]* [, (*method\_specification*) ] [,...] です。[表 8-1 \(312 ページ\)](#) を参照してください。

*file "input\_file"*  
クラスタを作成するためのオペレーティング・システムの入力ファイルです。

例 **例 1** “mycluster” という名前の新しいクラスタを作成します。sybcluster は、クラスタを作成するために必要な情報の入力を要求します。

```
create cluster mycluster
```

**例 2** mycluster1.xml ファイルで提供される設定情報を使用して “mycluster1” という名前の新しいクラスタを作成します。

```
create cluster mycluster1 file mycluster1.xml
```

使用法

- クラスタを作成する場合、sybcluster は次の情報の入力を要求します。
  - クラスタ名 (指定されていない場合)
  - インスタンスの数
  - クラスタのインストール・モード (プライベートまたは共有)
  - マスタ、クォーラム、PCI、systemdb、sybsysprocs、テンポラリー・データベース・デバイスへの完全なパス
  - interfaces ファイルへのパス (LDAP が設定されておらず、共有インストールである場合)
  - トレース・フラグ (オプション)
  - dataserver 設定ファイルへの完全なパス (共有インストールの場合)
  - プライマリ相互接続プロトコルとセカンダリ相互接続プロトコル
  - インスタンスのホスト名、ポート番号、プライベート・アドレス、ログ・ファイルのロケーション、起動時の引数  
 プライベート・インストールの場合、sybcluster は、\$SYBASE インストール・ディレクトリ、Adaptive Server ホーム・ディレクトリ、dataserver 設定ファイルのロケーション、interfaces ファイルのロケーション (LDAP が設定されていない場合) の入力も要求します。
- sybcluster は、システム上に Veritas Cluster Server (VCS) を検出すると、デバイスを VCS によって管理するかどうかの確認を要求します。
- クラスタの作成と確認が終了すると、create cluster は I/O フェンシング・チェックを行うかどうかの確認を要求します。I/O フェンシング・チェックは、各デバイスに I/O フェンシング機能があるかどうかを調べるためのものです (『インストール・ガイド』の I/O フェンシングの説明を参照)。

## create xpserver

**説明** クラスタの各インスタンスに XP Server を作成します。

**構文** `create xpserver`

**例** “mycluster” の各インスタンスに XP Server を作成します。

```
create xpserver
```

```
Enter the XP Server port number for instance "ase1":
Enter the XP Server port number for instance "ase2":
Enter the XP Server port number for instance "ase3":
The XP Server was successfully defined for each instance.
```

**使用法** `create xpserver` は、クラスタの各ノードについて XP Server の受信ポートの入力を要求します。XP Server を作成するために必要な他の情報は、クラスタ設定ファイルから読み込まれます。



## deploy plugin

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 説明    | クラスタの 1 つのインスタンスの設定情報を Unified Agent に追加します。Adaptive Server プラグインまたは sybcluster ユーティリティを使用せずにクラスタを作成した場合、またはクラスタの Unified Agent 設定を再作成する必要がある場合に、クラスタを管理する目的で Unified Agent を設定するときには使用できます。クラスタのインスタンスの設定は、Unified Agent プラグインを展開することによって実行されます。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 構文    | <pre> deploy plugin   [ login login_name ]   [ password password ]   [ agent agent_spec ]   [ discovery discovery_spec ] </pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| パラメータ | <p><b>login login_name</b><br/>Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのログインです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。</p> <p><b>password password</b><br/>Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのパスワードです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。</p> <p><b>agent agent_spec</b><br/>Unified Agent を実行しているクラスタのノードと、sybcluster が Unified Agent に接続するために使用するポート番号を識別するためのエージェント指定です。フォーマットは、“node_name:port_number [, node_name:port_number ] [...]” です。デフォルトのポート番号は “9999” です。</p> <p><b>discovery discovery_spec</b><br/>要求されたクラスタに関するエージェントを識別するために使用される検出方法です。フォーマットは、“method[(method_specification)] [, ( method_specification ) [...]” です。検出方法の詳細については、表 8-1 を参照してください。</p> |

**例** **例 1** ホスト “system1501” にある UAF エージェントを使用してプラグインを展開します。

```
deploy plugin agent system1501
```

**sybcluster** はクラスタ名、クラスタのノード番号、インストール・モード、クォーラム・デバイスのフル・パス、環境シェル・スクリプトのパス、Adaptive Server ホーム・ディレクトリの入力を要求します。

**例 2** プラグインを展開し、検出を使用してエージェントを識別します。

```
deploy plugin discovery udp
```

**使用法**

- **deploy plugin** が実行されると、**sybcluster** は次の情報の入力を要求します。
  - クォーラム・デバイスのパス。
  - Sybase ホーム・ディレクトリのパス。
  - インストール・モード (プライベートまたは共有)。デフォルトは共有です。
  - Sybase 環境スクリプトのロケーション。これは、“sh” または “bash” などの “*file\_name*” 構文を使用してロードできるシェル・スクリプトである必要があります (たとえば、*SYBASE.sh*)。
  - Adaptive Server ソフトウェア・ディレクトリのロケーション。デフォルトは <*sybase\_home\_directory*>/ASE-15\_0 です。Adaptive Server ソフトウェア・ディレクトリのロケーションを入力するときには、フル・パスを指定します。**\$\$SYBASE** は使用しないでください。
- **dataserver** のログインとパスワードは、クラスタを管理しているすべての Adaptive Server プラグインを更新する **login** コマンドを使用して設定されます。

## diagnose cluster

**説明** クラスタが正常に動作していることを確認するための一連のチェックを実行します。

**構文** `diagnose cluster`

**例** “mycluster” が正常に動作していることを確認します。

```
diagnose cluster

Cluster name.....mycluster
Maximum instances.....4
Cluster node count.....1
Instances defined.....4
Is cluster locked.....Yes
JDBC connection available.....1 ase1 Yes
JDBC connection available.....2 ase2 Yes
JDBC connection available.....3 ase3 Yes
JDBC connection available.....4 ase4 Yes
Instance Public Network.....1 ase1 on blade1 (10.22.79.39) Reachable:Yes
Instance Public Network.....2 ase2 on blade1 (10.22.79.39) Reachable:Yes
Instance Public Network.....3 ase3 on blade1 (10.22.79.39) Reachable:Yes
Instance Public Network.....4 ase4 on blade1 (10.22.79.39) Reachable:Yes
Has private Primary network...No
Has private Secondary network..No
Network ports required/instance 20
Minimum port allowed.....1025
Maximum port allowed.....65535

Current port strategy.....Public primary and secondary unique.
...The ports are sequenced primary followed by the next instance primary.
...When the primaries are are completed the secondary ports follow the same
pattern.

Recommended port strategy.....Public primary and secondary unique.
...The ports are sequenced primary followed by the next instance primary.
...When the primaries are are completed the secondary ports follow the same
pattern.
```

**使用法** `diagnose cluster` は次のことをチェックします。

- Unified Agent がクラスタの各インスタンスで実行されているか。
- クラスタ内のインスタンスの数がインスタンスの最大数として設定された値を超えていないかどうか。
- クォーラム・ファイルが存在するかどうか。

- すべてのインスタンスが `interfaces` ファイルで定義されており、ポート番号が競合していないかどうか。
- プライマリ・プロトコルとセカンダリ・プロトコルの指定が重複していないかどうか。
- 各 `$SYBASE` ディレクトリが共有されているかどうか。

## diagnose instance

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 説明    | インスタンスが正常に設定されていることを確認するための一連のチェックを実行します。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 構文    | <code>diagnose instance [instance_name]</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| パラメータ | <i>instance_name</i><br>インスタンスの名前です。インスタンス名が入力されていないと、 <code>sybcluster</code> はデフォルト値を使用します。                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 例     | “mycluster” にある “ase1” の設定情報を検証して表示します。<br><br><pre>diagnose instance ase1  Cluster name ..... mycluster Instance id ..... 1 Instance name ..... ase1 Node name ..... blade1  Query port ..... 7101  JDBC connection available ....Yes  Instance Public Network ..... 1 ase1 on blade1 (10.33.108.139) Reachable:.....Yes  Minimum port allowed.....1025 Maximum port allowed .....65535  Instance port range ..... 1 Primary ase1 17100 to 17115 (16) Okay Instance port range ..... 1 Secondary ase1 17165 to 17180 (16) Okay</pre> |
| 使用法   | クラスタが正常に設定されていることを確認するには、 <code>diagnose cluster</code> を使用します。                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

## disconnect

|     |                                                   |
|-----|---------------------------------------------------|
| 説明  | 現在のクラスタのすべての接続を閉じ、 <b>sybcluster</b> を未接続状態に戻します。 |
| 構文  | <b>disconnect</b>                                 |
| 使用法 | 既存のクラスタに再接続するには、 <b>connect</b> を使用します。           |

## drop backupserver

**説明** ノードまたはクラスタから Backup Server を削除します。クラスタに複数の Backup Server を設定している場合は、すべての Backup Server が削除されます。

**構文** drop backupserver

**例 1** 単一の Backup Server を削除します。

```
drop backupserver

Do you want to drop the Backup Server from:
 1. Selected nodes
 2. Cluster
Enter choice:1
Do you want to drop Backup Server from node "blade1"? [N] y
Do you want to drop Backup Server from node "blade2"? [N]
The Backup Server has been dropped from selected nodes.
```

**例 2** クラスタから Backup Server を削除します。

```
drop backupserver

Do you want to drop the Backup Server from:
 1. Selected nodes
 2. Cluster
Enter choice:2
Are you sure you want to drop Backup Server mycluster_BS from cluster
mycluster?(Y or N):[N] y
The Backup Server has been dropped.
```

**例 3** クラスタに複数の Backup Server を設定している場合に、すべての Backup Server を削除します。

```
drop backupserver

Multiple Backup Server are defined for the cluster.This command will drop
all of them.
Are you sure you want to continue?(Y/N):[N] y
The Backup Server has been dropped.
```

**使用法** drop backupserver は、クラスタから Backup Server を削除するときに使  
用します。

## drop cluster

|     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 説明  | クラスタから各インスタンスを削除し、次にクラスタ設定ファイルからクラスタ定義を削除します。クラスタに関連する通常のファイルと、クラスタを管理するクラスタ・エージェント・プラグインも削除します。 <b>drop cluster</b> を使用するには、クラスタが <b>Down</b> ( 停止 ) している必要があります。                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 構文  | <b>drop cluster</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 例   | 現在のクラスタからすべてのインスタンスを削除し、そのクラスタを削除します。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 使用法 | <pre>drop cluster</pre> <ul style="list-style-type: none"><li>• <b>sybcluster</b> は、クラスタを削除する前に確認を要求します。</li><li>• <b>drop cluster</b> は次のように動作します。<ul style="list-style-type: none"><li>• <b>interfaces</b> ファイル、設定ファイル、指定されたデータ・デバイスからクラスタとインスタンス・エントリを削除します。</li><li>• クォーラム・デバイスを未使用としてマーク付けします。</li><li>• クラスタの <b>UAF</b> エージェント・プラグインを停止して削除します。</li></ul></li><li>• 特定のファイル・システムのロックにより、<b>drop cluster</b> を使用しても、<b>UAF</b> プラグインが削除されない場合があります。<br/><b>\$SYBASE_UA/nodes/*/plugins/&lt;cluster_name&gt;</b> ディレクトリが削除されていることを確認してください。このディレクトリが残っている場合は、削除してください。</li></ul> |



## drop instance

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 説明    | クラスタ設定ファイルからインスタンスを削除し、Unified Agent Framework (UAF) と検出サービスを更新します。また、クラスタに対して削除対象のインスタンスを通知し、インスタンスと interfaces ファイル・エントリを削除します。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 構文    | <code>drop instance [ instance_name ]</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| パラメータ | <i>instance_name</i><br>クラスタのインスタンスを指定します。インスタンス名が指定されない場合、sybcluster は sybcluster コマンド・ラインで指定されたデフォルトのインスタンス名を使用します。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 例     | 現在のクラスタから “ase3” インスタンスを削除します。<br><pre>drop instance ase3</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| 使用法   | <ul style="list-style-type: none"><li>• <code>drop instance</code> を使用する前に、次の操作を行ってください。<ul style="list-style-type: none"><li>• 削除対象以外のインスタンスを少なくとも 1 つ、クラスタ内で起動します。</li><li>• 削除対象のインスタンスを停止します。</li><li>• インスタンスに固有の情報を手動で削除します。<code>drop instance</code> は、ローカル・システム・テンポラリ・データベースを自動的に削除します。</li></ul></li><li>• sybcluster は、インスタンスを削除する前に確認を要求します。</li><li>• クラスタ内の最後のインスタンスは削除できません。<code>drop cluster</code> を使用する必要があります。</li><li>• <code>drop instance</code> は、interfaces ファイルにあるインスタンスへの参照とクォーラム・デバイスにあるインスタンス・エントリを削除し、インスタンスを削除したことをクラスタに通知します。</li><li>• <code>drop instance</code> は、削除対象のインスタンスに 1 つまたは複数の Backup Server が設定されている場合、そのエントリを削除します。</li><li>• <code>drop instance</code> は、XP Server を削除し、そのインスタンスに対して 1 つまたは複数の Backup Server が設定されている場合は、それも削除します。</li></ul> |

## drop xpserver

**説明** クラスタの各インスタンスから XP Server を削除します。

**構文** drop xpserver

**例** “mycluster” から XP Server を削除します。

```
drop xpserver
```

```
Are you sure you want to drop the XP Servers from cluster
mycluster"?{Y or N):[N] y
```

```
The XP Servers have been dropped for all instances.
```

**使用法** drop xpserver は、クラスタから XP Server を削除するときに使用します。

## exit

|     |                                                                                                                                                                                                      |
|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 説明  | sybcluster ユーティリティを終了します。                                                                                                                                                                            |
| 構文  | exit                                                                                                                                                                                                 |
| 使用法 | <ul style="list-style-type: none"><li>• exit と quit の動作は同じです。どちらも sybcluster ユーティリティを終了します。</li><li>• sybcluster との接続中に停止したエージェントがあった場合、接続の状態を説明するエラー・メッセージが表示されることがあります。これらのメッセージは無視できます。</li></ul> |

## help

|     |                                                                    |
|-----|--------------------------------------------------------------------|
| 説明  | 現在使用できる <b>sybcluster</b> の対話型コマンドのリストを表示します。                      |
| 構文  | <b>help</b>                                                        |
| 使用法 | 現在使用できる対話型コマンドのリストは、 <b>sybcluster</b> がクラスタに接続されているかどうかに応じて変化します。 |

## localize

**説明** デフォルトの言語、文字セット、ソート順の現在の値を表示します。デフォルト値の変更と、言語の追加または削除が可能です。

**構文** `localize`

**例** この例では、デフォルトのローカライゼーション値が表示され、次に変更のための入力が必要されます。デフォルトの言語を中国語に、デフォルトの文字セットを `euqgb` に、デフォルトのソート順を `bin_euqgb` に変更します。

```
localize

Current default locale properties are:
Default Language - portuguese
Default Charset - mac
Default SortOrder - Binary ordering, for use with the Macintosh character
set(mac).

Options for default Language are:
1. spanish
2. portuguese
3. german
4. us_english
5. thai
6. french
7. japanese
8. chinese
9. korean
10. polish
Enter the number representing the language to be set as defaults:[2] 8

Options for default charsets are:
1. gb18030
2. euqgb
3. uttf8
Enter the number representing the charset to be set as default:[1] 2

Options for sort orders are:
1. Binary ordering, for the EUC GB2312-80 character set (euqgb).
Enter the number representing the sort order to be set as default [1]

Do you want to install any language?[Y] n
Do you want to remove any language?[N]
The cluster mycluster was successfully localized with default language
chinese, charset euqgb, sortorder bin_euqgb
```

### 使用法

- 現在のデフォルトのローカライゼーション値がそれぞれのプロンプトの後に表示されます。現在の値を受け入れるには、数字ではなく復帰改行を押します。
- デフォルト言語のオプションには、`$SYBASE_ASE`に含まれるすべての言語が表示されます。選択されたデフォルト言語が設定されていない場合は、`localize`を使用してその言語を設定または削除してください。
- クラスタ内のすべてのインスタンスに対して新しい値を一貫させるには、ローカライゼーション値を変更した後でクラスタを再起動します。

## quit

説明

sybcluster ユーティリティを終了します。

構文

quit

使用法

exit と quit は、どちらも sybcluster ユーティリティを終了します。

## set backupserver

**説明** クラスタ内の指定したノードにある Backup Server の受信ポート番号を変更します。

**構文** set backupserver

**例** **例 1** “mycluster” の “blade1” にある Backup Server の受信ポート番号を変更します。

```
set backupserver
```

```
Backup Server is configured on the following nodes:
```

1. blade1:3001
2. blade2:3002
3. blade3:3003

```
Do you want to change the Backup Server port on any node?{Y}
```

```
Enter the number representing the node whose port you want to change:1
```

```
Enter the Backup Server port number for node "blade1":4001
```

```
Backup Server was successfully modified as per new properties.
```

**例 2** “mycluster” に複数の Backup Server が設定されている場合に、1つまたは複数の Backup Server の受信ポート番号を変更します。

```
set backupserver
```

```
Multiple Backup Servers are configured for the cluster.Their configuration is as follows:
```

```
Backup Server Policy:Dedicated
```

1. Backup Server name:ase1\_BS  
Configured for blade1:23001  
Log file location:/remote/sybase/ASE-15\_0/install/ase1\_BS.log
2. Backup Server name:ase2\_BS  
Configured for blade2:23002  
Log file location:/remote/sybase/ASE-15\_0/install/ase2\_BS.log
3. Backup Server name:ase3\_BS  
Configured for blade3:23003  
Log file location:/remote/sybase/ASE-15\_0/install/ase3\_BS.log

```
Do you want to edit any Backup Server (y/n)? [Y]
```

```
Enter the number representing the Backup Server you want to edit:[1]
```

```
Enter the new port for Backup Server "ase1_BS":24001
```

```
Do you want to edit any more Backup Servers? [N]
```

```
Backup Server "ase1_BS" successfully updated.
```

**使用法** 新しい受信ポート番号を設定すると、Adaptive Server はまずそのポート番号がすでに使用中でないかどうかをチェックします。



## set cluster

**説明** クラスタの設定値を変更します。`set cluster login` を除いたどの `set cluster` コマンドを実行する場合も、クラスタが停止している必要があります。

**構文**

```
set cluster {
 maxinst max_num_instances |
 traceflags trace_flag[, trace_flag[,...]]
 { primary | secondary } protocol udp |
 login login_name [password password] }
```

**パラメータ**

`maxinst max_instances`  
 クラスタ内で実行できるインスタンスの最大数を指定します。

`traceflags trace_flag[, trace_flag[,...]]`  
 クラスタの起動時に設定されるトレース・フラグを指定します。

`login login_name [ password password ]`  
 Unified Agent がクラスタにログインしたり、停止やその他の特定のタスクを実行したりする際に使用するユーザ名とパスワードを指定します。このログインには `sa_role` が必要です。デフォルトでは、Unified Agent は “sa” ログインをパスワードなしで使用します。このパスワードを変更するには、`set cluster login` を使用します。『Cluster ユーザーズ・ガイド』を参照してください。

---

**注意** `set cluster login` は、Unified Agent がクラスタへのログインに使用するログインまたはパスワードを変更するためにのみ使用できます。`sybcluster` が Unified Agent にログインするときのログインまたはパスワードを変更するには、Agent Management Console Sybase Central ブラウザを使用してください。

---

`{ primary | secondary } protocol udp`  
 プライマリ・インタフェースまたはセカンダリ・インタフェースのプライベート・ネットワークのプロトコルを設定します。

**例 1** “mycluster” のインスタンスの最大数を 4 に変更します。

```
set cluster maxinst 4
```

**例 2** トレース・フラグ 15506 を追加します。

```
set cluster traceflags 15506
```

**例 3** “sa” ユーザ名のパスワードを変更します。

```
set cluster login sa password abcde
```

**使用法** クラスタが Down ( 停止 ) していることを確かめるには、`show cluster status` と入力します。

## set instance

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 説明    | インスタンスのプロパティを設定します。インスタンスは、Down (停止) している必要があります。                                                                                                                                                                                                                                                                                                                                                                                                              |
| 構文    | <pre>set instance <i>instance_name</i> logpath <i>path</i> set instance <i>instance_name</i> startargs <i>values</i> set instance <i>instance_name</i> {primary   secondary} port <i>port_range</i> set instance <i>instance_name</i> {primary   secondary} address <i>ip_address</i></pre>                                                                                                                                                                    |
| パラメータ | <p><i>logpath</i> <i>logfile_path</i><br/>インスタンス・ログ・ファイルのパスを指定します。</p> <p><i>instance_name</i><br/>インスタンスを指定します。</p> <p><i>startargs</i> <i>startup_args</i><br/>インスタンスを起動するときの引数を指定します。</p> <p>{ primary   secondary } address <i>ip_address</i><br/>インスタンスのプライマリ IP アドレスまたはセカンダリ IP アドレスを指定します。</p> <p>{ primary   secondary } port <i>port_range</i><br/>インスタンスのプライマリ・ポート範囲またはセカンダリ・ポート範囲を指定します。<i>port_range</i> のフォーマットは、<i>start_num end_num</i> です。</p> |
| 例     | プライマリ・インタフェースの受信ポートのポート範囲を変更します。<br><pre>set instance primary port 7777</pre>                                                                                                                                                                                                                                                                                                                                                                                  |
| 使用法   | インスタンスが Down (停止) していることを確かめるには、 <code>show cluster status</code> と入力します。                                                                                                                                                                                                                                                                                                                                                                                      |

