

SYBASE®

クイック・リファレンス・ガイド

**Adaptive Server® Enterprise**

15.5

ドキュメント ID : DC32974-01-1550-01

改訂 : 2009 年 11 月

Copyright © 2010 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。

#### マニュアルの注文

マニュアルの注文を承ります。ご希望の方は、サイベース株式会社営業部または代理店までご連絡ください。マニュアルの変更は、弊社の定期的なソフトウェア・リリース時のみ提供されます。

Sybase の商標は、**Sybase trademarks ページ** (<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。

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 Sun Microsystems, Inc. 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。

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.

# クイック・リファレンス・ガイド

トピック名	
<a href="#">データ型</a>	1
<a href="#">データ型と暗号化カラム</a>	3
<a href="#">標準規格と準拠レベル</a>	4
<a href="#">Adaptive Server のグローバル変数</a>	4
<a href="#">予約語</a>	9
<a href="#">関数</a>	12
<a href="#">コマンド</a>	24
<a href="#">対話型 dbsql コマンド</a>	55
<a href="#">システム・プロシージャ</a>	57
<a href="#">カタログ・ストアド・プロシージャ</a>	82
<a href="#">拡張ストアド・プロシージャ</a>	83
<a href="#">dbcc ストアド・プロシージャ</a>	84
<a href="#">システム・テーブル</a>	86
<a href="#">DBCC テーブル</a>	92
<a href="#">ユーティリティ</a>	101

## データ型

データ型の詳細については、『リファレンス・マニュアル：ビルディング・ブロック』を参照してください。

データ型 ( 種類別 )	同義語	範囲	記憶サイズ ( バイト数 )
真数値：整数			
bigint		$2^{63} \sim -2^{63} - 1$ (-9,223,372,036,854,775,808 ~ +9,223,372,036,854,775,807) の間の整数値。	8
int	integer	$2^{31} - 1$ (2,147,483,647) ~ $-2^{31}$ (-2,147,483,648)	4
smallint		$2^{15} - 1$ (32,767) ~ $-2^{15}$ (-32,768)	2
tinyint		0 ~ 255 ( 負の数は使用できない )	1
unsigned bigint		0 ~ 18,446,744,073,709,551,615 の間の整数値	8
unsigned int		0 ~ 4,294,967,295 の間の整数値	4
unsigned smallint		0 ~ 65535 の間の整数値	2

## データ型

データ型 (種類別)	同義語	範囲	記憶サイズ (バイト数)
真数値: 小数			
numeric (p, s)		$10^{38} - 1 \sim -10^{38}$	17 ~ 2
decimal (p, s)	dec	$10^{38} - 1 \sim -10^{38}$	17 ~ 2
概数値			
float (precision)		マシンに依存する	default precision < 16 の場合は 4, default precision >= 16 の場合は 8
double precision		マシンに依存する	8
real		マシンに依存する	4
通貨			
smallmoney		-214,748.3647 ~ 214,748.3648	4
money		922,337,203,685,477.5807 ~ -922,337,203,685,477.5808	8
日/時			
smalldatetime		1900 年 1 月 1 日 ~ 2079 年 6 月 6 日	4
datetime		1753 年 1 月 1 日 ~ 9999 年 12 月 31 日	8
date		0001 年 1 月 1 日 ~ 9999 年 12 月 31 日	4
time		12:00:00AM ~ 11:59:59:999PM	4
bigdatetime		0001 年 1 月 1 日 ~ 9999 年 12 月 31 日および 12:00.000000AM ~ 11:59:59.999999 PM	8
bigtime		12:00:00.000000 AM ~ 11:59:59.999999 PM	8
文字			
char(n)	character	pagesize	n
varchar(n)	character varying, char varying	pagesize	実際のエントリの長さ
unichar	Unicode 文字	pagesize	$n * @@unicharsize$ ( $@@unicharsize$ は 2)
univarchar	Unicode 文字 varying, char varying	pagesize	実際の文字数 * $@@unicharsize$
nchar(n)	national character, national char	pagesize	$n * @@ncharsize$
nvarchar(n)	nchar varying, national char varying, national character varying	pagesize	$@@ncharsize * 文字数$
text		$2^{31} - 1$ (2,147,483,647) バイト以下	初期化前は 0、初期化後は 2K の倍数
unitext		1 - 1,073,741,823	初期化前は 0、初期化後は 2K の倍数

データ型 (種類別)	同義語	範囲	記憶サイズ (バイト数)
バイナリ			
binary(n)		pagesize	<i>n</i>
varbinary(n)		pagesize	実際のエントリの長さ
image		2 <sup>31</sup> -1 (2,147,483,647) バイト以下	初期化前は 0、初期化後は 2K の倍数
ビット			
bit		0 または 1	1 (1 バイトで 8 つまでの bit カラムを保持する)

## データ型と暗号化カラム

次の表は、暗号化カラムでサポートされているデータ型と、ディスク上におけるデータ型の暗号化カラムの長さを示しています。

データ型	暗号化データの実際の長さ (init_vector なし)	暗号化データの実際の長さ (init_vector 使用)
date	4	17
time	4	17
smalldatetime	4	17
bigdatetime	8	
bigtime	8	
datetime	8	17
smallmoney	4	17
money	8	17
bit	8	17
bigint	8	17
unsigned bigint	8	17
unichar(10)	2 (1 unichar 文字)	33
unichar(10)	20 (10 unichar 文字)	33
univarchar(20)	20 (10 unichar 文字)	49

## 標準規格と準拠レベル

以下は、Transact-SQL データ型の ANSI SQL 標準規格と準拠レベルを示しています。標準規格と規格への準拠の詳細については、『リファレンス・マニュアル：ビルディング・ブロック』を参照してください。

Transact-SQL – ANSI SQL データ型 char, varchar, smallint, int, bigint, decimal, numeric, float, real, date, time, double precision

Transact-SQL 拡張機能 – ユーザ定義データ型 binary, varbinary, bit, nchar, datetime, smalldatetime, tinyint, unsigned smallint, unsigned int, unsigned bigint, money, smallmoney, text, unitext, image, nvarchar, unichar, univarchar, sysname, longsysname, timestamp

## Adaptive Server のグローバル変数

以下は、Adaptive Server のグローバル変数と各変数の簡単な説明です。詳細については、『リファレンス・マニュアル：ビルディング・ブロック』を参照してください。

<code>@@active_instances</code>	クラスタ内のアクティブなインスタンスの数を返す。
<code>@@authmech</code>	ユーザの認証に使用するメカニズムを示す。
<code>@@bootcount</code>	インストールが起動された回数を返す。
<code>@@boottime</code>	Adaptive Server が最後に起動された日付と時刻を返す。
<code>@@bulkarraysize</code>	バルク・コピー・インタフェースを使用して転送される前にローカル・サーバ・メモリにバッファリングされるローの数を返す。
<code>@@bulkbatchsize</code>	バルク・インタフェースを使用して <code>select into proxy_table</code> 経由でリモート・サーバに転送されるローの数を返す。
<code>@@char_convert</code>	文字セット変換が有効でない場合は 0 を返す。文字セット変換が有効な場合は 1 を返す。
<code>@@cis_rpc_handling</code>	<code>cis rpc handling</code> がオフの場合は 0 を返す。 <code>cis rpc handling</code> がオンの場合は 1 を返す。
<code>@@cis_version</code>	コンポーネント統合サービスの日付とバージョンを返す。
<code>@@client_csexpansion</code>	サーバの文字セットをクライアントの文字セットに変換するときに使用する拡張係数を返す。
<code>@@client_csid</code>	クライアントの文字セットが初期化されていない場合は -1 を返す。クライアントの文字セットが初期化されている場合は <code>syscharsets</code> から接続用のクライアントの文字セット ID を返す。

---

<code>@@client_csname</code>	クライアントの文字セットが初期化されていない場合は NULL を返す。クライアントの文字セットが初期化されている場合は、接続用の文字セットの名前を返す。
<code>@@clusterboottime</code>	クラスタが最初に起動された日付と時刻を返す (クラスタの起動を最初に開始したインスタンスが停止している場合を含む)。
<code>@@clustercoordid</code>	現在のクラスタ・コーディネータのインスタンス ID を返す。
<code>@@clustermode</code>	文字列 “shared-disk cluster” を返す。
<code>@@clustername</code>	クラスタの名前を返す。
<code>@@cmpstate</code>	高可用性環境における Adaptive Server の現在のモードを返す。
<code>@@connections</code>	ユーザ・ログインが試行された回数を返す。
<code>@@cpu_busy</code>	Adaptive Server が最後に起動してから、CPU が Adaptive Server の作業に費やした時間 (チック単位) を返す。
<code>@@cursor_rows</code>	カーソル結果セットに含まれるローの総数を示す。
<code>@@curloid</code>	カーソルが開いていない、最後に開いたカーソルを構成するローがない、または最後に開いたカーソルが閉じられたか割り付け解除された。
<code>@@datefirst</code>	<code>tinyint</code> で表される各週の指定された最初の曜日を返す。 <code>@@datefirst</code> の現在の値を返す。
<code>@@dbts</code>	現在のデータベースのタイムスタンプを返す。
<code>@@error</code>	システムによって生成された最新のエラー番号を返す。
<code>@@errorlog</code>	Adaptive Server エラー・ログが置かれているディレクトリを、 <code>\$\$SYBASE</code> ディレクトリの相対パスとしてフル・パスで返す。
<code>@@failedoverconn</code>	プライマリ・コンパニオンへの接続がフェールオーバーして、セカンダリ・コンパニオン・サーバが実行中の場合は、0 より大きい値を返す。
<code>@@fetch_status</code>	フェッチ操作が成功した場合は 0 を返し、フェッチ操作が成功しなかった場合は -1 を返す。-2 は今後のために予約済み。
<code>@@guestuserid</code>	guest ユーザの ID を返す。
<code>@@hacmpservername</code>	高可用性設定のコンパニオン・サーバの名前を返す。
<code>@@haconnection</code>	接続でフェールオーバー・プロパティが有効になっている場合は、0 より大きい値を返す。
<code>@@heapmemsize</code>	ヒープ・メモリ・プールのサイズ (バイト単位) を返す。
<code>@@identity</code>	生成された最新の IDENTITY カラム値を返す。
<code>@@idle</code>	Adaptive Server が最後に起動してからアイドル状態になっていた時間 (チック単位) を返す。
<code>@@invaliduserid</code>	無効なユーザ ID に対して -1 の値を返す。

<code>@@instanceid</code>	実行元のインスタンスの ID を返す。
<code>@@instancename</code>	実行元のインスタンスの名前を返す。
<code>@@invaliduserid</code>	無効なユーザ ID に対して -1 の値を返す。
<code>@@io_busy</code>	Adaptive Server が入出力操作に費やした時間 ( チック単位 ) を返す。
<code>@@isolation</code>	現在の Transact-SQL プログラムのセッション固有の独立性レベルの値 (0、1、または 3) を返す。
<code>@@instanceid</code>	Job Scheduler が実行されている、または有効化後に実行されるインスタンスの ID。
<code>@@kernel_addr</code>	カーネル領域を含む最初の共有メモリ領域の開始アドレスを返す。
<code>@@kernel_size</code>	最初の共有メモリ領域の一部であるカーネル領域のサイズを返す。
<code>@@langid</code>	<code>syslanguages.langid</code> に指定されている、現在使用中の言語のサーバワイドの言語 ID を返す。
<code>@@language</code>	<code>syslanguages.name</code> に指定されている、使用中の言語名を返す。
<code>@@lastlogindate</code>	<code>datetime</code> データ型を含む。その値は、現在のセッションが確立される前のログイン・アカウントの <code>lastlogindate</code> カラム内の値。
<code>@@lock_timeout</code>	現在の <code>lock_timeout</code> の設定 ( ミリ秒単位 ) を返す。
<code>@@maxcharlen</code>	Adaptive Server のデフォルト文字セット中の 1 文字の最大長 ( バイト単位 ) を返す。
<code>@@max_connections</code>	現在のコンピュータ環境の Adaptive Server に対して、同時に確立できる接続の最大数を返す。
<code>@@maxgroupid</code>	最大のグループ・ユーザ ID を返す。
<code>@@maxpagesize</code>	サーバの論理ページ・サイズを返す。
<code>@@max_precision</code>	サーバによって設定された <code>decimal</code> および <code>numeric</code> データ型で使用される精度レベルを返す。この値は定数 38。
<code>@@maxspid</code>	<code>spid</code> の最大有効値を返す。
<code>@@maxsuid</code>	最大のサーバ・ユーザ ID を返す。
<code>@@maxuserid</code>	最大のユーザ ID を返す。
<code>@@mempool_addr</code>	グローバル・メモリ・プールのテーブルのアドレスを返す。
<code>@@min_poolsize</code>	名前付きキャッシュ・プールの最小サイズ ( キロバイト ) を返す。
<code>@@mingroupid</code>	最小のグループ・ユーザ ID を返す。
<code>@@minspid</code>	<code>spid</code> の最小値である 1 を返す。
<code>@@minsuid</code>	最小のサーバ・ユーザ ID を返す。
<code>@@minuserid</code>	最小のユーザ ID を返す。

---

<code>@@monitors_active</code>	<code>sp_sysmon</code> によって表示されるメッセージ数を減らす。
<code>@@ncharsize</code>	現在のサーバのデフォルト文字セット中の 1 文字の最大長 (バイト単位) を返す。
<code>@@nestlevel</code>	現在のネスト・レベルを返す。
<code>@@nodeid</code>	現在のインストール環境の 48 ビットのノード識別子を返す。
<code>@@optgoal</code>	クエリの最適化に使用される現在の <code>optimization goal</code> 設定を返す。
<code>@@options</code>	セッションの <code>set</code> オプションの 16 進表現を返す。
<code>@@opttimeoutlimit</code>	クエリの最適化に使用される現在の <code>optimization timeout limit</code> 設定を返す。
<code>@@pack_received</code>	Adaptive Server が読み込んだ入力パケット数を返す。
<code>@@pack_sent</code>	Adaptive Server が書き込んだ出力パケット数を返す。
<code>@@packet_errors</code>	パケットの読み込みおよび書き込み中に Adaptive Server が検出したエラーの数を返す。
<code>@@pagesize</code>	サーバの仮想ページ・サイズを返す。
<code>@@parallel_degree</code>	現在の最大並列度の設定を返す。
<code>@@probesuid</code>	プローブ・ユーザ ID に対して 2 の値を返す。
<code>@@procid</code>	現在実行中のプロシージャのストアド・プロシージャ ID を返す。
<code>@@quorum_physname</code>	クォーラム・デバイスの物理パスを返す。
<code>@@recovery_state</code>	戻り値に基づいて、Adaptive Server がリカバリ中かどうかを示す。
<code>@@repartition_degree</code>	現在の動的な <code>repartitioning degree</code> 設定を返す。
<code>@@resource_granularity</code>	クエリの最適化に使用される最大リソース使用量ヒントの設定を返す。
<code>@@rowcount</code>	最後のクエリによる影響を受けたローの数を返す。
<code>@@scan_parallel_degree</code>	ノンクラスタード・インデックス・スキャンの現在の最大並列度の設定を返す。
<code>@@servername</code>	Adaptive Server の名前を返す。
<code>@@setrowcount</code>	<code>set rowcount</code> の現在の値を返す。
<code>@@shmem_flags</code>	共有メモリ領域のプロパティを返す。
<code>@@spid</code>	現在のプロセスのサーバ・プロセス ID を返す。
<code>@@sqlstatus</code>	<code>fetch</code> 文の実行結果であるステータス情報 ( 例外の警告 ) を返す。
<code>@@ssl_ciphersuite</code>	現在の接続で SSL が使用されない場合に NULL を返し、それ以外の場合に現在の接続での SSL ハンドシェイクに選択された暗号スイートの名前を返す。
<code>@@stringsize</code>	<code>toString()</code> メソッドから返される文字データのサイズを返す。
<code>@@system_busy</code>	Adaptive Server でシステム・タスクが実行されていた時間 ( チック単位 ) 。

<code>@@sys_tempdbid</code>	実行中のインスタンスの有効なローカル・システム・テンポラリ・データベースのデータベース ID を返す。
<code>@@system_view</code>	セッション固有のシステム・ビュー設定として“instance”または“cluster”を返す。
<code>@@tempdbid</code>	セッションに割り当てられたテンポラリ・データベースの有効なテンポラリ・データベース ID (dbid) を返す。
<code>@@textcolid</code>	<code>@@textptr</code> が参照するカラムのカラム ID を返す。
<code>@@textdataptnid</code>	<code>@@textptr</code> によって参照されるカラムが含まれるテキスト・パーティションのパーティション ID を返す。
<code>@@textdbid</code>	<code>@@textptr</code> が参照するカラムを持つオブジェクトを格納しているデータベースの ID を返す。
<code>@@textobjid</code>	<code>@@textptr</code> が参照するカラムを持つオブジェクトのオブジェクト ID を返す。
<code>@@textptnid</code>	<code>@@textptr</code> によって参照されるカラムが含まれるデータ・パーティションのパーティション ID を返す。
<code>@@textptr</code>	プロセスが最後に挿入または更新した、 <code>text</code> 、 <code>unitext</code> 、または <code>image</code> のテキスト・ポインタを返す ( <code>textptr</code> 関数とは異なる)。
<code>@@textptr_parameters</code>	<code>textptr_parameters</code> の現在のステータスがオフの場合は 0 を返す。 <code>textptr_parameters</code> の現在のステータスがオンの場合は 1 を返す。
<code>@@textsize</code>	<code>select</code> が返す <code>text</code> 、 <code>unitext</code> 、または <code>image</code> データのバイト数に対する制限を返す。
<code>@@textts</code>	<code>@@textptr</code> が参照するカラムのテキスト・タイムスタンプを返す。
<code>@@thresh_hysteresis</code>	スレッシュホールドをアクティブ化するために必要な空き領域の減少量を返す。
<code>@@timeticks</code>	チックごとのマイクロ秒数を返す。
<code>@@total_errors</code>	読み込みおよび書き込み中に Adaptive Server が検出したエラーの数を返す。
<code>@@total_read</code>	Adaptive Server によるディスク読み込み回数を返す。
<code>@@total_write</code>	Adaptive Server によるディスク書き込み回数を返す。
<code>@@tranchained</code>	Transact-SQL プログラムの現在のトランザクション・モードが非連鎖モードの場合は 0 を返し、連鎖モードの場合は 1 を返す。
<code>@@trancount</code>	現在のユーザ・セッションにおけるトランザクションのネスト・レベルを返す。
<code>@@transactional_rpc</code>	リモート・サーバへの RPC がトランザクション指向の場合は 0 を返し、非トランザクション指向の場合は 1 を返す。
<code>@@transtate</code>	現在のユーザ・セッションで文を実行した後のトランザクションの現在の状態を返す。
<code>@@unicharsize</code>	<code>unichar</code> の文字のサイズである 2 を返す。
<code>@@user_busy</code>	Adaptive Server でユーザ・タスクが実行されていた時間 (チック単位)。

---

<code>@@version</code>	Adaptive Server の現在のリリースの日付、バージョン文字列などを返す。
<code>@@version_number</code>	Adaptive Server の現在のリリースのバージョン全体を整数で返す。
<code>@@version_as_integer</code>	Adaptive Server の現在のリリースの最新のアップグレード・バージョン番号を整数で返す。

## 予約語

この項では、さまざまな予約語を示します。詳細については、『リファレンス・マニュアル：ビルディング・ブロック』を参照してください。

### Transact-SQL 予約語

以下は、Adaptive Server によって予約されているキーワード (SQL コマンドの構文の一部) を示します。

A	add, all, alter, and, any, arith_overflow, as, asc, at, authorization, avg
B	begin, between, break, browse, bulk, by
C	cascade, case, char_convert, check, checkpoint, close, clustered, coalesce, commit, compute, confirm, connect, constraint, continue, controlrow, convert, count, count_big, create, current, cursor
D	database, dbcc, deallocate, declare, decrypt, default, delete, desc, deterministic, disk distinct, drop, dummy, dump
E	else, encrypt, end, endtran, errlvl, errordata, errorexit, escape, except, exclusive, exec, execute, exists, exit, exp_row_size, external
F	fetch, fillfactor, for, foreign, from
G	goto, grant, group
H	having, holdlock
I	identity, identity_gap, identity_start, if, in, index, inout, insensitive, insert, install, intersect, into, is, isolation
J	jar, join
K	key, kill
L	level, like, lineno, load, lock
M	materialized, max, max_rows_per_page, min, mirror, mirrorexit, modify
N	national, new, noholdlock, nonclustered, nonscrollable, non_sensitive, not, null, nullif, numeric_truncation

O	of, off, offsets, on, once, online, only, open, option, or, order, out, output, over
P	partition, perm, permanent, plan, prepare, primary, print, privileges, proc, procedure, processexit, proxy_table, public
Q	quiesce
R	raiserror, read, readpast, readtext, reconfigure, references, remove, reorg, replace, replication, reservepagegap, return, returns, revoke, role, rollback, rowcount, rows, rule
S	save, schema, scroll, scrollable, select, semi_sensitive, set, setuser, shared, shutdown, some, statistics, stringsize, stripe, sum, syb_identity, syb_restree, syb_terminate
T	table, temp, temporary, textsize, to, tracefile, tran, transaction, trigger, truncate, tsequal
U	union, unique, unpartition, update, use, user, user_option, using
V	values, varying, view
W	waitfor, when, where, while, with, work, writetext
X	xmlextract, xmlparse, xmltest, xmlvalidate

## ANSI SQL 予約語

以下は、Adaptive Server によって予約されていない ANSI SQL キーワードを示します。

A	absolute, action, allocate, are, assertion
B	bit, bit_length, both
C	cascaded, case, cast, catalog, char, char_length, character, character_length, coalesce, collate, collation, column, connection, constraints, corresponding, cross, current_date, current_time, current_timestamp, current_user
D	date, day, dec, decimal, deferrable, deferred, describe, descriptor, diagnostics, disconnect, domain
E	end-exec, exception, extract
F	false, first, float, found, full
G	get, global, go
H	hour
I	immediate, indicator, initially, inner, input, insensitive, int, integer, interval
J	join
L	language, last, leading, left, local, lower
M	match, minute, module, month

---

N	names, natural, nchar, next, no, nullif, numeric
O	octet_length, outer, output, overlaps
P	pad, partial, position, preserve, prior
R	real, relative, restrict, right
S	scroll, second, section, semi_sensitive, session_user, size, smallint, space, sql, sqlcode, sqlerror, sqlstate, substring, system_user
T	then, time, timestamp, timezone_hour, timezone_minute, trailing, translate, translation, trim, true
U	unknown, upper, usage
V	value, varchar
W	when, whenever, write, year
Z	zone

## ANSI SQL の予約語となる可能性のあるワード

ISO/IEC 9075:1989 標準を使用する場合は、以下のワードの使用は避けてください。これらのワードは、ANSI SQL の予約語になる可能性があります。

A	after, alias, async
B	before, boolean, breadth
C	call, completion, cycle
D	data, depth, dictionary
E	each, elseif, equals
G	general
I	ignore
L	leave, less, limit, loop
M	modify
N	new, none
O	object, oid, old, operation, operators, others
P	parameters, pendant, preorder, private, protected
R	recursive, ref, referencing, resignal, return, returns, routine, row
S	savepoint, search, sensitive, sequence, signal, similar, sqlexception, structure
T	test, there, type