## set xpserver port

- 説明** クラスタ内の指定したノードにある XP Server の受信ポート番号を変更します。
- 構文** `set xpserver port`
- 例** “ase2” と “ase3” の受信ポートを変更せずに、“mycluster” の “blade1” にある “ase1” インスタンスの XP Server の受信ポートを変更します。
- ```
set xpserver port

Enter the XP Server port number for instance "ase1" [3002]:4002
Enter the XP Server port number for instance "ase2" [3002]:<CR>
Enter the XP Server port number for instance "ase3" [3002]:<CR>
```
- 使用法** 1 つまたは複数のインスタンスについて、XP Server の受信ポート番号を変更できます。

show agents

説明	使用できる UAF エージェントの情報を表示します。
構文	<pre>Show agents [login <i>login_name</i>] [password <i>password</i>] [agent "agent_spec[, agent_spec[...]]"] [discovery "discovery_spec[, discovery_spec[...]]"]</pre>
パラメータ	<p>login <i>login_name</i> Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのログインです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。</p> <p>password <i>password</i> Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのパスワードです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。</p> <p>agent <i>agent_spec</i> Unified Agent を実行しているクラスタのノードと、sybcluster が Unified Agent に接続するために使用するポート番号を識別するためのエージェント指定です。フォーマットは、“node_name:port_number[, node_name:port_number][...]” です。デフォルトのポート番号は “9999” です。</p> <p>discovery <i>discovery_spec</i> 要求されたクラスタに関係するエージェントを識別するために使用される検出方法です。フォーマットは、“method[(method_specification)][, (method_specification)[...]” です。検出方法の詳細については、表 8-1 を参照してください。</p>
例	<p>UAF エージェントの情報を表示します。</p> <pre>show agents Agent Information:service:jmx:rmi:///jndi/rmi://blade1:9985/agent -----</pre>

```
Node Name:      blade1
Agent Port:     9985
Agent Version:  2.5.0
Agent Build:    977
```

```
OS Name:        Linux
OS Version:     2.6.9-42.ELsmp
OS Architecture:amd64
```

Agent Service Info:

```
Agent Service (Agent) Build:977 Status:running
BootstrapService (BootstrapService) Build:<unavailable> Status:running
Configuration Service (ConfigService) Build:977 Status:running
Deployment Service (DeploymentService) Build:<unavailable> Status:running
Environment Service (EnvironmentDiscoveryService) Build:977 Status:running
File Transfer Service (FileTransferService) Build:977 Status:running
Plugin Registration Service (PluginRegisterService) Build:977 Status:running
RMI Service (RMIService) Build:977 Status:running
Remote Shell Service (RemoteShellService) Build:977 Status:running Security
Service (SecurityService) Build:977 Status:running Self Discovery Service
(SelfDiscoveryService) Build:977 Status:running Service Registration Service
(ServiceRegistrationService) Build:977 Status:running Session Service
(SessionService) Build:977 Status:running Sybase Home Service (SybaseHomeService)
Build:14 Status:running
```

Agent Plugin Info:

```
ASE Cluster Agent Plugin (com.sybase.ase.cluster) Version:15.1.0 Build:85 Instance:
  1 Status:running
  Cluster Name:marion
  Env Shell:      /job1/miso/betaR1/SYBASE.sh Shell Type:sh
  Sybase Home:    /job1/miso/betaR1
  ASE Home:       /job1/miso/betaR1/ASE-15_0
  ASE Version:Adaptive Server Enterprise/15.0.1/EBF 14721 Cluster
Edition/B/x86_64/Enterprise Linux/asecluster3/2360/64-bit/FBO/Fri Jul 20 10:04:16
2007
  ASE Login:      sa
  Update Time:   60 seconds
  Last Update:   2007-09-28 22:09:02 -0700
```

使用法 `show agents` は、クラスタに接続する前にアクティブになっています。

show backupserver config

説明 Backup Server が設定されているノード、関連する受信ポート番号、Backup Server ポリシーを表示します。

構文 show backupserver config

例 複数の Backup Server が設定されている “mycluster” の設定情報を表示します。

```
show backupserver config
```

```
Multiple Backup Servers are configured for cluster.Their configuration is as follows:
```

```
Backup Server policy:Dedicated
```

1. Backup Server for ase1:ase1_BS
Configured on (host:port) - blade1:23001
2. Backup Server for ase2:ase2_BS
Configured on (host:port) - blade2:23002
3. Backup Server for ase3:ase3_BS
Configured on (host:port) - blade3:23003

使用法

- show backupserver config コマンドは、Backup Server の設定情報を表示するために使用します。
- 複数の Backup Server を設定している場合、show backupserver config には Backup Server ポリシーが含まれます。

show cluster

説明 クラスタについての設定、ログ、ステータス情報を表示します。

構文

```
Show
    Cluster
    Config
    Template
    Log
    [Errors]
    [MinSeverity severity_level]
    [StartDate [date_string]]
    [EndDate [date_string]]
    [Last number_of_lines]
    Status
```

パラメータ

status

クラスタのステータス情報を表示します。値は次のとおりです。

- Up (起動)
- Down (停止)
- Undefined (未定義)
- Invalid (無効)
- Start (開始)
- Init (初期化)
- Quiesce (静止)

log

クラスタ内のすべてのインスタンスのログを表示します。

errors [*minseverity severity_level*]

エラーのログ・ファイル・エントリを表示します。オプションで、表示するエラー・エントリの重大度レベルの下限を設定できます。

注意 エラーの *severities_level* は、Adaptive Server のエラー・メッセージの属性であり、sybcluster のメッセージの属性ではありません。

startdate [*date_string*]

指定した日付以降に発生したログ・ファイル・エントリを表示します。date_string のフォーマットは、mm:dd:yy です。

startdate または enddate date_string が指定されていない場合、デフォルトは現在の日付 (今日) になります。

enddate [*date_string*]

指定した日付以前に発生したログ・ファイル・エントリを表示します。

last *num_lines*

表示する行数をログ・ファイルの最終行から数えた行数に制限します。

config

クラスタの次の設定情報を表示します。

- インスタンスの最大数
- インストール・モード：共有またはプライベート
- プライマリ・プロトコルとセカンダリ・プロトコル
- トレース・フラグの設定
- クォーラム・デバイスのロケーションと名前
- LDAP 情報 (LDAP が設定されている場合)
- マスタ・デバイスのロケーションと名前

template

フォーマットされたクラスタの設定情報を表示します。

例 **例 1** デフォルト・クラスタの現在の設定とその他の情報を表示します。

```
show cluster status
```

Id	Name	Node	State	Heartbeat
1	ase1	blade1	Up	Yes
2	ase2	blade2	Up	Yes
3	ase3	blade3	Down	No

例 2 共有インストール・モードに設定されたデフォルト・クラスタの設定情報 (LDAP が設定されている場合は、それを含む) を表示します。

```
show cluster config

**Cluster configuration for "mycluster" **
Installation Mode shared
Interfaces Path "/work2/sybase/ASE-15_0/"
Trace Flags:
    15556
Maximum Instances "4"
Quorum "/dev/raw/raw101"
```



```

Master Device
  "/dev/raw/raw102"
logfile ase1 /work2/sybase/ASE-15_0/install/
  ase1.log
run_parameters ase1 null
logfile ase2 /work2/sybase/ASE-15_0/install/
  ase2.log
run_parameters ase2 null

Primary Interconnect "udp"
  Server[1]ase1 tigger.sybase.com 26016 26031
  Server[2]ase2 christopher.sybase.com 26032 26047
Secondary Interconnect "udp"
  Server[1]ase1 tigger.sybase.com 26081 26096
  Server[2]ase2 christopher.sybase.com 26097 26112

```

例3 プライベート・インストール・モードに設定されたデフォルト・クラスタの設定情報を表示します。

```

show cluster config

**Cluster configuration for "localcluster" **
  Installation Mode "private"
  Trace Flags:
  There are no trace flags
  Maximum Instances "4"
  Quorum "/dev/raw/raw101"
  Master Device "/dev/raw/raw102"
  logfile ase1 /remote/work2/sybase/ASE-15_0/install/ase1.log
  run_parameters ase1 null
  logfile ase2 /work2/sybase/ASE-15_0/install/ase2.log
  run_parameters ase2 null

Primary Interconnect "udp"
  Server[1]ase1 tigger.sybase.com 26016 26031
  Server[2]ase2 christopher.sybase.com 26032 26047
Secondary Interconnect "udp"
  Server[1]ase1 tigger.sybase.com 26081 26096
  Server[2]ase2 christopher.sybase.com 26097 26112
LDAP server blade1 2250

```

使用法

`show cluster status` は、`show instance` コマンドの実行結果をクラスタ内の各インスタンスについて表示します。

show instance

説明 インスタンスについての情報を表示します。

構文

```
show instance [instance_name] {
    config |
    status |
    log
    [[ errors ] minseverity severity_level] |
    [ startdate [ date_string ] ] |
    [ enddate [ date_string ] ] |
    [ last num_lines ] ] }
```

パラメータ

instance_name

クラスタにあるインスタンスのユニークな名前を指定します。

status

インスタンスのステータス情報を表示します。値は次のとおりです。

- Up (起動)
- Down (停止)
- Undefined (未定義)
- Invalid (無効)
- Start (開始)
- Init (初期化)
- Quiesce (静止)

log

インスタンス・ログを表示します。

errors [minseverity *severity_level*]

エラーのログ・ファイル・エントリを表示します。オプションで、表示するエラー・エントリの重大度レベルの下限を設定できます。

注意 エラーの *severities_level* は、Adaptive Server のエラー・メッセージの属性であり、sybcluster のメッセージの属性ではありません。

startdate [*date_string*]

指定した日付以降に発生したログ・ファイル・エントリを表示します。*date_string* のフォーマットは、mm:dd:yy です。

startdate または *enddate* *date_string* が指定されていない場合、*date_string* のデフォルトは現在の日付になります。

`enddate [date_string]`

指定した日付以前に発生したログ・ファイル・エントリを表示します。フォーマットは次のとおりです。mm/dd/yy。

`last num_lines`

表示する行数をログ・ファイルの最終行から数えた行数に制限します。

例

例 1 “ase1” についての情報を表示します。

```
show instance ase1 status
```

```
Id      Name      State
-----
1       ase1      Down
```

例 2 “ase1” の設定情報を表示します。

```
show instance ase1 config
```

```
Instance:ase1 at blade6:25001
```

```
Private Primary Network
Address:blade1
Port Range:2541 - 2556
Sybase home:/sybase/sybase_sdc
ASE home:/sybase/sybase_sdc/ASE-15_0
Config file:/sybase/sybase_sdc/ase1.cfg
```

```
Private Secondary Network
Address:blade1
Port Range:2557 - 2572
```

```
Log Path:/blade1/sybase/
ASE-15_0/install/mycluster_ase1.log
```

使用法

- `show instance status` は、指定されたインスタンスについて、次の 7 つの異なるステータスの中から 1 つを表示します。
 - Down (停止)
 - Init (初期化)
 - Invalid (無効)
 - Quiesce (静止)
 - Start (開始)
 - Undefined (未定義)
 - Up (起動)

- `show instance config` は、インストール・モードがプライベートの場合、次の情報を表示します。
 - `$SYBASE` パス
 - ASE パス
 - サーバ設定ファイルのパス

show membership mode

説明	クラスタの現在のメンバシップ・モードを表示します。メンバシップ・モードは、現在のクラスタで Veritas Cluster Integration がサポートされているかどうかを示します。
構文	<code>show membership mode</code>
使用法	<ul style="list-style-type: none">• <code>show membership mode</code> の値は次のとおりです。<ul style="list-style-type: none">• <code>vcs</code> — 現在のクラスタで VCS がサポートされています。• <code>native</code> — 現在のクラスタでは VCS はサポートされていません。• クラスタが <code>vcs</code> メンバシップ・モードで実行されている場合、サーバおよびクラスタを停止または起動する際には、必ず VCS の停止および起動のメカニズムを使用してください。

show session

説明 現在の検出およびエージェント情報を表示します。

構文 `show session`

例 エージェントのステータス情報を表示します。

```
show session

Session information
-----

Sybase sybcluster Command Line Utility/15.0.1/CE GA
2/S/jdk1.4.2/sybclustermain/129/Mon Aug 13 09:59:51 PDT 2007

Connected Cluster:mycluster

Default Cluster:
  Default Instance:

Agent Specifications:
  [1]: oddjob:7171

Discovery Specifications:

Agent Connections:1

  Connection[1] URL:rmi://oddjob:7171

    Node Name:      oddjob1
    Agent Port:     7171
    Agent Version:  2.5.0
    Agent Build:    980
    OS Name:        Linux
    OS Version:     2.6.9-42.ELsmp
    OS Architecture:amd64

    Agent Service Info:
      Agent Service (Agent) Build:980 Status:running
      BootstrapService (BootstrapService) Build:
        <unavailable> Status:running
      Configuration Service (ConfigService) Build:
        980 Status:running
      Deployment Service (DeploymentService) Build:
        19 Status:running
      Environment Service(EnvironmentDiscoveryService)
        Build:980 Status:running
      File Transfer Service (FileTransferService)
```

```
Build:980 Status:running
Plugin Registration Service
  (PluginRegisterService) Build:980 Status:
  running
RMI Service (RMIService) Build:980 Status:
  running
Remote Shell Service (RemoteShellService) Build:
  980 Status:running
  Security Service (SecurityService) Build:980
  Status:running
Self Discovery Service (SelfDiscoveryService)
  Build:980 Status:running
Service Registration Service
  (ServiceRegistrationService) Build:980
  Status:running
  Session Service (SessionService) Build:980
  Status:running

Sybase Home Service (SybaseHomeService) Build:
  14 Status:running

Agent Plugin Info:

ASE Cluster Agent Plugin(com.sybase.ase.cluster)
  Version:15.0.1 Build:129 Instance:1
  Status:running
Cluster Name:mycluster
  Env Shell:/oddjob1/work2/
  sybase_sybclustermain_mycluster_vu/SYBASE.sh
  Shell Type:sh
  Sybase Home:/oddjob1/
  work2/sybase_sybclustermain_mycluster_vu
  ASE Home:/oddjob1/work2/
  sybase_sybclustermain_mycluster_vu/ASE-15_0
ASE Version:Adaptive Server Enterprise/
  15.0.1/EBF 14721 Cluster Edition/B/x86_64/
  Enterprise Linux/asecluster3/2381/64-bit/
  FBO/Mon Nov 12 07:44:23 2007
  ASE Login:sa
Update time:300 seconds
Last Update:2007-11-13 15:27:39 -0800
```

使用法 `sybcluster show session` コマンドは、現在のクラスタについての情報を表示するために使用します。

show xpserver

説明 XP Server の名前、受信ポート番号、ノード名、各ノードに設定されているインスタンス名を表示します。

構文 show xpserver

例 XP Server の名前、受信ポート番号、ノード名、インスタンス名を表示します。

```
show xpserver config
```

```
**XP Server configuration for the cluster**
```

XPServer Name	Port	Host	Node
ase1_XP	4010	blade1	ase1
ase2_XP	4011	blade2	ase2
ase3_XP	4012	blade3	ase3

使用法 sybcluster show xpserver コマンドは、XP Server についての情報を表示するために使用します。

shutdown cluster

- 説明** クラスタのインスタンス・リストにある各インスタンスに対して、クラスタ設定ファイルで指定された順序で Transact-SQL `shutdown` コマンドを実行することでクラスタを停止します。
- 構文** `shutdown cluster [nowait]`
- パラメータ** `nowait`
現在実行中のトランザクションまたは文が終了するのを待たずに、クラスタをただちに停止します。デフォルトでは、`sybcluster` は、クラスタを停止する前に、すべてのトランザクションと文が実行されるのを待ちます。
- 例** 現在のクラスタを停止します。

```
shutdown cluster
```

```
INFO - ...  
INFO - 01:00:00000:00117:2007/06/02 00:23:53.56 kernel ueshutdown:exiting  
INFO - 01:00:00000:00117:2007/06/02 00:23:53.56 kernel SySAM:Checked in  
license for 1 ASE_CORE (2007.1031/31-oct-2007/1293 6876 8FE7 E217).
```

- 使用法**
- `sybcluster` は、クラスタを停止する前に確認を要求します。
 - クラスタが VCS によって管理されている場合、`shutdown cluster` は失敗します。クラスタを停止するには、VCS の停止メカニズムを使用する必要があります。

shutdown instance

説明 Transact-SQL shutdown コマンドを実行することによって、インスタンスを停止します。

構文 shutdown instance [*instance_name*] [*nowait*]

パラメータ *instance_name*
クラスタにあるインスタンスのユニークな名前です。

nowait
現在実行中のトランザクションまたは文が終了するのを待たずに、インスタンスをただちに停止します。

例 現在実行中のトランザクションまたは文が終了するのを待ってから、インスタンス “ase1” を停止します。

```
shutdown instance ase1
```

```
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 kernel shutdown server ase1
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 Server SHUTDOWN by request.
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 ASE is terminating this
process
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 shut down local cluster
server.
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 kernel coordinator to be
shutdown, newcoo is 0.
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 kernel Single server
cluster.
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 kernel cipcnode_down():Node
1 down event.
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 server ASE shutdown by
request.
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 kernel ueshutdown:exitng
INFO - 01:00:00000:00113:2007/06/02 00:31:24/14 kernel SySAM:Checked in
license for 1 ASE_CORE (2007.1031.31-oct-2007/1293 6876 8FE7 E 217).
```

使用法

- クラスタにある最後のインスタンスを停止すると、クラスタも停止します。
- **sybcluster** は、インスタンスを停止する前に確認を要求します。
- クラスタが VCS によって管理されている場合、**shutdown instance** は失敗します。インスタンスを停止するには、VCS の停止メカニズムを使用する必要があります。

start cluster

説明 クラスタ内のすべてのインスタンスを起動します。

構文 `start cluster`

例 現在のクラスタを起動します。

```
start cluster
INFO - [cluster boot log]
...
INFO - 02:00:00000:00002:2007/06/02 00:21:53.56 server 'ase1' (ID=1).
INFO - 02:00:00000:00002:2007/06/02 00:21:53.56 server Master device
      size:
      80 megabytes, or 40960 virtual pages.
```

使用法 クラスタを起動するには、クラスタに接続している必要があります。

start instance

説明 インスタンスを起動します。

構文 `start instance [instance_name] [unlock]`

パラメータ *instance_name*

クラスタにあるインスタンスのユニークな名前を指定します。インスタンス名が入力されていない場合、`sybcluster` は、`sybcluster` コマンド・ラインで指定されたインスタンス、または `use` コマンドで指定されたインスタンスを使用します。

`unlock`

予期せずに終了したクラスタのロックを解除します。`unlock` を使用するには、クラスタが **Down** (停止) している必要があります。

警告! クラスタ内のすべてのインスタンスが停止していることが確認できないかぎり、`unlock` パラメータを使用しないでください。

使用法 `start instance unlock` を使用するには、インスタンスが **Down** (停止) している必要があります。

upgrade server

説明	<p>ノンクラスタード Adaptive Server を Adaptive Server Cluster Edition にアップグレードし、1 つのインスタンスを持つクラスタを作成します。アップグレードを実行するには、コマンド・ラインのプロンプトに応答するか、または入力ファイルを使用します。</p> <p>Adaptive Server バージョン 12.5 以降であれば、Cluster Edition にアップグレードできます。</p>
構文	<pre>upgrade server server_name [login login_name] [password password] [agent agent_spec] [discovery discovery_spec] [file input_file_name] [checkonly]</pre>
パラメータ	<p><i>server_name</i> ノンクラスタード Adaptive Server の名前です。</p> <p><i>login login_name</i> Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのログインです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。</p> <p><i>password password</i> Unified Agent フレームワークにある Sybase Common Security Infrastructure の管理エージェントのパスワードです。インストール後のデフォルトのユーザ名は “uafadmin” で、パスワードは設定されていません。これは、Agent 設定にある簡易ログイン・モジュールです。オペレーティング・システムのログインなど、認証と許可のためのさまざまなメカニズム用にユーザ名とパスワードを設定できます。</p> <p><i>agent agent_spec</i> Unified Agent を実行しているクラスタのノードと、sybcluster が Unified Agent に接続するために使用するポート番号を識別するためのエージェント指定です。ノンクラスタード Adaptive Server をアップグレードすると、ノードが 1 つだけ作成されます。<i>agent_spec</i> のフォーマットは、“<i>node_name:port_number</i>” です。デフォルトのポート番号は “9999” です。</p>

discovery *discovery_spec*

要求されたクラスタに関係するエージェントを識別するために使用される検出方法です。フォーマットは、“*method[method_specification]*”です。検出方法の詳細については、[表 8-1](#) を参照してください。

file *file_name*

サーバのアップグレードに必要な値を格納した入力ファイルです。

checkonly

アップグレードの準備ができているかどうかを判断するために、ノンクラスタード Adaptive Server のチェックを行います。

例

“myserver” を Cluster Edition にアップグレードします。

```
upgrade server

Enter the name of the cluster:new_cluster
Enter the existing Sybase installation directory for server myservers:
Enter the name of the subdirectory containing the ASE installation for server
myservers:
Enter the name of the subdirectory containing the OCS installation for server
myservers:
Enter the name of an sa login on server exit:[sa]
Enter a password:
Cluster new_cluster - Enter the maximum number of instances:[4]
Verifying the supplied agent specifications...
  1>tigger 9999 2.5.0 Linux
Enter the number representing the cluster node 1 [1]
Will this cluster be configured using private SYBASE installations?(Y/N)
...
```