U	under
V	variable, virtual, visible
W	wait, without

## 関数

以下は、組み込み関数の概要と構文です。詳細については、『リファレンス・マニュアル：ビルディング・ブロック』を参照してください。

abs	式の絶対値を返します。 <code>abs(<i>numeric_expression</i>)</code>
acos	余弦が指定されている角度 (ラジアン) を返します。 <code>acos(<i>cosine</i>)</code>
ascii	式の先頭文字の ASCII コードを返します。 <code>ascii(<i>char_expr</i>   <i>uchar_expr</i>)</code>
asehostname	Adaptive Server が動作する物理ホストまたは仮想ホストを返します。 <code>asehostname</code>
asin	正弦が指定されている角度 (ラジアン) を返します。 <code>asin(<i>sine</i>)</code>
atan	正接が指定されている角度 (ラジアン) を返します。 <code>atan(<i>tangent</i>)</code>
atn2	正弦と余弦が指定されている角度 (ラジアン) を返します。 <code>atn2(<i>sine</i>, <i>cosine</i>)</code>
avg	すべて (または重複を除くすべて) の値の平均値を返します。 <code>avg([all   distinct] <i>expression</i>)</code>
audit_event_name	監査イベントの説明を返します。 <code>audit_event_name(<i>event_id</i>)</code>
authmech	指定したログイン済みサーバのプロセス ID で使用されている認証メカニズムを判断します。 <code>authmech ([<i>spid</i>])</code>
biginttohex	指定した整数値に相当する、プラットフォームの影響を受けない 8 バイトの 16 進文字列を返します。 <code>biginttohex (<i>integer_expression</i>)</code>
bintostr	一連の 16 進数を同等の英数字または <code>varbinary</code> データの文字列に変換します。 <code>select bintostr(<i>sequence of hexadecimal digits</i>)</code>

---

case	<p>条件付き SQL 式をサポートします。</p> <pre> case   when <i>search_condition</i> then <i>expression</i>   [<i>when search_condition then expression</i>]...   [<i>else expression</i>] end </pre> <ul style="list-style-type: none"> <li>case と値の構文 <pre> case <i>expression</i>   when <i>expression</i> then <i>expression</i>   [<i>when expression then expression</i>]...   [<i>else expression</i>] end </pre> </li> </ul>
cast	<p>指定された値を別のデータ型に変換して返します。</p> <pre>cast (<i>expression as datatype</i> [(<i>length</i>   <i>precision</i> [, <i>scale</i>]])</pre>
ceiling	<p>指定された値以上の最小の整数を返します。</p> <pre>ceiling(<i>value</i>)</pre>
char	<p>整数に相当する文字を返します。</p> <pre>char(<i>integer_expr</i>)</pre>
char_length	<p>式の文字数を返します。</p> <pre>char_length(<i>char_expr</i>   <i>uchar_expr</i>)</pre>
charindex	<p>式の開始位置を表す整数を返します。</p> <pre>charindex(<i>expression1</i>, <i>expression2</i>)</pre>
coalesce	<p>条件付き SQL 式をサポートします。これは case 式の代替コマンドです。</p> <pre>coalesce(<i>expression</i>, <i>expression</i> [, <i>expression</i>]...)</pre>
col_length	<p>カラムの定義済みの長さを返します。</p> <pre>col_length(<i>object_name</i>, <i>column_name</i>)</pre>
col_name	<p>指定されたテーブル ID とカラム ID に対応するカラム名を返します。</p> <pre>col_name(<i>object_id</i>, <i>column_id</i> [, <i>database_id</i>])</pre>
compare	<p>代替照合規則をもとに、2つの文字列を直接比較できるようにします。</p> <pre>compare ({<i>char_expression1</i> <i>uchar_expression1</i>},          {<i>char_expression2</i> <i>uchar_expression2</i>},          [{<i>collation_name</i>   <i>collation_ID</i>}]</pre>
convert	<p>指定した値を別のデータ型または異なる datetime 表示フォーマットに変換して返します。</p> <pre>convert (<i>datatype</i> [(<i>length</i>)   (<i>precision</i> [, <i>scale</i>])]         [null   not null], <i>expression</i> [, <i>style</i>])</pre>
cos	<p>指定した角度の余弦を返します。</p> <pre>cos(<i>angle</i>)</pre>
cot	<p>指定した角度の余接を返します。</p> <pre>cot(<i>angle</i>)</pre>

---

count	null でない値の数 (重複する値を除く) または選択したローの数を整数で返します。 <code>count([all   distinct] expression)</code>
count_big	-null でない値の数 (重複する値を除く) または選択されたローの数を bigint の値で返します。 <code>count_big([all   distinct] expression)</code>
count_bigdatetime	現在の時刻を表す bigtime 値をマイクロ秒の精度で返します。 <code>current_bigdatetime()</code>
count_bigtime	現在の時刻を表す bigtime 値をマイクロ秒の精度で返します。 <code>current_bigtime()</code>
current_date	現在の日付を返します。 <code>current_date()</code>
current_time	現在の時刻を返します。 <code>current_time()</code>
curunreservedpgs	指定したディスク区分の空きページ数を返します。 <code>curunreservedpgs (dbid, lstart, unreservedpgs)</code>
data_pages	指定したテーブル、インデックス、またはパーティションで使用されているページ数を返します。 <code>data_pages(dbid, object_id [, indid [, ptnid]])</code>
datachange	<code>update statistics</code> が最後に実行された後にデータ分配で行われた変更の量を測定します。 <code>datachange(object_name, partition_name, column_name)</code>
datalength	指定したカラムや文字列の実際の長さをバイト単位で返します。 <code>datalength(expression)</code>
dateadd	特定の年数、四半期数、時間数、その他の日付要素を、指定した日付に加算または減算して計算した日付を返します。 <code>dateadd(date_part, integer, {date   time   bigtime   datetime,   bigdatetime})</code>
datediff	2つの日付の差を返します。 <code>datediff(datepart, {date, date   time, time   bigtime, bigtime   datetime, datetime   bigdatetime, bigdatetime})</code>
datename	指定した date または time の指定した datepart を文字列として返します。 <code>datename(datepart {date   time   bigtime   datetime   bigdatetime})</code>
datepart	指定した date の指定した datepart (最初の引数) を整数として返します。 <code>datepart(date_part {date   time   datetime   bigtime   bigdatetime})</code>
day	指定した日付の datepart の日を表す整数を返します。 <code>day(date_expression)</code>

---

db_attr	指定したデータベースの <code>durability</code> , <code>dml_logging</code> , <code>template</code> の設定を返します。 <code>db_attr('database_name'   database_ID   NULL, 'attribute')</code>
db_id	指定したデータベースの ID 番号を返します。 <code>db_id(database_name)</code>
db_instanceid	クラスタ環境のみ – 指定したローカル・テンポラリ・データベースの所有インスタンスの ID を返します。指定したデータベースがグローバル・テンポラリ・データベースまたは非テンポラリ・データベースである場合は NULL を返します。
db_name	ID 番号が指定されているデータベースの名前を返します。 <code>db_name([database_id])</code>
db_recovery_status	クラスタ環境のみ – 指定したローカル・テンポラリ・データベースの所有インスタンスの ID を返します。指定したデータベースがグローバル・テンポラリ・データベースまたは非テンポラリ・データベースである場合は NULL を返します。 <code>db_recovery_status([database_ID   database_name])</code>
degrees	ラジアン数が指定されている角度の大きさを度数単位で返します。 <code>degrees(numeric)</code>
derived_stat	指定したオブジェクトとインデックスに対して得られた統計を返します。 <code>derived_stat("object_name"   object_id, index_name   index_id, ["partition_name"   partition_id], "statistic")</code>
difference	2 つの <code>soundex</code> 値間の差を表す整数を返します。 <code>difference(expr1,expr2)</code>
exp	定数を指定した累乗にした値を返します。 <code>exp(approx_numeric)</code>
floor	指定した値以下の最大の整数を返します。 <code>floor(numeric)</code>
get_appcontext	指定したコンテキストの属性値を返します。 <code>get_appcontext ("context_name", "attribute_name")</code>
getdate	システムの現在の日付と時刻を返します。 <code>getdate()</code>
getutcdate	日時を UTC (万国標準時) で返します。 <code>getutcdate()</code>
has_role	指定した役割がユーザに付与されているかどうかを示す情報を返します。 <code>has_role ("role_name", option)</code>
hash	固定長のハッシュ値式を生成します。 <code>hash(expression , [algorithm])</code>

---

hashbytes	固定長のハッシュ値式を生成します。 <code>hashbytes(<i>algorithm</i>, <i>expression</i>[, <i>expression...</i>], <i>using options</i>)</code>
hextobigint	16 進文字列に相当する <b>bigint</b> 値を返します。 <code>hextobigint(<i>hexadecimal_string</i>)</code>
hextoint	指定した 16 進文字列に相当する、プラットフォームの影響を受けない整数値を返します。 <code>hextoint(<i>hexadecimal_string</i>)</code>
host_id	クライアント・コンピュータのオペレーティング・システムで使用されている、現在の Adaptive Server クライアントのプロセス ID を返します。 <code>host_id()</code>
host_name	クライアント・プロセスの現在のホスト・コンピュータ名を返します。 <code>host_name()</code>
identity_burn_max	指定したテーブルの ID 消去最大値を追跡します。 <code>identity_burn_max(<i>table_name</i>)</code>
index_col	指定したテーブルまたはビューのインデックス付きカラムの名前を返します。 <code>index_col(<i>object_name</i>, <i>index_id</i>, <i>key_#</i>, <i>user_id</i>)</code>
index_colorder	カラム順を返します。 <code>index_colorder(<i>object_name</i>, <i>index_id</i>, <i>key_#</i>, <i>user_id</i>)</code>
index_name	インデックス ID、データベース ID、およびインデックスが定義されているオブジェクトを指定すると、インデックスの名前を返します。 <code>index_name(<i>dbid</i>, <i>objid</i>, <i>indid</i>)</code>
instance_id	クラスタ・エディションのみ – 指定したインスタンスの ID を返します。 <i>name</i> の値を指定しない場合は、発行元のインスタンスの ID を返します。 <code>instance_id(<i>[name]</i>)</code>
instance_name	クラスタ・エディションのみ – 指定した ID を持つ Adaptive Server の名前を返します。 <i>id</i> の値を指定しない場合は、発行元の Adaptive Server の名前を返します。 <code>instance_name(<i>[id]</i>)</code>
inttohex	指定した整数値に相当する、プラットフォームの影響を受けない 16 進文字列を返します。 <code>inttohex(<i>integer_expression</i>)</code>
isdate	入力式が有効な <b>datetime</b> 値かどうかを確認します。 <code>isdate(<i>character_expression</i>)</code>
isnumeric	<b>numeric</b> データ型かどうかを確認します。 <code>isnumeric (<i>character_expression</i>)</code>
is_quiesced	データベースが <b>quiesce database</b> モードであるかどうかを示します。 <code>is_quiesced(<i>dbid</i>)</code>

---

is_sec_service_on	セキュリティ・サービスが有効な場合は 1 を返し、無効な場合は 0 を返します。 <code>is_sec_service_on(<i>security_service_nm</i>)</code>
isnull	<i>expression1</i> の値が NULL のとき、 <i>expression2</i> に指定した値と置き換えます。 <code>isnull(<i>expression1</i>, <i>expression2</i>)</code>
isnumeric	式が有効な numeric データ型かどうかを確認します。 <code>isnumeric (<i>character_expression</i>)</code>
lc_id	クラスタ環境のみ – 指定した名前の論理クラスタの ID を返します。名前を指定しない場合は、現在の論理クラスタの ID を返します。 <code>lc_id(<i>logical_cluster_name</i>)</code>
lc_name	クラスタ環境のみ – 指定した ID の論理クラスタの名前を返します。ID を指定しない場合は、現在の論理クラスタの名前を返します。 <code>lc_name(<i>[logical_cluster_ID]</i>)</code>
lct_admin	ラストチャンス・スレッシュホールド (LCT) を管理し、LCT の現在の値を返し、LCT に達したトランザクション・ログに含まれるトランザクションをアポートします。 <code>lct_admin({'lastchance'   'logfull'   'reserved_for_rollbacks'},           <i>database_id</i>   'reserve', {<i>log_pages</i>   0 }             'abort', <i>process-id</i> [, <i>database-id</i>])</code>
left	文字列の左端から指定された数の文字を返します。 <code>left(<i>character_expression</i>, <i>integer_expression</i>)</code>
len	後続空白を除き、指定した文字列式の文字数 (バイト数ではない) を返します。 <code>len(<i>string_expression</i>)</code>
license_enabled	機能のライセンスが有効になっている場合は 1 を返し、有効になっていない場合は 0 を返します。また、無効なライセンス名が指定された場合は NULL を返します。 <code>license_enabled("ase_server"   "ase_ha"   "ase_dtm"   "ase_java"                     "ase_asm")</code>
list_appcontext	現在のセッション内にある全コンテキストの属性をすべてリストします。 <code>list_appcontext(<i>["context_name"]</i>)</code>
lockscheme	指定したオブジェクトのロック・スキームを文字列として返します。 <code>lockscheme(<i>object_name</i>)</code> <code>lockscheme(<i>object_id</i> [, <i>db_id</i>])</code>
log	指定した数値の自然対数を返します。 <code>log(<i>approx_numeric</i>)</code>
log10	指定した数の常用対数 (底が 10 の対数) を返します。 <code>log10(<i>approx_numeric</i>)</code>

---

lower	指定した式に相当する小文字の文字値を返します。 <code>lower(char_expr   uchar_expr)</code>
ltrim	指定した式から先行空白を削除して返します。 <code>ltrim(char_expr   uchar_expr)</code>
max	式内の最高値を返します。 <code>max(expression)</code>
min	カラム内の最小値を返します。 <code>min(expression)</code>
month	指定した日付の <code>datepart</code> の月を表す整数を返します。 <code>month(date_expression)</code>
mut_excl_roles	2つの役割の間の相互排他性についての情報を返します。 <code>mut_excl_roles(role1, role2 [membership   activation])</code>
newid	指定した引数に基づいて、人間が判読できるグローバルにユニークな識別子 (GUID) を2通りの異なるフォーマットで生成します。 <code>newid(optionflag)</code>
next_identity	次の <code>insert</code> に使用可能な次の <code>identity</code> 値を取得します。 <code>next_identity(table_name)</code>
nullif	条件付き SQL 式をサポートします。 <code>nullif(expression1, expression2)</code>
object_attr	セッション、テーブル、データベース全体の設定に応じて、テーブルの現在のロギング・モードをレポートします。 <code>object_attr(table_name, string)</code>
object_id	指定したオブジェクトのオブジェクト ID を返します。 <code>object_id(object_name)</code>
object_name	指定したオブジェクト ID を持つオブジェクトの名前を返します。名前は最長で 255 バイトです。 <code>object_name(object_id[, database_id])</code>
object_owner_id	オブジェクトの所有者 ID を返します。 <code>object_owner_id(object_id[, database_id])</code>
pagesize	指定したオブジェクトのページ・サイズをバイト数で返します。 <code>pagesize(object_name[, ])</code> <code>pagesize(object_id[, db_id[, index_id]])</code>
partition_id	指定したデータまたはインデックス・パーティション名のパーティション ID を返します。 <code>partition_id(table_name, partition_name[, index_name])</code>

---

partition_name	指定したデータまたはインデックス・パーティション ID のパーティション名を返します。 <code>partition_name(indid, ptnid [, dbid])</code>
partition_object_id	指定したパーティション ID とデータベース ID のオブジェクト ID を表示します。 <code>partition_object_id(partition_id [, database_id])</code>
pssinfo	プロセス・ステータス構造からの情報を返します。 <code>pssinfo(spids   0, 'pss_field')</code>
patindex	指定したパターンが最初に検出された先頭位置を返します。 <code>patindex("%pattern%", char_expr   uchar_expr [, using {bytes   characters   chars}])</code>
pi	定数値 (円周率) 3.1415926535897936 を返します。 <code>pi()</code>
power	指定した数字を指定の累乗で計算した結果の値を返します。 <code>power(value, power)</code>
proc_role	指定した役割がユーザに付与されているかどうかを示す情報を返します。 <code>proc_role("role_name")</code>
radians	指定した度数の角度の大きさをラジアンで返します。 <code>radians(numeric)</code>
rand	0 と 1 の間の乱数値を返します。 <code>rand([integer])</code>
rand2	指定したシード値を使用して生成され、 <code>select</code> リストでの使用時に返された各ローに対して計算された、0 と 1 の間の乱数値を返します。 <code>rand2([integer])</code>
replicate	指定した式を指定した回数だけ繰り返した結果の文字列を返します。 <code>replicate(char_expr   uchar_expr, integer_expr)</code>
reserve_identity	あるプロセスが <code>identity</code> 値のブロックを予約し、そのプロセスがこの値を利用できるようにします。 <code>reserve_identity (table_name, number_of_values)</code>
reserved_pages	データベース、オブジェクト、またはインデックス用に予約されているページ数をレポートします。 <code>reserved_pages(dbid, object_id[, indid[, ptnid]])</code>
reverse	指定した文字列の文字順を逆にして返します。 <code>reverse(expression   uchar_expr)</code>
right	式の右端から、指定した文字数の部分を返します。 <code>right(expression, integer_expr)</code>

---

<code>rm_appcontext</code>	特定のアプリケーション・コンテキストまたはすべてのアプリケーション・コンテキストを削除します。 <code>rm_appcontext("context_name", "attribute_name")</code>
<code>role_contain</code>	<code>role2</code> に <code>role1</code> が含まれている場合は 1 を返します。 <code>role_contain("role1", "role2")</code>
<code>role_id</code>	指定した名前を持つ役割のシステム役割 ID を返します。 <code>role_id("role_name")</code>
<code>role_name</code>	指定したシステム役割 ID の名前を返します。 <code>role_name(role_id)</code>
<code>round</code>	指定した数値を、指定した小数桁数に丸めた値を返します。 <code>round(number, decimal_places)</code>
<code>row_count</code>	指定したテーブル内のローの予測数を返します。 <code>row_count(dbid, object_id [,ptnid])</code>
<code>rtrim</code>	指定した式から後続ブランクを削除した結果を返します。 <code>rtrim(char_expr   uchar_expr)</code>
<code>sdc_intempdbconfig</code>	クラスタ環境のみ – システムが現在テンポラリ・データベース設定モードである場合は 1 を返し、それ以外の場合は 0 を返します。 <code>sdc_intempdbconfig()</code>
<code>set_appcontext</code>	ユーザ・セッション用のアプリケーション・コンテキスト名、属性名、属性値を設定します。それらは、指定したアプリケーションの属性によって定義されます。 <code>set_appcontext("context_name", "attribute_name", "attribute_value")</code>
<code>show_role</code>	ログインの現在有効なシステム定義役割を表示します。 <code>show_role()</code>
<code>show_sec_services</code>	セッションに有効なセキュリティ・サービスをリストします。 <code>show_sec_services()</code>
<code>sign</code>	指定した値の符号 (正数と 0 の場合は 1、負数の場合は -1) を返します。 <code>sign(numeric)</code>
<code>sin</code>	指定した角度 (ラジアン) の正弦を返します。 <code>sin(approx_numeric)</code>
<code>sortkey</code>	照合動作に基づいて結果を並べ替えるために使用できる値を生成します。 <code>sortkey(char_expression   uchar_expression [, {collation_name   collation_ID}])</code>
<code>soundex</code>	式の発音を表す 4 文字コードを返します。 <code>soundex(char_expr   uchar_expr)</code>
<code>space</code>	指定した数のシングルバイト・スペースからなる文字列を返します。 <code>space(integer_expr)</code>

---

spid_instance_id	<p>クラスタ・エディションのみ – 指定したプロセス ID (spid) を実行しているインスタンスの ID を返します。</p> <p><code>spid_instance_id(spид_value)</code></p>
square	<p>指定した値の平方を <code>float</code> で表したものを返します。</p> <p><code>square(numeric_expression)</code></p>
sqrt	<p>指定した数値の平方根を返します。</p> <p><code>sqrt(approx_numeric)</code></p>
stddev	<p><code>stddev_samp</code> のエイリアスです。</p>
stdev	<p><code>stddev_samp</code> のエイリアスです。</p>
stddevp	<p><code>stddev_pop</code> のエイリアスです。</p>
stddev_pop	<p>1 つの数値式で構成される母標準偏差を <code>double</code> 型として計算します。</p> <p><code>stddev_pop ([ all   distinct ] expression )</code></p>
stddev_samp	<p>1 つの数値式で構成される標本標準偏差を <code>double</code> 型として計算します。</p> <p><code>stddev_samp ([ all   distinct ] expression )</code></p>
str	<p>指定した数値に相当する文字を返します。</p> <p><code>str(approx_numeric[, length [, decimal]])</code></p>
str_replace	<p>最初の文字列式内に出現する 2 番目の文字列式のすべてのインスタンスを、3 番目の文字列式で置き換えます。</p> <p><code>str_replace("string_expression1", "string_expression2", "string_expression3")</code></p>
strtobin	<p>一連の英数字をそれに相当する 16 進数に変換します。</p> <p><code>select strtobin("string of valid alphanumeric characters")</code></p>
stuff	<p>1 つの文字列から指定した数の文字を削除して、代わりに別の文字列を入れて作成した文字列を返します。</p> <p><code>stuff(char_expr1   uchar_expr1, start, length, char_expr2   uchar_expr2)</code></p>
substring	<p>指定した数の文字を文字列から取り出して作った文字列を返します。</p> <p><code>substring(expression, start, length)</code></p>
sum	<p>値の合計を返します。</p> <p><code>sum([all   distinct] expression)</code></p>
suser_id	<p><code>syslogins</code> テーブルからサーバ・ユーザの ID 番号を返します。</p> <p><code>suser_id([server_user_name])</code></p>
suser_name	<p>現在のサーバ・ユーザの名前、または指定したサーバ ID を持つユーザの名前を返します。</p> <p><code>suser_name([server_user_id])</code></p>
syb_quit	<p>接続を終了します。</p> <p><code>syb_quit()</code></p>