サーバのアップグレードとクラスタの作成には同じ情報が必要になります。「[create cluster](#)」([326 ページ](#)) または使用しているプラットフォームの『インストール・ガイド』を参照してください。

使用法

- `upgrade server` は、次の値の入力を要求します。
 - ノンクラスタード Adaptive Server の Sybase インストール・ディレクトリ
 - ノンクラスタード Adaptive Server のリリース・ホーム・ディレクトリ
 - ノンクラスタード Adaptive Server の Open Client ホーム・ディレクトリ
 - インストール・モード (プライベートまたは共有)

- sybcluster が VCS サブシステムを検出した場合、クラスタに VCS 統合を含めるかどうか
 - interfaces ファイルへのパス (LDAP が設定されていない場合)
 - クラスタの最初のインスタンスの名前
 - クラスタの作成に必要なその他の値
- **checkonly** オプションを使用すると、アップグレード手順は実行されません。このオプションは、サーバでアップグレードの準備ができたかどうかを確認するためのものです。**checkonly** によって発見されたエラー状態は、実際にアップグレードを実行する前にすべて解決する必要があります。

use

説明	デフォルトのインスタンスを指定します。
構文	<code>use instance_name</code>
使用法	<code>use</code> は、 <code>sybcluster</code> コマンド・ラインで指定されたインスタンス名を上書きします。

この章では、`sybmigrate` について説明します。

トピック	ページ
概要	369
作業を始める前に	373
マイグレーション・プロセス	376
暗号化カラムに対する <code>sybmigrate</code> の使用	406
マイグレーション後のアクティビティ	407
Replication Server ドメインのデータベースのマイグレート	408
制限事項	416
トラブルシューティングとエラー・メッセージ	418

概要

`sybmigrate` は、サーバ間でデータをマイグレートするために使用するマイグレーション・ツールです。

`sybmigrate` のデフォルトでは、暗号化カラムは暗号テキスト形式でマイグレートされます。このため、ソースでのデータの復号化とターゲットでの暗号化によるオーバーヘッドが回避されます。場合によっては、`sybmigrate` でマイグレート方法として `reencrypt` が選択され、ソースでのデータの復号化とターゲットでの暗号化が行われることもあります。

注意 ある Adaptive Server からより新しいバージョンの別の Adaptive Server にマイグレートするときは、ターゲット・サーバ上のワーク・データベースのサイズと位置を指定してください。

`sybmigrate` の利点

`sybmigrate` には、次の利点があります。

- データベース・アプリケーションのページ・サイズを変更できる。
- 管理しやすく円滑なマイグレーション・プロセスを提供する。
- 可変ページ・サイズ機能を利用して、ユーザ・データが格納されている既存のデータベースで、Adaptive Server バージョン 12.5 以降の利点を完全に活用できる。

sybmigrate でマイグレートできるデータ

マイグレーション・プロセスの設定部分で、次のサーバ・データがターゲット Adaptive Server にマイグレートされます。

- リモート・サーバ
- ログイン
- ログイン属性
- サーバの役割
- ログインの役割
- 役割の属性
- ユーザ
- 代替ユーザ
- 役割
- パーミッション
- リモート・ログイン
- 外部ログイン属性
- タイマ
- リソースの制限
- 複写の属性
- 表示レベルの属性
- master データベースのユーザ・メッセージ
- master データベースの Java クラス
- master データベースの JAR ファイル

- プロキシ・オブジェクト

マイグレーション・プロセスのマイグレーション部分で、次のデータベース固有のデータがターゲット・データベースにマイグレートされます。

- デフォルト
- ユーザ定義データ型
- ルール
- ユーザ・テーブル
- ユーザ・テーブル・データ
- ビュー
- トリガ
- インデックス
- ストアド・プロシージャ
- 拡張ストアド・プロシージャ
- ユーザ
- ログイン
- 役割
- リモート・サーバ
- データベース・データ
 - ユーザ
 - 代替ユーザ
 - 役割
 - 役割の属性
 - パーミッション
 - ユーザ・メッセージ
 - Java クラス
 - JAR ファイル
- デフォルト
- ルール

- ユーザ定義の型
- テーブル
- インデックス
- 参照制約
- ビュー
- ストアド・プロシージャ
- トリガ

sybmigrate でマイグレートできないデータ

次のデータは手動でマイグレートしてください。

- テーブルレベルのロック・プロモーション属性
- ユーザ定義のスレッシュホールド
- `sysqueryplans` に置かれている抽象プラン定義
- `model` データベースを除くすべてのシステム・データベース
- キャッシュ・バインド、リカバリ順序、`sp_logiosize` で指定された関連ログ I/O サイズなどの必須データベース・オプション
- プロキシ・データベース
- エンジン・グループ
- エンジン・バインド
- 実行クラス
- キャッシュ設定
- 監査テーブルと監査設定
- サーバワイドのロー・ロックの拡大設定

- アクセス・ルール

注意 データ・マイグレーションを開始する前にアクセス・ルールを削除します。アクセス・ルールが存在すると、データベース所有者がテーブル内のすべてのローにアクセスすることができず、完全なデータ・マイグレーションを実行できません。

- 隠し SQL テキストがあるコンパイル済みオブジェクト
- ユーザ定義のセグメント
- 制約はマイグレートされますが、制約の名前でユーザ定義のメッセージ番号にバインドされている場合は、バインドを手動で再作成してください
- dbcc tune を使用して作成した ascinserts、indextrips、oamtrips、datatrips、sortbufsize などのオブジェクトの設定
- デバイス定義
- SQLJ 関数
- 外部ファイルのプロキシ・テーブル
- 監査オプションと監査イベント
- サーバ設定
- データベース・サスペクト・スレッショルド
- リカバリ順序

作業を始める前に

sybmigrate を実行するには、送信元 Adaptive Server に JRE 1.4、jConnect™ for JDBC™ 6.0、ddlgen コンポーネント、コンポーネント統合サービスが必要です。

sybmigrate を実行するには、サーバ間の接続が必要であるため、2つの Adaptive Server が稼働していなければなりません。適切なライセンスがあることを確認してください。

依存性

マイグレーション・プロセスを開始する前に、ターゲット Adaptive Server 上でデータベース、デバイス、セグメントを作成します。サーバとキャッシュの設定も、ターゲット Adaptive Server にあらかじめインストールされていなければなりません。

ddlgen を使用して、対応するスクリプトを送信元 Adaptive Server から抽出し、それを必要に応じて修正してから、ターゲット Adaptive Server に適用します。詳細については、「[ddlgen](#)」(183 ページ)を参照してください。

インストール

sybmigrate は、Adaptive Server ソフトウェアの一部としてインストールされます。Adaptive Server のインストール方法については、使用しているプラットフォームの『インストール・ガイド』を参照してください。

パーミッション

マイグレーション・プロセスの設定部分では、システム管理者ログインが必要です。プロセスのその他の部分では、sybmigrate を実行するのに、“sa_role”と “sso_role”の権限が必要です。

ターゲット・ログイン・アカウントの変更

異なるプラットフォーム間でマイグレートした場合、ログイン・パスワードには互換性がありません。ただし、sybmigrate を使用すると、次の2つの方法のいずれかで、マイグレーション・プロセスの設定セッション中にターゲット Adaptive Server ログイン・アカウントのパスワードを変更できます。

- sybmigrate によってターゲット・サーバにパスワードを生成します。プロセスが完了すると、sybmigrate はマイグレーション中に使用したパスワードのリストを出力します。

- ユーザ名とパスワードの組み合わせを格納したパスワード・ファイルを提供します。sybmigrate は、ターゲット・サーバでこれらのパスワードを設定します。その後、システム管理者が sp_password を実行して、パスワード・ファイルに含まれていない各ログインの新しいパスワードを作成します。

注意 マイグレーション・プロセスが完了した後は、システム管理者がターゲット Adaptive Server でパスワードを手動で変更します。システム管理者は、新しいログインと、マイグレーション・プロセスでリセットされていない各ログインに対して sp_password を発行します。

sybmigrate では、パスワード・オプションの変更に加え、ターゲット Adaptive Server アカウントのロックとロック解除もできます。システム管理者はこのオプションを使用して、マイグレーション・プロセス中にユーザがターゲット Adaptive Server にログインしないようにできます。

プラットフォーム

sybmigrate は、UNIX プラットフォームと Windows プラットフォームのどちらでも動作します。

- UNIX の場合、実行ファイルは `$SYBASE/$SYBASE_ASE/bin/sybmigrate` にあります。
- Windows の場合、実行ファイルは `%SYBASE%\¥%SYBASE_ASE%\¥bin ¥sybmigrate.bat` にあります。

環境設定

以下の環境変数を正しく設定してください。

SYBMIGRATE_MEMORY を除いて、これらの環境変数は、インストール・プロセスで作成される SYBASE.csh ファイルまたは SYBASE.sh ファイルに定義されます。

- SYBASE – Sybase リリース・パスのロケーションを定義します。
- SYBASE_ASE – Adaptive Server コンポーネント・ディレクトリのロケーションを定義します。

- **SYBASE_JRE** – Java Runtime Environment のロケーションを定義します。通常、この変数は Adaptive Server リリース領域の `$$SYBASE/shared/jre-1_4` に設定されます。この環境変数は、`JAVA_HOME` を無効にします。

`SYBASE_JRE` は、デフォルトで `$$SYBASE/shared/jre142` (UNIX の場合) または `%SYBASE%\$Shared\Sun\jre142` (Windows の場合) です。

- **SYBMIGRATE_MEMORY** – Java 仮想マシン (JVM: Java Virtual Machine) を呼び出すときに使用するメモリの量を指定します。この環境変数には、メモリの量をメガバイト単位で指定します。`SYBMIGRATE_MEMORY` を設定しない場合、JVM はデフォルトのメモリ設定の 512MB を使用します。

`sybmigrate` が多数のスレッドを使用する場合、または多数のテーブルやインデックスを並列処理する場合は、クライアント側の JVM に割り付けるメモリの量を増やします。

プロキシ・テーブルのマイグレート

`sybmigrate` は、プロキシ・テーブルのマイグレーションをサポートしています。プロキシ・テーブルのマイグレーションを計画している場合は、マイグレーションを開始する前に次の点を確認してください。

- プロキシ・テーブル定義に含まれるリモート・サーバがターゲット・サーバの `interface` ファイルに存在することを確認します。
- DDL 実行時間を検証するためには、マイグレーションの実行時にリモート・サーバにアクセスできる必要があります。

マイグレーション・プロセス

`sybmigrate` を実行する目的は、送信元 Adaptive Server に存在するすべてのオブジェクトとユーザ・データをマイグレートすることです。ただし、マイグレーションを実行するときは、サーバワイドのデータをマイグレートしてから、ユーザ・データまたはユーザ・オブジェクトを個々のデータベースにマイグレートする必要があります。

オブジェクトの階層は、オブジェクトが再作成される順序を示します。通常、**master** データベースからサーバワイドのオブジェクトが最初に作成されます。デフォルト言語やデフォルト文字などの独立オブジェクトが、個々のデータベースからのデータの前にターゲット・サーバにマイグレートされます。

マイグレーション・プロセスの概要

マイグレーション・プロセスは、送信元 **Adaptive Server** とターゲット **Adaptive Server** の設定、マイグレーション・パスの設定、オブジェクトのマイグレート、マイグレートしたオブジェクトの検証というステップからなっています。

設定セッションでは、送信元データベースからターゲット・データベースへのマイグレーション・パスを確立します。また、レポジトリ・データベースとワーク・データベースを作成し、サーバ・データをマイグレートするためのオプションを登録します。設定セッションを実行できるのは、“sa” ログインのみです。

マイグレート・セッションでは、オブジェクトとデータを送信元データベースからターゲット・データベースにマイグレートします。

検証セッションでは、マイグレートしたオブジェクトを検証します。この検証によって、送信元データベースからターゲット・データベースにマイグレートしたデータとオブジェクトの整合性が確認されます。

サーバ全体をマイグレートする場合、**sybmigrate** を実行してもアーカイブ・データベースはマイグレートされません。

sybmigrate は、アーカイブ・データベースがマイグレーション対象として明確に選択されている場合のみ、アーカイブ・データベースをマイグレートします。アーカイブ・データベースをターゲット・サーバにマイグレートすると、**sybmigrate** によりターゲット・サーバにはアーカイブ・データベースではなく従来のデータベースが作成されます。

マイグレーション前の考慮事項

送信元 **Adaptive Server** からターゲット **Adaptive Server** にデータをマイグレートするには、両方の **Adaptive Server** が稼働していなければなりません。

`sybmigrate` の実行は、データ・マイグレーションの前に、ターゲット Adaptive Server のインストールと設定が完了していることを想定しています。`srvbuild` または `syconfig` を使用して、新しい Adaptive Server を必要な論理ページ・サイズで作成します。

マイグレーションの前にターゲット Adaptive Server を作成して送信元 Adaptive Server を設定するときには、以下のことを考慮してください。

- `sybmigrate` では、`allow resource limits` を 0 に設定する必要があります。
- ターゲット・サーバにメタデータがすでに存在する場合、サーバ・データをマイグレートすることはできません。
- データのマイグレート先となる新しい Adaptive Server を異なる論理ページ・サイズで作成するときは、インバウンド・データを収容できるように、ターゲット Adaptive Server 上のデータベースのサイズを適切に調整してください。論理ページ・サイズが大きい Adaptive Server にデータをマイグレートする場合は、このことが特に重要です。

ターゲット・データベース上で使用できる領域の量を確認するには、領域見積もりレポート `space_est` を使用します。`space_est` の詳細については、「[sybmigrate の起動](#)」(385 ページ) を参照してください。

- マイグレーション・プロセスの速度を上げるために、同じサーバ内で複数の `sybmigrate` セッションを実行できます。ただし、同じ送信元/ターゲット・データベース・パス上で複数の `sybmigrate` セッションを実行することはできません。
- ターゲット・データベース上でセグメントを手動で作成してから、テーブルとインデックスをマイグレートしてください。
- `sybmigrate` のデータ転送率は `CIS bulk insert array size` で設定します。`CIS bulk insert array size` のデフォルト設定は 50 ローです。つまり、CIS によって 50 ロー分のデータがバッファされてから、ターゲット Adaptive Server に転送されます。

スループットを向上させるには、`CIS bulk insert array size` を大きな値に設定します。

ただし、`CIS bulk insert array size` の値を大きくすると、送信元 Adaptive Server がローカル・バッファ用にオペレーティング・システムのメモリを使用します。このため、オペレーティング・システムのメモリが過剰に消費される場合があります。

`CIS bulk insert array size` のデフォルト値を大きくする場合は、過度に大きくしないことをおすすめします。詳細については、CIS のマニュアルを参照してください。

- 転送されるテーブルに `text` カラム、`image` カラム、または `Java ADT` カラムがある場合、`CIS bulk insert array size` はデータ・スループットに影響しません。テーブルに `text`、`image`、または `Java ADT` カラムがある場合は、そのテーブルのすべてのデータは一度に1ローずつマイグレートされます。また、配列のバッファリングは行われません。
- `CIS bulk transfer` を使用してデータ・マイグレーションを実行する場合、送信元 `Adaptive Server` の `CIS packet size` 設定パラメータの値がデータ転送速度に影響することがあります。送信元 `Adaptive Server` の `CIS packet size` の推奨値は、ターゲット `Adaptive Server` の論理ページ・サイズ (2K、4K、8K、または 16K) です。
- ターゲット `Adaptive Server` 上の `max packet size allowed` は、送信元 `Adaptive Server` 上の `CIS packet size` と同じ値でなければなりません。

`max packet size allowed` の詳細については、『システム管理ガイド』を参照してください。

- `sybmigrate` のパフォーマンスを最大にするには、ターゲット `Adaptive Server` の `additional network memory` 設定パラメータをデフォルト値より大きな値に設定します。

`additional network memory` の詳細については、『システム管理ガイド』を参照してください。

- 上記のすべての考慮事項は、`max memory` 設定パラメータに影響します。データをマイグレートする前に、`max memory` が十分に大きい値に設定されていることを確認してください。
- マイグレートされるデータには、サーバ・データ、データベース・データ、ユーザ・オブジェクトの3種類があります。メタデータ (サーバ・データとデータベース・データ) をマイグレートするには、マイグレートするメタデータが、これまで使用してきた残りのデータと矛盾しないように、ターゲット `Adaptive Server` を新規にインストールしてください。

ユーザ・オブジェクトだけをマイグレートする場合は、これまで使用してきた `Adaptive Server` を使用できます。ただし、ユーザ・データの場合は、ターゲット・テーブルが空でなければなりません。

- データをマイグレートする前に、データのマイグレート先となるデータベースをターゲット `Adaptive Server` 上に作成します。データベースは送信元 `Adaptive Server` 上のデータベースと同じ名前にしてください。

- **Adaptive Server** 内部変換ルーチンがない文字セットを変換できるように、ターゲット **Adaptive Server** で `enable unicode conversions` を 1 に設定します。
- ターゲット **Adaptive Server** 上の名前付きキャッシュとバッファ・プールのサイズを確認してください。`sybmigrate` では、キャッシュ設定はマイグレートされません。`ddlgen` で生成される情報を使用して、それをターゲット **Adaptive Server** に適用するか、大きいページ・サイズに対応できるようにメモリの量を大きく設定できます。

ただし、`sybmigrate` はキャッシュ・バインドをマイグレートするため、必要なキャッシュがターゲット **Adaptive Server** にはない場合は、マイグレーション・ログに警告が生成されます。

- 必要な言語をターゲット **Adaptive Server** にインストールしてから `sybmigrate` を実行します。デフォルト言語は、送信元 **Adaptive Server** 上とターゲット **Adaptive Server** 上で同じにしてください。

送信元 **Adaptive Server** 上のユーザ・メッセージが、ターゲット **Adaptive Server** にインストールされていない場合は、`sybmigrate` を実行するとユーザ・メッセージのマイグレーションがアポートされ、エラーが報告されます。

- **Java** カラムをマイグレートする場合は、マイグレーションを実行する前に送信元 **Adaptive Server** 上とターゲット **Adaptive Server** 上で **Java** を有効にしてください。次のように入力します。

```
sp_configure 'enable java', 1
```

- マイグレーションを実行するには、送信元 **Adaptive Server** とターゲット **Adaptive Server** を異なるローカル・サーバ名にします。ローカル・サーバ名を設定したら、サーバを再起動して変更内容を有効にしてください。
- シングルバイト文字セットを使用する **Adaptive Server** をマルチバイト文字セット (`utf8`) を使用する **Adaptive Server** にマイグレートするには、次の手順に従います。
 - a `sybmigrate` を使用して、同じシングルバイト文字セットを使用するサーバにマイグレートします。
 - b マイグレーションが完了したら、文字セットをマルチバイトに変更します。

`sybcluster` では、シングルバイト文字セットを使用する **Adaptive Server** からマルチバイト文字セットを使用する **Adaptive Server** にデータを直接マイグレートすることはできません。

パフォーマンスを向上させる設定とチューニング

サーバ・リソースに応じて `sybmigrate` と `Adaptive Server` を設定することにより、最大のパフォーマンスを実現できます。

`sybmigrate` の設定に関する考慮事項

コピー・スレッドとインデックス作成スレッドを使用してテーブルをマイグレートし、インデックスを再作成します。設定モードで `sybmigrate` を設定するときに、`COPY_THREADS` と `INDEX_THREADS` の値で `sybmigrate` によるデータのコピー速度とマイグレーション速度を上げることができます。

コピー・スレッドの数で、データ・マイグレーションを同時に行うテーブルの数を制御します。1つのコピー・スレッドが各テーブルに割り当てられます。スレッドが1つのタスクを完了すると、別のタスクに移ります。データベースのサイズおよび `Adaptive Server` のリソースに応じて、マイグレーション・プロセス中に使用するコピー・スレッドの数を大きくして、パフォーマンス向上を図ることができます。

注意 多数のオブジェクトを並列にマイグレートするときは、`SYBMIGRATE_MEMORY` の値をチェックして、`sybmigrate` に十分なメモリが割り付けられているかどうかを確認します。

インデックス・スレッドは、ターゲット `Adaptive Server` のテーブル上でインデックスを再作成するときに使用するスレッドの数を制御します。1つのテーブルで1つのスレッドを使用してインデックスを再作成します。テーブル上でインデックスが再作成されると、スレッドは次にマイグレートされたテーブルに移ります。タスクを持たないスレッドも存在します。インデックス作成スレッドの数はコピー・スレッドの数よりかなり少なくなります。

`INDEX_THREADS` を大きな値に設定した場合は、必ずターゲット `Adaptive Server` のソート・バッファも大きな値に設定します。インデックス・スレッドを使用すると、ターゲット・データベースの領域を消費するため、指定した数のインデックス・スレッドに対応するようにターゲット・データベースの領域を適切に設定してください。また、クラスタード・インデックスを再作成する場合は、ターゲット・データベースの領域をさらに追加して設定してください。

Adaptive Server の設定に関する考慮事項

送信元 Adaptive Server とターゲット Adaptive Server には、マイグレーション・プロセスのパフォーマンスに影響するいくつかの設定パラメータがあります。

送信元 Adaptive Server 上の設定は以下の通りです

- **cis packet size** — ターゲット Adaptive Server の **max page size** と同じ値に設定します。
- **number of user connections** — **COPY_THREADS** と **INDEX_THREADS** の値に従って、複数のテーブルを同時にマイグレートできる値に設定します。
- **max parallel degree** — 1 つのテーブル内のパーティションの最大数より大きな値に設定します。データ・マイグレーションが並列に実行されます。**max parallel degree** が、分割されたテーブルに対応する大きい値に設定されていない場合は、テーブルがマイグレートされません。
- **number of worker processes** — 分割されたテーブルのデータ・マイグレーションでは、1 つのパーティションに 1 つのワーカー・スレッドが必要です。したがって、それぞれ p 個のパーティションに分割された t 個のテーブルを同時にマイグレートする場合、送信元 Adaptive Server 上で合計 $t \times p$ 個のワーカー・スレッドを設定します。
- **cis bulk insert batch size** — データ転送トランザクションがコミットされる前のロー数を制御します。デフォルト値は 0 です。データをマイグレートするときにデータの整合性を保証するには、デフォルト値 - を使用するのが最も安全ですが、送信元 Adaptive Server 上でページ・ロックとロー・ロックの数が多くなります。ロックの数を減らすには、この値を大きくします。

cis bulk insert batch size の値を大きくした場合、プロセスでエラーが発生すると、データ・マイグレーションの一部だけが完了します。この場合は、ターゲット・テーブルを手動でトランケートして、**sybmigrate** を再起動します。

- **cis bulk insert array size** — 一度にバルク・コピーされるロー数を制御します。デフォルトは、1 バッチで 50 ローです。データを高速でマイグレートする場合は、この値を大きくします。

テーブルに **text** カラムまたは **image** カラムが含まれている場合は、**cis bulk insert array size** の値に関係なく、データは一度に 1 ローずつ転送されます。

ターゲット Adaptive Server 上では、以下の設定パラメータが `sybmigrate` のパフォーマンスに影響します。

- `max network packet size` — `max page size` 以上の値に設定します。
- `number of user connections` — 複数のテーブルの並列マイグレーションと分割されたテーブルのマイグレーションに対応する値に設定します。

分割されたテーブルのデータを並列に転送するには、送信元 Adaptive Server 上でワーカー・プロセスが必要ですが、ターゲット Adaptive Server 上ではユーザ接続が必要です。分割されたテーブルをマイグレートする場合は、ターゲット Adaptive Server 上の `number of user connections` を、送信元 Adaptive Server 上の `number of worker processes` と同じ値に設定します。

- `number of sort buffers` — マイグレーション・プロセス中は、デフォルト値の 500 で十分です。`sybmigrate` でインデックスを再構築するとき、特に、分割されたテーブルのインデックスをマイグレートする場合は、この値を大きくします。

回避すべきエラー

データ・マイグレーション・プロセスを開始する前に、`sybmigrate` は以下のエラー状態をチェックします。これらの状態が 1 つでも検出されると、マイグレーション・プロシージャがアボートされます。

- ターゲット・テーブルにデータが存在する — すでにデータが入っているテーブルにデータをマイグレートしようとする、`sybmigrate` が失敗します。
- ターゲット・テーブルにインデックスが存在する — ターゲット・テーブルにインデックスが存在すると、`sybmigrate` が低速 `bcp` で動作します。すべてのインデックスを手動で削除してから、データ・マイグレーションを開始してください。
- 送信元テーブルとターゲット・テーブルの分割数が一致しない — 送信元テーブル上の分割数とターゲット・テーブル上の分割数が一致しない場合は、データ・マイグレーションが失敗します。`sybmigrate` はデータをマイグレートするだけで、テーブルの分割にデータを再分配しません。

マイグレーション用従属オブジェクトの自動選択

自動選択機能を使用すると、マイグレーション用の従属オブジェクトが `sybmigrate` によって選択されます。自動選択機能は、従属オブジェクトが存在するかどうかを調べ、従属オブジェクトが存在する場合はターゲット `Adaptive Server` に自動的にマイグレートします。マイグレーションを順調に行うために、この機能を使用することをおすすめします。

アーカイブ・データベースのマイグレート

サーバ全体をマイグレートする場合、`sybmigrate` を実行してもアーカイブ・データベースはマイグレートされません。

`sybmigrate` は、アーカイブ・データベースがマイグレーション対象として明確に選択されている場合のみ、アーカイブ・データベースをマイグレートします。アーカイブ・データベースをターゲット・サーバにマイグレートすると、`sybmigrate` によりターゲット・サーバにはアーカイブ・データベースではなく従来のデータベースが作成されます。

アーカイブ・データベースが含まれる `Adaptive Server` のアップグレード

アーカイブ・データベースはアップグレードできません。データベース・ダンプを以前のバージョンの `Adaptive Server` から、新しいバージョンの `Adaptive Server` にホストされているアーカイブ・データベースにロードする場合、`online database` を実行するとデータベースは内部的にアップグレードされません。

アーカイブ・データベースが含まれる `Adaptive Server` をアップグレードした場合、アーカイブ・データベースを除くすべてのデータベースがアップグレードされます。アーカイブ・データベースは、古いバージョンの `Adaptive Server` のままになります。

すでにアップグレードされたデータベースから生成されたダンプを使用して、アーカイブ・データベースを再ロードすることをおすすめします。

`Adaptive Server` のアップグレード方法の詳細については、使用しているプラットフォームの『インストール・ガイド』を参照してください。

アーカイブ・データベースが含まれる Adaptive Server のダウングレード

アーカイブ・データベースがサポートされないバージョンの Adaptive Server にダウングレードするときは、次の点に注意してください。

- アーカイブ・データベースを含む Adaptive Server を、アーカイブ・データベースがサポートされないバージョンの Adaptive Server にダウングレードする場合は、ダウングレードする前にアーカイブ・データベースを削除することが推奨される。

新しい `sysaltusages` テーブルを消去するには、ダウングレード手順を実行する前にスクラッチ・データベースを削除する。スクラッチ・データベースが削除された場合は、`sysaltuages` により問題が発生することはない。

- Backup Server バージョン 15.0 ESD #2 以降は、ダンプをアーカイブ・データベースにロードできるように、圧縮を新しいフォーマットで書き込む (`with compression = compression_level`)。そのため、アーカイブ・データベースのアクセスがサポートされないバージョンの Adaptive Server に圧縮ダンプをロードする必要がある場合は、圧縮ダンプが作成されたのと同じバージョンの Backup Server を使用して圧縮データベース・ダンプをロードする。以前のバージョンの Backup Server では、新しいフォーマットの圧縮データベース・ダンプはサポートされない。

圧縮せずにダウングレードする場合は、Backup Server に関する注意は不要です。

sybmigrate の起動

警告！ sybmigrate は、マイグレーション中に送信元 Adaptive Server とターゲット Adaptive Server 上でアクティビティが行われないことを想定しています。マイグレーション・プロセス (設定、マイグレート、検証) 中にオブジェクトの作成、修正、または削除が行われた場合、Sybase はマイグレーションの整合性を保証できません。

sybmigrate の GUI バージョンとリソース・ファイル・バージョンのどちらを実行する場合も、以下の関連コマンド・ライン引数を使用して起動します。

```
sybmigrate [-v] [-h] [-f]
            [-D 1 | 2 | 3 | 4]
            [-I interfaces file]
            [-r input resource file]
```

```

[-m setup | migrate | validate | report ]
[-rn status | space_est | repl | diff | password ]
[-l log file ]
[-t output template resource file ]
[-J client_charset ]
[-z language ]
[-T trace_flags ]
[-Tase trace_flags ]
[-f ]

```

各パラメータの意味は次のとおりです。

- `-v` は、バージョンの文字列を印刷して、終了します。
- `-h` は、ヘルプ情報と構文の使用方法を印刷して、終了します。
- `-f` は、ロック・セッションを無効にします。

`sybmigrate` でセッションが不適切に終了した場合は、`-f` を使用して、作成された送信元データベースとターゲット・データベースのバインドを無効にし、`sybmigrate` の 1 セッションだけが送信元/ターゲット・データベース・パス上で実行されるようにします。

- `-D` は、`sybmigrate` のデバッグ・レベルを設定します。デフォルトのデバッグ・レベルは 2 です。
- `-I` は、サーバ名を検索する `interfaces` ファイルを指定します。`interfaces` ファイルのロケーションを指定しない場合、UNIX では `$$SYBASE/interfaces`、Windows では `%SYBASE%\$ini %sql.ini` が使用されます。

注意 LDAP エントリが `$$SYBASE/$SYBASE_OCS/config/libtcl.cfg` (Unix の場合) または `%SYBASE%\$SYBASE_OCS%\$ini %libtcl.cfg` (Windows の場合) に定義されている場合は、`sybmigrate` を無効にし、`-I` 引数を指定して `interfaces` ファイルを使用できます。

- `-r` は、マイグレーション・プロセスでリソース・ファイル・モードを使用することを指定します。`-r` パラメータを使用して入力リソース・ファイルを指定しない場合、`sybmigrate` は GUI モードで動作します。

`-r` パラメータを使用する場合は、`-m` 引数を使用して、実行するオペレーションのタイプとして `setup`、`migrate`、`validate`、または `report` を指定する必要があります。リソース・ファイル・モードで、マイグレーション・プロセス全体を実行するか、プロセスの一部だけを実行することができます。

- `-m` は、実行するオペレーションのタイプを指定します。

- **setup** – レポジトリとマイグレーション作業データベースを設定し、サーバワイドのデータをマイグレートします。
- **migrate** – データとオブジェクトのマイグレーションを実行します。
- **validate** – マイグレートしたオブジェクトを検証します。
- **report** – 5つのレポートのいずれかを実行します。レポートはGUIモードとリソース・ファイル・モードで実行できます。以下のレポートを実行できます。

- **status** – マイグレート・オブジェクト・ステータス・レポートは、マイグレートされたオブジェクトに関する情報を提供します。このレポートを実行するには、次のコマンドを発行します。

```
sybmigrate -r resource file -m report -rn
status
```

- **space_est** – ターゲット・データベース領域見積もりレポートを使用して、ターゲット・データベースに十分なリソースが割り当てられているかどうか確認します。リソース・ファイル・モードで、次のコマンドを発行して**space_est**レポートを実行します。

```
sybmigrate -r resource file -m report -rn
space_est
```

- **repl** – 複製レポートを使用して、マイグレートされた明示的な複製オブジェクトをチェックし、複製システムのタイプを確認し、ユーザがターゲット Adaptive Server と Replication Server 上で実行する SQL コマンドを生成します。**repl**レポートを実行するには、次のコマンドを発行します。

```
sybmigrate -r resource file -m report -rn repl
```

- **diff** – 送信元データベースとターゲット・データベース間のオブジェクトを確認します。ユーザは、サーバとデータベースの情報またはメタデータを除き、個々のオブジェクトまたはデータベース全体に対してこのレポートを実行できます。**diff**レポートはいつでも実行できます。**diff**レポートを実行するために、設定セッションを実行する必要はありません。**diff**レポートの実行時、送信元データベース名とターゲット・データベース名が同じである必要はありません。

diff レポートでは、次のオブジェクト・タイプに関する情報が出力されます。

- サーバ情報 — 送信元 Adaptive Server とターゲット Adaptive Server 間で master データベースのシステム・カタログのロー・カウントを比較します。このタスクは検証セッションと似ています。
- データベース情報 — 送信元 Adaptive Server とターゲット Adaptive Server 間でユーザ・データベースのシステム・カタログのロー・カウントを比較します。このタスクは検証セッションと似ています。
- DDL オブジェクト — オブジェクトが送信元 Adaptive Server とターゲット Adaptive Server のどちらに存在するかをレポートに表示します。オブジェクトが両方のデータベースに存在する場合、そのオブジェクトはレポートに表示されません。
- ユーザ・テーブル・データ — 送信元 Adaptive Server とターゲット Adaptive Server にあるユーザ・テーブルのロー・カウントを比較します。テーブルが送信元データベースとターゲット・データベースのどちらかにしか存在しない場合、そのテーブルはレポートに表示されません。
- password — 変更されたパスワード用のファイルを作成します。このレポートはシステム管理者だけが実行できます。
- -m は、生成するレポートのタイプを指定します。-m を指定しない場合、5 つのレポートがすべて実行されます。
- -l は、マイグレーション・プロセスの出力が格納されるユーザ定義のログ・ファイルを指定します。-l を使用しない場合、`$$SYBASE/$SYBASE_ASE/init/logs` または作業ディレクトリにログが格納されます。
- -t は、`sybmigrate` のリソース・ファイル・モードでこれ以降のマイグレーションに使用する出力テンプレート・リソース・ファイルの生成を指定します。

-t を指定するには、ログイン情報を指定する -r 引数を使用して、`sybmigrate` を起動する必要があります。この引数を使用するには、-m を使用して、生成するリソース・ファイルのタイプも指定する必要があります。

注意 -t はリソース・ファイル・モードでのみ使用できます。

- -J は、Adaptive Server 接続に使用する文字セットを指定します。
- -z は、Adaptive Server 接続に使用する言語を指定します。
- -T は、コマンド・ライン・トレース・フラグを設定します。コマンド・ライン・トレース・フラグは次のとおりです。
 - DUMP_SQL — `sybmigrate` によって発行されるすべてのクエリをログ・ファイルに出力することを指定します。
 - NO_SORTED_DATA — クラスタード・インデックスのあるテーブルをターゲット・サーバに順番にコピーすることを指定するデフォルトを無効にします。クラスタード・インデックスは、`with_sorted_data` オプションを使用して再作成されます。
 - LEAVE_PTBL_ON_ERROR — 障害発生時にプロキシ・テーブルを削除しないことを指定します。
 - SKIP_CONFIG_CHECK — 設定の互換性検査を実行しないことを指定します。
 - SKIP_PARTITION_CHECK — パーティションの互換性検査を実行しないことを指定します。
 - DUMP_DDL — DDL コマンドをログ・ファイルに出力することを指定します。
 - DUMP_DEPEND_OBJECT — `auto_select_dependent_objects` オプションの使用時に、従属オブジェクトとして追加されたオブジェクトのリストを `sybmigrate` で出力することを指定します。
 - ONE_WORK_THREAD — スキーマ作成スレッドの現在の設定を無効にして、ワーカー・スレッドを1つ使用することを指定します。
 - ALLOW_DATA_AND_INDEX — すべてのテーブルの作成後にインデックスを作成するデフォルトの動作を無効にします。インデックスは、リソースが利用可能になったときに作成されます。

- `-Tase` は、`sybmigrate` がオープンしたすべての Adaptive Server 接続に対する Adaptive Server トレース・フラグ (`dbcc traceon` を使用してオンにしたフラグ) を実行します。カンマで区切ったリストで トレース・フラグを指定してください。

`sybmigrate` を実行するときは、設定、マイグレート、検証という 3 つのフェーズに分けてマイグレーション・プロセスを実行します。

GUI モード

マイグレーション・プロセスでは、GUI モードまたはリソース・ファイル・モードを使用できます。マイグレーション・プロセスの一部を GUI モードで実行し、一部をリソース・ファイル・モードで実行することもできます。

設定

データをマイグレートする前に、送信元 Adaptive Server とターゲット Adaptive Server を指定し、サーバに含まれる送信元データベースとターゲット・データベースの間のパスを登録します。これを行うには、`-m setup` コマンド・ライン・オプションを使用するか、[セッション・タイプ] ウィンドウでプロンプトが表示されたときに [マイグレーション用ソース・データベースを設定] を選択して、`sybmigrate` を起動します。

❖ 送信元 Adaptive Server とターゲット Adaptive Server を指定し、送信元データベースとターゲット・データベースの間のパスを登録する

1 [ASE に接続] ウィンドウで、マイグレーション・プロセスに使用する送信元 Adaptive Server とターゲット Adaptive Server を指定できます。

- [サーバ] フィールドのドロップダウン・メニューから選択できます。メニューに表示されるのは、デフォルトの `interfaces` ファイル (UNIX では `$SYBASE/interfaces`、Windows では `%SYBASE%\$ini ¥sql.ini`) にある Adaptive Server、または `-I` コマンド・ライン引数で指定した `interfaces` ファイルにある Adaptive Server のリストです。

`interfaces` ファイルを使用しない場合は、`-I` コマンド・ライン引数を使用できません。送信元 Adaptive Server とターゲット Adaptive Server を `host:port` 形式で指定してください。

- 設定フェーズでは、システム管理者としてサーバにログインします。[ログイン]フィールドに“sa”と入力し、パスワードを入力して、[接続]を選択してください。

注意 `sybmigrate` を実行できるのは、一度に1セッションだけです。したがって、同じ送信元 Adaptive Server とターゲット Adaptive Server 上で別のユーザが `sybmigrate` を実行している場合は、「設定セッションのロック：前回の設定が異常に終了したか、別の設定セッション実行中です。上書きしますか?」というエラー・メッセージが表示されます。前のセッションがクラッシュしたか中断された可能性がある場合、そのセッション・ロックを無効にすることができます。

設定プロセスとマイグレーション・プロセスに進む前に、ほかのユーザが `sybmigrate` を実行していないことを確認してください。複数のユーザが同時に `sybmigrate` を実行している場合は、データの整合性が保証されません。

- 2 [セッション・タイプ] ウィンドウで、実行するオペレーションのタイプを選択するように指示されます。以下の選択肢がありません。
 - マイグレーションに使用する送信元データベースの設定
 - データベースのオブジェクトとデータのマイグレート
 - マイグレートしたオブジェクトとデータの検証
 - Reports — [Reports] を選択すると、[Reports type] ウィンドウが表示されます。`status`、`space_est`、`repl`、`diff`、`password` から選択できます。領域見積もりレポートまたは複写レポートを選択した場合は、[Report Paths] ウィンドウでレポートを実行するデータベース・パスを選択するよう要求されます。

送信元 Adaptive Server とターゲット Adaptive Server の間で設定セッションが完了していない場合、パスワード・レポート、ステータス・レポート、複写レポートは選択できません。

`setup`、`migrate`、`validate`、または `reports` を指定する `-m` オプションを使用して `sybmigrate` を起動した場合は、このウィンドウが表示されません。
- 3 マイグレーションに向けてデータベースを準備するには、Setup ウィザードを使用します。Setup ウィザードでは、次のような複数のウィンドウが表示されます。
 - [Choose Database] ウィンドウ

このウィンドウでは、送信元 Adaptive Server 内の送信元データベースと、ターゲット Adaptive Server 内のターゲット・データベースを選択するよう要求されます。この選択により、**sybmigrate** で送信元 Adaptive Server のデータをターゲット Adaptive Server のどこに格納するかを指定します。

注意 送信元データベースとターゲット・データベースは同じ名前であればなりません。

[ソース・データベース] ドロップダウン・リストに、送信元 Adaptive Server にあるデータベースのリストが表示されます。

[ターゲット・データベース] ドロップダウン・リストに、ターゲット Adaptive Server で使用できるデータベースのリストが表示されます。**sybmigrate** を実行するには、ターゲット Adaptive Server でデータベースを作成してから、マイグレーション・プロセスを開始する必要があります。

「マイグレーション・パス」は、選択した送信元データベースとターゲット・データベースの組み合わせです。

- [Configure DDL threads]

指定したマイグレーション・パスのターゲット・サーバで、データベース・オブジェクトを作成するときに使用するスレッドの数を選択します。

- [Configure copy threads]

マイグレーション・パスの送信元データベースからターゲット・データベースにデータをコピーするときに使用するスレッドの数を選択します。複数のエンジンを備えたシステムに対応できる十分な数のスレッドを使用してください。

- [Configure index threads]