---

syb_sendmsg	<p>UNIXのみ – UDP (ユーザ・データグラム・プロトコル) ポートにメッセージを送信します。</p> <p><code>syb_sendmsg ip_address, port_number, message</code></p>
sys_tempdbid	<p>クラスタ環境のみ – 指定したインスタンスの有効なローカル・システム・テンポラリ・データベースの ID を返します。<code>instance_id</code> を指定しない場合は、現在のインスタンスの有効なローカル・システム・テンポラリ・データベースの ID を返します。</p> <p><code>sys_tempdbid(instance_id)</code></p>
tan	<p>指定した角度 (ラジアン) の正接を返します。</p> <p><code>tan(angle)</code></p>
tempdb_id	<p>指定のセッションが割り当てられているテンポラリ・データベースをレポートします。</p> <p><code>tempdb_id()</code></p>
textptr	<p><code>text</code>、<code>image</code>、または <code>unitext</code> カラムの先頭ページを指すポインタを返します。</p> <p><code>textptr(column_name)</code></p>
textvalid	<p>指定した <code>text</code> カラムまたは <code>unitext</code> カラムを指すポインタが有効な場合は 1 を返し、無効な場合は 0 を返します。</p> <p><code>textvalid("table_name.column_name", textpointer)</code></p>
to_unichar	<p>整数式の値を持つ <code>unichar</code> 式を返します。</p> <p><code>to_unichar(integer_expr)</code></p>
tran_dumpable_status	<p><code>dump transaction</code> を使用できるかどうかを示す <code>true</code> または <code>false</code> の値を返します。</p> <p><code>tran_dumpable_status("database_name")</code></p>
tsequal	<p>ローがブラウズ用に選択された後で変更された場合に、ローを更新できないようにするため <code>timestamp</code> 値を比較します。</p> <p><code>tsequal(browsed_row_timestamp, stored_row_timestamp)</code></p>
uhighsurr	<p><code>start</code> の位置の Unicode 値が、サロゲート・ペアの上位サロゲート (ペアの中で最初に指定されている) の場合は 1 を返します。それ以外の場合は 0 を返します。</p> <p><code>uhighsurr(uchar_expr, start)</code></p>
ulowsurr	<p><code>start</code> の位置の Unicode 値が、サロゲート・ペアの下位サロゲート (ペアの中で 2 番目に指定されている) の場合は 1 を返します。それ以外の場合は 0 を返します。</p> <p><code>ulowsurr(uchar_expr, start)</code></p>
upper	<p>指定した文字列に相当する大文字を返します。</p> <p><code>upper(char_expr)</code></p>
uscalar	<p>式の最初の Unicode 文字に対する Unicode スカラ値を返します。</p> <p><code>uscalar(uchar_expr)</code></p>

---

used_pages	<p>テーブル、インデックス、または特定のパーティションで使用されるページ数をレポートします。</p> <p><code>used_pages(<i>dbid</i>, <i>object_id</i> [, <i>indid</i> [, <i>ptnid</i>]])</code></p>
user	<p>現在のユーザの名前を返します。</p> <p><code>user</code></p>
user_id	<p>指定したユーザの ID、またはデータベース内の現在のユーザの ID を返します。</p> <p><code>user_id(<i>user_name</i>)</code></p>
user_name	<p>指定したユーザまたは現在のユーザの、データベース内の名前を返します。</p> <p><code>user_name(<i>user_id</i>)</code></p>
valid_name	<p>指定した文字列が有効な識別子でない場合は 0 を返し、有効な識別子の場合は 0 以外の値を返します。</p> <p><code>valid_name(<i>character_expression</i> [, <i>maximum_length</i>])</code></p>
valid_user	<p>指定した ID が、Adaptive Server 上の少なくとも 1 つのデータベースで有効なユーザかエイリアスのときは 1 を返し、そうでないときは 0 を返します。</p> <p><code>valid_user(<i>server_user_id</i>)</code></p>
var	<p><code>var_samp</code> のエイリアスです。</p>
var_pop	<p>1 つの数値式で構成される母統計分散を <b>double</b> 型として計算します。</p> <p><code>var_pop ( [ <i>all</i>   <i>distinct</i> ] <i>expression</i> )</code></p>
var_samp	<p>1 つの数値式で構成される標本の統計分散を <b>double</b> 型として計算し、数値セットの分散を返します。</p> <p><code>var_samp ( [ <i>all</i>   <i>distinct</i> ] <i>expression</i> )</code></p>
variance	<p><code>var_samp</code> のエイリアスです。</p>
varp	<p>1 つの数値式で構成される母統計分散を <b>double</b> 型として計算します。varp は <code>var_pop</code> のエイリアスです。</p>
workload_metric	<p>クラスタ環境のみ - 指定したインスタンスの現在の負荷測定基準のクエリを実行するか、指定したインスタンスの測定基準を更新します。</p> <p><code>workload_metric( <i>instance_id</i>   <i>instance_name</i> [, <i>new_value</i> ] )</code></p>
xa_bqual	<p>ASCII XA トランザクション ID の bqual コンポーネントのバイナリ・バージョンを返します。</p> <p><code>xa_bqual(<i>xactname</i>, 0)</code></p>
xa_gtrid	<p>ASCII XA トランザクション ID の gtrid コンポーネントのバイナリ・バージョンを返します。</p> <p><code>xa_gtrid(<i>xactname</i>, <i>int</i>)</code></p>

**xmltable** XML ドキュメントからデータを抽出し、SQL テーブルとして返します。

```

xmltable_expression ::= xmltable
    ( row_pattern passing xml_argument
      columns column_definitions
      options_parameter )
row_pattern ::= character_string_literal
xml_argument ::=
    xml_expression | column_reference | variable_reference
column_definitions ::=
    column_definition [ { , column_definition } ]
    column_definition ::=
        ordinality_column | regular_column

ordinality_column ::= column_name datatype for ordinality
regular_column ::=
    column_name datatype [ default literal ] [ null | not null ]
    [ path column_pattern ]
column_pattern ::= character_string_literal
options_parameter ::= [,] option option_string
options_string ::= basic_string_expression

```

- 抽出テーブルの構文 – SQL の from 句から SQL テーブルを返します。

```

from_clause ::= from table_reference [ , table_reference ] ...
table_reference ::= table_view_name | ANSI_join | derived_table
table_view_name ::= See the select command in Reference Manual
    Volume 2, "Commands."
ANSI_join ::= See the select command in Reference Manual
    Volume 2, "Commands."
derived_table ::=
    (subquery) as table_name [ (column_name [ , column_name ] ...) ]
    xmltable_expression as table_name

```

**year** 指定した日付の **datepart** の年を表す整数を返します。

```
year(date_expression)
```

## コマンド

以下は、Adaptive Server のコマンドの概要と構文です。詳細については、『リファレンス・マニュアル：コマンド』を参照してください。

**alter database** データベースに加え、アーカイブ・データベースの変更済みページ・セクションに割り当てられた空き領域を増やします。

```

alter database database_name
[on {default | database_device} [= size]
  [, database_device [= size]]...]
[log on {default | database_device} [= size]
  [, database_device [= size]]...]
set { [durability = { no_recovery | at_shutdown | full}]
      [, dml_logging = {full | minimal}]
      [, template = { database_name | NULL}}]
[with override]
[for load]
[for proxy_update]

```

alter encryption key

暗号化キーの現在のパスワードを変更します。

```
alter encryption key [[database.][owner.].]keyname
{ [ as | not default ]
  [ [ with passwd
      'password' | system_encr_passwd | login_passwd ]
    modify encryption
      [ with passwd
          'password' | system_encr_passwd | login_passwd ]
  ] with passwd 'password'
  add encryption [ with passwd 'password' ]
  for user user_name
  [ for login_association | for recovery ]
  | drop encryption for
    { user user_name | recovery }
  | [ with passwd 'password' ]
    recover encryption with passwd 'password'
  | modify owner user_name
}
```

alter role

役割間の相互排他的な関係を定義します。役割のパスワードの追加、削除、変更を行います。特定の役割に許可されるパスワード有効期間、最短のパスワード長、ログイン失敗の最大回数を指定します。役割をロックおよびロック解除するためにも使用します。

```
alter role role1 {add | drop} exclusive
{membership | activation} role2

alter role role_name [add passwd "password" |
drop passwd] [lock | unlock]

alter role {role_name | "all overrides"}
set {passwd expiration | min passwd length |
max failed_logins} option_value
```

alter table

- 新しいカラムをテーブルに追加します。既存のカラムを削除または変更します。制約を追加、変更、または削除します。既存のテーブルのプロパティを変更します。テーブルのトリガを有効化または無効化します。
- 計算カラムの追加、削除、変更をサポートし、実体化されたプロパティ、null 入力可能性、または既存の計算カラム定義の変更を可能にします。
- テーブルの分割や再分割を指定のパーティション方式で行ったり、パーティションが既に存在するテーブルにパーティションを追加したりします。テーブル・パーティション変更の各構文を次に示します。「パーティションに使用する alter table 構文」を参照してください。

```
alter table [[database.][owner.].]table_name
{add column_name datatype}
[default {constant_expression | user | null}]
[identity | null | not null]
[off row | in row]
[[constraint constraint_name]
{{unique | primary key}
  [clustered | nonclustered]
  [asc | desc]
  [with {fillfactor = pct,
        max_rows_per_page = num_rows,
        reservepagegap = num_pages}
    transfer table [on | off]]]
[on segment_name]
| references [[database.].owner.].ref_table
```

```

        [(ref_column)]
        [match full]
    | check (search_condition)
    [encrypt [with [database.owner.] keyname]
     [decrypt_default {constant_expression | null}]]
    [, next_column]...
| add [constraint constraint_name]
     {unique | primary key}
     [clustered | nonclustered]
     (column_name [asc | desc][, column_name [asc | desc]...])
     [with {fillfactor = pct,
           max_rows_per_page = num_rows,
           reservepagegap = num_pages}]
     [on segment_name]
    | foreign key (column_name [{,column_name}...])
     references [(database.owner.)ref_table
                [(ref_column [{, ref_column}...])]
                [match full]]
    | check (search_condition)
    | set dml_logging = {full | minimal | default}
    | drop {column_name [, column_name]...
          | constraint constraint_name}
    | modify column_name
          [datatype [null | not null]]
          [[encrypt [with keyname] [decrypt_default [value]]
           | decrypt
          ]
          [, next_column]...
| replace column_name
     default {constant_expression | user | null}
     | decrypt_default {constant_expression | null}
     | drop decrypt_default|
lock {allpages | datarows | datapages} }
| with exp_row_size=num_bytes
| partition number_of_partitions
| unpartition
| partition_clause
| add_partition_clause

```

パーティションに使用する alter table 構文：

```

partition_clause ::=
    partition by range (column_name[, column_name]...)
        ([partition_name] values <= ({constant | MAX}
        [, {constant | MAX}] ...) [on segment_name]
        [, [partition_name] values <= ({constant | MAX}
        [, {constant | MAX}] ...) [on segment_name]]...)

| partition by hash (column_name[, column_name]...)
    { (partition_name [on segment_name]
      [, partition_name [on segment_name]]...)
      | number_of_partitions
        [on (segment_name [, segment_name]...)]}

| partition by list (column_name)
    ([partition_name] values (constant[, constant] ...)
     [on segment_name]
     [, [partition_name] values (constant[, constant] ...)
     [on segment_name]] ...)

| partition by roundrobin
    { (partition_name [on segment_name]
      [, partition_name [on segment_name]]...)
      | number_of_partitions
        [on (segment_name [, segment_name]...)]}

```

```

add_partition_clause ::=
    add partition
        { ([partition_name] values <= ({constant | MAX}
            [, {constant | MAX}] ...)
            [on segment_name]
            [, [partition_name] values <= ({constant | MAX}
            [, {constant | MAX}] ...)
            [on segment_name]) ...)}
        | ([partition_name] values (constant[, constant] ...)
            [on segment_name]
            [, [partition_name] values (constant[, constant] ...)
            [on segment_name]) ...)}

```

計算カラムに使用する alter table 構文

```

alter table
    add column_name {compute | as}
        computed_column_expression...
        [materialized | not materialized]
    drop column_name
    modify column_name {null | not null |
        {materialized | not materialized} [null | not null] |
        {compute | as} computed_column_expression
        [materialized | not materialized]
        [null | not null]}

```

パーティションの削除に使用する alter table 構文

```

alter table table_name drop partition
    partition_name [, partition_name]...

```

パーティションに使用する alter table 構文 :

```

partition_clause ::=
    partition by range (column_name[, column_name]...)
        ([partition_name] values <= ({constant | MAX}
            [, {constant | MAX}] ...) [on segment_name]
            [, [partition_name] values <= ({constant | MAX}
            [, {constant | MAX}] ...) [on segment_name])...)}
        | partition by hash (column_name[, column_name]...)
            { ([partition_name] [on segment_name]
                [, partition_name [on segment_name])...}
                | number_of_partitions
                [on (segment_name [, segment_name]...)]}
        | partition by list (column_name)
            ([partition_name] values (constant[, constant] ...)
                [on segment_name]
                [, [partition_name] values (constant[, constant] ...)
                [on segment_name]) ...)}
        | partition by roundrobin
            { ([partition_name] [on segment_name]
                [, partition_name [on segment_name])...}
                | number_of_partitions
                [on (segment_name [, segment_name]...)]}

add_partition_clause ::=
    add partition
        { ([partition_name] values <= ({constant | MAX}
            [, {constant | MAX}] ...)
            [on segment_name]
            [, [partition_name] values <= ({constant | MAX}
            [, {constant | MAX}] ...)
            [on segment_name]) ...)}

```

	<pre> [ , [partition_name] values &lt;= ({constant   MAX}   [, {constant   MAX}] ...)   [on segment_name] ...)    ([partition_name] values (constant[, constant] ...)   [on segment_name]   [, [partition_name] values (constant[, constant] ...)   [on segment_name] ...]) </pre>
begin...end	<p>フロー制御言語がグループ全体に作用するように、一連の SQL 文を囲みます。</p> <pre> begin   statement block end </pre>
begin transaction	<p>ユーザ定義のトランザクションの開始点をマーク付けします。</p> <pre> begin tran[saction] [transaction_name] </pre>
break	<p>while ループを終了させます。break は、多くの場合 if 文によってアクティブ化されます。</p> <pre> while logical_expression   statement break   statement continue </pre>
checkpoint	<p>すべてのダーティページをデータベース・デバイスに書き込みます。</p> <pre> checkpoint [all   [dbname[, dbname, dbname, .....]] </pre>
close	<p>カーソルを非アクティブにします。</p> <pre> close cursor_name </pre>
commit	<p>ユーザ定義のトランザクションの終了点をマーク付けします。</p> <pre> commit [tran   transaction   work] [transaction_name] </pre>
compute 句	<p>クエリ結果の追加ローとして表示される合計値を生成します。</p> <pre> start_of_select_statement compute row_aggregate (column_name)   [, row_aggregate (column_name)]... [by column_name [, column_name]...] </pre>
connect to...disconnect	<p>コンポーネント統合サービスのみ – 指定されたサーバに接続します。また、接続されているサーバとの接続を解除します。別のサーバへのパススルーを作成する場合：</p> <pre> connect to server_name disconnect   [from ASE]   [all]   [connection_name] </pre>

- CIS を使用せずに Adaptive Server に対して新しい JDBC 接続を開きます。引数は、どのような順序で指定してもかまいません。引数を指定しない場合、接続パラメータの入力を求めるプロンプトが表示されます。

```
connect
  [to ASE engine_name]
  [database database_name]
  [as connection_name]
  [user user_id]
  [identified by password]]]
```

- Adaptive Server に対して新しい JDBC 接続を開きます。CIS は使用しません。

```
connect using connect_string
```

continue

while ループを再開します。continue は、多くの場合 if 文によってアクティブ化されます。

```
while boolean_expression
  statement
break
  statement
continue
```

create archive database

アーカイブ・データベースを作成します。

```
create archive database db_name
  [on db_device [= size]
  [, db_device [= size] ] ...]
  with scratch_database = db_name
```

create database

新しいデータベースを作成します。ノンクラスタード環境の構文：

```
create [inmemory] [temporary] database database_name
  [use database_name as template]
  [on {default | database_device} [= size]
  [, database_device [= size]]...]
  [log on database_device [= size]
  [, database_device [= size]]...]
  [with {dbid = number, default_location = "pathname", override}
  | [.,]durability = { no_recovery
  | at_shutdown
  | full} ]
  }...
  [for {load | proxy_update}]
```

クラスタ環境の構文：

```
create [ [ global | system ] temporary ] database database_name
  [ for instance instance_name ]
  [on {default | database_device} [= size]
  [, database_device [= size]]...]
  [log on database_device [= size]
  [, database_device [= size]]...]
  [with {override | default_location = "pathname"}]
  [for {load | proxy_update}]
```

create default

挿入時に、明示的に与えられた値がない場合、1つのカラムに挿入する値を指定します。

```
create default [owner.]default_name
  as constant_expression
```

create encryption key

キーのユーザ指定のパスワードを作成します。

```
create encryption key [ db.[owner .] ] keyname [as default]
  [ for algorithm_name ]
  [ with { [ keylength num_bits ]
  [ passwd 'password_phrase' ]
  [ init_vector {NULL | random } ]
  [ pad {NULL | random } ] } ]
```

create existing table

コンポーネント統合サービスのみ – プロキシ・テーブルを作成し、次にリモート・テーブルからメタデータを検索して格納し、そのデータをプロキシ・テーブルに配置します。こうすると、リモート・ロケーションのテーブル、ビュー、またはプロシージャにプロキシ・テーブルをマップできるようになります。

```
create existing table table_name (column_list)
  [ on segment_name ]
  [[external {table | procedure | file | connection_type}] at pathname
  [column delimiter "string"]]
```

create function

指定値を返す保存された Transact-SQL ルーチンであるユーザ定義関数を作成します。

```
create function [ owner_name. ] function_name
  ( [ { @parameter_name [as] parameter_datatype [= default ] }
  [ ,...n ] ] )
  returns return_datatype
  [ with recompile ]
  as
  [begin]
  function_body
  return scalar_expression
  [end]
```

create function (SQLJ)

Java の静的メソッドに SQL ラップを追加することにより、ユーザ定義の関数を作成します。メソッドによって定義された値を返すことができます。

```
create function [owner.]sql_function_name
  ([ sql_parameter_name sql_datatype
  [(length) (precision[, scale])]
  [[, sql_parameter_name sql_datatype
  [(length) (precision[, scale])]
  ...]])
  returns sql_datatype
  [(length) (precision[, scale])]
  [modifies sql data]
  [returns null on null input |
  called on null input]
  [deterministic | not deterministic]
  [exportable]
  language java
  parameter style java
  external name 'java_method_name
  [(java_datatype[, java_datatype
  ...])]'
```

---

create index

テーブル内の 1 つ以上のカラム ( 計算カラムとそれ以外のカラムの両方 ) にインデックスを作成します。

```
create [unique] [clustered | nonclustered] index index_name
on [[database.] owner.] table_name
(column_expression [asc | desc]
 [, column_expression [asc | desc]]...)
[with {fillfactor = pct,
      max_rows_per_page = num_rows,
      reservepagegap = num_pages,
      consumers = x, ignore_dup_key, sorted_data,
      [ignore_dup_row | allow_dup_row],
      statistics using num_steps values}]
[on segment_name]
[index_partition_clause]
```

- インデックス・パーティションを作成します。

```
index_partition_clause::=
[[local index partition_name [on segment_name]
 [, partition_name [on segment_name]...]]
```

- 関数ベース・インデックスを作成します。

```
create [unique | nonclustered] index index_name
on [[database.] owner.] table_name
(column_expression [asc | desc]
 [, column_expression [asc | desc]]...
```

create plan

抽象プランを作成します。

```
create plan query plan
[into group_name]
[and set @new_id]
```

create procedure

ユーザが提供する 1 つまたは複数のパラメータを付けることができるストアード・プロシージャまたは拡張ストアード・プロシージャを作成します。

```
create procedure [owner.]procedure_name[;number]
[[(@parameter_name datatype [(length) | (precision [, scale])]
 [= default][output]
 [, @parameter_name datatype [(length) | (precision [, scale])]
 [= default][output]...]]]
[with recompile]
as {SQL_statements | external name dll_name}
```

create procedure (SQLJ)

SQL ラップを Java の静的メソッドに追加して、SQLJ ストアド・プロシージャを作成します。

```
create procedure [owner.]sql_procedure_name
(((in | out | inout) sql_parameter_name
 sql_datatype [(length) |
 (precision[, scale])]
 [=default]
 ...))
[, [(in | out | inout) sql_parameter_name
 sql_datatype [(length) |
 (precision[, scale])]
 [=default]
 ...)]
[modifies sql data]
[dynamic result sets integer]
[deterministic | not deterministic]
language java
```

	<pre>parameter style java external name 'java_method_name   [(<i>java_datatype</i> [, <i>java_datatype</i> ...])]'</pre>
create proxy_table	<p>コンポーネント統合サービスのみ – カラム・リストを指定しないでプロキシ・テーブルを作成します。</p> <pre>create proxy_table <i>table_name</i> [external [table   directory   file]] at <i>pathname</i> [column delimiter "&lt;string&gt;"]</pre>
create role	<p>ユーザ定義の役割を作成し、作成時に、特定の役割に許可されるパスワード有効期間、最短のパスワード長、ログイン失敗の最大回数を指定します。</p> <pre>create role <i>role_name</i> [with passwd "<i>password</i>" [, {passwd expiration   min passwd length   max failed_logins} <i>option_value</i>]]</pre>
create rule	<p>特定のカラムまたはユーザ定義データ型の任意のカラムに対し、受け入れ可能な値の領域を指定し、アクセス・ルールを作成します。</p> <pre>create [[and   or] access]] rule [owner.]<i>rule_name</i> as <i>condition_expression</i></pre>
create schema	<p>データベース・ユーザ用の新しいテーブル、ビュー、パーミッションを作成します。</p> <pre>create schema authorization <i>authorization_name</i> create_object_statement [create_object_statement ...] [permission_statement ...]</pre>
create service	<p>提供された SQL 文を指定された名前とパラメータを持つストアド・プロシージャでラップします。</p> <pre>create service <i>service-name</i> [secure <i>security_options</i> ] [, userpath <i>path</i>] [, alias <i>alias-name</i>] type { xml   raw   soap } [[(@<i>parameter_name</i> datatype [(length )   (precision [, scale ])] [= default][output] [, @<i>parameter_name</i> datatype [(length )   (precision [, scale ])] [= default][output]...])] as <i>SQL_statements</i> <i>security_options</i> ::= ( <i>security_option_item</i> [<i>security_option_item</i>])</pre>
create table	<p>新しいテーブルを作成し、必要に応じて整合性制約を作成します。テーブルの作成時に、計算カラム、暗号化カラムと暗号化カラムの復号化デフォルト値、テーブルのパーティション・プロパティを定義します。</p> <pre>create table [[database.]owner.]<i>table_name</i> (<i>column_name</i> <i>datatype</i> [default {<i>constant_expression</i>   user   null}] [<i>identity</i>   null   not null]] [off row   [in row [(<i>size_in_bytes</i>)]]] [[constraint <i>constraint_name</i>] {{unique   primary key} [clustered   nonclustered] [asc   desc] [with {fillfactor = <i>pct</i>, max_rows_per_page = <i>num_rows</i>,} reservepagegap = <i>num_pages</i>] dml_logging = {full   minimal}]</pre>

```

        transfer table [on | off]
    [on segment_name]
    | references [[database.]owner.]ref_table
        [(ref_column)]
        [match full]
        | check (search_condition)]}]
[[encrypt [with key_name]
  [decrypt default_constant_expression | null]]
[[constraint [[database.]owner.]key_name
  {unique | primary key}
  [clustered | nonclustered]
  (column_name [asc | desc]
    [{, column_name [asc | desc]}...])
  [with {fillfactor = pct
    max_rows_per_page = num_rows,
    reservepagegap = num_pages}]
  [on segment_name]
  | foreign key (column_name [{, column_name}...])
  references [[database.]owner.]ref_table
    [(ref_column [{, ref_column}...])]
    [match full]
    | check (search_condition ...)]
  [{, {next_column | next_constraint}...)]
[lock {datarows | datapages | allpages}]
[with { max_rows_per_page = num_rows,
  exp_row_size = num_bytes,
  reservepagegap = num_pages,
  identity_gap = value}]
[on segment_name]
[partition_clause]
[ [ external table ] at pathname ]
[for load]