指定したマイグレーション・パスのターゲット・サーバで、インデックスを作成するときに使用するスレッドの数を選択します。複数のエンジンを備えたシステムに対応できる十分な数のスレッドを使用してください。

テーブルの並列転送に使用するスレッドの数はユーザが制御します。複数のテーブルを同時に転送する場合、テーブルごとに 1 つのサーバ間 CIS 接続が必要です。

分割されていないテーブル (各テーブルに1つのパーティションが含まれる) でデータ・マイグレーションを実行するとします。このようなテーブルをマイグレートするときは、1つのサーバ間接続が確立されます。この接続では、送信元 Adaptive Server 上で1つのユーザ接続を使用し、ターゲット Adaptive Server 上でも1つのユーザ接続を使用します。

n 分割されたテーブルのデータ・マイグレーションを実行する場合、 n 並列度で並列にデータ転送が行われます。このデータ転送には、送信元サーバ上で n 個のワーカー・プロセスと、ターゲット・サーバ上で $2n$ 個のユーザ接続が必要です。

たとえば、マイグレートする n 分割されたテーブルが 10 個あるとします。sybmigrate で4つのスレッドを使用する場合、少なくとも4つのワーカー・プロセスと8個のユーザ接続を使用するように送信元 Adaptive Server を設定します。ターゲット Adaptive Server では、少なくとも8個のユーザ接続を使用するように設定します。

注意 設定セッションで各プロパティに割り当てる値がデフォルト値になります。マイグレート・セッションまたは検証セッションでは、デフォルト値を一時的に無効にできます。これらの値は、Adaptive Server で使用可能なリソースに制限してください。

- [Configuring the work database]

sybmigrate では、マイグレーション・プロセスで1つ以上のワーク・データベースが必要です。[データベース・サイズの作成] フィールドには、デフォルト値がメガバイト単位で表示されます。このデフォルトは、前のウィンドウで指定したコピー・スレッドとインデックス作成スレッドの数に基づいています。このデフォルトは最小値です。この値を大きくすることはできますが、これ以上小さな値にすることはできません。

[Device] フィールドでは、ワーク・データベースを作成するデバイスを指定できます。

注意 バージョン 12.0 ~ 12.5.0 の送信元 Adaptive Server からマイグレートする場合、**sybmigrate** を実行するには、ターゲット・サーバ上にもワーク・データベースが必要です。ウィザードでは、ターゲット・ワーク・データベースについても同じ情報を入力するよう要求されます。

- [Current paths]

選択済みのマイグレーション・パスを確認します。マイグレーション・パスを右クリックすると、編集オプションと削除オプションが表示されます。

パスを追加するには、[Add Migration Path] を選択します。後でパスを追加する場合は、**sybmigrate** を設定モードで再実行します。

- [Configure repository]

sybmigrate では、すべてのマイグレーション・パスのマイグレーションを追跡するために、送信元サーバ上にレポジトリ・データベースを作成します。デフォルト・データベース・サイズは最小値です。この値を大きくすることはできませんが、これ以上小さな値にすることはできません。

- [Migration of server-wide data]

ログイン情報など、システム・カタログ内の情報をマイグレートするかどうかを選択できます。オプションは次のとおりです。

- [はい] – 設定フェーズの終わりにサーバ全体のデータがマイグレートされます。
- [いいえ] – サーバ全体のデータはマイグレートされません。データベース・マイグレーションを開始する前であれば、いつでもこのウィンドウに戻ってデータのマイグレートを選択できます。
- [未決定] – 後でこのウィンドウに戻って、別のマイグレーション・オプションを選択できます。ただし、[はい]か[いいえ]を選択するまでマイグレーション・フェーズを開始できません。[未決定]は、マイグレーション・プロセスを設定しておいて、実際のデータ・マイグレーションを後日実行する場合に役立ちます。

ログインや他のサーバ全体の情報について、ターゲット・サーバがすでに設定されている場合には、このオプションのデフォルトは[いいえ]です。

[オプション]ボタンには、ログイン・アカウントを処理するための高度なオプションが用意されています。このオプションでは、以下を指定できます。

- マイグレーション後にログイン・アカウントをロックするかどうか
- プラットフォーム間でのマイグレーション時にログイン・パスワードを処理する方法
 - 変更なし - 同じプラットフォームへのマイグレーション時に使用 (デフォルト)
 - ランダム・パスワードの生成
 - ファイル内のリストからのパスワードの割り当て
- [Summary]

選択したオプションの概要を表示します。[完了]をクリックして、選択した設定タスクを実行します。

- 4 [設定の進行状況] ウィンドウに、設定フェーズの進行状況が表示されます。

この間に、レポジトリ・データベースが作成され、データベース・スキーマがインストールされ、選択した各パスに作業データベースが作成され、ユーザの選択に基づいてサーバ・データがマイグレートされます。この後で、**sybmigrate** を設定モードで実行すると、データ・マイグレーション用の新しいパスが作成されません。新しいパスを作成しない場合は、**sybmigrate** を設定モードで2回以上実行する必要はありません。

[ログの表示] をクリックして、ログに記録された進行状況を表示できます。設定プロセスが完了すると、[現在のタスク] ウィンドウに **DONE** と表示され、ログには **SETUP_COMPLETE** と示されます。[クローズ] をクリックして、ログと [設定の進行状況] ウィンドウを終了します。

- 5 [ASE に接続] ウィンドウに戻ります。[終了] を選択して **sybmigrate** を終了します。データ・マイグレーション・プロセスのマイグレーション・フェーズを開始するには、**sybmigrate** を一度終了し、マイグレート・モードで再起動します。

マイグレート

設定が完了したら、データのマイグレートを開始できます。-m migrate コマンド・ライン・オプションを使用して sybmigrate を再起動するか、GUI ウィンドウから [データベース・オブジェクトとデータのマイグレート] オプションを選択します。

- 1 [ASE に接続] ウィンドウで、接続する送信元 Adaptive Server とターゲット Adaptive Server を選択します。
- 2 -m migrate コマンド・ライン引数を使用して sybmigrate を起動しなかった場合は、[セッション・タイプ] ウィンドウでセッション・タイプを選択します。
- 3 [オブジェクトの選択] ウィンドウで、マイグレートするデータベース・データのタイプを選択できます。

[オブジェクトの選択] ウィンドウでは、[Setting] メニュー・バーからコピー・スレッド、インデックス作成スレッド、ワーカー・スレッドのパラメータを設定できます。

[オブジェクトの選択] ウィンドウでは、オブジェクト・ツリー・ノードを右クリックして、選択したオブジェクトの従属オブジェクトを sybmigrate で自動選択するよう要求することもできます。

データベース・データ・フォルダを展開すると、設定中に作成した各パスに対応するファイルがあります。各ファイルで、特定のデータベースからマイグレートするデータを選択できます。次のデータを選択できます。

- データベース・データ

注意 データベース・データのマイグレートを選択する場合は、そのデータをすべてマイグレートしてください。データベース・データの一部を選択解除すると、データベース・データをマイグレートするかどうかを尋ねるエラー・メッセージが表示されます。

設定時にサーバ・データをマイグレートしなかった場合、[データベース・データ] という選択肢は無効になります。

- デフォルト
- ルール
- ユーザ定義データ型
- テーブル

- インデックス
- 参照制約
- ビュー
- ストアド・プロシージャ
- トリガ

これらのオブジェクトの[ステータス]フィールドに、データが正しくマイグレートされたかどうかを示されます。[成功]は、データがすでにマイグレートされたことを示します。[初期状態]は、マイグレーションがまだ開始されていないことを示します。マイグレートしたデータにエラーが見つかった場合は、[ステータス]フィールドを[初期状態]にリセットして、データを再びマイグレートすることができます。検証プロセスは、正しくマイグレートされたオブジェクトに対してのみ実行されるため、一部のデータが正しくマイグレートされていないときに検証プロセスを開始するには、[ステータス]フィールドを[成功]にリセットします。“Work in Progress”は、オブジェクトはマイグレートするように選択されているが、`sybmigrate`の異常終了を引き起こすエラーがあったため、マイグレーションが行われていないことを意味します。

サーバ・データがマイグレートするように選択されているかどうかを表示できます。ただし、マイグレーション・プロセスのこの時点では、サーバ・データはすでにマイグレートされているため、情報の確認に過ぎません。

マイグレートするデータを選択したら、[マイグレート]をクリックします。

検証

検証フェーズはマイグレート・フェーズと同様です。ウィンドウで同じ情報を指定するように指示されますが、ここではマイグレートするデータを選択するのではなく、検証するデータを選択します。

正しくマイグレートされたオブジェクトだけが、検証の対象になります。

マイグレーションと検証の進行状況

sybmigrate では、[Migration/validation] 画面にマイグレーションと検証の進行状況が常に表示されます。この画面には、マイグレーションの進行状況、レポートされたメッセージや、タスク・タイプごとに保留中のオブジェクト、失敗したオブジェクト、成功したオブジェクトの概数が表示されます。

[キャンセル] ボタンをいつでも選択できます。このボタンを選択すると、進行中の実行内容が適切に停止されます。

リソース・ファイル・モード

リソース・ファイル・モードに対して次の変更を行う必要があります。

- `data_copy_thread`、`create_index_thread`、`work_thread` の各属性は、`sybmigrate` の設定セッション、マイグレート・セッション、検証セッションで認識されます。設定セッションでは、これらの値がレポジトリ・データベースに記録され、マイグレート・セッションと検証セッションのデフォルト値として使用されます。マイグレート・セッションと検証セッションでは、新しい値を指定してデフォルト値を無効にできます。
- `lock_account` は新しいログイン・アカウント管理機能です。`lock_account` は、ログイン情報のコピー後にターゲット Adaptive Server の全アカウントのロックまたはロック解除を `sybmigrate` に指示します。有効な値は “Yes” と “No” です。“Yes” に設定すると、`sybmigrate` はターゲット Adaptive Server のアカウントをロックします。`lock_account` をアクティブにするには、設定セッションで `migrate_server_data` を “Yes” に設定します。

`lock_account` 属性が設定されていない場合、ターゲット・ログイン・アカウントに対して何も実行されません。
- `login_password_file` は、ターゲット Adaptive Server のパスワードの変更をサポートするために追加されました。設定セッションで、`login_password_file` は、入力パスワード・ファイルまたは値 “<generate>” を取得します。“<generate>” は、パスワード・ファイルからパスワードを読み込む代わりに、パスワードを生成するように `sybmigrate` に指示する特殊キーです。設定セッション中にこの属性がリソース・ファイル内に設定されていない場合は、ターゲット Adaptive Server のログイン・パスワードは変更されません。`login_password_file` をアクティブにするには、設定セッションで `migrate_server_data` を “Yes” に設定します。

- パスワード・ファイルは、プレーン・テキスト形式にする必要があります。このファイルの内容は、ログイン名カラムとパスワード文字列カラムという2つのカラムで構成されます。カラム間はタブまたはスペースで区切ります。“#”で始まる行はコメントです。
- `auto_select_dependent_objects` は、マイグレーション・セッションと検証セッションで使用できる新しい値です。この属性は、マイグレーションと検証用の従属オブジェクトを自動選択するように `sybmigrate` に指示します。この属性の有効な値は “Yes” または “No” です。デフォルト値は “No” です。
- `source_ase`、`source_ase_login`、`source_ase_password`、`target_ase`、`target_ase_login`、`target_ase_password` の各属性がリソース・ファイルにない場合、`sybmigrate` はこれらの属性を指定するようユーザーに要求します。
- リソース・ファイルのデータベース・セクションにオブジェクトまたは SQL を指定しない場合、すべてのオブジェクトとタイプが選択されます。

たとえば、次のリソース・ファイルでは、`pubs2` データベースと `pubs3` データベースからすべてのオブジェクト・タイプ (デフォルト、ルール、テーブルなど) がマイグレートされます。

```
[server]
source_ase=tho:5002
source_ase_login=sa
source_ase_password=

target_ase=tho:6002
target_ase_login=sa
target_ase_password=

[database]
source_database_name=pubs2
target_database_name=pubs2

[database]
source_database_name=pubs3
target_database_name=pubs3
```

リソース・ファイル・モードは非対話型モードです。リソース・ファイルには、マイグレーションに必要な情報がすべて入っています。GUI がサポートされていない場合、またはバッチ・ファイルを実行する必要がある場合に、リソース・ファイル・モードを使用できます。

マイグレートするオブジェクト・タイプ属性をリソース・ファイルに指定しない場合、**sybmigrate** はデータベース全体をマイグレートします。

送信元 Adaptive Server またはターゲット Adaptive Server のログインかパスワードをリソース・ファイルに指定しない場合、**sybmigrate** はこの情報をユーザに要求します。

sybmigrate を非対話型モードで実行するためのリソース・ファイルのフォーマットを次に示します。リソース・ファイルを作成するには、ファイルにすべての値を入力します。

```
#
# This is a sample Migration Tool resource file.
# This resource file will migrate objects in pubs2,
# pubs3, and foo databases.
#

#####
# Server wide information
#####
[server]
# "<host name>:<port number>" or just server name.
source_ase=tho:5002
source_ase_login=sa
source_ase_password=

# "<host name>:<port number>" or just server name.
target_ase=tho:6002
target_ase_login=sa
target_ase_password=

# Repository database setup attributes.This is
required with "setup" mode.
# Repository database size in MB.
repository_database_size=7
# Device used to create the "sybmigrate" database.
repository_device=master

# Migrate server wide data - logins, roles, remote servers, etc...
# valid only with "setup" mode, default is yes
migrate_server_data=yes

# Tell sybmigrate to lock or unlock all login accounts on the
# target Adaptive Server.Valid values are "yes" and "no":
# "yes" to lock and "no" to unlock.This is only valid if
# "migrate_server_data" is set to "yes" and run in "setup" mode.
# If this attribute is not specified, target Adaptive Server login
```



```
# accounts are not change.
lock_account=no

# Change target Adaptive Server login passwords.This is only valid
# if "migrate_server_data" is set to "yes" and run in "setup" mode.
# If this attribute is not specified, target Adaptive Server login
# accounts are not change.
# The valid values are "<generate>" and password file.
# "<generate>" instructs sybmigrate to use random passwords.
# Password file instructs sybmigrate to use the passwords from
# this file.
# The content of the password file consists of two columns:
# the login name column and the password string column.
# The separator between the columns are tabs and or spaces.
login_password_file=<generate>

#####
# Database information
#####

#
# Migrate the "pubs2" database objects
#
[database]
# Specify the source target database to migrate.
source_database_name=pubs2
target_database_name=pubs2

# Migrate database data, valid only if "migrate_server_data"
# was set to "yes" in "setup" mode.This is default to yes.
migrate_database_data=yes

# Work database setup attributes.This is required with "setup" mode.
# Work database size in MB.
work_database_size=5
# Device used to create the work database.
work_database_device=master

# Number of threads use to do user table data copy
data_copy_thread=5

# Number of thread use to create indexes.
create_index_thread=1

# Number of thread use to do ddl migration/validation
work_thread=10
```

```
# Automatically select the dependent objects for migration and
# validation. Valid values are "yes" or "no".
auto_select_dependent_objects=yes

#
## Migrate objects
#
# These attributes specify the list of DDL object to
# migrate or validate. User can directly specify the
# list of DDL object or ask Migration tool to query the
# list. Directly specifying the list has the higher
# precedence. The SQL command will ignore if the list
# is given.
#
# Note:
# * The SQL command for the "_list_from_sql" attributes
# must return column <object name> or columns <user
# name> and <object name>
# * Index type must also specify the table name. For

# example, "<table>.<index name>" for
# "index_create_list" attribute or columns <table>,
# <index name> for "index_create_list_from_sql"
# attribute.
# * Value "<ALL_OBJECTS>" can be used on any of the
# attributes to specify all objects for the type.
# * If none of these attributes are given, all objects
# and data are migrated.
#
user_defined_type_create_list=
id
dbo.tid

default_create_list_from_sql=
select user_name(uid), name from sysobjects
where type = 'D'

rule_create_list=
pub_idrule, title_idrule

table_create_list=
publishers
titles
dbo.authors
dbo.titleauthor
```

```
dbo.roysched
stores
dbo.sales
dbo.salesdetail
dbo.discounts
dbo.au_pix
blurbs

table_migrate_list=
dbo.publishers titles dbo.authors dbo.titleauthor
dbo.roysched
stores dbo.sales dbo.salesdetail dbo.discounts au_pix
dbo.blurbs

index_create_list=
dbo.authors.auidind
dbo.authors.aumind
publishers.pubind
roysched.titleidind
sales.salesind
salesdetail.titleidind

salesdetail.salesdetailind
titleauthor.taind
titleauthor.auidind
titleauthor.titleidind
titles.titleidind
titles.titleind

trigger_create_list=
deltitle
totalsales_trig

store_procedure_create_list_from_sql=
select name from sysobjects where type = 'P'

view_create_list_from_sql=<ALL_OBJECTS>

referential_constraint_create_list_from_sql=<ALL_OBJECTS>

logical_key_create_list_from_sql=<ALL_OBJECTS>

#####

#
# Migrate the "pubs3" database objects
```

```
#
[database]
source_database_name=pubs3
target_database_name=pubs3

# Migrate database data - user, etc.
migrate_database_data=yes

# These two attributes valid only with "setup" mode
work_database_size=5
work_database_device=master

# Number of threads use to do user table data copy
data_copy_thread=5

# Number of thread use to create indexes.
create_index_thread=1

# Number of thread use to do ddl migration/validation
work_thread=10

# Migrate objects

user_defined_type_create_list=<ALL_OBJECTS>

default_create_list=<ALL_OBJECTS>

rule_create_list=<ALL_OBJECTS>

table_create_list=
dbo.authors
publishers
dbo.titles
dbo.roysched
stores
dbo.sales
dbo.store_employees
salesdetail
dbo.titleauthor
dbo.discounts
blurbs

table_migrate_list_from_sql=<ALL_OBJECTS>

index_create_list=<ALL_OBJECTS>
```

```
trigger_create_list=<ALL_OBJECTS>

store_procedure_create_list=<ALL_OBJECTS>

view_create_list=<ALL_OBJECTS>

referential_constraint_create_list_from_sql=<ALL_OBJECTS>

logical_key_create_list_from_sql=<ALL_OBJECTS>

#####

#
# Migrate all the "foo" database objects with default settings.
#
[database]
source_database_name=foo
target_database_name=foo

# Migrate database data - user, etc.
migrate_database_data=yes

# These two attributes valid only with "setup" mode

work_database_size=5
work_database_device=master

# Number of threads use to do user table data copy
data_copy_thread=5

# Number of thread use to create indexes.
create_index_thread=1

# Number of thread use to do ddl migration/validation
work_thread=10
```

暗号化カラムに対する `sybmigrate` の使用

暗号化カラムが含まれるデータベースでは、`sybmigrate` によって次の項目がマイグレートされます。

- 1 システム暗号化パスワード – システム暗号化パスワードをマイグレートしないように指定すると、`sybmigrate` は、暗号テキストを直接マイグレートする代わりに、`reencrypt` 方法を使用して暗号化カラムをマイグレートします。
- 2 デフォルトでの暗号テキスト形式の暗号化カラム – ソースでのデータの復号化とターゲットでの再暗号化によるオーバーヘッドが回避されます。ただし、場合によっては、`sybmigrate` でマイグレート方法として `reencrypt` が選択され、ソースでのデータの復号化とターゲットでの再暗号化が行われることもあります。
- 3 暗号化キー – マイグレートするキーをユーザが選択できます。また、現在のデータベースのカラムの暗号化に使用されたキーは、`sybmigrate` によって自動的に選択されます。ユーザがシステム暗号化パスワードのマイグレーションを選択した場合は、`sybmigrate` が暗号化キーを実際の値でマイグレートします。
sysencryptkeys システム・テーブルのキー値は、システム暗号化パスワードを使用して暗号化されています。このような値がマイグレートされます。ユーザがシステム暗号化パスワードをマイグレートしなかった場合、`sybmigrate` はキーを名前でマイグレートします。これは、ターゲットで適切に復号化できないキーのマイグレートを回避するためです。キーを名前でマイグレートすると、ソースとは異なるキー値のキーがターゲットで作成されます。
- 4 データ – デフォルトでは、データは暗号テキスト形式で転送されます。暗号テキスト・データは、異なるオペレーティング・システムにマイグレートできます。文字データの場合は、ターゲット・サーバーでソース・サーバと同じ文字セットが使用されている必要があります。

`sybmigrate` は、データベースを作業の単位として実行されます。ソース・サーバのデータベースのデータが、別のデータベースのキーで暗号化されている場合は、キーのデータベースを最初にマイグレートします。

`sybmigrate` で、マイグレートしたデータの再暗号化が選択されるのは次の場合です。

- 現在のデータベースのいずれかのキーがマイグレーションの対象として選択されていない、またはいずれかのキーがすでにターゲット・サーバに存在している場合。ターゲットのキーがソースのキーと同一である保証はありません、そのためマイグレートするデータを再暗号化する必要があります。
- システム・パスワードがマイグレーションの対象として選択されなかった場合。ターゲットのシステム・パスワードがソースと異なるときは、キーを値でマイグレートできません。また、データも暗号テキストとしてマイグレートできません。
- ユーザが次のフラグを使用する場合。

```
sybmigrate -T 'ALWAYS_REENCRYPT'
```

データの再暗号化によってパフォーマンスが低下する場合があります。再暗号化モードでマイグレーションを実行すると、この影響に関するメッセージがマイグレーション・ログ・ファイルに書き込まれます。

暗号化カラムをマイグレートするには、`sa_role` と `sso_role` の両方を有効にしてください。

マイグレーション後のアクティビティ

- `sybmigrate` は、このマニュアルに示すオブジェクトのマイグレーションだけをサポートしています。ターゲット `Adaptive Server` を完全に機能させるには、その他のスキーマ・オブジェクトと設定情報を手動でマイグレートします。
- インデックスを再構築するときに、インデックスの統計が自動的に再作成されます。ただし、`sybmigrate` で、インデックスのないカラムから統計を再作成することはできません。インデックス統計に対するユーザ定義のステップ値は、マイグレーションでは保持されません。送信元サーバ側の統計と同様のターゲット・サーバ側の統計を取得するには、`optdiag` を使用して、インデックスのないカラムに統計が含まれるテーブルを指定します。インデックスのないカラムの中で統計を含むカラムを判別したら、統計を手動で更新します。

- マイグレーション・ログに記録されている、先頭に “attention” という単語が付いたユーザの注意を促すメッセージを確認します。
- オブジェクト・マイグレーション・ステータス・レポートを実行して、すべてのオブジェクトがマイグレートされたかどうかを確認します。

Replication Server ドメインのデータベースのマイグレート

Replication Server ドメインには、次のデータベースが1つ以上含まれています。

- プライマリ・データベース
- 複写データベース
- Replication Server システム・データベース (RSSD: Replication Server System Database)

注意 RSSD には、Replication Server システム・テーブルが格納されます。また、RSSD がプライマリ・データベースまたは複写データベースでもかまいません。

これらのデータベースはすべてマイグレートできますが、マイグレーションを確実に成功させるために、プロセスに追加の手順が必要となります。

マイグレーション前の手順

各データベースの複写が完了していることを確認してからマイグレーションを開始してください。これは次のことを意味します。

- プライマリ・データベースの場合 – サブスクライブするすべてのデータベースにすべての変更内容が適用されている。
- 複写データベースの場合 – データベースがサブスクライブするすべての変更内容が適用されている。

注意 Replication Server のインバウンド・キューとアウトバウンド・キューのすべてのトランザクションを適用してください。マイグレーション後に、Adaptive Server トランザクション・ログに残されたデータをリストアすることはできません。

❖ データをマイグレートする前

- 1 Replication Server にログインし、ログ転送を中断します。次のように入力します。

```
suspend log transfer from server.database
```

- 2 Adaptive Server にログインし、RepAgent を停止します。次のように入力します。

```
use database  
sp_stop_rep_agent database
```

- 3 複写データベースへの DSI 接続をすべて中断します。Replication Server にログインし、次のように入力します。

```
suspend connection to server.database
```

- 4 Replication Server をハイバネーション・モードにします。次のように入力します。

```
sysadmin hibernate_on, replication_server
```

マイグレーション・プロセスを開始する前に、複写情報がログに記録されます。マイグレーション後の手順で複写情報をリストアするために必要な情報は、このログから取得できます。詳細については、「[マイグレーション後の手順](#)」(409 ページ)を参照してください。

マイグレーション後の手順

マイグレーション後に、データベース内の複写情報をリストアします。これらの手順は repl レポートで生成できます。

送信元データベースとターゲット・データベース間でページ・サイズを変更する場合は、「[論理ページ・サイズを変更した場合のシステム・テーブルの修正](#)」(411 ページ)の指示にも従ってください。

Replication Server は、*server_name.database_name* によってすべての接続を識別します。マイグレーション後に、ターゲット・サーバ(マイグレート先のサーバ)の名前を送信元サーバ(マイグレート元のサーバ)の名前に変更してください。

プライマリ・データベースのリストア

RSSD がプライマリ・データベースである場合は RSSD も含め、すべてのプライマリ・データベースで次の手順に従います。

マイグレーション中にページ・サイズを変更する場合は、「[論理ページ・サイズを変更した場合のシステム・テーブルの修正](#)」(411 ページ)の説明に従って、rs_lastcommit システム・テーブルと rs_threads システム・テーブルも変更してください。

❖ プライマリ・データベースのリストア

- 1 元のプライマリ・データベースでウォーム・スタンバイがオンになっていた場合は、スタンバイ・ステータスをリストアします。次のように入力します。

```
sp_reptostandby database_name, status
```

スタンバイ・ステータスは送信元データベースのマイグレーション・ログに保存されています。

- 2 生成 ID を 1 だけ増やします。次のように入力します。

```
dbcc settrunc ("ltm", "gen_id", gen_id)
```

現在の生成 ID は送信元データベースのマイグレーション・ログで確認できます。

- 3 セカンダリ・トランケーション・ポイントをリセットします。

```
dbcc settrunc ("ltm", "valid")
```

- 4 このデータベースの Replication Server のロケータ値をゼロにします。次のように入力します。

```
rs_zeroltm server, database_name
```

- 5 このデータベースがウォーム・スタンバイ設定のアクティブ接続である場合は、プライマリ・データベースをダンプし、そのダンプをスタンバイ・データベースにロードすることによって、スタンバイ・データベースを実体化し直します。詳細については、Replication Server のマニュアルを参照してください。

- 6 プライマリ・データベース上で RepAgent を起動します。次のように入力します。

```
sp_start_rep_agent database_name
```

- 7 Replication Server にログインし、ログ転送を再開します。

```
resume log transfer from server.database
```

複写データベースのリストア

マイグレーション中にページ・サイズを変更しない場合は、複写データベースに必要なマイグレーション後の手順はありません。

ページ・サイズを変更した場合は、「[論理ページ・サイズを変更した場合のシステム・テーブルの修正](#)」(411 ページ) の手順に従います。

RSSD のリストア

RSSD をリストアする手順は次のとおりです。

❖ RSSD のリストア

- 1 RSSD がプライマリ・データベースの場合は、「[プライマリ・データベースのリストア](#)」(410 ページ) の手順に従います。

ページ・サイズを変更した場合は、指示に従って `rs_lastcommit` システム・テーブルと `rs_threads` システム・テーブルを必ず変更してください。

- 2 Replication Server のハイバネーションをオフにします。Replication Server にログインし、次のように入力します。

```
sysadmin hibernate_off replication_server
```

論理ページ・サイズを変更した場合のシステム・テーブルの修正

マイグレーション中に論理ページ・サイズを変更した場合は、`rs_lastcommit` システム・テーブルと `rs_threads` システム・テーブルを変更してページ・サイズの変更に対応します。

ページ・サイズを変更したすべてのデータベースで、次の手順に従います。

❖ `rs_lastcommit` システム・テーブルと `rs_threads` システム・テーブルの変更

- 1 `rs_lastcommit` テーブルを変更します。次のように入力します。

```
declare @pad8_size integer
declare @alter_cmd varchar(200)

select @pad8_size = (@@maxpagesize / 2)
- (select sum(A.length) from
syscolumns A, sysobjects B
where A.id = B.id
and B.name = 'rs_lastcommit')
+ (select A.length from
```

```

syscolumns A, sysobjects B
where A.id = B.id
and B.name = 'rs_lastcommit'
and A.name = 'pad8')

select @alter_cmd = "alter table rs_lastcommit "
+ "modify pad8 char("
+ convert(varchar(100), @pad8_size)
+ ")"
execute (@alter_cmd)
go

```

- 2 `rs_threads` テーブルを変更します。次のように入力します。

```

declare @pad4_size integer
declare @alter_cmd varchar(200)

select @pad4_size = (@@maxpagesize / 2)
- (select sum(A.length) from
syscolumns A, sysobjects B
where A.id = B.id
and B.name = 'rs_threads')
+ (select A.length from
syscolumns A, sysobjects B
where A.id = B.id
and B.name = 'rs_threads'
and A.name = 'pad4')

select @alter_cmd = "alter table rs_threads "
+ "modify pad4 char("
+ convert(varchar(100), @pad4_size)
+ ")"
execute (@alter_cmd)
go

```

ログ

マイグレーション・ツール・ログでは、複製オブジェクトに関する情報が次の見出しで始まります。

```
=== Replication Information for Database 'pdb1' ===
```

次に示すのは、`pdb1` という名前のプライマリ・データベースのサンプル・ログ・ファイルです。

```
sp_repostandby 'pdb1' is NONE.
```

データベースのスタンバイ・ステータスが NONE でない場合は、マイグレーション後の手順の説明に従って、スタンバイ・ステータスを使用します。

```
sp_config_rep_agent 'pdb1'
```

`sp_config_rep_agent` は、現在の RepAgent 設定を要求します。マイグレーション・ツールは RepAgent 設定を自動的にリストアします。このログを使用して RepAgent 設定を確認できます。

Parameter name	Default	Config Value	Run value
priority	5	5	5
fade timeout	30	30	30
scan timeout	15	15	15
retry timeout	60	60	60
rs username	n/a	rs1_user	rs1_user
trace flags	0	8194	8194
batch ltl	true	true	true
rs servername	n/a	rs1	rs1
send buffer size	2k	2k	2k
trace log file	n/a	n/a	n/a
connect database	n/a	pdb1	pdb1
connect dataserver	n/a	pds1	pds1
can batch size	1000	1000	1000
security mechanism	n/a	n/a	n/a
msg integrity	false	false	false
unified login	false	false	false
kip ltl errors	false	false	false
msg origin check	false	false	false
short ltl keywords	false	false	false
msg confidentiality	false	false	false
data limits filter mode	stop	stop	stop
msg replay detection	false	false	false
mutual authentication	false	false	false
send structured oqids	false	false	false
send warm standby xacts	false	false	false
msg out-of-sequence check	false	false	false
skip unsupported features	false	false	false
send maint xacts to replicate	false	false	false

(28 rows affected)

これは、明示的に複製されたテーブルのリストです。`sybmigrate` は、明示的に複製されたテーブルの複製ステータスを自動的にリストアします。ログのこの部分を使用して、明示的に複製されたテーブルの複製ステータスを確認できます。

```
sp_setreptable
```

```

Name                                     Repdef Mode
-----
t1                                       owner_off
t2                                       owner_on
(2 rows affected)

```

これは明示的に複製されたストアド・プロシージャのリストです。マイグレーション・ツールは、明示的に複製されたストアド・プロシージャの複製ステータスを自動的にリストアします。ログのこの部分を使用して、明示的に複製されたストアド・プロシージャの複製ステータスを確認できます。

```

sp_setrepproc
Name                                     Type          Log Mode
-----
p1                                       function     log_sproc
p2                                       function     log_current
p3                                       table        log_sproc
p4                                       table        log_current
(4 rows affected)

```

これはセカンダリ・トランケーション・ページに関する情報です。マイグレーション後の手順で、**generation_id** カラムが必要になります。

```

dbcc gettrunc
secondary_trunc_page  secondary_trunc_state  dbrepstat  generation_id
      database_id      database_name          ltl_version
-----
621                   1                      167        0
      6                pdb1                   400
(1 row affected)

```

This appears to be a replicated primary database. Make sure the post processing steps for a replicated primary database are performed. Please consult the manuals for the steps that need to be performed.

次に示すのは、データベースが複製データベースの場合のサンプル・ログ・エントリです。

This appears to be a replicate database. If the pagesize is greater than 2K, make sure the post processing steps for a replicate database are performed. Please consult the manuals for the steps that need to be performed.

次に示すのは、RSSD データベースのサンプル・ログ・エントリです。

This appears to be a replication system database

Make sure the post processing steps for a replication system database are performed. Please consult the manuals for the steps that need to be performed

データベースには3つのカテゴリを表示できるため、1つのデータベースに対して3つのログがすべて存在する場合があります。

ワイド・データをサポートするデータベースのマイグレート

Adaptive Server バージョン 12.5 以降では、Replication Server バージョン 12.1 以前で処理できない長さのデータが生成されることがあります。RepAgent から Replication Server 12.1 以前にワイド・データが渡されると、Replication Server スレッドが停止する場合があります。

RepAgent は、ログ転送言語 (LTL: Log Transfer Language) を使用して Replication Server と通信します。表 9-1 に示すように、RepAgent は Replication Server に接続すると、LTL のバージョンを返します。

表 9-1 : Replication Server と LTL のバージョン

Replication Server のバージョン	LTL バージョン
12.1 以前	< 400
12.5 以降	>= 400

Replication Server が返した LTL のバージョンが 400 未満の場合、RepAgent は `data limits filter mode` オプションの設定を使用して、ワイド・データを処理する方法を決定します。

`data limits filter mode` オプションは、`sp_config_rep_agent` を使用して設定できます。`data limits filter mode` の値は次のとおりです。

- **stop** — RepAgent は、Replication Server が処理できない長さのデータを検出すると停止します (LTL のバージョンが 400 未満の場合のデフォルト)。
- **skip** — RepAgent は、Replication Server で処理できない長さのデータを無視し、情報メッセージをログに記録します。
- **truncate** — Replication Server がデータを処理できるように、RepAgent はワイド・データをトランケートします。テーブルまたはストアド・プロシージャに 250 を超えるカラムまたはパラメータがある場合、最初の 250 のカラムまたはパラメータだけが送信されます。カラムまたはパラメータが 255 バイトを超える場合は、最初の 255 バイトだけが送信されます。

- off – RepAgent はワイド・データを Replication Server に送信しません。そのため、Replication Server スレッドが停止する場合があります。

表 9-2 は、Replication Server 12.1 以前と Replication Server 12.5 以降の列数と列幅の制限を示します。

表 9-2 : Replication Server 列数と列幅の制限

プロパティ	Replication Server 12.1 以前	Replication Server 12.5 以降
列数	250	65535
列幅	255	65535

制限事項

注意 sybmigrate でサーバ・データをマイグレートするときは、ターゲット Adaptive Server カタログにデフォルト・データのみが含まれるようにする必要があります。デフォルト・データは、Windows マシンと UNIX マシンで異なります。このため、UNIX マシンから Windows マシンにマイグレートするときに問題が発生します。UNIX マシンから Windows マシンに正常にマイグレートするには、ターゲット Windows マシンの XP Server 名と mon_user ログインを削除します。

高可用性

高可用性環境では、データ・マイグレーションがサポートされていません。高可用性環境を停止してから、データベース・マイグレーションを開始してください。

- ❖ **高可用性環境を停止してからデータベース・マイグレーションを開始する**
 - 1 プライマリ Adaptive Server とセカンダリ Adaptive Server の組み合わせを解除します。
 - 2 プライマリ送信元 Adaptive Server とセカンダリ送信元 Adaptive Server のデータを、プライマリ・ターゲット Adaptive Server とセカンダリ・ターゲット Adaptive Server に別々にマイグレートします。
 - 3 ターゲット Adaptive Server を高可用性環境用に設定します。

警告！ 高可用性機能を実行するには、プライマリ Adaptive Server とセカンダリ Adaptive Server を同じ論理ページ・サイズに設定します。

その他の制限事項

- **sybmigrate** は、DTM/XA 環境用の特殊な処理を実行しません。オープン・トランザクションと準備済みで未処理のトランザクションのステータスについて考慮する必要があります。特殊な処理が必要な場合は、手動で実行してください。
- **sybmigrate** でさまざまなオブジェクトの依存性を確実に判別することはできません。**sybmigrate** は、オブジェクトをマイグレートする順序を、ほかのオブジェクトに対する依存性に基づいて作成することはありません。ビューはほかのビューに依存する場合があります。依存されているビューがマイグレートされていない場合は、ビューが再作成されません。ストアド・プロシージャとトリガが依存するデータがマイグレートされていない場合は、これらのマイグレーションが失敗する場合があります。データベース間に依存性がある場合は、関連するオブジェクトのマイグレーションを調整する必要があります。選択した組み合わせに依存性がある場合は、その依存性が考慮されます。ただし、選択した組み合わせの外部との間に依存性がある場合は、マイグレーションで2回以上 **sybmigrate** を実行する必要があります。このため、データ・マイグレーションを正しく完了するのに、部分的な再試行が必要な場合があります。
- **Adaptive Server** バージョン 12.5.3 以降を使用すると、ターゲット・サーバ上のワーク・データベースのサイズとロケーションを指定できます。ソース・サーバの **Adaptive Server Enterprise** のバージョンが 12.0 以降で 12.5.0.1 より前の場合は、データベースまたはサーバをマイグレートするときに、ターゲット・サーバ上のワーク・データベースのサイズとロケーションを指定してください。
- 送信元データベースとターゲット・データベースの名前は同じでなければなりません。**ddlgen** で生成される SQL スキーマにあるオブジェクトは、送信元 **Adaptive Server** 名で修飾します。
- **sybmigrate** は、マイグレーション・アクティビティの監査をサポートしていません。
- コンパイル済みオブジェクト (プロシージャ、ビュー、ルール、デフォルト) の名前を変更しても、**syscomments** のオブジェクト名は更新されません。

マイグレーション中、**DDLGen** はテキストの古い名前を使用して **syscomments** のオブジェクトを問い合わせます。**DDL** マイグレーション中に、テキストのこの古い名前が **sybmigrate** で問題となります。

トラブルシューティングとエラー・メッセージ

この項では、一般的なエラーとその対処方法、およびさまざまなエラー・メッセージとその意味について説明します。

オブジェクトのマイグレーション失敗

最初の試みで、オブジェクトのマイグレーションが失敗する場合があります。マイグレーションが失敗すると、必ず自動的に `sybmigrate` が再試行されます。ただし、まだマイグレートされていないオブジェクトに依存するオブジェクトをマイグレートしようとすると、マイグレーションが失敗します。

オブジェクトのマイグレーション失敗を防止するには、マイグレーションするオブジェクトの依存性を調べます。たとえば、トリガが定義されているテーブルがマイグレートされていない場合は、トリガをマイグレートすることはできません。同様に、ほかのビューまたはテーブルに基づいてビューを作成した場合、これらのオブジェクトがマイグレートされていないと、ビューのマイグレーションが失敗します。

データベースのマイグレーション開始

マイグレーション・プロセスの設定フェーズで、サーバ・データをマイグレートするかどうかを選択するよう求められます。[はい]、[いいえ]、または[未決定]を選択します。

[未決定]を選択すると、マイグレーション・プロセスを設定しておいて、後で都合のよいときにマイグレーションを実行できます。[未決定]を選択した場合は、サーバ・データをマイグレートするかどうかを指定するまで、データベースのマイグレーションを開始できません。

設定時にサーバ・データをマイグレートしないことを指定した場合、マイグレーション中にデータベース・データをマイグレートできません。GUI モードでは、この制限を無効にできます。

接続の拒否とサーバへの接続を確保できない

これらのエラー・メッセージは、次の2つの理由により生成されます。

- 送信元 Adaptive Server またはターゲット Adaptive Server が稼働していないため、`sybmigrate` が接続を確立できない。
- `number of user connections` 設定パラメータで、送信元 Adaptive Server とターゲット Adaptive Server に十分なリソースが提供されていない。

送信元サーバからターゲット・サーバにアクセスできない

`interfaces` ファイルは、送信元 Adaptive Server を起動するために使用します。このファイルに、ターゲット Adaptive Server を示すエントリがあるかどうか確認してください。

現在のログインで、送信元 Adaptive Server からターゲット Adaptive Server にアクセスできるかどうか確認してください。

sybmigrate がマイグレーション中にハングする

マイグレーション・プロセス中に *sybmigrate* がハングする場合は、`$$SYBASE/$SYBASE_ASE/init/logs` にある *sybmigrate* ログで、エラーまたは例外をチェックしてください。

Adaptive Server ログもチェックしてください。Adaptive Server ログの領域がデータベース上で不足している場合は、データベース・サイズを増やし、`sp_thresholdaction` ストアド・プロシージャをインストールして、ログがいっぱいになった時点で `dump tran` を実行します。

2 つのデータベースのマージ

送信元 Adaptive Server 上の 2 つのデータベースを、ターゲット Adaptive Server 上で 1 つのデータベースにマージするには、次の手順に従います。

- 最初のデータベースを設定してマイグレートします。
- 最初のデータベースをマイグレートした後で、ターゲット・データベースの名前を変更して、2 番目の送信元データベースと同じ名前にします。
- 2 番目のデータベースを設定してマイグレートします。

注意 ターゲット・データベース上にユーザ、役割、およびその他のデータベース・データが存在する場合、2 番目のデータベースのデータをマイグレートすることはできません。ユーザ・データはマイグレートできます。

マイグレーション後の失敗クリーンアップ

sybmigrate が予期せず失敗した場合は、失敗した領域で *sybmigrate* を再実行します。それ以上進行しないで再度失敗した場合は、送信元 Adaptive Server とターゲット Adaptive Server をクリーンアップし、マイグレーションを再び開始します。送信元 Adaptive Server とターゲット Adaptive Server の両方で、以下のアクションを実行します。