```

#### パーティションの構文

```

partition_clause ::=
    partition by range (column_name[, column_name]...)
      ((partition_name) values <= ({constant | MAX}
        [, {constant | MAX}] ...)[on segment_name]
        [, (partition_name) values <= ({constant | MAX}
          [, {constant | MAX}] ...)[on segment_name]...))

    | partition by hash (column_name[, column_name]...)
      { (partition_name [on segment_name]
        [, partition_name [on segment_name]...]
        | number_of_partitions
          [on (segment_name [, segment_name]...)]}

    | partition by list (column_name)
      ((partition_name) values (constant[, constant] ...)
        [on segment_name]
        [, (partition_name) values (constant[, constant] ...)
          [on segment_name] ...))

    | partition by roundrobin
      { (partition_name [on segment_name]
        [, partition_name [on segment_name]...]
        | number_of_partitions
          [on (segment_name [, segment_name]...)]}

```

計算カラムの構文

```
create table [database.[owner].] table_name
    (column_name {compute | as}
        computed_column_expression [materialized | not materialized])
```

仮想ハッシュ・テーブルを作成するための構文

```
create table [database.[owner].]table_name
...
    | {unique | primary key}
    using clustered
    (column_name [asc | desc] [{, column_name [asc | desc]}...] =
    (hash_factor [{, hash_factor}...] )
    with max_num_hash_values key
```

create trigger

整合性の制約の実現に頻繁に使用されるストアド・プロシージャの一種である、トリガを作成します。

```
create trigger [owner.]trigger_name
    on [owner.]table_name
    {for | instead of} {insert , update , delete}
    as SQL_statements
```

if update 句を使用する場合 :

```
create trigger [owner.]trigger_name
    on [owner.]table_name
    {for {insert , update} | instead of {insert, update, delete}}
    as
        [if update (column_name)
            [{and | or} update (column_name)]...]
        SQL_statements
        [if update (column_name)
            [{and | or} update (column_name)]...
        SQL_statements]...
```

create view

ビューを作成します。

```
create view [owner .]view_name
    [(column_name[, column_name] ...)]
    as
    select [distinct] select_statement
    [with check option]
```

dbcc

データベースの一貫性チェックは、データベースの論理的および物理的な一貫性のチェックを行い、統計、計画、修復機能を提供します。

```
dbcc addtempdb( dbid |database_name )
dbcc checkalloc [(database_name[, fix | nofix])]
dbcc checkcatalog [(database_name[, fix])
dbcc checkdb [(database_name[, skip_ncindex])]
dbcc checkindex ({table_name | table_id}, index_id
    [, bottom_up[, partition_name | partition_id]])
dbcc checkstorage [(database_name)]
dbcc checktable (table_name | table_id
    [, skip_ncindex | fix_spacebits | "check spacebits" |
    bottom_up | NULL[, partition_name | partition_id])
dbcc checkverify (dbname[, tblname[, ignore_exclusions]])
dbcc complete_xact (xid, [{"commit", "1pc"} | "rollback"])
```

---

```

dbcc forget_xact (xid)
dbcc dbrepair (database_name, dropdb)
dbcc engine ({offline, [enginenum] | "online"})
dbcc fix_text ({table_name | table_id})
dbcc indexalloc (table_name | table_id, index_id
[, optimized | fast | NULL [, fix | nofix | NULL
[, partition_name | partition_id]])
dbcc monitor (increment, <group name>)
dbcc monitor (decrement, <group name>)
dbcc monitor (reset, <group name>)
dbcc pravailabletempdbs
dbcc rebuild_text (table_name | table_id | "all" [, column[, text_page
[, data_partition_name | data_partition_id]])
dbcc reindex ({table_name | table_id})
dbcc serverlimits
dbcc stackused
dbcc tablealloc (table_name | table_id [, full | optimized | fast | NULL
[, fix | nofix | NULL [, data_partition_name | data_partition_id]])
dbcc textalloc (table_name | table_id [, full | optimized | fast | NULL
[, fix | nofix | NULL [, data_partition_name | data_partition_id]])
dbcc {traceon | traceoff} (flag [, flag ...])
dbcc tune ({ascinserts, {0 | 1} , table_name |
cleanup, {0 | 1} |
cpuaffinity, start_cpu {, on | off} |
des_greedyalloc, dbid, object_name,
" {on | off}" | deviochar vdevno, "batch_size" |
doneinproc {0 | 1})
dbcc upgrade_object [ ( dbid | dbname
[, [database.owner].compiled_object_name' |
'check' | 'default' | 'procedure' | 'rule' |
'trigger' | 'view'
[, 'force' ] ] )

```

クラスタ専用の dbcc 構文：

```

dbcc nodetraceon(trace_flag_number)
dbcc nodetraceoff(trace_flag_number)
dbcc set_scope_in_cluster("cluster"|"instance"|"scope")
dbcc quorum

```

deallocate cursor

カーソルをアクセスできない状態にし、そのカーソルにコミットされているメモリ・リソースをすべて解放します。

```
deallocate [cursor] cursor_name
```

declare

バッチまたはプロシージャのローカル変数の名前とデータ型を宣言します。

変数宣言

```
declare @variable_name datatype
[, @variable_name datatype]...
```

変数割り当て

```
select @variable = {expression | select_statement}
[, @variable = {expression | select_statement} ...]
[from table_list]
[where search_conditions]
[group by group_by_list]
[having search_conditions]
[order by order_by_list]
[compute function_list [by by_list]]
```

declare cursor

select 文をカーソル名に関連付けることで、カーソルを定義します。

```
declare cursor_name
[semi_sensitive | insensitive] [scroll | no scroll]
cursor for select_statement
[for {read only | update [of column_name_list]]]
```

delete

テーブルからローを削除します。

```
delete
[top unsigned_integer]
[from] [[database.]owner.]{view_name|table_name}
[where search_conditions]
[plan "abstract plan"]

delete [[database.]owner.]{table_name | view_name}
[from [[database.]owner.]{view_name [readpast]]
table_name
[(index {index_name | table_name}
[prefetch size][lru|mru])]]
[readpast]
[, [[database.]owner.]{view_name [readpast]]
table_name
[(index {index_name | table_name}
[prefetch size][lru|mru])]]
[readpast]} ...]
[where search_conditions]
[plan "abstract plan"]

delete [from] [[database.]owner.]{table_name|view_name}
where current of cursor_name
```

delete statistics

sysstatistics システム・テーブルから統計情報を削除します。

```
delete [shared] statistics table_name
[ partition data_partition_name ]
[(column_name[, column_name] ...)]
```

disk init

物理デバイスまたはファイルを Adaptive Server で使用できるようにします。

```
disk init
name = "device_name",
physname = { 'physical_name' | 'cache_name' }
skip_alloc={true | false},
[vdevno = virtual_device_number ,]
size = number_of_blocks
[, type = 'inmemory' ]
[, vstart = virtual_address
cntritype = controller_number
[, dsync = {true | false}]
[, directio = {true | false}]
[, instance = "instance_name"]
```

---

disk mirror	<p>プライマリ・デバイスで障害が発生した場合、ただちに処理を引き継ぐソフトウェア・ミラーを作成します。</p> <pre> disk mirror   name = "device_name",   mirror = "physicalname"   [, writes = {serial   noserial}]   [, clear = {TRUE   FALSE}] </pre>
disk refit	<p><code>sysdevices</code> に格納されている情報に基づいて、<code>master</code> データベースの <code>sysusages</code> と <code>sysdatabases</code> のシステム・テーブルを再構築します。</p> <pre> disk refit </pre>
disk reinit	<p><code>master</code> データベースの <code>sysdevices</code> システム・テーブルを再構築します。</p> <pre> disk reinit   name = "device_name",   physname = "physicalname",   [, vdevno = virtual_device_number .,]   size = number_of_blocks   [, vstart = virtual_address     , cntrtype = controller_number]   [, dsync = {true   false}]   [, directio = {true   false}]   [, instance = "instance_name"] </pre>
disk remirror	<p>ディスク・ミラーリングが、ミラーリングされたデバイスの障害により停止したか、または <code>disk unmirror</code> コマンドによって一時的に無効になった後に、ディスク・ミラーリングを再起動します。</p> <pre> disk remirror   name = "device_name" </pre>
disk resize	<p>Adaptive Server が使用するデバイスのサイズを動的に拡大します。</p> <pre> disk resize   name = "device_name",   size = additional_space </pre>
disk unmirror	<p><code>disk mirror</code> コマンドによって開始されたディスク・ミラーリングを中断して、ハードウェアのメンテナンス、またはハードウェア・デバイスの変更を可能にします。</p> <pre> disk unmirror   name = "device_name"   [, side = {"primary"   secondary}]   [, mode = {retain   remove}] </pre>
drop database	<p>Adaptive Server から 1 つまたは複数のデータベースを削除します。</p> <pre> drop database database_name [, database_name] ... </pre>
drop default	<p>ユーザ定義のデフォルトを削除します。</p> <pre> drop default [owner.]default_name   [, [owner.]default_name] ... </pre>
drop encryption key	<p>テーブル所有者が <code>alter table</code> に <code>decrypt</code> オプションを指定して、カラムの暗号化または暗号化キーを削除できるようにします。</p> <pre> drop encryption key [database.[owner].]keyname </pre>

drop function	現在のデータベースから 1 つまたは複数のユーザ定義関数を削除します。 <pre>drop function{ [ owner_name .] function_name } [ , ...n ]</pre>
drop function (SQLJ)	SQLJ 関数を削除します。 <pre>drop func[tion] [owner.]function_name [, [owner.]function_name ] ...</pre>
drop index	現在のデータベース内のテーブルからインデックスを削除します。 <pre>drop index table_name.index_name [, table_name.index_name] ...</pre>
drop procedure	プロシージャを削除します。 <pre>drop proc[edure] [owner.]procedure_name [, [owner.]procedure_name] ...</pre>
drop role	ユーザ定義の役割を削除します。 <pre>drop role role_name [with override]</pre>
drop rule	ユーザ定義のルールを削除します。 <pre>drop rule [owner.]rule_name[, [owner.]rule_name] ...</pre>
drop service	現在のデータベースからユーザ定義 Web サービスを削除します。 <pre>drop service service-name</pre>
drop table	データベースからテーブルの定義とそのデータ、インデックス、パーティション・プロパティ、トリガ、暗号化プロパティ、パーミッションすべてを削除します。 <pre>drop table [[database.]owner.]table_name [, [[database.]owner.]table_name] ...</pre>
drop trigger	トリガを削除します。 <pre>drop trigger [owner.]trigger_name [, [owner.]trigger_name] ...</pre>
drop view	1 つまたは複数のビューを現在のデータベースから削除します。 <pre>drop view [owner.]view_name [, [owner.]view_name] ...</pre>
dump database	<b>load database</b> で読み込むことのできる形式で、トランザクション・ログを含むデータベース全体のバックアップ・コピーを作成します。 <pre>dump database database_name to [compress::[compression_level::]]stripe_device [at backup_server_name] [density = density_value, blocksize = number_bytes, capacity = number_kilobytes, dumpvolume = volume_name, file = file_name] with verify[= header   full] [stripe on [compress::[compression_level::]]stripe_device [at backup_server_name] [density = density_value, blocksize = number_bytes, capacity = number_kilobytes, dumpvolume = volume_name, file = file_name]]</pre>

```

[[stripe on [compress::[compression_level::]]stripe_device
  [at backup_server_name]
  [density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  dumpvolume = volume_name,
  file = file_name]]...]
[with {
  density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  compression = compress_level
  dumpvolume = volume_name,
  file = file_name,
  [dismount | nodismount],
  [nounload | unload],
  passwd = password,
  retaindays = number_days,
  [noinit | init],
  notify = {client | operator_console}
}]

```

Tivoli Storage Manager からバックアップ・サービスが提供されている場合に、データベースをコピーします。

```

dump database database_name
to "syb_tsm::object_name"
  [blocksize = number_bytes]
[stripe on "[syb_tsm::]object_name"
  [blocksize = number_bytes]]...]
[with {
  blocksize = number_bytes,
  compression = compress_level,
  passwd = password,
  [noinit | init],
  notify = {client | operator_console},
  verify[= header | full]
}]

```

dump transaction

トランザクション・ログのコピーを作成し、無効な部分を削除します。定期的なログのダンプを作成する場合：

```

dump tran[saction] database_name
to [compress::[compression_level::]]stripe_device
  [at backup_server_name]
  [density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  dumpvolume = volume_name,
  file = file_name]
[stripe on [compress::[compression_level::]]stripe_device
  [at backup_server_name]
  [density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  dumpvolume = volume_name,
  file = file_name]]
[[stripe on [compress::[compression_level::]]stripe_device
  [at backup_server_name]
  [density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  dumpvolume = volume_name,
  file = file_name]]...]

```

```
[with {
  density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  compression = compress_level,
  dumpvolume = volume_name,
  file = file_name,
  [dismount | nodismount],
  [nounload | unload],
  retaindays = number_days,
  [noinit | init],
  notify = {client | operator_console},
  standby_access}}
```

バックアップ・コピーを作成しないでログをトランケートします。

```
dump tran[saction] database_name
with truncate_only
```

容量一杯となったログをトランケートします (最後の手段として使用します)。

```
dump tran[saction] database_name
with no_log
```

データベース・デバイスで障害が発生した後にログをバックアップします。

```
dump tran[saction] database_name
to [compress::compression_level::] stripe_device
  [at backup_server_name]
  [density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  dumpvolume = volume_name,
  file = file_name]
[[stripe on [compress::compression_level::] stripe_device
  [at backup_server_name]
  [density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  dumpvolume = volume_name,
  file = file_name]]
[[[stripe on [compress::compression_level::] stripe_device
  [at backup_server_name]
  [density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  dumpvolume = volume_name,
  file = file_name]]...]]
[with {
  density = density_value,
  blocksize = number_bytes,
  capacity = number_kilobytes,
  compression = compress_level,
  dumpvolume = volume_name,
  file = file_name,
  [dismount | nodismount],
  [nounload | unload],
  retaindays = number_days,
  [noinit | init],
  no_truncate,
  notify = {client | operator_console}}}]
```

---

Tivoli Storage Manager からバックアップ サービスが提供されている場合に、トランザクション・ログをコピーします。

```
dump transaction database_name
to "syb_tsm::object_name"
  [blocksize = number_bytes]
[stripe on "[syb_tsm::object_name"
  [blocksize = number_bytes]...]
[with {
  blocksize = number_bytes,
  compression = compress_level,
  passwd = password,
  [noinit | init],
  notify = {client | operator_console},
  verify = {header | full}
}]
```

execute

プロシージャを実行するか、または Transact-SQL コマンドを動的に実行します。

```
[exec[ute]] [@return_status =]
[[[server .]database.]owner.]procedure_name[:number]
  [[@parameter_name =] value |
  [ @parameter_name =] @variable [output]
  [, [ @parameter_name =] value |
  [ @parameter_name =] @variable [output]...]]
[with recompile]
```

• または exec[ute] ("*string*" | *char\_variable* [+ "*string*" | *char\_variable*]...)

fetch

カーソル結果セットからローまたはローのセットを返します。

```
fetch [next |prior | first | last | absolute
  fetch_offset | relative fetch_offset ]
[from] cursor_name
[ into fetch_target_list ]
```

goto label

ユーザ定義のラベルへ分岐します。

```
label:
goto label
```

grant

個々のユーザ、ユーザ・グループ、役割にパーミッションを割り当てます。また、ユーザ、システム役割、またはユーザ定義役割に役割を割り当てます。データベース・オブジェクトへのアクセス・パーミッションを付与する場合は、次のように入力します。

```
grant {all [privileges]} permission_list
on {table_name [(column_list)]
  | view_name [(column_list)]
  | stored_procedure_name}
to {public | name_list | role_list}
[with grant option]
```

組み込み関数を使用するパーミッションを付与する場合：

```
grant select
on [builtin] builtin
to { name_list | role_list }
```

特定のコマンドの実行パーミッションを付与する場合：

```
grant {all [privileges] | command_list}
to {public | name_list | role_list}
```

特定の dbcc コマンドへのアクセスを付与する場合：

```
grant dbcc {dbcc_command [on {all | database}]
[, dbcc_command [on {all | database}], ...]}
to {user_list | role_list }
```

暗号化キーの作成パーミッションを付与する場合：

```
grant create encryption key to {user_list | role_list | group_list}
```

テーブルまたはテーブル内のカラム・リストの decrypt パーミッションを付与する場合：

```
grant decrypt on [ owner.]tablename[(columnname [{,columnname}])]
to {user | group | role}
```

特定のシステム・テーブルにデフォルトのパーミッションを付与する場合：

```
grant default permissions on system tables
```

ユーザまたは役割に対して役割を付与する場合：

```
grant {role role_granted [, role_granted ...]}
to grantee [, grantee...]
```

サーバ・ユーザ ID を他のサーバ・ログインに切り替え、その使用をターゲット・ログインの役割に基づいて制限する場合：

```
grant set proxy to role_list
[restrict role role_list | all | system]
```

group by 句と having 句

テーブルをグループに分類したり、having 句の条件に一致するグループだけを返したりするために、select 文内で使用します。

```
Start of select statement
[group by [all] aggregate_free_expression
[, aggregate_free_expression]...]
[having search_conditions]
End of select statement
```

if...else

SQL 文の実行に条件を設定します。

```
if logical_expression [plan "abstract plan"]
statements
[else
[if logical_expression] [plan "abstract plan"]
statement]
```

insert

テーブルまたはビューに新しいローを追加します。

```
insert [into] [database.[owner.]]{table_name|view_name}
[(column_list)]
{values (expression [, expression]...)
|select_statement [plan "abstract plan"]}
```

kill

プロセスを強制終了します。

```
kill spid with statusonly
```

load database

dump database で作成されたトランザクション・ログなどのユーザ・データベースのバックアップ・コピーをロードし、データベース・ダンプとともにロードされたアーカイブ・データベースをマテリアライズします。定期的なデータベースのロードを設定する場合は、次のようにします。

- 定期的なデータベースのロードを設定します。

```
load database database_name
from [compression=]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]
with verify only [= header | full]
[stripe on [compression=]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]
[[stripe on [compression=]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]]...]
[with {
density = density_value,
blocksize = number_bytes,
compression,
dumpvolume = volume_name,
file = file_name,
[dismount | nodismount],
[nounload | unload],
passwd = password,
notify = {client | operator_console},
[override]]}]
```

- バックアップをロードしないでヘッダ情報またはファイル情報を調べます。

```
load database database_name
from [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]
[stripe on [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]
[[stripe on [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]]...]
[with {
density = density_value,
blocksize = number_bytes,
compression,
dumpvolume = volume_name,
file = file_name,
[dismount | nodismount],
[nounload | unload],
passwd = password,
```

```
listonly [= full],
headeronly,
notify = {client | operator_console}
}}
```

- アーカイブ・データベースをマテリアライズします。

```
load database database_name
from dump_device
[[stripe on stripe_device ...]
[with [norecovery,][passwd=password]]
```

- サイトで Tivoli Storage Manager がライセンスされている場合はデータベースのコピーをロードします。

```
load database database_name
from syb_tsm::[[-S source_sever_name][-D source_database_name]
::]object_name [blocksize = number_bytes]
[stripe on syb_tsm::[[-S source_sever_name]
[-D source_database_name]::]object_name
[blocksize = number_bytes]
[[stripe on syb_tsm::[[-S source_sever_name]
[-D source_database_name]::]object_name
[blocksize = number_bytes]...]]
[with {
blocksize = number_bytes,
passwd = password,
listonly [= full],
headeronly,
notify = {client | operator_console},
[[verifyonly | verify] [= header | full]]
}]
```

load transaction

dump transaction によって作成されたトランザクション・ログのバックアップ・コピーをロードします。定期的なログのロードを設定する場合は、次のようにします。

定期的なログのロードを設定します。

```
load tran[saction] database_name
from [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]
[stripe on [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]
[[stripe on [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]]...]
[with {
density = density_value,
blocksize = number_bytes,
compression,
dumpvolume = volume_name,
```

```

file = file_name,
[dismount | nodismount],
[nounload | unload],
notify = {client | operator_console}
}}]

```

バックアップ・ログをロードしないでヘッダ情報またはファイル情報を調べます。

```

load tran[saction] database_name
from [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]
[stripe on [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]
[[stripe on [compress::]stripe_device
[at backup_server_name]
[density = density_value,
blocksize = number_bytes,
dumpvolume = volume_name,
file = file_name]]...]
[with {
density = density_value,
blocksize = number_bytes,
compression,
dumpvolume = volume_name,
file = file_name,
[dismount | nodismount],
[nounload | unload],
listonly [= full],
headeronly,
notify = {client | operator_console}
until_time = datetime}}]

```

アーカイブ・データベースにトランザクション・ログをロードします。

```

load tran[saction] database_name
from dump_device
[[stripe on stripe_device] ...]