送信元 Adaptive Server :

- テンポラリ作業データベース `mtpdb%` を削除します。
- レポジトリ・データベース `sybmigratedb` を削除します。
- すべてのリモート・サーバ `mtrs%` を削除します。

ターゲット Adaptive Server :

- サーバ・データをマイグレートした場合は、`srvbuild` または `syconfig` を使用して、ターゲット Adaptive Server を再構築します。
- ターゲット・データベースを再作成します。

1つのデータベースの再マイグレート

特定のデータベースを再マイグレートするには、次の手順に従います。

- 1 `sybmigrate` を起動します。
- 2 設定セッション時に [パスの設定] ウィンドウで、再実行するマイグレーション・パスを右クリックします。
- 3 ポップアップ・メニューで、[マイグレーション・パスの削除] を選択します。
- 4 ターゲット・データベース上で、マイグレートしたデータとオブジェクトをクリーンアップまたは削除するか、ターゲット・データベースを削除して再作成します。
- 5 設定モードから `sybmigrate` を再起動して実行します。

個々のオブジェクトの再作成

個々のオブジェクトを再作成するには、次の操作を行います。

- 1 ターゲット Adaptive Server で、再作成するオブジェクトを削除します。
- 2 マイグレーション・セッションで `sybmigrate` を起動し、[オブジェクト選択のマイグレート] ウィンドウに移動します。作成するオブジェクトを強調表示して、右クリックします。
- 3 ポップアップ・メニューから、[オブジェクトを初期状態にリセット] を選択します。
- 4 マイグレーション・プロセスを完了します。

接続失敗

送信元 Adaptive Server とターゲット Adaptive Server が稼働しているにもかかわらず、接続失敗を示すエラー・メッセージが表示される場合は、不適切な文字セットを使用している可能性があります。

`sybmigrate` を実行するときは、デフォルトの文字セットを使用してください。-J *charset* オプションを使用して `sybmigrate` を実行し、使用する文字セットを変更します。

JVM 共有クラスのメモリ不足

サーバ・ログに次のエラーが記録されている場合は、`size of shared class heap` 設定パラメータを大きい値に設定してください。

```
01:00000:00036:2002/01/28 14:17:05.63 server Java VM
Host:Memory allocation request failed because of
insufficient memory in Jvm Shared Class.
```

プロシージャ・キャッシュのメモリが不足

インデックスのマイグレーション中に、`there is not enough memory in the procedure cache` というエラー・メッセージが表示された場合は、`sp_configure procedure cache size` を使用してプロシージャ・キャッシュを増やします。

java.lang 関連のエラー

Adaptive Server への接続時に、
`java.lang.NoClassDefFoundError: com/sybase/jdbcx/SybDriver`
というエラー・メッセージが表示された場合は、jConnect 6.0 が
`$SYBASE` ディレクトリ (`$SYBASE/jConnect-6_0`) にインストールされて
いるかどうか確認してください。

索引

記号

- <(リダイレクト・イン)、isql 16
- >(リダイレクト・アウト)、isql 16
- ¥¥0 (null) 文字ターミネータ、対話型 bcp 78

A

- Adaptive Server
 - アップグレード (sqlupgrade) 254, 287
 - アップグレード、リソース・ファイル (sqlupgraderes) 288
 - 実行 (dataserver) 173
 - 実行 (sqlsrv) 279
 - 設定、マイグレーション 382
 - プロセスのロールバック 94
- Adaptive Server のアップグレード 254, 287
 - 「sqlupgrade ユーティリティ・コマンド」参照 287
- additional network memory** パラメータ 379
- allow_dup_row オプション、create index、bcp 98
- ASCII フォーマット、bcp 68
- ascii_7 文字セット
 - defncopy 210

B

- Backup Server
 - 「backupsrv ユーティリティ・コマンド」参照 133
- backup Server 133
- backupsrv ユーティリティ・コマンド 133
 - interfaces ファイル 137
 - interfaces ファイル、名前とロケーション 133
 - LC_ALL 環境変数 134
 - エラー・メッセージ 138

- エラー・ログ・ファイル 134
- 起動、サーバ 137
- 言語、デフォルト 135
- サーバ接続、個数 133
- サーバ名、指定 133
- 定義 133
- トレース・フラグ 138
- 必要なパーミッション 138
- フル・パス名、指定 134
- 文字セット、デフォルト 135
- bbatch_size パラメータ 94
- bcp<default para font** 65
- bcp の低速バージョン 52
 - デッドロック、インデックス・ページ 56
- bcp ユーティリティ・コマンド 100, 139
 - allow_dup_row オプション、create index 98
 - ASCII フォーマット 68
 - bbatch_size パラメータ 94
 - dump database 100
 - IDENTITY カラム 62
 - load database 100
 - load transaction 100
 - non-iso_1 データ・ファイル 92
 - インデックスとトリガ、削除 53
 - インデックスの作成 98
 - エラー・ファイル 97
 - エラー、データ変換 97
 - オペレーティング・システムによる違い 68
 - 記憶タイプ 77
 - 言語、使用、代替 92
 - 向上、パフォーマンス、オペレーション 96
 - 高速バージョン 53
 - 高速バージョン、データのリカバリ性 52
 - コピー・アウト、データ 85, 88, 96
 - コピー・イン、データ 53, 57, 88, 92
 - コピー・イン、バッチ、テーブルのロー 94
 - 最高速度、有効化 99

- システム・データ・フォーマット (SDF) 出力 86
 - 使用例 148
 - 設定パラメータ 96
 - 説明 44
 - 挿入 100
 - 速度、インデックスとトリガ 50
 - 速度、モード 46
 - ターミネータ 45
 - プロンプト、応答。「対話型 bcp」参照 69, 70
 - 対話型モード 70, 79
 - 定義 139
 - 低速バージョン 52
 - 低速バージョンとデッドロック、インデックス・ページ 56
 - データ型 77
 - データ記憶サイズ 72
 - データ整合性 98
 - データ転送、準備 48
 - データ転送、プログラム間 45
 - データに必要な領域 53
 - データの消失とダンプ 50
 - データ・リカバリに関する警告 146
 - テーブルのデフォルト、コピー、データ 99
 - デフォルト値、データ 99
 - デフォルト、カラムとデータ型 99
 - 特殊文字、処理 147
 - トリガとデータのコピー 99
 - トリガ、起動されない、ターゲット・テーブル 50
 - ネイティブ・ファイル・フォーマット 67
 - ネイティブ・フォーマット・オプション 68
 - バイナリ形式 45
 - パスワードの暗号化 146
 - バッチ・オペレーション 94
 - バッチ・サイズ、設定 142
 - パフォーマンスの問題 63, 99
 - 非対話型 67
 - 必要なパーミッション 47, 74
 - ファイル記憶タイプ 71
 - フィールド・ターミネータ 69, 72
 - フィールド長 68
 - フォーマット・ファイル 81, 85
 - プレフィクス長 71
 - プロセスのロールバック 94
 - プロンプト 71
 - 分割されたテーブル 54, 64, 95
 - ページ割り付け、増加 57
 - 他の Adaptive Server ユーティリティ 99
 - 文字セット、デフォルト 144
 - 文字データ型以外のデータ型から文字データ型への変換、デフォルトのフィールド長 76
 - 文字フォーマット 68
 - 文字フォーマット・ファイル 67
 - 文字フォーマット、使用可能 45
 - 文字フォーマット、デフォルト 69
 - ユニークな IDENTITY カラム値 63
 - リカバリ性の向上、コピー・イン、データ 94
 - ルール、コピー、データ 99
 - ロー・ターミネータ 69
 - ロー・レベルのアクセス・ルールの使用 93
- C**
- char データ型、bcp 67
 - charset ユーティリティ・コマンド 170
 - 設定 170
 - 定義 170
 - 必要なパーミッション 170
 - CIS bulk insert array size**
 - マイグレーション用の設定 378, 382
 - CIS packet size**
 - マイグレーション 379
 - マイグレーション用の設定 382
 - create index コマンド、bcp、重複ロー 98
- D**
- dataserver ユーティリティ・コマンド 173
 - 定義 173
 - パスワード、新規生成 180
 - 必要なパーミッション 180, 284
 - buildmaster ユーティリティ・コマンド。
 - 「dataserver」参照

- default network packet size 設定パラメータ 96
- defnccopy ユーティリティ・コマンド 209
- ascii_7 文字セット 210
 - Report Workbench、非互換性 209
 - 暗号化されたテキスト 214
 - 失敗、コメントが長い ため 213
 - 定義 209
 - パスワードとクラッシュ 209
 - 例 212
- DISPLAY 環境変数
- 設定、dsedit 112
- dscp ユーティリティ・コマンド 215
- exit コマンド 108
 - help 101, 109
 - quit コマンド 108
 - エン트리、新規作成、コピー 106
 - 起動 101
 - コマンド、一覧 108
 - サーバ・エントリの内容、表示 107
 - サーバ・エントリ、コピー 106
 - サーバ・エントリ、削除 108
 - サーバ・エントリ、追加 104
 - サーバ・エントリ、表示 107
 - サーバ・エントリ、変更 105
 - サーバの属性 103
 - セッション間でのコピー 106
 - セッション、オープン 102
 - セッション、クローズ 102
 - セッション、表示 102
 - セッション、別のセッションへのコピー 107
 - 定義 101, 215
 - 必要なパーミッション 215
 - 例 215
- dsedit に関するトラブルシューティング 123
- dsedit ユーティリティ・コマンド 216
- [Directory Service Session] 画面 114
 - DISPLAY 環境変数 112
 - libtcl.cfg ファイル 113
 - [Select a Directory Service] 画面 112
 - SYBASE 環境変数、libtcl.cfg の場所の特定 113
 - オープン、sql.ini ファイル 113
- 起動 112
- コマンド・ライン引数 111
 - サーバ・エントリ、コピー 119, 120, 122, 123
 - サーバ・エントリ、削除 118
 - サーバ・エントリ、追加 117, 120
 - サーバ・エントリ、追加、変更、削除ができない 124
 - サーバ・エントリ、名前の変更 118
 - サーバ・エントリ、変更 117, 120
 - サーバの属性 116
 - 実行、Windows エクスプローラ 111
 - 実行、コマンド・プロンプト 111
 - セッション、オープン 114
 - セッション、クローズ 114
 - 説明 111
 - 定義 216
 - トラブルシューティング 111, 123
 - ネットワーク・トランスポート・アドレス、編集 120
 - ネットワーク・トランスポート、追加 120, 123
 - 必要なパーミッション 112, 216
 - リモート・マシン、実行 112
 - ¥xd2 Unable to open X display¥xd3、エラー・メッセージ 123
- DSLISEN 環境変数
- backupsver 137
- dump database コマンド
- bcp 100
- dump transaction コマンド
- bcp 100
 - エラー・メッセージ、dump database の使用を指示 52
- ## E
- exit コマンド
- dscp 108
 - isql 10

G

GUI モード
マイグレーション 390

H

help コマンド (dscp) 101, 109

I

IDENTITY カラム
bcp 63
パラレル bcp 62, 63
ignore_dup_key オプション、create index、bcp 98
image データ型
対話型 bcp 74
insert コマンド
バルク・コピー、比較 100
interfaces ファイル
backupserver 137
backupserver、指定、名前とロケーション 133
dscp セッション、オープン 102
dscp、表示および編集 215
dsedit、表示および編集 216
dsedit、表示および編集、GUI 216
dsedit、編集 114
オープン、編集 112
interfaces ファイル
sybmigrate 386
iso_1 文字セット 92
isql 出力のフォーマット 10
isql ユーティリティ・コマンド 225
<< symbol 234
>(リダイレクト・アウト)記号 16
exit コマンド 10
quit コマンド 10
reset コマンド 12
Transact-SQL、使用 10
カラム・セパレータ文字 11
行番号、削除 11, 16
クエリ・バッファ、リセット 12

クエリ、編集 12
コマンド・ターミネータ 10
コマンド・ターミネータ、変更 13
コマンド・ターミネータ、リセット 227
出力ファイル 11
出力、フォーマット 10
定義 225
統計 14
統計値オプションとコマンド・ターミネータ
14
入力のエコー 16
入力ミスの訂正 12
ネットワーク・パケット・サイズ、指定 15
ネットワーク・パケット・サイズ、設定 15
パケット・サイズ、設定 15
文の最大サイズ 10
ホスト・ファイル、読み込み 16
例 226
<(リダイレクト・イン)記号 16
SQL パーサ・ユーティリティ。「isql ユーティリ
ティ・コマンド」参照 225

J

Java Runtime Environment のマイグレーション
376
Java カラムのマイグレーション 380

L

LANG 環境変数、backupserver 134
langinstall ユーティリティ・コマンド 238, 240
定義 238
必要なパーミッション 239
LC_ALL 環境変数、backupserver 134
load database コマンドと bcp 100
load transaction コマンド
bcp 100

M

- max memory、マイグレーション 379
- max network packet size 設定パラメータ 96
- max network packet size、マイグレーション用の設定 383
- max packet size allowed、sybmigrate 379
- max parallel degree、マイグレーション用の設定 382

N

- null 値 97
 - 対話型 bcp 74
- null 文字ターミネータ (¥¥0)、対話型 bcp 78
- number of pre-allocated extents 57
- numeric データ型
 - bcp、変換、文字記憶領域 76
 - オペレーティング・システム・フォーマット 68
 - カラムの位取り 85
 - カラムの精度 85

O

- optdiag コマンド、変更 246
- optdiag ユーティリティ・コマンド 241, 249
 - 定義 241, 249
 - 必要なパーミッション 247

P

- preupgrade ユーティリティ・コマンド 249

Q

- QPTune
 - collect アクション 267
 - compare アクション 267
 - fix_stats アクション 266
 - アプリケーションまたはクエリをチューニン

- グするための簡単な start アクション 266
- 欠落統計を修正するための start_stats アクション 264
- 構文 262
- チューニングのためのカスタムの start アクション 266
- パーミッション 268
- パラメータ 262
- 説明 262
- QPTune の collect アクション 267
- QPTune の compare アクション 267
- QPTune のカスタムの start アクション 266
- QPTune の簡単な start アクション 266
- quit コマンド
 - dscp 108
 - isql 10

R

- Replication Server、複写データを Adaptive Server にマイグレートする 408
- Report Workbench、defncopy との非互換性 209
- reset コマンド (isql) 12

S

- Security Mechanism、サーバの属性 (dscp) 104
- [Select a Directory Service] 画面 112
- select into コマンド、バルク・コピーとの比較 100
- [Server Entry Editor] ウィンドウ 115
- Server Name、サーバ属性 (dscp) 103
- Server Object Version、サーバの属性 (dscp) 103
- Server Service、サーバの属性 (dscp) 103
- Server Status、サーバの属性 (dscp) 103
- showserver ユーティリティ・コマンド 269
 - 定義 269
 - 例 269
- SPX/IPX アドレス、interfaces エントリ 121
- sql.ini ファイル
 - dsedit セッション、オープン 113
 - エントリ、コピー 119, 122, 123
 - エントリ、削除 118

- エントリ、追加 117
- エントリ、名前の変更 118
- エントリ、変更 117
- sqldbgr ユーティリティ 270
- sqlloc ユーティリティ・コマンド 276, 277
 - 定義 276
 - 必要なパーミッション 277
- sqllocres ユーティリティ・コマンド 278
 - 定義 278
 - 必要なパーミッション 278
- sqlsrvr ユーティリティ・コマンド 279
 - 定義 279
- sqlupgrade ユーティリティ・コマンド 254, 287
 - 定義 287
 - 必要なパーミッション 287
- sqlupgraderes ユーティリティ・コマンド 288
 - 定義 288
 - 必要なパーミッション 288
- srvbuild ユーティリティ・コマンド 289, 290
 - 定義 289
 - 必要なパーミッション 290
- srvbuildres ユーティリティ・コマンド 291, 292
 - 定義 291
 - 必要なパーミッション 292
- startserver ユーティリティ・コマンド 293
 - backupserver 137
 - runserver ファイル、例 294
 - 定義 293
 - 例 293
- SYBASE 環境変数 375
- SYBASE 環境変数、dsedit 113
- SYBASE_ASE 環境変数 375
- SYBASE_JRE 環境変数 376
- sybcluster コマンド
 - add backupserver 319
 - add instance 320
 - connect 322
 - create backupserver 324
 - create cluster 326
 - create xpserver 328
 - deploy plugin 329
 - diagnose cluster 331
 - diagnose instance 333
 - disconnect 334
 - drop backupserver 335
 - drop cluster 336
 - drop instance 337
 - drop xpserver 338
 - exit 339
 - localize 341
 - quit 343
 - set backupserver 344
 - set cluster 345
 - set instance 346
 - set xpserver port 347
 - show agents 348
 - show backupserver config 350
 - show cluster 351
 - show instance 354
 - show membership mode 357
 - show session 358
 - show xpserver 360
 - shutdown cluster 361
 - shutdown instance 362
 - start cluster 363
 - start instance 364
 - upgrade server 365
 - use 368
 - help 340
- sybcluster ユーティリティ 311
- sybdiag ユーティリティ・コマンド 296
 - 定義 296
- sybmigrate 304, 381, 386, 387
 - additional network memory パラメータ 379
 - CIS bulk insert array size 382
 - CIS bulk insert array size の設定 378
 - CIS bulk insert batch size の設定 382
 - CIS packet size の設定 379, 382
 - GUI モード 390
 - interfaces ファイル 386
 - Java Runtime Environment 376
 - Java カラム 380
 - JVM 用のメモリ 376
 - max packet size allowed の設定 379
 - max parallel degree の設定 382
 - number of sort buffers の設定 383
 - number of user connections の設定 382, 383
 - number of worker processes の設定 382
 - Replication Server データをマイグレートする

408
sybmigrate ログ 412
 依存性 374
 インストール 374
 インデックス・スレッド 381
 インデックスの再作成 407
 オブジェクト階層 376
 オブジェクトの再作成 420
 オブジェクトのマイグレーション失敗 418
 回避すべきエラー 383
 環境変数 375
 起動 385
 言語 380
 言語の指定 389
 検証セッション 377, 397
 高可用性環境でのマイグレーション 416
 構文 385
 コピー・スレッド 381
 コマンド・ライン・トレース・フラグ 389
 コンポーネント 373
 コンポーネント・ディレクトリ 375
 サーバ・データのマイグレーション 370
 実行ファイル 375
 出力テンプレート・リソース・ファイル
 388
 設定セッション 377, 390
 設定とチューニング 381
 送信元 Adaptive Server の設定 382
 データベース・データのマイグレーション
 371
 データベースの再マイグレート 420
 デバッグ・レベル 386
 トラブルシューティングとエラー・メッセー
 ジ 418
 バージョンの文字列 386
 パーミッション 374
 複写データベースに対するマイグレーション
 後の手順 409
 複写データベースのリストア 411
 複数のセッション 378
 2つのデータベースのマージ 419
 プライマリ・データベースのリストア 410

プラットフォーム 375
 ヘルプ情報 386
 マイグレーション後のアクティビティ 407
 マイグレーション失敗のクリーンアップ 419
 マイグレーション前の考慮事項 377
 マイグレート・セッション 377, 396
 文字セットの指定 389
 文字セット変換 380
 ユーザ定義のログ・ファイル 388
 リソース・ファイル・モード 398
 リリース・パス 375
 レポート 304, 387
 ロック・セッションの無効化 386
sybmigrate ユーティリティ・コマンド 303
 定義 303
SYBMIGRATE_MEMORY 環境変数
 マイグレーション 376
sybmigrate、**max memory** の設定 379
sybmigrate、**max network packet size** の設定 383
sybmigrate、マイグレートされないデータ 372
sybmultbuf、起動、Backup Server 134

T

t 304
 TCP/IP アドレス、interfaces エントリ 120
 text データ型
 対話型 bcp 74
 Transact-SQL
 使用、isql 10
 Transport Address、サーバの属性 (dscp) 104
 Transport Type、サーバの属性 (dscp) 103

X

XP Server、手動で起動 309, 310
 「xpserver ユーティリティ・コマンド」参照
 309
 xpserver ユーティリティ・コマンド 309, 310
 定義 309
 必要なパーミッション 310

あ

アーカイブ・データベースのダウングレード 385
 アーカイブ・データベースのマイグレート 377, 384
 アーカイブ・データベースへのアクセス
 ダウングレード 385
 マイグレート 377, 384
 アクション
 collect 267
 compare 267
 fix_stats 266
 カスタム start 266
 簡単な start 266
 欠落統計を修正するための start_stats 264
 アップグレード、Adaptive Server、リソース・ファイル 288
 「sqlupgraderes ユーティリティ・コマンド」参照 288
 アプリケーション・プログラム、コピー、データ 85
 暗号化された隠しテキスト 214

い

印刷できない文字、ホスト・ファイル 75
 インストール
 言語 (sqllocres) 278
 言語、GUI (sqlloc) 276, 277
 言語、新規 (langinstall) 238, 240
 ソート順 (sqllocres) 278
 ソート順、GUI (sqlloc) 276
 文字セット (sqllocres) 278
 文字セット、GUI (sqlloc) 276, 277
 インデックス
 bcp、削除、使用前 53
 速度低下、bcp 50
 引用符 (" "), 特殊文字を囲む 125
 引用符 (' '), 特殊文字を囲む 125

え

エラー
 オペレーション、保存、コピー・アウト中 98

オペレーション、保存、コピー・イン中 97
 データ変換 97, 98
 エラー・ファイル
 bcp 97, 98
 コピー・アウト 98
 エラー・メッセージ
 backupserver 138
 dsedit と X 画面 123
 dump database、使用、dump transaction 52
 エラー・ログ・ファイル
 backupserver 134
 ¥ (円記号)
 エスケープ、特殊文字 125
 円記号 (¥)
 エスケープ、特殊文字 125
 ターミネータ、対話型 bcp 78
 ¥ (円記号) データ・フィールド・ターミネータ、対話型 bcp 78

お

オープン
 セッション、dscp 102
 セッション、dsedit 114
 オブジェクト
 マイグレーション後の再作成 420
 マイグレーション失敗 418
 オペレーティング・システム
 コマンド・プレフィクス (!!)(isql) 235
 数値データ型のフォーマット 68
 文字データ型以外のデータ型のフォーマット 73
 オペレーティング・システム・ファイル
 ネイティブ・フォーマット 67
 必要なパーミッション、コピー、テーブル 47, 157

か

改行ターミネータ (¥¥n)
 bcp 69
 対話型 bcp 78

隠しテキスト、暗号化された 214
 数(量)
 サーバ接続、backupserver 133
 カラム
 null 97
 固定長と可変長 74
 セパレータ文字 (isql) 11
 データ型サイズ 75
 デフォルト値、bcp 99
 カラム精度、記憶フォーマット numeric または decimal 85
 カラムの位取り、記憶フォーマット numeric または decimal 85
 環境変数
 リリース・パス、**sybmigrate** 375
 DSLISITEN (backupserver) 137
 LANG (backupserver) 135
 LC_ALL (backupserver) 135
 SYBASE_ASE 375
 SYBASE_JRE 環境変数 376
 感嘆符 (!) オペレーティング・システム・コマンド・プレフィクス (isql) 235
 !! (感嘆符) オペレーティング・システム・コマンド・プレフィクス (isql) 235
 カンマ区切り出力 85, 87

き

記憶タイプ、bcp での使用 77
 起動
 dscp ユーティリティ 101
 dsedit ユーティリティ・コマンド 111, 112
 XP Server、手動 (xpserver) 309, 310
 サーバ (startserver) 293
 行番号、削除、isql 11, 16

く

クイック・リファレンス
 dscp コマンド 108
 クエリ・バッファ、リセット 12

け

警告
 bcp データの消失 50
 bcp、データ・リカバリ 146
 defncopy の失敗、コメントが長いた 213
 欠落統計
 fix_stats アクション 266
 start_stats アクション 264
 言語
 インストールおよび変更 (sqllocres) 278
 インストールおよび変更、GUI (sqlloc) 276, 277
 インストール、新規 (langinstall) 238, 240
 代替言語、bcp 92
 マイグレーション 380
 マイグレーション用の指定 389

こ

高可用性
 マイグレーション 416
 高速バージョン、bcp 53
 コピー・イン、データ 53
 データのリカバリ性 52
 構文
 sybmigrate 385
 固定長フィールド 85
 コピー
 新しいエントリ、dscp 106
 サーバ・エントリ、dscp 106
 サーバ・エントリ、dsedit 119, 120, 122, 123
 セッション間、dscp 106
 別のセッション、dscp 107
 コピー・アウト、データ、対話型 bcp 85, 88
 text および image データ 96
 エラー・ファイル 98
 固定長フィールド 85
 デリミタ 87
 他のソフトウェア 85
 エクスポート、データ。「えコピー・アウト、データ、対話型 bcp」参照 85
 コピー・イン、データ

使用手順、高速バージョンの bcp 53
 バッチ・オペレーション 94
 パラレル bcp、要件 58, 63
 分割されたテーブル 55
 分割、ランダムな使用 55
 リカバリ性の向上、ロールバック後 94
 コピー・イン、データ、対話型 bcp 88, 92
 エラー・ファイル 98
 データ型の互換性、失敗 90
 デリミタ 90
 フィールド長 88
 コピー、定義 209
 コマンド・ターミネータ (isql) 10
 統計値オプションとの関係 14

サ

サーバ
 backupsrvr、起動 137
 backupsrvr、指定、サーバ名 133
 startserver ユーティリティ・コマンド 293
 startserver、起動 293
 sybdiag ユーティリティ・コマンド 296
 sybmigrate ユーティリティ・コマンド 303
 速度のモード、bcp 46
 マイグレート 303
 サーバ・エントリ
 dscp、表示 107
 コピー、dsedit 119, 120, 122, 123
 サーバ・エントリ、追加 120
 削除、dseditdsedit 118
 追加、dsedit 117
 内容、表示、dscp 107
 名前の変更、dsedit 118
 変更、dsedit 117, 120
 サーバ接続
 backupsrvr、接続数 133
 サーバ・データ、マイグレーション 370
 サーバのカラム
 順序 84
 名前 85
 サーバの属性 (dscp) 103

サイズ
 text または image データ 96
 データ記憶領域、bcp 72
 パケット・サイズ 96
 削除
 サーバ・エントリ (dscp) 108
 作成
 新規サーバ (srvbuild) 289, 290
 新規サーバ、リソース・ファイルを使用
 (srvbuildres) 291, 292

シ

出力フォーマット、データ。「コピー・アウト、
 データ、対話型 bcp」参照 85
 システム・データ・フォーマット (SDF) 出力、bcp
 86
 実行、Adaptive Server 173, 279
 「dataserver ユーティリティ・コマンド」参照
 173
 「sqlsrvr ユーティリティ・コマンド」参照 279
 指定、サーバ名、backupsrvr 133
 指定、フル・パス名、Backup Server 134
 手動マイグレーション、マイグレートされない
 データ 372
 新規サーバ、作成 289, 290
 「srvbuild ユーティリティ・コマンド」参照
 289
 新規サーバ、作成、リソース・ファイルを使用
 291, 292
 「srvbuildres ユーティリティ・コマンド」参照
 291
 シンボル、フィールド・ターミネータ (bcp) 78
 ストアド・プロシージャ
 コピー、defncopy 209

セ

セキュリティ、注意 125
 セッション
 表示 (dscp) 102
 セッションのクローズ
 dscp 102
 dsedit 114

そ

ソート順

- インストールおよび変更 (sqllocres) 278
- インストールおよび変更、GUI (sqlloc) 276

た

- ターミネータ (bcp) 84
 - 定義 45
 - フィールドとロー 77, 79
 - 変更 69
 - 他のプログラム、使用 78
- 対話型 bcp 70, 79
 - image データ型、デフォルト 74
 - null 値 74
 - null、見えない文字ターミネータ (¥¥0) 78
 - 暗黙の変換、データ型 72, 73
 - 円記号ターミネータ (¥) 78
 - 記憶長 75, 77
 - コピー・アウト、データ 85, 88
 - コピー・イン、データ 88, 92
 - コピー、データ、他のソフトウェア 70
 - ターミネータ、改行 (¥¥n) 78
 - ターミネータ、表形式データの準備 78
 - ターミネータ、フィールドとロー 77, 79
 - タブ・データ・フィールド・ターミネータ (¥¥) 78
 - データへの埋め込み、スペース 75
 - テキスト・データ、コピー、デフォルト 74
 - デフォルト、プロンプト 70, 79
 - パイナリ・データ、デフォルト 74
 - ファイル記憶タイプ 70, 73
 - フィールド・ターミネータ 78, 79
 - フィールド長 75, 77
 - フィールド、プレフィクス長 73, 75
 - 復帰改行データ・フィールド・ターミネータ (¥¥r) 78
 - 文字フォーマット・ファイル、ターミネータ 78
 - ロー・ターミネータ 78, 79

対話型 SQL パーサ 225

- 「isql ユーティリティ・コマンド」参照 225
- タブ・データ・フィールド・ターミネータ (¥¥)、対話型 bcp 78

ち

注意

- ascii_7 文字セットの互換性 210
- defncopy と Report Workbench 209
- null ターミネータとターミネータなし 79
- オペレーティング・システムによる違い、bcp 68
- 隠しテキスト、暗号化された 214
- セキュリティ 125
- 低速バージョンの bcp とデッドロック 56
- 特殊文字、bcp での処理 147
- トリガ、bcp で起動されない、ターゲット・テーブル 50
- パスワード、使用 125
- バッチ・サイズ、設定、bcp 142

つ

追加

- サーバ・エントリ、dscp 104
- サーバ・エントリ、dsedit 117, 120
- ネットワーク・トランスポート・アドレス、dsedit 120, 123

て

定義

- backupserver ユーティリティ・コマンド 133
- bcp ユーティリティ・コマンド 139
- charset ユーティリティ・コマンド 170
- dataserver ユーティリティ・コマンド 173
- defncopy ユーティリティ・コマンド 209
- コピー。「defncopy ユーティリティ・コマンド」参照 209

- dscp ユーティリティ・コマンド 215
- dsedit ユーティリティ・コマンド 216
- isql ユーティリティ・コマンド 225
- optdiag ユーティリティ・コマンド 241, 249
- showserver ユーティリティ・コマンド 269
- sqlloc ユーティリティ・コマンド 276
- sqllocres ユーティリティ・コマンド 278
- sqlsrvr ユーティリティ・コマンド 279
- sqlupgrade ユーティリティ・コマンド 287
- sqlupgraderes ユーティリティ・コマンド 288
- srvbuild ユーティリティ・コマンド 289
- srvbuildres ユーティリティ・コマンド 291
- startserver ユーティリティ・コマンド 293
- sybdiag ユーティリティ・コマンド 296
- sybmigrate ユーティリティ・コマンド 303
- xpsrvr ユーティリティ・コマンド 309
- ディレクトリ・サービス
 - エントリ、コピー 119, 122, 123
 - エントリ、削除 118
 - エントリ、追加 117
 - エントリ、名前の変更 118
 - エントリ、変更 117
 - オープン、dsedit 113
- データ
 - 移動、Adaptive Server コマンド 99
 - インポートとエクスポート、bcp 44
 - 埋め込み、スペース、対話型 bcp 75
 - 転送、別のプログラム、bcp を使用 45
 - 必要なパーミッション、テーブルへのコピー 47, 157
 - 解析。「フィールド・ターミネータ」参照 69
 - 変換エラー 97, 98
 - 変更、Adaptive Server コマンド 100
 - リカバリ性 52
- データ型
 - bcp ファイル記憶タイプ 70, 73
 - bcp フォーマット・ファイル 81
 - bcp、使用 77
 - char 67
 - 暗黙の変換 72, 73
 - 記憶領域 (SYB) 83
 - コピーと互換性 90
 - デフォルト値、bcp 99
 - フィールド長、対話型 bcp 72, 79
 - 文字データ型以外のデータ型から文字データ型への変換、デフォルトのフィールド長、bcp 76
- データ転送、デフォルト・フォーマット、bcp 67
- データへのスペースの埋め込み、bcp 77
- データベース・オブジェクト
 - コピー、bcp を使用 139
- データベース管理システム、その他 85
- データベース・データ、マイグレーション 371
- データベースの再マイグレート 420
- データベースのマージ、マイグレーション時 419
- データベース、コピー、bcp 100
- テーブル
 - bcp のプロンプト 71
 - bcp 文字セット、デフォルト 144
 - 必要なパーミッション、オペレーティング・システム・ファイルへのコピー 47, 157
 - 必要なパーミッション、コピー、データ 47, 157
- テーブルのロー
 - コピー・イン、バッチ 94
 - 長さ 60
- データのインポート。「コピー・イン、データ、対話型 bcp」参照 88
- デバッグ・ユーティリティ 270
- デフォルト
 - bcp データの変換 76
 - bcp プロンプト 70, 79
 - 言語、backupserver 135
 - コピー、defncopy 209
 - テーブルへのコピー、データを使用 98
 - プロンプト、対話型 bcp 70, 79
 - 文字セット、backupserver 135
 - 文字セット、bcp 144
- デリミタ
 - コピー・アウト、データ 87
 - コピー・イン、データ 90
- 転送、データ、プログラム、bcp を使用 45

と

- 統計
 - isql 14
 - 表示 (optdiag) 241, 249
 - ロード、更新された統計値 (optdiag) 241, 249
- 特殊文字
 - bcp、処理 147
 - ユーティリティ・コマンド、特殊文字の使用 125
- " " (特殊文字を囲む) 125
- “ ” (特殊文字を囲む) 125
- トラブルシューティング、マイグレーション 418
- トリガ
 - bcp、削除、使用前 53
 - コピー、defncopy 209
 - 速度低下、bcp 50
 - テーブル、コピー、データ 98
- トレース・フラグと backupserver 138

な

- 名前の変更、サーバ・エントリ、dsedit 118

に

- 入力のエコー (isql) 16

ね

- ネイティブ・ファイル・フォーマット
 - bcp 67
- ネイティブ・フォーマット・ファイル 67
- ネットワーク・トランスポート 115
- ネットワーク・トランスポート・アドレス
 - dsedit、編集 120
 - 追加、dsedit 120, 123
- ネットワーク・パケット・サイズ
 - isql、指定 15

は

- バイナリ・データ
 - bcp 45
 - デフォルト、対話型 bcp 74
- パケット・サイズ、ネットワーク
 - bcp、指定 96
 - isql、指定 15
 - 指定、isql 15
- パスワード
 - bcp の暗号化 146
 - defncopy 209
 - 新規、生成 180
 - 注意、使用について 125
- バックアップ、バルク・コピーとの比較 100
- バッファ、クエリ 12
- パフォーマンス
 - bcp の問題 63, 99
 - isql ネットワーク・パケット・サイズ 15
 - バルク・コピーとパケット・サイズ 96
- パラメータ
 - hide-vcc 155
- パラレル bcp
 - IDENTITY カラム 62, 63
 - 構文 61
 - コピー、特定の分割 58, 62
 - 別の使用方法 60, 63
- バルク・コピー・プロセス
 - 「bcp ユーティリティ・コマンド」参照 139
- 番号
 - 行番号、isql からの削除 11, 16

ひ

- 必要なパーミッション
 - backupserver ユーティリティ・コマンド 138
 - bcp ユーティリティ・コマンド 47, 74
 - charset ユーティリティ・コマンド 170
 - dataserver ユーティリティ・コマンド 180, 284
 - dscp ユーティリティ・コマンド 215
 - dsedit ユーティリティ・コマンド 216
 - dsedit、必要なパーミッション 112
 - langinstall ユーティリティ・コマンド 239

- optdiag ユーティリティ・コマンド 247
 - sqlloc ユーティリティ・コマンド 277
 - sqllocres ユーティリティ・コマンド 278
 - sqlupgrade ユーティリティ・コマンド 287
 - sqlupgraderes ユーティリティ・コマンド 288
 - srvbuild ユーティリティ・コマンド 290
 - srvbuildres ユーティリティ・コマンド 292
 - xpserver ユーティリティ・コマンド 310
 - オペレーティング・システム・ファイル、コピー、
テーブル 47, 157
 - テーブル、コピー、データ 47, 157
 - 必要な領域と bcp の手順 53
 - ビュー、コピー、dfncopy 209
 - 表形式の出力 85, 88
 - 表形式のデータ、コピー 79
 - 表示、interfaces ファイル、GUI 216
 - 「dsedit ユーティリティ・コマンド」参照 216
 - 表示、サーバ 269
 - 「showserver ユーティリティ・コマンド」参照
269
- ふ**
- ファイル
 - エラー (bcp) 97
 - デフォルト・フォーマット、bcp 67
 - ネイティブ・フォーマット 67
 - 「フォーマット・ファイル、bcp」参照 81
 - フィールド・ターミネータ 69
 - bcp 72
 - 対話型 bcp 78, 79
 - フィールド長
 - コピー・イン 88
 - 対話型 bcp 75, 77
 - プレフィクス長、整数値 84
 - フォーマット・ファイル、bcp 81, 85
 - TDS バージョン番号 82
 - カラムの位取り 85
 - カラムの精度 85
 - カラム番号 82
 - サーバ・カラム名 85
 - サーバのカラム順 84
 - 使用されるフォーマット 81
 - 使用される要素 81
 - ターミネータ 84
 - プレフィクス長、整数値 84
 - ホスト・ファイルのカラム順 82
 - ホスト・ファイルのデータ型の記憶フォーマット (SYB) 83
 - ホスト・ファイルのデータ長 84
 - 保存 81
 - レコード数 82
 - 複写データベース
 - マイグレーション後の手順 409
 - マイグレーション後のリストア 411
 - 復帰改行データ・フィールド・ターミネータ (¥¥r)、対話型 bcp 78
 - プライマリ・データベース、マイグレーション後のリストア 410
 - フル・パス名、指定
 - Backup Server 134
 - プレフィクス長 84
 - プレフィクス長、bcp 71
 - プレフィクス・フィールド長、対話型 bcp 73, 75
 - プロンプト
 - bcp。「対話型 bcp」参照 70
 - プロンプト、bcp ユーティリティ・コマンド 71
 - 分割されたテーブル、bcp 54, 64
 - コピー、データ、方法 55
 - コピー、データ、ランダム 55
 - ページ割り付け、増加 57
- へ**
- 変更
 - 言語 (sqllocres) 278
 - 言語、GUI (sqlloc) 276, 277
 - サーバ・エントリ (dscp) 105
 - サーバ・エントリ、dsedit 117, 120
 - ソート順 (sqllocres) 278
 - ソート順、GUI (sqlloc) 276, 277
 - 文字セット (sqllocres) 278
 - 文字セット、GUI (sqlloc) 276, 277
 - 編集
 - interfaces ファイル 112, 114

interfaces ファイル、GUI (dsedit) 216
 ネットワーク・トランスポート・アドレス、
 dsedit 120
 編集、interfaces ファイル、GUI
 「dsedit ユーティリティ・コマンド」参照
 216

ほ

ホスト・ファイル
 isql と読み込み 16
 カラム順 82
 対話型 bcp のネイティブ・フォーマット 75
 データ型の記憶フォーマット (SYB) 83
 データ長 84
 保存
 エラー、コピー・アウト・オペレーション中
 98
 エラー、コピー・イン・オペレーション中
 97
 フォーマット・ファイル (bcp) 81

ま

マイグレーションのエラー・メッセージ 418
 マイグレーション用の **enable unicode
 conversions** 380
 マイグレーション用の **number of sort buffers**
 383
 マイグレーション用の **number of user connections**
 382, 383
 マイグレーション用の **number of worker
 processes** 382
 マイグレーション用のインデックスの再作成
 407
 マイグレーション用のオブジェクト階層 376
 マイグレーション用のコンポーネント・ディレ
 クトリ 375
 マイグレーション用のロック・セッション 386
 マイグレート
 サーバ (sybmigrate) 303
 マイグレート・セッション、**sybmigrate** 377

も

文字セット
 backupserver、デフォルト 135
 bcp、デフォルト 144
 インストールおよび変更 (sqllocres) 278
 インストールおよび変更、GUI (sqlloc) 276,
 277
 プラットフォームのデフォルト 92
 変換、文字データ型以外のデータ 75
 マイグレーション用の指定 389
 ロード、charset 170
 文字データ型以外のデータ型、オペレーティ
 ング・システム・フォーマット 73
 文字フォーマット
 bcp 45
 ターミネータ 78
 デフォルト、bcp 67, 69

ゆ

ユーティリティ
バルク・コピー (bcp) 65
 ユーティリティ・コマンド
 特殊文字、使用 125
 ユーティリティ、他の Adaptive Server ユーティ
 リティ、bcp 99

り

リソース・ファイル・モード、マイグレーション
 398
 リダイレクト・アウト記号 (>)、isql 16
 リダイレクト・イン記号 (<)、isql 16
 リモート・マシン 112
 リモート・マシン、dsedit ユーティリティの使用
 112

る

ルール
 defncopy、コピー 209
 テーブル、コピー、データ 98

れ

例

- bcp ユーティリティ・コマンド 148
- defncopy ユーティリティ・コマンド 212
- dscp ユーティリティ・コマンド 215
- isql ユーティリティ・コマンド 226
- showserver ユーティリティ・コマンド 269
- startserver ユーティリティ・コマンド 293, 294

ろ

ロー・ターミネータ

bcp 69

対話型 bcp 78, 79

ロード、文字セット 170

「charset ユーティリティ・コマンド」参照
170

ロールバック・プロセス 94

ロー・レベルのアクセス・ルールと bcp 93

ロー、bcp、不正なテーブル 98

ロー、テーブル

長さ 60

バルク・コピー、失敗 94

ログが記録されないトランザクション 52

ロケーション、backupsrv エラー・ログ・ファイル 134

わ

検証セッション、sybmigrate 377