```

Tivoli Storage Manager のみ サイトで Tivoli Storage Manager がライセンスされている場合はトランザクション・ログのコピーをロードします。

```

load transaction database_name
from syb_tsm::[[-S source_sever_name]][-D source_database_name]
::]object_name [blocksize = number_bytes]
[stripe on syb_tsm::[[-S source_sever_name]
[-D source_database_name>::]object_name
[blocksize = number_bytes]]
[[stripe on syb_tsm::[[-S source_sever_name]
[-D source_database_name>::]object_name
[blocksize = number_bytes]]...]
[with {
blocksize = number_bytes,
passwd = password,
listonly [= full],
headeronly,
}

```

```

notify = {client | operator_console},
until_time = datetime
}]

```

lock table

トランザクション内でテーブルを明示的にロックします。

```

lock table table_name in {share | exclusive} mode
[wait [numsecs] | nowait]

```

mount

データベースを移送先またはセカンダリの Adaptive Server に接続します。

```

mount database all | database_mapping[, database_mapping, ...]
from "manifest_file"
[using device_mapping [, device_mapping...]
[with listonly]

```

```

database_mapping:

```

```

    origdbname as newdbname
    | newdbname = origdbname
    | origdbname
    | newdbname

```

```

device_mapping

```

```

    logical_device_name as new_physical_name
    | new_physical_name = logical_device_name
    | original_physical_name
    | new_physical_name

```

online database

通常のロード・シーケンスの後でデータベースを誰でも使用できるようにします。

```

online database database_name [for standby_access]

```

open

処理のためにカーソルを開きます。

```

open cursor_name

```

order by 句

クエリ結果を、指定されたカラムのソート順で返します。

```

[Start of select statement]
[order by
    {table_name.| view_name.}
    column_name | select_list_number | expression}
    [asc | desc]
    [, {table_name.| view_name.}
    column_name | select_list_number | expression}
    [asc | desc]]...]
[End of select statement]

```

prepare transaction

サーバがトランザクションをコミットする準備ができていようかどうかを確認するために、-2 フェーズ・コミット・アプリケーションで DB-Library によって使用されます。

```

prepare tran[saction]

```

print

ユーザ定義のメッセージをユーザの画面に表示します。

```

print
    {format_string | @local_variable |
    @@global_variable}
    [, arg_list]

```

---

quiesce database	<p>指定されたデータベースのリストへの更新処理を中断および再開します。</p> <pre>quiesce database <i>tag_name</i> hold <i>database_list</i> [for external dump] [to <i>manifest_file</i> [with override]]</pre> <p>または</p> <pre>quiesce database <i>tag_name</i> release</pre>
raiserror	<p>ユーザ定義のエラー・メッセージをユーザの画面に表示し、エラー条件が発生したことを記録するシステム・フラグを設定します。</p> <pre>raiserror <i>error_number</i> [<i>format_string</i>   <i>@local_variable</i>] [, <i>arg_list</i>] [with errordata <i>restricted_select_list</i>]</pre>
readtext	<p>指定のオフセット位置から開始して、指定のバイト数または文字数の <i>text</i> 値、<i>unitext</i> 値、<i>image</i> 値を読み込みます。</p> <pre>readtext [<i>database.</i>]<i>owner.</i><i>table_name.column_name</i> <i>text_pointer</i> offset size [holdlock   noholdlock] [readpast] [using {bytes   chars   characters}] [at isolation { [read uncommitted   0]   [read committed   1]   [repeatable read   2]   [serializable   3]}}</pre>
reconfigure	<p>このコマンドには何の効果もありますが、既存のスクリプトを修正しないで実行できるようにするために残されています。</p> <pre>reconfigure</pre>
remove java	<p>Java クラスがデータベース内にインストールされている場合に、1つまたは複数の Java-SQL クラス、パッケージ、JAR をデータベースから削除します。</p> <pre>remove java class <i>class_name</i> [, <i>class_name</i>]...   package <i>package_name</i> [, <i>package_name</i>]...   jar <i>jar_name</i> [, <i>jar_name</i>]... [retain classes]</pre>
reorg	<p>使用するオプションに従って、ページの未使用領域の再利用またはローの転送の削除を行ったり、テーブルのすべてのローを新しいページに書き換えたりします。</p> <pre>reorg compact <i>table_name</i> [partition <i>partition_name</i>] [with {resume, time = <i>no_of_minutes</i>}]</pre> <pre>reorg forwarded_rows <i>table_name</i> [partition <i>partition_name</i>] [with {resume, time = <i>no_of_minutes</i>}]</pre> <pre>reorg rebuild <i>table_name</i> [<i>index_name</i> [partition <i>index_partition_name</i>]]</pre> <pre>reorg reclaim_space <i>table_name</i> [<i>index_name</i>] [partition <i>partition_name</i>] [with {resume, time = <i>no_of_minutes</i>}]</pre>
return	<p>バッチまたはプロシージャを無条件に終了し、オプションでリターン・ステータスを提供します。</p> <pre>return [<i>integer_expression</i>] [plan "<i>abstract_plan</i>"]</pre>

revoke

ユーザ、グループ、役割に付与されたパーミッションまたは役割を取り消します。データベース・オブジェクトにアクセスするパーミッションを取り消す場合：

```
grant option for
    {all [privileges] | permission_list}
on {table_name [(column_list)]
    | view_name [(column_list)]
    | stored_procedure_name}
from {public | name_list | role_list}
[cascade]
```

組み込み関数を選択するパーミッションを取り消す場合：

```
revoke select
    on [builtin] builtin
to { name_list | role_list }
```

データベース・オブジェクトの作成、**set proxy** の実行、または **set session authorization** の実行を行うパーミッションを取り消す場合：

```
revoke {all [privileges] | command_list}
from {public | name_list | role_list}
```

ユーザや別の役割から役割を取り消す場合：

```
revoke role {role_name [, role_list ...]} from
    {grantee [, grantee ...]}
```

一部の **dbcc** コマンドでのアクセスを取り消す場合：

```
revoke dbcc {dbcc_command [on {all | database}]
    [, dbcc_command [on {all | database}], ...]}
from {user_list | role_list}
```

他のユーザ、グループ、役割から、暗号化キーを作成するパーミッションを取り消す場合：

```
revoke create encryption key from user | role | group
```

テーブルまたはテーブル内のカラム・リストの **decrypt** パーミッションを取り消す場合：

```
revoke decrypt on [owner.] tablename[(columnname [{,columnname}])]
from user | group | role
```

**public** からデフォルトのパーミッションを取り消す場合：

```
revoke default permissions on system tables
```

rollback

ユーザ定義トランザクションを、トランザクション内の指定したセーブポイントまで、またはトランザクションの開始点までロールバックします。

```
rollback [tran | transaction | work]
    [transaction_name | savepoint_name]
```

rollback trigger

トリガによって実行された処理と、トリガを起動させたデータ変更処理をロールバックします。また、オプションで **raiserror** 文を実行します。

```
rollback trigger
    [with raiserror_statement]
```

save transaction

トランザクション内にセーブポイントを設定します。

```
save transaction savepoint_name
```

---

select

データベース・オブジェクトからローを取得します。

```
select ::=
  select [all | distinct]
  [top unsigned_integer]
  select_list
  [into_clause]
  [from_clause]
  [where_clause]
  [group_by_clause]
  [having_clause]
  [order_by_clause]
  [compute_clause]
  [read_only_clause]
  [isolation_clause]
  [browse_clause]
  [plan_clause]

select_list ::=
into_clause ::=
  into [(database.] owner.] table_name
  [(colname encrypt [with [database.[owner].]keyname] [,
  colname encrypt_clause ...])]
  [[external table at
  'server_name.[database].[owner].object_name'
  | external directory at 'pathname'
  | external file at 'pathname' [column delimiter 'string']]
  [on segment_name]
  dml_logging = (full | minimal)
  [partition_clause]
  [lock {datarows | datapages | allpages}]
  [with [, into_option[, into_option] ...]]

  | into existing table table_name

partition_clause ::=
  partition by range (column_name[, column_name]...)
  ((partition_name) values <= ({constant | MAX}
  [, {constant | MAX}] ...) [on segment_name]
  [, [partition_name] values <= ({constant | MAX}
  [, {constant | MAX}] ...) [on segment_name]]...)

  | partition by hash (column_name[, column_name]...)
  { (partition_name [on segment_name]
  [, partition_name [on segment_name]]...)
  | number_of_partitions
  [on (segment_name [, segment_name]...)]}

  | partition by list (column_name)
  ((partition_name) values (constant[, constant] ...)
  [on segment_name]
  [, [partition_name] values (constant[, constant] ...)
  [on segment_name]] ...)

  | partition by roundrobin
  { (partition_name [on segment_name]
  [, partition_name [on segment_name]]...)
  | number_of_partitions
  [on (segment_name [, segment_name]...)]}

into_option ::=
  | max_rows_per_page = num_rows
  | exp_row_size = num_bytes
```

```

| reservepagegap = num_pages
| identity_gap = gap
from_clause ::=
from table_reference [, table_reference]...
table_reference ::=
table_view_name | ANSI_join
table_view_name ::=
[[database.]owner.]{table_name | view_name}
[as] [correlation_name]
[(index {index_name | table_name})]
[parallel [degree_of_parallelism]]
[fetch size][r | mru]
[holdlock | noholdlock]
[readpast]
[shared]
ANSI_join ::=
table_reference join_type join table_reference
join_conditions
join_type ::= inner | left [outer] | right [outer]
join_conditions ::= on search_conditions
where_clause ::=
where search_conditions
group_by_clause ::=
group by [all] aggregate_free_expression
[, aggregate_free_expression]...
having_clause ::=
having search_conditions
order_by_clause ::=
order by sort_clause [, sort_clause]...
sort_clause ::=
{[[database.]owner.]{table_name.|view_name.}}column_name
| select_list_number
| expression }
[asc | desc]
compute_clause ::=
compute row_aggregate (column_name)
[, row_aggregate (column_name)]...
[by column_name [, column_name]...]
read_only_clause ::=
for {read only | update [of column_name_list]}
isolation_clause ::=
at isolation
{read uncommitted | 0}
| {read committed | 1}
| {repeatable read | 2}
| {serializable | 3}
browse_clause ::=
for browse
plan_clause ::=
plan "abstract plan"

```

---

set

ユーザの作業セッション中の Adaptive Server クエリ処理オプションを設定します。一部のオプションは、トリガまたはストアド・プロシージャ内部で設定されます。

```
set advanced_aggregation on/off
set @variable = expression [, @variable = expression...]
set ansinull { on | off }
set ansi_permissions { on | off }
set arithabort [ arith_overflow | numeric_truncation ] { on | off }
set arithignore [arith_overflow] { on | off }
set bulk array size number
set bulk batch size number
set {chained, close on endtran, nocount, noexec, parseonly,
    self_recursion, showplan, sort_resources} {on | off}
set char_convert {off | on [with {error | no_error}]} |
    charset [with {error | no_error}]
set cis_rpc_handling { on | off }
set [clientname client_name | clienthostname host_name
    | clientapplname application_name]
set cursor rows number for cursor_name
set {datefirst number, dateformat format, language language}
set delayed_commit {on | off | default}
set deferred_name_resolution { on | off }
set dml_logging {minimal | default}
set encryption passwd 'password_phrase'
    for {key | column} {keyname | column_name}
set export_options {on | off}
set fipsflagger { on | off }
set flushmessage { on | off }
set fmtonly {on | off}
set forceplan { on | off }
set identity_insert [database.owner.]table_name {on | off}
set identity_update table_name {on | off}
set index_union on | off
set literal_autoparam on | off
set lock {wait [numsecs] | nowait}
set metrics_capture on | off
set offsets {select, from, order, compute, table,
    procedure, statement, param, execute} {on | off}
set option show
set opttimeoutlimit
set parallel_degree number
set plan {dump | load} [group_name] {on | off}
set plan exists check { on | off }
set plan for show
```

```

set plan optgoal {allrows_mix | allrows_dss}
set plan opttimeoutlimit number
set plan replace { on | off }
set prefetch [on|off]
set print_minlogged_mode_override
set proc_output_params {on | off}
set proc_return_status {on | off}
set process_limit_action { abort | quiet | warning }
set proxy login_name
set quoted_identifier { on | off }
set repartition_degree number
set repthreshold number
set resource_granularity number
set role {"sa_role" | "sso_role" | "oper_role" |
         role_name [with passwd "password"]} {on | off}
set {rowcount number, textsize number}
set scan_parallel_degree number
set session authorization login_name
set switch [serverwide] {on | off} trace_flag [,trace_flag,] [with option [, option]]
set show_exec_info ["on" | "off"]
set show_sqltext {on | off}
set statistics {io, subquerycache, time, plancost} {on | off}
set statistics simulate {on | off}
set strict_dtm_enforcement { on | off }
set string_truncation { on | off }
set system_view {instance | cluster | clear}
set textsize {number}
set tracefile [filename] [off] [for spid]
set transaction isolation level {
    [read uncommitted | 0] |
    [read committed | 1] |
    [repeatable read | 2] |
    [serializable | 3]}
set transactional_rpc { on | off }

```

setuser データベース所有者が別のユーザになり代わることを可能にします。  
 setuser ["*user\_name*"]

shutdown コマンドの発行元 Adaptive Server、そのローカル Backup Server、またはリモート Backup Server を停止します。  
 shutdown [*srvname*] [with {wait [= "*hh:mm:ss*"] | nowait}]]  
 クラスタの構文：  
 shutdown {cluster | [*instance\_name*]} [with {wait | nowait}]

transfer table	<p>テーブルの増分転送を開始します。</p> <pre>transfer table [[db.]owner.]table [to   from] destination_file [ for { ase   bcp   iq   csv } ] [ with {column_separator=string}, {column_order=option}, {encryption=option}, {row_separator=string}, {resend=id}, {progress=sss}, {tracking_id=nnn} {sync = true   false}], {fixed_length = true   false} , null_byte = true   false}]</pre>
truncate table	<p>テーブルまたはパーティションからすべてのローを削除します。</p> <pre>truncate table [[database.]owner.]table_name [partition partition_name]</pre>
union 演算子	<p>複数のクエリ結果を結合する単一の結果セットを返します。all キーワードを指定しないかぎり、重複するローは結果セットから削除されます。</p> <pre>select [top unsigned_integer] select_list [into clause] [from clause] [where clause] [group by clause] [having clause] [union [all] select [top unsigned_integer] select_list [from clause] [where clause] [group by clause] [having clause]]... [order by clause] [compute clause]</pre>
unmount	<p>データベースを停止し、Adaptive Server から削除します。さらにデバイスを非アクティブ化し、削除します。</p> <pre>unmount database dbname_list to manifest_file</pre>
update	<p>データを追加したり、既存のデータを変更することで、既存のロー内のデータを変更します。</p> <pre>update [top unsigned_integer] [[database.]owner.]{table_name   view_name} set [[{database.]owner.]{table_name. view_name.} column_name1 = {expression1   NULL   (select_statement)} ] variable_name1 = {expression1   NULL   (select_statement)} [, column_name2 = {expression2   NULL   (select_statement)}]... [, variable_name2 = {expression2   NULL   (select_statement)}]...  [from [[database.]owner.]{view_name [readpast] table_name [(index {index_name   table_name} [fetch size][r u mru])]} [readpast] [, [[database.]owner.]{view_name [readpast] table_name [(index {index_name   table_name} [fetch size][r u mru])]} [readpast] ...] [where search_conditions] [plan "abstract plan"]  update [[database.]owner.]{table_name   view_name} set [[{database.]owner.]{table_name. view_name.} column_name1 = {expression1   NULL   (select_statement)} ]</pre>

	<pre> variable_name1 =   {expression1   NULL   (select_statement)} [, column_name2 =   {expression2   NULL   (select_statement)}]... [, variable_name2 =   {expression2   NULL   (select_statement)}]... where current of cursor_name         </pre>
update all statistics	<p>指定したテーブルの統計情報をすべて更新します。update all statistics は 1 つのデータ・パーティションに対して実行できます。</p> <pre> update all statistics table_name [partition data_partition_name]         </pre>
update index statistics	<p>インデックス上にある、すべてのカラムの統計を更新します。</p> <pre> update index statistics   table_name [[partition data_partition_name]     [index_name [partition index_partition_name]]]   [using step values]   [with consumers = consumers][, sampling=N percent]         </pre>
update statistics	<p>指定したインデックス、またはインデックス、テーブル、パーティション内のすべてのカラムのキー値の分布に関する情報を更新し、グローバル・ノンクラスタード・インデックスのデータ変更カウンタをリセットします。</p> <pre> update statistics table_name   [[partition data_partition_name] [(column_list)]     index_name [partition index_partition_name]]   [using step values]   [with consumers = consumers][, sampling=N percent]         </pre>
update table statistics	<p><b>sysstabstats</b> テーブルに格納されている統計を更新します。</p> <pre> update table statistics table_name   [ partition data_partition_name ]   [index_name [partition index_partition_name]]         </pre>
use	<p>作業するデータベースを指定します。</p> <pre> use database_name         </pre>
waitfor	<p>文ブロック、ストアド・プロシージャ、またはトランザクションを実行するための、特定の時間、時間間隔、またはイベントを指定します。</p> <pre> waitfor {delay time   time time   errorexit   processexit   mirrorexit}         </pre>
where 句	<p><b>select</b>、<b>insert</b>、<b>update</b>、または <b>delete</b> の各文で探索条件を設定します。</p> <pre> where [not] expression comparison_operator expression where {[not] expression comparison_operator expression}   {...} where [not] expression [not] like "match_string"   [escape "escape_character "] where [not] expression is [not] null where [not] expression [not] between expression and expression where [not] expression [not] in ({value_list   subquery}) where [not] exists (subquery) where [not] expression comparison_operator {any   all} (subquery) where [not] column_name join_operator column_name where [not] logical_expression where [not] expression {and   or} [not] expression         </pre>

---

while	文または文ブロックの繰り返し実行の条件を設定します。 <pre>while <i>logical_expression</i> [plan "<i>abstract plan</i>"] <i>statement</i></pre>
writetext	既存の <b>text</b> カラム、 <b>unitext</b> カラム、 <b>image</b> カラムで、最小限のログを取る対話型の更新を実行します。 <pre>writetext [[<i>database.</i>]owner.]<i>table_name.column_name</i> text_pointer [readpast] [with log] <i>data</i></pre>

## 対話型 dbsql コマンド

以下は、Adaptive Server の対話型 **dbsql** コマンドの構文と概要です。詳細については、『リファレンス・マニュアル：コマンド』を参照してください。

clear	対話型 SQL ウィンドウ枠をクリアします。 <pre>clear</pre>
configure	[Interactive SQL オプション] ダイアログを開きます。 <pre>configure</pre>
connect	データベースへの接続を確立します。 <pre>connect [ to <i>engine_name</i> ] [database <i>database_name</i>] [as <i>connection_name</i>] [user] <i>user_id</i> identified by password <i>engine_name, database_name, connection_name, user_id,</i> <i>password</i> :{<i>identifier</i>   <i>string</i>   <i>hostvar</i>}  connect using <i>connect_string</i> : {<i>identifier</i>   <i>string</i>   <i>hostvar</i>}</pre>
disconnect	データベースへの現在の接続を切断します。 <pre>disconnect [{<i>identifier</i>   <i>string</i>   <i>hostvar</i>}   current   all]</pre>
exit	対話型 SQL を終了します。 <pre>{exit   quit   bye} [{<i>number</i>   <i>connection_variable</i>}]</pre>
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。 <pre>input into [ <i>owner.</i>]<i>table_name</i> [ from <i>filename</i>   prompt] [ format { <i>ascii</i>   <i>dbase</i>   <i>dbasell</i>   <i>dbaselll</i>   <i>excel</i>   <i>fixed</i>   <i>foxpro</i>   <i>lotus</i> }] [ escape character <i>character</i> ] [ escapes { on   off }] [ by order   by name ] [ delimited by <i>string</i> ] [ column widths (integer , ... ) ] [ nostrip ] [ ( <i>column_name</i> , ... ) ] [ encoding {<i>identifier</i>   <i>string</i>}]</pre>

---

output	データベース・テーブルから外部ファイルまたはキーボードにデータをエクスポートします。 output to <i>filename</i> [ append ] [ verbose ] [ format {ascii   dbase   dbaseII  dbaseIII   excel   fixed   foxpro   lotus   sql   xml}] [ escape character <i>character</i> ] [ escapes { on   off} ] [ delimited by string ] [ quote string [ all ] ] [ column widths (integer , ... ) ] [ hexadecimal { on   off   asis } ] [ encoding { <i>string</i>   <i>identifier</i> }]
parameters	対話型 SQL コマンド・ファイルにパラメータを指定します。 parameters parameter1, parameter2, ...
read	対話型 SQL 文をファイルから読み込みます。 read [ encoding { <i>identifier</i>   <i>string</i> }] <i>file_name</i> [ <i>parameters</i> ]
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 connection { <i>identifier</i>   <i>string</i>   <i>hostvar</i> }
set option	対話型 SQL オプションの値を変更します。 set [ temporary] option [ { <i>identifier</i>   <i>string</i>   <i>hostvar</i> }.  public.] { <i>identifier</i>   <i>string</i>   <i>hostvar</i> } = [ <i>option_value</i> ] set permanent set
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。 start logging <i>file_name</i>
stop logging	現在のセッションで実行された SQL 文のログの記録を停止します。 stop logging
system	対話型 SQL から実行ファイルを起動します。 system '[ <i>path</i> ] <i>file_name</i> '

---

## システム・プロシージャ

以下は、Adaptive Server のシステム・ストアド・プロシージャの構文と概要です。詳細については、『リファレンス・マニュアル：プロシージャ』を参照してください。

<code>sp_activeroles</code>	すべてのアクティブな役割を表示します。 <code>sp_activeroles [expand_down]</code>
<code>sp_add_qpgroup</code>	抽象プラン・グループを追加します。 <code>sp_add_qpgroup new_name</code>
<code>sp_add_resource_limit</code>	Adaptive Server ログインまたはアプリケーションがクエリ、クエリ・バッチ、トランザクションの実行に使用できるサーバ・リソース数の制限値を作成します。 <code>sp_add_resource_limit name, appname, rangename, limittype, limitvalue [, enforced [, action [, scope ]]]</code>
<code>sp_add_time_range</code>	名前付き時間範囲を Adaptive Server に追加します。 <code>sp_add_time_range name, startday, endday, starttime, endtime</code>
<code>sp_addalias</code>	ユーザをデータベース内で別のユーザとして認識されるようにします。 <code>sp_addalias loginame, name_in_db</code>
<code>sp_addauditrecord</code>	ユーザが、監査証跡にユーザ定義の監査レコード (コメント) を入力できるようにします。 <code>sp_addauditrecord [text [, db_name [, obj_name [, owner_name [, dbid [, objid]]]]]]</code>
<code>sp_addauditable</code>	監査機能のインストール後に、別のシステム監査テーブルを追加します。 <code>sp_addauditable devname</code>
<code>sp_addengine</code>	既存のエンジン・グループにエンジンを追加します。グループがなければ、エンジン・グループを作成してからエンジンを追加します。 <code>sp_addengine engine_number, engine_group [, instance_id]</code>
<code>sp_addexclass</code>	クライアント・アプリケーション、ログイン、ストアド・プロシージャにバインドできるユーザ定義の実行クラスを作成または更新します。 <code>sp_addexclass classname, priority, timeslice, engine_group [, instance_id]</code>
<code>sp_addextendedproc</code>	<b>master</b> データベース内に拡張ストアド・プロシージャ (ESP: extended stored procedure) を作成します。 <code>sp_addextendedproc esp_name, dll_name</code>
<code>sp_addexternlogin</code>	コンポーネント統合サービスのみ - CIS を介してリモート・サーバとの通信に使用する代替ログイン・アカウントとパスワードを作成します。 <code>sp_addexternlogin server, loginame, externname [, externpasswd] [rolename]</code>

sp_addgroup	データベースにグループを追加します。グループを使用すると、権利の付与および取り消しをまとめて行うことができます。 <code>sp_addgroup <i>grpname</i></code>
sp_addlanguage	代替言語における月と曜日の名前、およびその日付フォーマットを定義します。 <code>sp_addlanguage <i>language, alias, months, shortmons, days, datefmt, datefirst</i></code>
sp_addlogin	Adaptive Server に新しいユーザ・アカウントを追加します。また、作成時にパスワードの有効期間、パスワードの最小長、指定のログインで許容される失敗したログインの最大数を指定します。 <code>sp_addlogin <i>loginame, passwd</i> [, <i>defdb</i>] [, <i>deflanguage</i>] [, <i>fullname</i>] [, <i>passwdexp</i>] [, <i>minpwdlen</i>] [, <i>maxfailedlogins</i>] [, <i>auth_mech</i>]</code>
sp_addmessage	ストアド・プロシージャでの <code>print</code> と <code>raiserror</code> の呼び出し、および <code>sp_bindmsg</code> で使用するために、 <code>sysusermessages</code> にユーザ定義メッセージを追加します。 <code>sp_addmessage <i>message_num, message_text</i> [, <i>language</i>] [, <i>with_log</i>] [, <i>replace</i>]</code>
sp_addobjectdef	コンポーネント統合サービスのみ – ローカル・テーブルと外部記憶装置の間のマッピングを指定します。 <code>sp_addobjectdef <i>tablename, objectdef</i> [, "<i>objecttype</i>"]</code>
sp_addremotelogin	<code>master.dbo.sysremotelogins</code> にエントリを追加して、新しいリモート・サーバ・ユーザを認可します。 <code>sp_addremotelogin <i>remoteserver</i> [, <i>loginame</i>] [, <i>remotename</i>]</code>
sp_addsegment	データベースにデータベース・デバイスのセグメントを定義します。 <code>sp_addsegment <i>segname, dbname, devname</i></code>
sp_addserver	リモート・サーバまたはローカル・サーバの名前を定義します。 <code>sp_addserver <i>lname</i> [, <i>class</i>] [, <i>pname</i>]</code> <code>sp_addserver '<i>logical_server_name</i>', ASEnterprise, '<i>host:port:filter</i>'</code>
sp_addthreshold	データベース・セグメントの空き領域の使用状況をモニタするために、スレッシュホールドを作成します。 <code>sp_addthreshold <i>dbname, segname, free_space, proc_name</i></code>
sp_addtype	ユーザ定義データ型を作成します。 <code>sp_addtype <i>typename, phystype</i> [(<i>length</i>)   (<i>precision</i> [, <i>scale</i>])] [, "<i>identity</i>"   <i>nulltype</i>]</code>
sp_addumpdevice	Adaptive Server にダンプ・デバイスを追加します。 <code>sp_addumpdevice ("<i>tape</i>"   "<i>disk</i>"), <i>logicalname, physicalname</i> [, <i>tapesize</i>]</code>
sp_adduser	新しいユーザを現在のデータベースに追加します。 <code>sp_adduser <i>loginame</i> [, <i>name_in_db</i>] [, <i>grpname</i>]</code>
sp_altermessage	Adaptive Server エラー・ログ内の特定のシステム定義メッセージまたはユーザ定義メッセージのロギングを、有効にしたり無効にしたりします。 <code>sp_altermessage <i>message_id, parameter, parameter_value</i></code>

---

sp_audit	<p>システム・セキュリティ担当者による監査オプションの設定を可能にします。</p> <p style="padding-left: 2em;">sp_audit <i>option</i>, <i>login_name</i>, <i>object_name</i> [, <i>setting</i>]</p> <ul style="list-style-type: none"> <li>• または sp_audit 'restart'</li> </ul>
sp_autoconnect	<p>コンポーネント統合サービスのみ – 特定のユーザが使用するリモート・サーバへの接続パススルーを定義します。これによって、このユーザはログイン時に自動的にパススルー・モードを開始できます。</p> <p style="padding-left: 2em;">sp_autoconnect <i>server</i>, {true   false} [, <i>loginame</i>]</p>
sp_autoformat	<p>判読しやすい結果セット・データを生成し、可変長の文字データの幅を再フォーマットして非ブランク文字のみを表示します。</p> <p style="padding-left: 2em;">sp_autoformat <i>fulltablename</i> [, <i>selectlist</i>, <i>whereclause</i>, <i>orderby</i>]</p>
sp_bindcache	<p>データベース、テーブル、インデックス、text、または image をデータ・キャッシュにバインドします。</p> <p style="padding-left: 2em;">sp_bindcache <i>cachename</i>, <i>dbname</i> [, [<i>ownername</i>].<i>tablename</i> [, <i>indexname</i>   "text only"]]</p>
sp_bindefault	<p>ユーザ定義デフォルトをカラムまたはユーザ定義データ型にバインドします。</p> <p style="padding-left: 2em;">sp_bindefault <i>defname</i>, <i>objname</i> [, <i>futureonly</i>]</p>
sp_bindindexclass	<p>クライアント・アプリケーション、ログイン、ストアド・プロシージャに、実行クラスを対応させます。</p> <p style="padding-left: 2em;">sp_bindindexclass "<i>object_name</i>", "<i>object_type</i>", "<i>scope</i>", "<i>classname</i>"</p>
sp_bindmsg	<p>ユーザ・メッセージを、参照整合性制約または検査制約にバインドします。</p> <p style="padding-left: 2em;">sp_bindmsg <i>constrname</i>, <i>msgid</i></p>
sp_bindrule	<p>カラムまたはユーザ定義データ型にルールをバインドします。</p> <p style="padding-left: 2em;">sp_bindrule <i>rulename</i>, <i>objname</i> [, <i>futureonly</i>]</p>
sp_cacheconfig	<p>データ・キャッシュの作成、設定、再設定、削除を行い、データ・キャッシュについての情報を提供します。</p> <p style="padding-left: 2em;">sp_cacheconfig [<i>cachename</i> [, "<i>cache_size</i>{P   K   M   G}"] [, <i>logonly</i>   <i>mixed</i>   <i>inmemory_storage</i>] [, <i>strict</i>   <i>relaxed</i>]] [, "<i>cache_partition</i>={1   2   4   8   16   32   64}"] [, <i>instance</i> <i>instance_name</i>]</p>
sp_cachestrategy	<p>テーブル、インデックス、text オブジェクト、または image オブジェクトに対するプリフェッチ (大容量 I/O) と MRU キャッシュの置換方式を有効または無効にします。</p> <p style="padding-left: 2em;">sp_cachestrategy <i>dbname</i>, [<i>ownername</i>].<i>tablename</i> [, <i>indexname</i>   "text only"   "table only" [, {<i>prefetch</i>   <i>mru</i>}, {"on"   "off"}]]</p>
sp_changedbowner	<p>ユーザ・データベースの所有者を変更します。</p> <p style="padding-left: 2em;">sp_changedbowner <i>loginame</i> [, true]</p>
sp_changegroup	<p>ユーザのグループを変更します。</p> <p style="padding-left: 2em;">sp_changegroup <i>grpname</i>, <i>username</i></p>

sp_checknames	7ビットの ASCII 文字セット以外の文字が、現在のデータベースの名前に使用されていないかを検査します。 <code>sp_checknames [help   silent]</code>
sp_checkreswords	識別子として使われている Transact-SQL の予約語を検出して表示します。 <code>sp_checkreswords [user_name_param]</code>
sp_checksourc	コンパイル済みオブジェクトのソース・テキストと計算カラムのソース・テキストがあるかどうかを確認します。 <code>sp_checksourc [objname [, tabname [, username]]]</code>
sp_chgattribute	<code>max_rows_per_page</code> 、 <code>fillfactor</code> 、 <code>reservpagegap</code> 、 <code>exp_row_size</code> の各値を変更して、テーブルまたはインデックスの今後の領域を割り付けます。また、テーブルに <code>concurrency_opt_threshold</code> を設定します。オプティミスティック・インデックス・ロック用のユーザ・インタフェースを提供します。 <code>sp_chgattribute objname, {"max_rows_per_page"   "fillfactor"   "reservpagegap"   "exp_row_size"   "concurrency_opt_threshold"   "optimistic_index_lock"   "identity_burn_max"   "plldegree"}, value, optvalue</code> <code>sp_chgattribute objname, {"identity_gap", set_number   "dealloc_first_txtpg", value}</code>
sp_cleanpwdchecks	ユーザ定義のテーブルに格納されている、ログインおよびパスワード関連の属性を削除する日時と方法を定義できるようにします。 <code>sp_cleanpwdchecks, login_name</code>
sp_clearpsex	<code>sp_setpsex</code> によって設定された Adaptive Server セッションの実行属性をクリアします。 <code>sp_clearpsex spid, exeattr</code>
sp_clearstats	すべてのサーバ・ユーザまたは指定ユーザの新しいアカウント期間を開始します。 <code>sp_reportstats</code> を実行して、前回の期間の統計を表示します。 <code>sp_clearstats [loginame]</code>
sp_client_addr	クライアント・アプリケーションが付加されている各 Adaptive Server タスクの IP アドレスを、 <code>spid</code> とクライアント・ホスト名とともに表示します。 <code>sp_client addr [spid]</code>
sp_clusterlockusage	クラスタ環境内 - クラスタ内のフリー・ロック、使用中のロック、保持されているロックについてレポートします。 <code>sp_clusterlockusage</code>
sp_cluster	<b>クラスタ環境のみ</b> クラスタに関係する複数のプロシージャを実行します。接続を別の論理クラスタまたはインスタンスにマイグレートします。 <code>sp_cluster connection, migrate, lc_name, instance_name, "spid_list"</code> 新しいインスタンスへの以前の接続マイグレーションが待機中かどうかを調べ、待機中であればそのマイグレーションをキャンセルします。 <code>sp_cluster connection, ['migrate_status'   'migrate_cancel'] [, 'spid_list']</code>

---

未処理のまま残っているアクション ( キャンセルやタイミングの変更など ) を変更します。

```
sp_cluster logical, "action", lc_name, {
    cancel, action_handle |
    modify_time, action_handle, wait_option[, timeout] |
    release, action_handle }
```

論理クラスタにリソースまたは 1 つ以上のルートを追加します。

```
sp_cluster logical, "add", lc_name, {
    route, route_type, key_list |
    instance, instance_list |
    failover, instance_list }
```

論理クラスタ間でルートを移動します。

```
sp_cluster logical, "alter", lc_name, route, route_type, key_list
```

新しい論理クラスタを作成します。

```
sp_cluster logical, "create", lc_name
```

1 つまたは複数のインスタンスで論理クラスタを停止するか、論理クラスタ全体を停止して、インスタンスまたはクラスタを非アクティブな状態にします。

```
sp_cluster logical, "deactivate", lc_name, {
    "cluster" |
    "instance", instance_list }
[, wait_option[, timeout][, @handle output ]]
```

論理クラスタを削除するか、論理クラスタから 1 つまたは複数のリソースを削除します。

```
sp_cluster logical, "drop", lc_name,
{cluster | instance, instance_list |
 failover, instance_list |
 route, route_type, key_list }
```

元のベース・インスタンスを復活させて、手動フェールバックをリバースします。

```
sp_cluster logical, "failback", lc_name, {
    cluster[, wait_option[, timeout], @handle output ] ] |
    instance, from_instance_list, to_instance_list[, wait_option[,
    timeout], @handle output ] ] }
```

ベース・インスタンスからフェールオーバー・インスタンスへの手動フェールオーバーを開始します。

```
sp_cluster logical, "failover", lc_name, {cluster
    [, to_instance_list[, wait_option[, timeout], @handle output ] ] ]
    [, from_instance_list, to_instance_list[, wait_option[,
    timeout], @handle output ] ] }
```

接続グループを手動で集めて別の論理クラスタにマイグレートします。

```
sp_cluster logical, "gather", lc_name
```

`sp_cluster logical` の完全な構文を表示します。

```
sp_cluster logical, "help"
```

1つまたは複数のインスタンスで論理クラスタを停止するか、論理クラスタ全体を停止します。

```
sp_cluster logical, "offline", lc_name,
    {cluster | instance, instance_list}
    [, wait_option [, timeout [, @handle output ]]]
```

1つまたは複数のインスタンスでデフォルトの論理クラスタを開始します。

```
sp_cluster logical, "online", { lc_name [, instance_list] }
```

論理クラスタのルール (open 論理クラスタ、フェールオーバー・モード、システム・ビュー、起動モード、ロード・プロファイル) を設定します。

```
sp_cluster logical, "set", lc_name, { open
    | failover, failover_mode
    | system_view, view_mode
    | startup, { automatic | manual }
    | load_profile, profile_name }
    login_distribution, { affinity | "round-robin" }
```

論理クラスタに関する情報を表示します。

```
sp_cluster logical, "show"
    [, lc_name [, {action [, state] | route [, type [, key]]}] ]]
```

論理クラスタのロード・プロファイルを設定および管理します。

```
sp_cluster profile, [ "show" [, profile_name] ]
    | "create", profile_name
    | "drop", profile_name
    | "set", profile_name [, weight [, wt_metric [, wt_value] ] ]
    | threshold [, thr_metric [, thr_value] ] ]]
```

論理クラスタのロード・プロファイルを設定および管理します。

```
sp_cluster profile, [ "show" [, profile_name] ] | "create", profile_name | "drop",
    profile_name | "set", profile_name [, weight [, wt_metric [, wt_value] ] ] | threshold [,
    thr_metric [, thr_value] ] ]]
```

sp\_cmp\_all\_qplans

2つの抽象プラン・グループにあるすべての抽象プランを比較します。

```
sp_cmp_all_qplans group1, group2 [, mode]
```

sp\_cmp\_qplans

2つの抽象プランを比較します。

```
sp_cmp_qplans id1, id2
```

sp\_commonkey

2つのテーブルまたはビューの間の共通キーを定義します。

```
sp_commonkey tabaname, tabbname, col1a, col1b
    [, col2a, col2b, ..., col8a, col8b]
```

sp\_companion

高可用性システムのセカンダリ・コンパニオンとして Adaptive Server を設定し、コンパニオン・サーバをフェールオーバー・モードから別のモードへ移行させるなどのクラスタ・オペレーションを実行します。

```
sp_companion
    {server_name {, configure [, {with_proxydb | NULL}]
    [, srvlogin] [, server_password] [, cluster_login] [, cluserpassword]
    | drop | suspend | resume | prepare_failback | do_advisory}
    {, all | help | group_attribute_name | base_attribute_name}
```

---

sp_compatmode	フル互換モードを使用できるかどうかを確認します。 sp_compatmode
sp_configure	グループ別の設定パラメータ、その現在の値、デフォルト値以外の設定、一番最後に設定された値、この設定が使用するメモリ量を表示します。表示レベルがユーザの表示レベルと同等またはそれより低いパラメータだけを表示します。 sp_configure [configname [, configvalue]   group_name   non_unique_parameter_fragment] 'drop instance' [, instance_name] [display_nondefault_settings] sp_configure "configuration file", 0, {"write"   "read"   "verify"   "restore"} "file_name"
sp_copy_all_qplans	ある抽象プラン・グループのすべてのプランを別のグループにコピーします。 sp_copy_all_qplans src_group, dest_group
sp_copy_qplan	抽象プランを抽象プラン・グループにコピーします。 sp_copy_qplan src_id, dest_group
sp_countmetadata	Adaptive Server にあるインデックス、オブジェクト、またはデータベースの数を表示します。 sp_countmetadata "configname" [, dbname]
sp_cursorinfo	特定のカーソルの情報、または使用しているセッションでアクティブなすべての実行カーソルの情報をレポートします。 sp_cursorinfo [{cursor_level   null}] [, cursor_name]
sp_dbextend	データベース／セグメントのペアとデバイスに自動データベース拡張プロシージャをインストールします。個々のセグメントとデバイスにサイト固有のポリシーを定義します。また、大量のロードを処理する前にそのオペレーションを確認するため、データベース拡張構成の実行をシミュレートします。 sp_dbextend 'help' [, command] sp_dbextend [ ['set', ['threshold', dbname, segmentname, freespace   'database', dbname, segmentname { [ [ growby ] [, maxsize ] } ]   'device', devicename { [ [ growby ] [, maxsize ] } } ]   'clear', 'threshold', dbname, segmentname ] ] sp_dbextend 'clear', 'database' [, dbname [, segmentname ] ] sp_dbextend 'clear', 'device' [, devicename ] sp_dbextend 'modify', 'database', dbname, segmentname, { 'growby'   'maxsize' }, newvalue sp_dbextend 'modify', 'device', devicename, { 'growby'   'maxsize' }, newvalue sp_dbextend { 'list'   'listfull' } [, 'database' [, dbname [, segmentname [, order_by_clause ] ] ] ] sp_dbextend { 'list'   'listfull' } [, 'device' [, devicename [, order_by_clause ] ] ] sp_dbextend { 'list'   'listfull' } [, [ 'threshold' [, @dbname [, @segmentname ] ] ] ] sp_dbextend 'check', 'database' [, dbname [, segmentname ] ] sp_dbextend { 'simulate'   'execute' }, dbname, segmentname [, iterations ]

	<pre> sp_dbextend 'trace', {'on'   'off' } sp_dbextend 'reload [defaults]' sp_dbextend { 'enable'   'disable' }, 'database'       [, dbname [, segmentname ] ] sp_dbextend 'who' [, 'spid'   'block'   'all' ] </pre>
sp_dboption	<p>データベース・オプションを表示または変更します。非同期ログ・サービス機能を有効にします。</p> <pre> sp_dboption [dbname, optname, optvalue [, dockptf]] </pre>
sp_dbrecovery_order	<p>ユーザ・データベースをリカバリする順序を指定し、1つまたはすべてのデータベースのユーザ定義のリカバリ順をリストします。</p> <pre> sp_dbrecovery_order [database_name [, rec_order       [, force [ relax   strict ]]]] </pre>
sp_dbremap	<p>Adaptive Server に、<b>alter database</b> が行った変更を認識させます。Adaptive Server のメッセージで指示された場合のみ、このプロシージャを実行してください。</p> <pre> sp_dbremap dbname </pre>
sp_defaultloc	<p>コンポーネント統合サービスのみ – ローカル・データベース内での、オブジェクトのデフォルトの記憶位置を定義します。</p> <pre> sp_defaultloc dbname, defaultloc, defaulttype </pre>
sp_deletesmobj	<p>IBM Tivoli Storage Manager (TSM) から指定されたバックアップ・オブジェクトを削除します。</p> <pre> sp_deletesmob "syb_tsm", "server_name" { "database_name", "object_type",       "dump_type", "until_time", "bs_name" } </pre>
sp_depends	<p>指定したテーブルまたはビューに依存するデータベース内のデータベース・オブジェクトの依存性に関する情報や、指定したビュー、トリガ、またはプロシージャが依存するデータベース内のテーブルやビューを表示します。また、指定したカラムまたはテーブル内のすべてのカラムで定義されているテーブル・カラムの依存性に関する情報も表示します。</p> <pre> sp_depends objname[, column_name] </pre>
sp_deviceattr	<p>UNIX プラットフォームのみ – 既存のデータベース・デバイス・ファイルのデバイス・パラメータ設定を変更します。</p> <pre> sp_deviceattr logicalname, optname, optvalue </pre>
sp_diskdefault	<p>データベース・デバイスを指定しない場合や、<b>create database</b> または <b>alter database</b> コマンドで <b>default</b> を指定する場合に、データベースの記憶領域用にデータベース・デバイスが使用できるかどうかを指定します。</p> <pre> sp_diskdefault logicalname, {defaulton   defaultoff} </pre>
sp_displayaudit	<p>監査オプションのステータスを表示します。</p> <pre> sp_displayaudit ["procedure"   "object"   "login"   "database"   "global"         "default_object"   "default_procedure" [, "name"]] </pre>
sp_displaylevel	<p>sp_configure の出力に、Adaptive Server のどの設定パラメータを表示するかを設定します。</p> <pre> sp_displaylevel [loginame [, leve]] </pre>

---

sp_displaylogin	ログイン・アカウント情報を表示します。 <pre>sp_displaylogin [user_id   [loginname   wildcard]]</pre>
sp_displayroles	他の役割に付与されたすべての役割を表示します。または、役割の階層ツリー全体を表形式で表示します。 <pre>sp_displayroles [grantee_name [, mode]]</pre>
sp_downgrade	以前の 15.0.x リリースへのダウングレードの準備ができているかどうかを検証し、Adaptive Server 15.0.2 によって変更されたシステム・カタログの変更をダウングレードします。 <pre>sp_downgrade @cmd = {'prepare'   'downgrade'   'help'},   @toversion = 'n', @verbose = 0   1[, @override = 0   1]</pre>
sp_dropalias	sp_addalias によって設定されたエイリアス・ユーザ名 ID を削除します。 <pre>sp_dropalias loginname [, force]</pre>
sp_drop_all_qplans	抽象プラン・グループ内のすべての抽象プランを削除します。 <pre>sp_drop_all_qplans name</pre>
sp_drop_qpgroup	抽象プラン・グループを削除します。 <pre>sp_drop_qpgroup group</pre>
sp_drop_qplan	抽象プランを削除します。 <pre>sp_drop_qplan id</pre>
sp_drop_resource_limit	Adaptive Server から 1 つ以上のリソースの制限を削除します。 <pre>sp_drop_resource_limit { name, appname }   [, rangename, limittype, enforced, action, scope]</pre>
sp_drop_time_range	Adaptive Server からユーザ定義の時間範囲を削除します。 <pre>sp_drop_time_range name</pre>
sp_dropdevice	Adaptive Server のデータベース・デバイスまたはダンプ・デバイスを削除します。 <pre>sp_dropdevice logicalname</pre>
sp_dropengine	指定したエンジン・グループからエンジンを削除します。そのエンジンがグループ内の最後のエンジンである場合は、エンジン・グループを削除します。 <pre>sp_dropengine engine_number [, engine_group] [, instance_id]</pre>
sp_dropexeclass	ユーザ定義の実行クラスを削除します。 <pre>sp_dropexeclass classname</pre>
sp_dropextendedproc	ESP ( 拡張ストアド・プロシージャ ) を削除します。 <pre>sp_dropextendedproc esp_name</pre>
sp_dropexternlogin	コンポーネント統合サービスのみ - sp_addexternlogin によってあらかじめ定義されているリモート・ログインの定義を削除します。 <pre>sp_dropexternlogin server [, loginname [, rolename ]]</pre>
sp_dropglockpromote	テーブルまたはデータベースからのロック・プロモーション値を削除します。 <pre>sp_dropglockpromote {"database"   "table"}, objname</pre>

sp_dropgroup	データベースからグループを削除します。 sp_dropgroup <i>grpname</i>
sp_dropkey	sp_primarykey、sp_foreignkey、または sp_commonkey を使用して定義されていたキーを syskeys テーブルから削除します。 sp_dropkey <i>keytype, tablename</i> [, <i>deptabname</i> ]
sp_droplanguage	サーバから代替言語を削除し、そのローを master.dbo.syslanguages から削除します。 sp_droplanguage <i>language</i> [, <i>dropmessages</i> ]
sp_droplogin	master.dbo.syslogins 内のユーザ・エントリを削除することによって、Adaptive Server のユーザ・ログインを削除します。 sp_droplogin <i>loginame</i>
sp_dropmessage	ユーザ定義メッセージを sysusermessages から削除します。 sp_dropmessage <i>message_num</i> [, <i>language</i> ]
sp_dropobjectdef	コンポーネント統合サービスのみ – ローカル・オブジェクトに関する外部の記憶マッピングを削除します。 sp_dropobjectdef <i>tablename</i>
sp_droptellogin	リモート・ユーザ・ログインを削除します。 sp_droptellogin <i>remoteserver</i> [, <i>loginame</i> [, <i>remotename</i> ]]
sp_droprowlockpromote	データベースまたはテーブルから、ロー・ロック・プロモーション・スレッシュホールド値を削除します。 sp_droprowlockpromote { "database"   "table" }, <i>objname</i>
sp_dropsegment	データベースからセグメントを削除するか、特定のデータベース・デバイスからセグメントのマッピングを解除します。 sp_dropsegment <i>segname, dbname</i> [, <i>device</i> ]
sp_dropserver	登録されているサーバのリストからサーバを削除するか、リモート・ログインと外部ログインを同一の操作で削除します。 sp_dropserver <i>server</i> [, <i>droplogins</i> ]
sp_droptreshold	セグメントから空き領域スレッシュホールドを削除します。 sp_droptreshold <i>dbname, segname, free_space</i>
sp_droptype	ユーザ定義データ型を削除します。 sp_droptype <i>typename</i>
sp_dropuser	現在のデータベースからユーザを削除します。 sp_dropuser <i>name_in_db</i>
sp_dumpoptimize	データベースのダンプ・オペレーション中に Backup Server によってダンプされるデータの量を指定します。 sp_dumpoptimize [ 'archive_space = {maximum   minimum   default}' ] sp_dumpoptimize [ 'reserved_threshold = {nnn   default}' ] sp_dumpoptimize [ 'allocation_threshold = {nnn   default}' ]

---

sp_encryption	<p>暗号化の情報をレポートします。</p> <pre> sp_encryption help   helpkey sp_encryption help   helpkey [, key_name   wildcard] [, all_dbs   key_copy   display_cols] sp_encryption help   helpkey [, system_encr_passwd] [, display_keys] sp_encryption helpcol [, table_name   column_name ] sp_encryption helpuser [, user_name   wildcard ][, key_copy] sp_encryption system_encr_passwd, 'newpasswd' [, 'oldpasswd'] </pre>
sp_engine	<p>エンジンをオンラインまたはオフラインにできるようにします。</p> <pre> sp_engine {"online"   [offline   can_offline] [, engine_id]   ["shutdown", engine_id]} </pre>
sp_estspace	<p>テーブルとそのインデックスに必要な空き領域と、インデックスの作成に必要な時間を見積もります。</p> <pre> sp_estspace table_name, no_of_rows, fill_factor, cols_to_max, textbin_len, iosec, page_size </pre>
sp_export_qpgroup	<p>指定したユーザのすべてのプランと抽象プラン・グループをユーザ・テーブルにエクスポートします。</p> <pre> sp_export_qpgroup usr, group, tab </pre>
sp_extendsegment	<p>セグメントの範囲を別のデータベース・デバイスに拡張します。</p> <pre> sp_extendsegment segname, dbname, devname </pre>
sp_extengine	<p>EJB サーバを起動したり停止したりします。また、EJB サーバのステータス情報も表示します。</p> <pre> sp_extengine 'ejb_server', '{ start   stop   status }' </pre>
sp_extrapwdchecks	<p>パスワード複雑性チェックのユーザ定義ロジックを含んでいます。</p> <pre> sp_extrapwdchecks caller_password, new_password, login_name </pre>
sp_familylock	<p>並列に文を実行するファミリーが保持しているすべてのロックに関する情報をレポートします。</p> <pre> sp_familylock [fpid1 [, fpid2]] </pre>
sp_find_qplan	<p>クエリ・テキストまたはプラン・テキストから指定されたパターンによって、抽象プランを検索します。</p> <pre> sp_find_qplan pattern [, group ] </pre>
sp_fixindex	<p>システム・テーブルの矛盾したインデックスを修復します。</p> <pre> sp_fixindex database_name, table_name [, index_id   null] [, index_name   null] [, force_option] </pre>
sp_flushstats	<p>メモリ内記憶領域の統計を <b>sysabstats</b> システム・テーブルと <b>sysstatistics</b> システム・テーブルにフラッシュします。</p> <pre> sp_flushstats [objname] </pre>

sp_forceonline_db	以前リカバリを実行したときに疑わしいページとして suspect マークが付けられた、データベース内に格納されているすべてのページにアクセスできるようにします。 <code>sp_forceonline_db dbname, {"sa_on"   "sa_off"   "all_users"}</code>
sp_forceonline_object	以前リカバリを実行したときに疑わしいインデックスとして suspect マークが付けられたインデックスにアクセスできるようにします。 <code>sp_forceonline_object dbname, objname, indid, {sa_on   sa_off   all_users} [, no_print]</code>
sp_forceonline_page	リカバリによって疑わしいページとして suspect のマークが付けられたページにアクセスできるようにします。 <code>sp_forceonline_page dbname, pgid, {"sa_on"   "sa_off"   "all_users"}</code>
sp_foreignkey	現在のデータベース内にあるテーブルまたはビューの外部キーを定義します。 <code>sp_foreignkey tablename, pktablename, col1 [, col2] ...[, col8]</code>
sp_freedll	拡張ストアド・プロシージャの実行をサポートするために XP Server メモリ内に事前にロードされている DLL をアンロードします。 <code>sp_freedll dll_name</code>
sp_getmessage	print 文と raiserror 文で使用するために、sysmessages と sysusermessages から、格納されているメッセージ文字列を検索します。 <code>sp_getmessage message_num, result output [, language]</code>
sp_grantlogin	Windows のみ - 統合セキュリティ・モードまたは混合モード ( 名前付きパイプ使用 ) がアクティブなとき、Windows のユーザとグループに、Adaptive Server の役割または default パーミッションを割り当てます。 <code>sp_grantlogin {login_name   group_name} ["role_list"   default]</code>
sp_ha_admin	高可用性システムで設定された Adaptive Server で管理タスクを実行します。 <code>sp_ha_adminin [cleansessions   help]</code>
sp_help	データベース・オブジェクト、システム定義データ型またはユーザ定義データ型、ユーザ定義関数、計算カラムと関数ベースのインデックスに関する情報をレポートします。 <code>sp_help [objname]</code>
sp_help_resource_limit	リソース制限についてレポートします。 <code>sp_help_resource_limit [name [, appname [, limittime [, limitday [, scope [, action[, verbose]]]]]]]</code>
sp_help_qpgroup	抽象プラン・グループの情報をレポートします。 <code>sp_help_qpgroup [ group [, mode ]]</code>
sp_help_qplan	抽象プランについての情報をレポートします。 <code>sp_help_qplan id [, mode ]</code>
sp_helpapptrace	Adaptive Server がトレースしているセッションを判断します。 <code>sp_helpapptrace</code>

---

sp_helppartition	<p>テーブルまたはインデックスのパーティション関連情報をリストします。</p> <p><code>sp_helppartition [ <i>tablename</i> [, { null   <i>indexname</i>   'all' }], <i>partitionname</i> ] ] ]</code></p>
sp_helpcache	<p>データ・キャッシュにバインドされたオブジェクトについての情報、または指定されたキャッシュ・サイズに必要なオーバーヘッド量を表示します。</p> <p><code>sp_helpcache {<i>cache_name</i>   "cache_size[P   K   M   G]", 'instance <i>instance_name</i>'}</code></p>
sp_helpcomputedcolumn	<p>指定したテーブルの計算カラムに関する情報をレポートします。</p> <p><code>sp_helpcomputedcolumn (<i>tablename</i>)</code></p>
sp_helpconfig	<p>設定パラメータに関するヘルプ情報をレポートします。</p> <p><code>sp_helpconfig "<i>configname</i>", "<i>size</i>"</code></p>
sp_helpconstraint	<p>指定したテーブルで使用される整合性制約に関する情報をレポートします。</p> <p><code>sp_helpconstraint [<i>objname</i>][, <i>detail</i>]</code></p>
sp_helpdb	<p>特定のデータベースまたはすべてのデータベースについての情報をレポートします。</p> <p><code>sp_helpdb [<i>dbname</i> [, <i>order</i>]]</code></p>
sp_helpdevice	<p>特定のデバイスまたはすべての Adaptive Server のデータベース・デバイスとダンプ・デバイスについての情報をレポートします。</p> <p><code>sp_helpdevice [<i>devname</i>]</code></p>
sp_helpextendedproc	<p>現在のデータベース内の拡張ストアド・プロシージャと、それに対応した DLL ファイルを表示します。</p> <p><code>sp_helpextendedproc [<i>esp_name</i>]</code></p>
sp_helpexternlogin	<p>コンポーネント統合サービスのみ - 外部ログイン名についての情報をレポートします。</p> <p><code>sp_helpexternlogin [<i>server</i>[, <i>loginame</i>[, <i>rolename</i>]]]</code></p>
sp_helpgroup	<p>特定のグループまたは現在のデータベース内のすべてのグループに関する情報をレポートします。</p> <p><code>sp_helpgroup [<i>grpname</i>]</code></p>
sp_helpindex	<p>テーブルに作成されたインデックス、計算カラムのインデックス、関数ベースのインデックスに関する情報をレポートします。</p> <p><code>sp_helpindex <i>objname</i></code></p>
sp_helpjava	<p>データベースにインストールされている Java クラスと関連する JAR についての情報を表示します。</p> <p><code>sp_helpjava ["class"[, <i>java_class_name</i>[, "detail"   "depends"]]   "jar", <i>jar_name</i>[, "depends"]]</code></p>
sp_helpjoins	<p>2つのテーブルまたはビューから、互いをジョインする場合に効果的に使用できる可能性のあるカラムをリストします。</p> <p><code>sp_helpjoins <i>lefttab</i>, <i>righttab</i></code></p>

sp_helpkey	特定のテーブルまたはビューのプライマリ・キー、外部キー、共通キーに関する情報をレポートします。または現在のデータベース内にあるすべてのキーに関する情報をレポートします。 <code>sp_helpkey [tablename]</code>
sp_helplanguage	特定の代替言語またはすべての言語についての情報をレポートします。 <code>sp_helplanguage [language]</code>
sp_helplog	トランザクション・ログの最初のページが入っているデバイスの名前をレポートします。 <code>sp_helplog</code>
sp_helpobjectdef	コンポーネント統合サービスのみ – リモート・オブジェクト定義の所有者、オブジェクト、タイプについての情報をレポートします。 <code>sp_helpobjectdef [objname]</code>
sp_helpremotelogin	特定のリモート・サーバのログイン、またはすべてのリモート・サーバのログインについての情報をレポートします。 <code>sp_helpremotelogin [remoteserver [, remotename]]</code>
sp_helpprotect	データベース・オブジェクト、ユーザ、グループ、または役割のパーミッションについてレポートします。 <code>sp_helpprotect [name[, username[, "grant" ["none" "granted" "enabled" "role_name[, permission_name]]]]</code>
sp_helpsegment	特定のセグメントまたは現在のデータベース内のすべてのセグメントについての情報をレポートします。 <code>sp_helpsegment [segname]</code>
sp_helpserver	特定のリモート・サーバ、またはすべてのリモート・サーバについての情報をレポートします。 <code>sp_helpserver [server]</code>
sp_helpsort	Adaptive Server のデフォルトのソート順と文字セットを表示します。 <code>sp_helpsort</code>
sp_helptext	コンパイル済みオブジェクトのソース・テキストを表示します。また、ユーザ定義関数、計算カラム、または関数ベースのインデックス定義のテキストも表示します。 <code>sp_helptext objname[, grouping_num][, numlines[, printopts]]</code>
sp_helpthreshold	現在のデータベース内のすべてのスレッシュホールド、または特定のセグメントのすべてのスレッシュホールドに対応するセグメント、空き領域値、ステータス、ストア・プロシージャをレポートします。 <code>sp_helpthreshold [segname]</code>
sp_helpuser	現在のデータベース内の特定のユーザ、グループ、エイリアスに関する情報や、すべてのユーザに関する情報をレポートします。 <code>sp_helpuser [name_in_db]</code>

---

sp_hidetext	<p>指定したコンパイル済みオブジェクトのソース・テキストを隠します。また、計算カラムのテキストと関数ベースのインデックス・キーも隠します。</p> <pre>sp_hidetext [objname[, tablename[, username]]]</pre>
sp_import_qpgroup	<p>抽象プランをユーザ・テーブルから抽象プラン・グループにインポートします。</p> <pre>sp_import_qpgroup tab, usr, group</pre>
sp_indsuspect	<p>ソート順変更後のリカバリ時に、suspect (疑わしい) としてマーク付けされたインデックスがないか、ユーザ・テーブルを検査します。</p> <pre>sp_indsuspect [tab_name]</pre>
sp_jreconfig	<p>Java PCA/JVM を管理します。引数やディレクティブを有効または無効にしたり、設定値を変更したり、設定値をレポートしたりします。</p> <pre>sp_jreconfig {   add array_arg, new_string     array_clear array_arg     array_enable array_arg     array_disable array_arg     delete array_arg, string_value     disable { directive   argument   array_arg, string_value }     enable { directive   argument   array_arg, string_value }     list { list_type [, formatted ]   units   units, units_type[, formatted ] }     reload_config     report { directive[, formatted ]   directive, args[, formatted ]            argument[, formatted ] }     update { argument, old_value, new_value } }</pre>
sp_ldapadmin	<p>LDAP URL 検索文字列を作成またはリストします。LDAP URL 検索文字列またはログインを確認します。アクセス・アカウントとチューニング可能な LDAPUA 関連のパラメータを指定します。</p> <pre>sp_ldapadmin command [, option1 [, option2]]</pre> <p>有効な <i>command</i> [, <i>option1</i> [, <i>option2</i>]] オプションは、次のとおりです。</p> <pre>'set_primary_url', 'url' 'set_secondary_url', 'url' 'set_dn_lookup_url', 'url' 'set_secondary_dn_lookup_url', 'url' 'set_access_acct', 'distinguished_name', 'password' 'set_secondary_access_acct', 'distinguished_name', 'password' 'set_failback_interval', time_in_minutes 'suspend', {'primary'   'secondary'} 'activate', {'primary'   'secondary'} 'list' 'list_urls' 'list_access_acct' 'check_url', 'url' 'reinit_descriptors' 'check_login', 'name' 'set_timeout', timeout_in_milli_seconds 'set_log_interval', log_interval_in_minutes 'set_num_retries', num_retries 'set_max_ldapua_native_threads', max_ldapua_native_threads 'set_max_ldapua_desc', max_ldapua_desc 'set_abandon_ldapua_when_full', {true false} 'starttls_on_primary', {true false} 'starttls_on_secondary', {true false} 'help'</pre>

sp_listener	<p>任意のポートでの Adaptive Server のリスナを、エンジン単位で動的に起動したり停止したりします。</p> <pre>sp_listener "command", "server_name   network", engine   remaining</pre> <p>または</p> <pre>sp_listener "command", "[protocol:]machine:port", engine</pre>
sp_listsuspect_db	<p>リカバリ時に検出された破壊が原因で現在オフラインになっているページがある、すべてのデータベースをリストします。</p> <pre>sp_listsuspect_db</pre>
sp_listsuspect_object	<p>リカバリ時に検出された破壊が原因で現在オフラインになっている、データベース内のすべてのインデックスをリストします。</p> <pre>sp_listsuspect_object [dbname]</pre>
sp_listsuspect_page	<p>リカバリ時に検出された破壊が原因で現在オフラインになっている、データベース内のすべてのページをリストします。</p> <pre>sp_listsuspect_page [dbname]</pre>
sp_lmconfig	<p>Adaptive Server のライセンス管理に関連する情報を設定します。</p> <pre>sp_lmconfig [ 'edition' [ , edition_type ] [ , 'license type' [ , license_type_name ] [ , 'smtp host' [ , smtp_host_name ] [ , 'smtp port' [ , smtp_port_number ] [ , 'email sender' [ , sender_email_address ] [ , 'email recipients' [ , email_recipients ]]] [ , 'email severity' [ , email_severity ]]</pre>
sp_lock	<p>現在ロックを保持しているプロセスのオブジェクト名と ID をレポートします。</p> <pre>sp_lock [spid1[, spid2]]   [@verbose = int]</pre>
sp_locklogin	<p>ユーザがログインできないように Adaptive Server アカウントをロックするか、またはロックされたすべてのアカウントのリストを表示します。</p> <pre>sp_locklogin login   NULL   wildcard_string , "lock"   "unlock", [except_login_name   except_role_name] [, number_of_inactive_days]</pre> <ul style="list-style-type: none"> <li>• または sp_locklogin</li> </ul>
sp_logdevice	<p>ログおよびデータと同じデバイス上にあるデータベースのトランザクション・ログを、別のデータベース・デバイスに移動します。</p> <pre>sp_logdevice dbname, devname</pre>
sp_loginconfig	<p>Windows NT のみ – 1 つまたはすべての統合セキュリティ・パラメータの値を表示します。</p> <pre>sp_loginconfig ["parameter_name"]</pre>
sp_logininfo	<p>Windows NT のみ – sp_grantlogin によって Windows NT のユーザとグループに付与されているすべての役割を表示します。</p> <pre>sp_logininfo ["login_name"   "group_name"]</pre>

---

sp_logiosize	現在のデータベースのトランザクション・ログに対して I/O を行っているときに、Adaptive Server が使用するログ I/O サイズを、異なるメモリ・プールに変更します。  sp_logiosize ["default"   "size"   "all"]
sp_logintrigger	グローバル・ログイン・トリガを設定または表示します。  sp_logintrigger <global login trigger name>
sp_maplogin	外部ユーザを Adaptive Server のログインにマップします。  sp_maplogin (authentication_mech   null), (client_username   null), (action   login_name   null)
sp_metrics	クエリに関する QP 測定基準とそれらの統計をバックアップ、削除、フラッシュします。  sp_metrics ['backup' backup_group_ID   'drop', 'gid' [, 'id']   'flush'   'help', 'command']
sp_modify_resource_limit	リソース制限を変更します。この変更は、新しい制限値を指定するか、または制限値を超えた場合の動作を指定する ( またはその両方を指定する ) ことにより行われます。  sp_modify_resource_limit {name, appname} rangename, limittype, limitvalue, enforced, action, scope
sp_modify_time_range	名前付き時間範囲に対応する開始日、開始時刻、終了日、または終了時刻を変更します。  sp_modify_time_range name, startday, endday, starttime, endtime
sp_modifylogin	指定の Adaptive Server ログイン・アカウントについて、デフォルト・データベース、デフォルト言語、デフォルト役割のアクティブ化、ログイン・スクリプト、フルネーム、パスワード有効期間、最小パスワード長、許容される最大ログイン失敗回数を変更します。  sp_modifylogin {loginame   "all overrides"}, option, value
sp_modifystats	システム管理者や、このプロシージャを実行してターゲット・テーブルの統計を更新するパーミッションを付与されたすべてのユーザは、sysstatistics のカラムの密度値を変更できます。  sp_modifystats [database].[owner].table_name, { "column_group"   "all" }, MODIFY_DENSITY, { range   total }, { absolute   factor }, "value"  • または sp_modifystats [database].[owner].table_name, column_name, REMOVE_SKEW_FROM_DENSITY
sp_modifythreshold	別のスレッシュホールド・プロシージャ、空き領域レベル、またはセグメント名と対応させることによって、スレッシュホールドを変更します。  sp_modifythreshold dbname, segname, free_space [, new_proc_name][, new_free_space][, new_segname]

sp_monitor	<p>Adaptive Server についての統計情報を表示します。</p> <pre>sp_monitor [[connection   statement], [cpu   diskio   elapsed time]]            [event, [spid]]            [procedure, [dbname, [procname[, summary   detail]]]            [enable] [disable]            [help],            [deadlock][procstack]</pre>
sp_monitorconfig	<p>インデックス、オブジェクト、データベース用のメタデータ記述子に関するキャッシュ使用率の統計を表示します。</p> <pre>sp_monitorconfig "configname"[, "result_tbl_name"][, "full"]</pre>
sp_object_stats	<p>ロック競合、ロック待機時間、およびテーブルとインデックスのデッドロック統計を表示します。</p> <pre>sp_object_stats interval[, top_n[, dbname, objname[, rpt_option]]]</pre>
sp_options	<p>オプションの値を表示します。</p> <pre>sp_options [ [show   help              [, option_name   category_name   null              [, dflt   non_dflt   null [, spid]]]]]</pre>
sp_passthru	<p>コンポーネント統合サービスのみ - ユーザがリモート・サーバに SQL コマンド・バッファを渡せるようにします。</p> <pre>sp_passthru server, command, errcode, errmsg, rowcount            [, arg1, arg2, ... argn]</pre>
sp_password	<p>Adaptive Server のログイン・アカウントのパスワードを追加または変更します。</p> <pre>sp_password caller_passwd, new_passwd [, loginame, immediate]</pre>
sp_passwordpolicy	<p>sso_role を持つユーザがログインとパスワード・ポリシーのオプションを設定するときを使用できるインタフェース。</p> <p>新しいパスワードの複雑性オプションを指定、削除、リスト表示する場合：</p> <pre>sp_passwordpolicy {"set"   "clear"   "list"}, policy_option, option_value</pre> <p>パスワードの複雑性オプションを確認する場合：</p> <pre>sp_passwordpolicy 'validate password options'</pre> <p>ネットワーク・ログイン・パスワードを暗号化するための非対称キー・ペアを生成する場合：</p> <pre>sp_passwordpolicy "regenerate keypair"</pre> <p>パスワードを期限切れにする場合：</p> <pre>sp_passwordpolicy "expire role passwords", "[rolename   wildcard]" sp_passwordpolicy "expire login passwords", "[login_name   wildcard]" sp_passwordpolicy "expire stale login passwords", "datetime" sp_passwordpolicy "expire stale login passwords", "datetime"</pre>

---

sp_pciconfig	<p>Java PCI Bridge を管理します。引数やディレクティブを有効または無効にしたり、設定値を変更したり、設定値をレポートしたりします。</p> <pre> sp_pciconfig {     disable { directive   argument }       enable { directive   argument }       list { list_type [, formatted ]   units   units, units_type[, formatted ] }       report { directive[, formatted ]           directive, args[, formatted ]           argument[, formatted ] }       update { number_arg, old_value new_value } } </pre>
sp_placeobject	<p>テーブルまたはインデックスに対する今後の領域の割り当てを、特定のセグメントで行います。</p> <pre> sp_placeobject segname, objname </pre>
sp_plan_dbccdb	<p>新しく作成する dbccdb と dbccalt データベースのサイズの推奨値、dbccdb と dbccalt の適切なデバイスのリスト、ターゲット・データベースのキャッシュ・サイズと適切なワーカー・プロセス数を提示します。</p> <pre> sp_plan_dbccdb [dbname] </pre>
sp_poolconfig	<p>データ・キャッシュ内のメモリ・プールを作成、削除、サイズ変更し、情報を表示します。既存のキャッシュ内にメモリ・プールを作成したり、プール・サイズを変更する場合：</p> <pre> sp_poolconfig cache_name[, "mem_size [P   K   M   G]", "config_poolK"     [, "affected_pool K", instance instance_name] </pre> <p>プールのウォッシュ・サイズを変更する場合：</p> <pre> sp_poolconfig cache_name, "affected_poolK", "wash=size[P K M G]" </pre> <p>プールの非同期プリフェッチ率を変更する場合：</p> <pre> sp_poolconfig cache_name, "affected_poolK",     "local async prefetch limit=percent " </pre>
sp_post_xpload	<p>エンディアン・タイプの異なるプラットフォーム間で load database を実行した後、インデックスをチェックして再構築します。</p> <pre> sp_post_xpload </pre>
sp_primarykey	<p>テーブルまたはビューのプライマリ・キーを定義します。</p> <pre> sp_primarykey tablename, col1 [, col2, col3, ..., col8] </pre>
sp_processmail	<p>Windows NT のみ - Adaptive Server のメッセージ・インボックスにあるメッセージの読み込み、処理、送信、削除を実行します。</p> <pre> sp_processmail [subject] [, originator [, dbuser     [, dbname [, filetype [, separator]]]] </pre>
sp_procxmode	<p>ストアド・プロシージャに関連付けられている実行モードを表示または変更します。</p> <pre> sp_procxmode [procname [, tranmode]] </pre>

sp_querysmobj	<p><i>Tivoli Storage Manager</i> のみ – Adaptive Server バックアップ・オブジェクトのリストについて <i>Tivoli Storage Manager (TSM)</i> に問い合わせます。</p> <pre>sp_querysmobj "syb_tsm", "output_file", "server_name"               {, "database_name", "object_name", "dump_type",                 "until_time", "bs_name"}</pre>
sp_recompile	<p>指定したテーブルを使用するすべてのストアド・プロシージャとトリガが、次の実行時に再コンパイルされるように設定します。</p> <pre>sp_recompile objname</pre>
sp_refit_admin	<p>クラスタ環境の場合 – disk refit プロセスの現在のステータスの表示、disk refit プロセスのステータスの再設定、インスタンスの disk refit プロセスのスキップなど、さまざまな disk refit 関連アクションを実行するインタフェースを提供します。</p> <pre>sp_refit_admin ['help']   'status'   ['reset'   'skiprefit' [, instance_name]]                [  'removedevice', device_name]</pre>
sp_remap	<p>4.8 以降 10.0 の前までのリリースのストアド・プロシージャ、トリガ、ルール、デフォルト、ビューを再マップし、10.0 以降のリリースとの互換性を持たせます。</p> <pre>sp_remap objname</pre>
sp_remotoption	<p>リモート・ログイン・オプションを表示または変更します。</p> <pre>sp_remotoption [remoteserver[, loginname                 [, remotename[, optname[, optvalue]]]]]</pre>
sp_remotesql	<p>コンポーネント統合サービスのみ – リモート・サーバへの接続を確立し、クライアントからのクエリ・バッファをリモート・サーバに渡し、結果をクライアントに伝えます。</p> <pre>sp_remotesql server, query[, query2, ..., query254]</pre>
sp_rename	<p>現在のデータベース内のユーザ作成オブジェクトまたはユーザ定義データ型の名前を変更します。</p> <pre>sp_rename objname, newname [, "index"   "column"]</pre>
sp_rename_qpgroup	<p>抽象プラン・グループの名前を変更します。</p> <pre>sp_rename_qpgroup old_name, new_name</pre>
sp_renamedb	<p>ユーザ・データベースの名前を変更します。</p> <pre>sp_renamedb dbname, newname</pre>
sp_reportstats	<p>システム使用状況についての統計をレポートします。</p> <pre>sp_reportstats [loginame]</pre>
sp_revokelogin	<p><i>Windows NT</i> のみ – 統合セキュリティ・モードまたは混合モード (名前付きパイプ使用) がアクティブなとき、<i>Windows NT</i> のユーザとグループから Adaptive Server の役割とデフォルトのパーミッションを取り消します。</p> <pre>sp_revokelogin {login_name   group_name}</pre>
sp_role	<p>Adaptive Server ログイン・アカウントに対する役割を付与または取り消します。</p> <pre>sp_role {"grant"   "revoke"}, rolename, loginame</pre>

---

sp_sendmsg	UDP (User Datagram Protocol) ポートにメッセージを送信します。 sp_sendmsg <i>ip_address, port_number, message</i>
sp_serveroption	リモート・サーバ・オプションを表示または変更します。 sp_serveroption [ <i>server, optname, optvalue</i> ]
sp_set_qplan	対応するクエリを変更しないで、既存プランの抽象プランのテキストを変更します。 sp_set_qplan <i>id, plan</i>
sp_setlangalias	代替言語のエイリアスの割り当てまたは変更を行います。 sp_setlangalias <i>language, alias</i>
sp_setpglockpromote	データベース、テーブル、または Adaptive Server に対するロック・プロモーション・スレッシュホールドを設定または変更します。 sp_setpglockpromote {"database"   "table"}, <i>objname, new_lwm, new_hwm, new_pct</i> sp_setpglockpromote <i>server, NULL, new_lwm, new_hwm, new_pct</i>
sp_setpsex	セッションがアクティブなときに、セッションのカスタム実行属性を設定します。 sp_setpsex <i>spid, exeattr, value</i>
sp_setrowlockpromote	データロー・ロックされたテーブル、データベース内のデータロー・ロックされたすべてのテーブル、またはサーバ上のデータロー・ロックされたすべてのテーブルに使用される、ローロック・プロモーション・スレッシュホールドを設定または変更します。 sp_setrowlockpromote "server", NULL, <i>new_lwm, new_hwm, new_pct</i> sp_setrowlockpromote {"database"   "table"}, <i>objname, new_lwm, new_hwm, new_pct</i>
sp_setsuspect_granularity	ユーザ・データベースのリカバリ・フォールト・アイソレーション・モードを表示または設定します。このモードは、リカバリによってデータの破損が検出された場合に、どう処置するかを決定します。 sp_setsuspect_granularity [ <i>dbname</i> [, "database"   "page" [, "read_only"]]]
sp_setsuspect_threshold	データベース全体に疑わしいことを示す suspect マークを付ける前に、データベース内で Adaptive Server が許容できる疑わしいページの最大値を表示または設定します。 sp_setsuspect_threshold [ <i>dbname</i> [, <i>threshold</i> ]]
sp_setup_table_transfer	このデータベース内に spt_TableTransfer テーブルを作成するために、増分転送のマークが付いたテーブルを含んでいる各データベースで 1 回ずつ実行します。 sp_setup_table_transfer
sp_show_options	現在のセッションで設定されたすべてのサーバ・オプションを出力します。 sp_show_options

sp_showcontrolinfo	<p>エンジン・グループの割り当て、バインドされているクライアント・アプリケーション、ログイン、ストアド・プロシージャについての情報を表示します。</p> <p>sp_showcontrolinfo [object_type, object_name, spid]</p>
sp_showexeclass	<p>特定の実行クラスに対応する任意のエンジン・グループ内のエンジンと、実行クラス属性を表示します。</p> <p>sp_showexeclass [execlassname]</p>
sp_showplan	<p>現在の SQL 文または同じバッチの前の文について、すべてのユーザ接続の showplan 出力を表示します。</p> <p>sp_showplan spid, batch_id output, context_id output, stmt_num output</p> <ul style="list-style-type: none"> <li>• batch_id、context_id、または stmt_num を指定しないで、現在の SQL 文の showplan 出力を表示します。</li> </ul> <p>sp_showplan spid, null, null, null</p>
sp_showpsexexe	<p>Adaptive Server で実行されているすべてのクライアント・セッションの実行クラス、現在の優先度、結び付きを表示します。</p> <p>sp_showpsexexe [spid]</p>
sp_spaceusage	<p>テーブル、インデックス、トランザクション・ログの領域使用率に関するレポートを作成したり、データベースのテーブルおよびインデックスの断片化量の推定値を計算したりします。</p> <p>“help” アクションの構文は次のとおりです。</p> <pre>sp_spaceusage 'help'[, 'all'] sp_spaceusage 'help' [, {'display'   'display summary'                             'report'   'report summary'   'archive'}                     [, {'table'   'index'   'tranlog'}]]</pre> <p>display アクションの構文は次のとおりです。</p> <pre>sp_spaceusage 'display summary [using unit= {KB   MB   GB   PAGES} ]', {'table'   'index'}, name [,where_clause [,order_by[,command ] ] ] sp_spaceusage 'display [using unit={KB   MB   GB   PAGES} ]', {'table'   'index'}, name [,select_list [,where_clause [,order_by[,command ] ] ] ] sp_spaceusage 'display [using unit={KB   MB   GB   PAGES} ]', 'tranlog' [, name[,select_list[,where_clause [,order_by]]]]</pre> <p>“archive” アクションの構文は次のとおりです。</p> <pre>sp_spaceusage 'archive [ using_clause ]', {'table'   'index'}, name[,where_clause[,command ] ] sp_spaceusage 'archive [ using_clause ]', 'tranlog' [,name[,where_clause ] ]</pre> <p>“report” アクションの構文は次のとおりです。</p> <pre>sp_spaceusage 'report summary [ using_clause ]', {'table'   'index'}, name [,where_clause [,order_by[,from_date [,to_date]]]] sp_spaceusage 'report [ using_clause ]', {'table'   'index'}, name [,select_list[,where_clause [,order_by[,from_date [,to_date]]]]]</pre>

	<pre> sp_spaceusage 'report' [ using_clause ],     'tranlog' [, name     [,select_list[,where_clause [,order_by     [,from_date [,to_date]]]]]     using_clause = USING using_item [, using_item ...]     using_item = { unit={ KB   MB   GB   PAGES }       dbname=database_name   prefix=string } </pre>
sp_spaceused	<p>ロー数、データ・ページ数、インデックスのサイズ、指定のテーブルが使用する領域、現在のデータベース内のすべてのテーブルが使用する領域の概算を表示します。</p> <pre> sp_spaceused [objname [,1]] </pre>
sp_ssladmin	<p>Adaptive Server に対するサーバ認証のリストを追加、削除、表示します。</p> <pre> sp_ssladmin {[addcert, certificate_path [, password   NULL]]     [dropcert, certificate_path] [lscert] [help]} [lsciphers] [setciphers,     {"FIPS"   "Strong"   "Weak"   "All"   quoted_list_of_ciphersuites}] </pre>
sp_syntax	<p>使用しているサーバ上の製品とそれに対応する <code>sp_syntax</code> スクリプトに応じて、Transact-SQL 文、システム・プロシージャ、ユーティリティ、Adaptive Server のその他のルーチンの構文を表示します。</p> <pre> sp_syntax word [, mod][, language] </pre>
sp_sysmon	<p>パフォーマンス情報を表示します。</p> <pre> sp_sysmon begin_sample sp_sysmon { end_sample   interval }[, section[, applmon]]     [, 'cache wizard' [, top_N [, filter]]] </pre>
sp_tab_suspectptn	<p>suspect (疑わしい) パーティションを含んでいるテーブルをリストします。文字ベースのパーティション・キーを使用する範囲分割されたテーブルは、ソート順を変更すると suspect になる可能性があり、ハッシュ分割されたテーブルは、プラットフォーム間のダンプとロードを行うと suspect になる可能性があります。</p> <pre> sp_tab_suspectptn [table_name] </pre>
sp_tempdb	<p>複数のテンポラリ・データベースに関連するバインドを <code>sysattributes</code> で管理するためのバインド・インタフェースを提供します。</p> <pre> sp_tempdb [     [{"create"   "drop"}, "groupname"]       [{"add"   "remove"}, "tempdbname", "groupname"]       [{"bind", "objtype", "objname", "bindtype", "bindobj"         [, scope · hardness · ]}           {"unbind", "objtype", "objname" [, "scope" ] "instance_name"}]       [{"unbindall_db", "tempdbname"}]       [show [, "all"   "gr"   "db"   "login"   "app" [, "name" ]]]       [who, "dbname"]     [help]] </pre>
sp_tempdb_markdrop	<p>クラスタ環境の場合 - ローカル・システム・テンポラリ・データベースを削除状態にします。</p> <pre> sp_tempdb_markdrop database_name [, {'mark'   'unmark'}] </pre>

sp_thresholdaction	<p>ログ・セグメント上の空きページの数が増え、ラストチャンス・スレッシュホールドを下回る場合、このスレッシュホールドが別のプロシージャと対応していないかぎり、自動的に実行されます。Sybaseはこのプロシージャを提供していません。</p> <p><code>sp_thresholdaction @dbname, @segment_name, @space_left, @status</code></p>
sp_tran_dumpable_status	<p>データベースでトランザクションをダンプできない場合、ダンプできない理由が <code>sp_tran_dumpable_status</code> によって表示されます。</p> <p><code>sp_tran_dumpable_status [database_name]</code></p>
sp_transactions	<p>アクティブなトランザクションについての情報をレポートします。</p> <p><code>sp_transactions ["xid", xid_value]  </code>  <code>["state", {"heuristic_commit"   "heuristic_abort"</code>  <code> "prepared"   "indoubt"}] [, "xactname"]   ["gtrid", gtrid_value]</code></p>
sp_unbindcache	<p>データ・キャッシュから、データベース、テーブル、インデックス、text オブジェクト、image オブジェクトのバインドを解除します。</p> <p><code>sp_unbindcache dbname [, [owner.]tablename</code>  <code>[, indexname   "text only"]]</code></p>
sp_unbindcache_all	<p>キャッシュにバインドされているすべてのオブジェクトのバインドを解除します。</p> <p><code>sp_unbindcache_all cache_name</code></p>
sp_unbindefault	<p>作成済みのデフォルト値を、カラムまたはユーザ定義データ型からバインド解除します。</p> <p><code>sp_unbindefault objname [, futureonly]</code></p>
sp_unbindexclass	<p>指定したスコープのクライアント・アプリケーション、ログイン、ストアード・プロシージャにこれまで対応していた実行クラス属性を削除します。</p> <p><code>sp_unbindexclass object_name, object_type, scope</code></p>
sp_unbindmsg	<p>制約からユーザ定義メッセージのバインドを解除します。</p> <p><code>sp_unbindmsg constrname</code></p>
sp_unbindrule	<p>カラムまたはユーザ定義データ型からルールのバインドを解除します。</p> <p><code>sp_unbindrule objname [, futureonly [, "accessrule"   "all"]]</code></p>
sp_version	<p>最後に実行されたインストール・スクリプトのバージョン情報と、そのスクリプトが正常に実行されたかどうかを返します。</p> <p><code>sp_version [script_file, [all]]</code></p>
sp_volchanged	<p>ダンプまたはロード中に、要求されたボリューム処理をオペレータが実行したことを Backup Server に通知します。</p> <p><code>sp_volchanged session_id, devname, action[, fname [, vname]]</code></p>
sp_webservices	<p>Adaptive Server Web Services Engine で使用されるプロキシ・テーブルの作成と管理を行います。</p> <p>プロキシ・テーブルを作成します。</p> <p><code>sp_webservices 'add', 'wsdl_uri' [, sds_name]</code>  <code>[, 'method_name=proxy_table</code>  <code>[, 'method_name=proxy_table '* ' ]</code></p>

---

`sp_webservices` の使用方法を表示します。

```
sp_webservices help [, 'option']
```

WSDL ファイルにマップされているプロキシ・テーブルをリストします。

```
sp_webservices 'list' [, 'wsdl_uri'] [, sds_name]
```

タイムアウト設定を変更します。

```
sp_webservices 'modify', 'wsdl_uri', 'timeout=time'
```

WSDL ファイルにマップされているプロキシ・テーブルを削除します。

```
sp_webservices 'remove', 'wsdl_uri' [, sds_name]
```

ユーザ定義 *Web* サービスのオプション：

ユーザ定義 Web サービスのデータベース・エイリアスを作成します。

```
sp_webservices 'addalias' alias_name , database_name
```

ユーザ定義 Web サービスを展開します。

```
sp_webservices 'deploy', ['all' | 'service_name']
```

ユーザ定義 Web サービスのデータベース・エイリアスを削除します。

```
sp_webservices 'dropalias' alias_name
```

ユーザ定義 Web サービスの WSDL ファイルにマップされているプロキシ・テーブルをリストします。

```
sp_webservices 'listudws' [, 'service_name']
```

ユーザ定義 Web サービスの 1 つまたは複数のデータベース・エイリアスをリストします。

```
sp_webservices 'listalias'
```

ユーザ定義 Web サービスの展開を解除します。

```
sp_webservices 'undeploy', ['all' | 'service_name']
```

`sp_who`

現在のすべての Adaptive Server ユーザとプロセス、または特定のユーザとプロセスの情報についてレポートします。

```
sp_who [loginame | "spid"]
```

## カタログ・ストアド・プロシージャ

以下は、Adaptive Server のカタログ・ストアド・プロシージャの構文と概要です。詳細については、『リファレンス・マニュアル：プロシージャ』を参照してください。

sp_column_privileges	テーブルまたはビュー内の 1 つまたは複数のカラムのパーミッション情報を返します。 <pre>sp_column_privileges table_name [, table_owner                              [, table_qualifier [, column_name]]]</pre>
sp_columns	1 つまたは複数のカラムに保管できるデータ型の情報を返します。 <pre>sp_columns table_name [, table_owner ]            [, table_qualifier] [, column_name]</pre>
sp_databases	Adaptive Server 内にあるデータベースのリストを返します。 <pre>sp_databases</pre>
sp_datatype_info	特定の ODBC データ型またはすべての ODBC データ型の情報を返します。 <pre>sp_datatype_info [data_type]</pre>
sp_fkeys	create table または alter table コマンドによって作成された、現在のデータベース内の外部キー制約の情報を返します。 <pre>sp_fkeys phtable_name [, phtable_owner]          [, phtable_qualifier] [, fhtable_name]          [, fhtable_owner] [, fhtable_qualifier]</pre>
sp_pkeys	create table コマンドまたは alter table コマンドによって 1 つのテーブル用に作成された、プライマリ・キー制約の情報を返します。 <pre>sp_pkeys table_name [, table_owner][, table_qualifier]</pre>
sp_server_info	Adaptive Server の属性名と現在の値のリストを返します。 <pre>sp_server_info [attribute_id]</pre>
sp_special_columns	テーブルまたはビュー内のローを一意に識別する最適カラム・セットを返します。また、トランザクションによってロー内の値が更新されると自動的に値が生成される timestamp カラムのリストを返します。 <pre>sp_special_columns table_name [, table_owner]                   [, table_qualifier] [, col_type]</pre>
sp_sproc_columns	ストアド・プロシージャの入力パラメータとリターン・パラメータについての情報を返します。 <pre>sp_sproc_columns procedure_name [, procedure_owner]                  [, procedure_qualifier] [, column_name]</pre>
sp_statistics	1 つのテーブルのインデックス・リストを返します。 <pre>sp_statistics table_name [, table_owner]               [, table_qualifier] [, index_name] [, is_unique]</pre>
sp_stored_procedures	1 つまたは複数のストアド・プロシージャの情報を返します。 <pre>sp_stored_procedures [sp_name [, sp_owner [, sp_qualifier]]]</pre>

---

sp_table_privileges	テーブルまたはビューのすべてのカラムの権限情報を返します。 sp_table_privileges <i>table_name</i> [, <i>table_owner</i> [, <i>table_qualifier</i> ]]
sp_tables	from 句内に表示できるオブジェクトのリストを返します。 sp_tables [ <i>table_name</i> ] [, <i>table_owner</i> ] [, <i>table_qualifier</i> ] [, <i>table_type</i> ]

## 拡張ストアド・プロシージャ

以下は、Adaptive Server の拡張ストアド・プロシージャの構文と概要です。詳細については、『リファレンス・マニュアル：プロシージャ』を参照してください。

xp_cmdshell	Adaptive Server を実行するホスト・システム上で、ネイティブなオペレーティング・システムのコマンドを実行します。 xp_cmdshell <i>command</i> [, <i>no_output</i> ] [return_status   no_wait]
xp_deletemail	Windows NT のみ – Adaptive Server のメッセージ・インボックスからメッセージを削除します。 xp_deletemail [ <i>msg_id</i> ]
xp_enumgroups	Windows NT のみ – 指定の Windows NT ドメインのグループを表示します。 xp_enumgroups [ <i>domain_name</i> ]
xp_findnextmsg	Windows NT のみ – Adaptive Server のメッセージ・インボックスから次のメッセージの識別子を取得します。 xp_findnextmsg @ <i>msg_id</i> = @ <i>msg_id</i> output [, <i>type</i> ] [, unread_only = {true   false}]
xp_logevent	Windows NT のみ – Adaptive Server 内から Windows NT のイベント・ログにユーザ定義イベントをロギングします。 xp_logevent <i>error_number</i> , <i>message</i> [, <i>type</i> ]
xp_readmail	Windows NT のみ – Adaptive Server のメッセージ・インボックスからメッセージを読み込みます。 xp_readmail [ <i>msg_id</i> ] [, <i>recipients</i> output] [, <i>sender</i> output] [, <i>date_received</i> output] [, <i>subject</i> output] [, <i>cc</i> output] [, <i>message</i> output] [, <i>attachments</i> output] [, suppress_attach = {true   false}] [, peek = {true   false}] [, unread = {true   false}] [, <i>msg_length</i> output] [, <i>bytes_to_skip</i> [output]] [, <i>type</i> [output]]

xp_sendmail	<p><i>Windows NT</i> のみ – 指定した受信者にメッセージを送信します。メッセージはテキストまたは Transact-SQL クエリの結果のどちらかです。</p> <pre>xp_sendmail recipient [, recipient] ... [ , subject] [ , cc_recipient] ... [ , bcc_recipient] ... [ , {query   message}][ , attachname] [ , attach_result = {true   false}] [ , echo_error = {true   false}] [ , include_file [ , include_file] ... ] [ , no_column_header = {true   false}] [ , no_output = {true   false}] [ , width][ , separator][ , dbuser][ , dbname][ , type] [ , include_query = {true   false}]</pre>
xp_startmail	<p><i>Windows NT</i> のみ – Adaptive Server のメール・セッションを開始します。</p> <pre>xp_startmail [mail_user] [ , mail_password]</pre>
xp_stopmail	<p><i>Windows NT</i> のみ – Adaptive Server のメール・セッションを停止します。</p> <pre>xp_stopmail</pre>

## dbcc ストアド・プロシージャ

以下は、Adaptive Server の dbcc ストアド・プロシージャの構文と概要です。詳細については、『リファレンス・マニュアル：プロシージャ』を参照してください。

sp_dbcc_alterws	<p>指定した作業領域のサイズを指定した値に変更して、作業領域を初期化します。</p> <pre>sp_dbcc_alterws dbname, wsname, "wssize[K M]"</pre>
sp_dbcc_configreport	<p>指定したデータベースについて、dbcc checkstorage オペレーションで使用される設定情報を提供するレポートを生成します。</p> <pre>sp_dbcc_configreport [dbname]</pre>
sp_dbcc_createws	<p>指定したセグメントとデータベース上に、指定したタイプとサイズで作業領域を作成します。</p> <pre>sp_dbcc_createws dbname, segname, [wsname], wstype, "wssize[K M]"</pre>
sp_dbcc_deletedb	<p>指定したターゲット・データベースに関するすべての情報を dbccdb から削除します。</p> <pre>sp_dbcc_deletedb [dbname   dbid]</pre>
sp_dbcc_deletehistory	<p>指定した日付と時刻より前にターゲット・データベースについて実行された dbcc checkstorage オペレーションの結果を削除します。</p> <pre>sp_dbcc_deletehistory [cutoffdate] [ , dbname   dbid]</pre>
sp_dbcc_differentialreport	<p>2 つの dbcc オペレーションの間で発生した I/O 統計とフォールトの変化を示すレポートを生成します。</p> <pre>sp_dbcc_differentialreport [dbname [ , objectname]], [db_op] [ , "date1" [ , "date2"]]</pre>

---

sp_dbcc_evaluatedb	ターゲット・データベースの設定情報を再計算して、現在の設定情報と比較します。 sp_dbcc_evaluatedb [dbname]
sp_dbcc_exclusions	checkverify と sp_dbcc_faultreport で使用される永続的な除外リストをユーザが作成および管理できるようにします。 sp_dbcc_exclusions dbname, op, type, exclusion_list
sp_dbcc_faultreport	ターゲット・データベースの指定したオブジェクトについて、指定した日付に実行された dbcc checkstorage オペレーションのフォールト統計のレポートを生成します。 sp_dbcc_faultreport [report_type [, dbname [, objectname [, date [, hard_only [, exclusion_mode[, exclusion_faults [, exclusion_tables [, exclusion_combo [, display_recommendations [, opid [, fault_type_in]]]]]]]]]]]]]]]
sp_dbcc_fullreport	指定した date 日以前の database.object_name に対し、sp_dbcc_summaryreport、sp_dbcc_configreport、sp_dbcc_statisticsreport、sp_dbcc_faultreport short を実行します。 sp_dbcc_fullreport [dbname [, objectname [, date]]]
sp_dbcc_help_fault	指定されたフォールト・タイプと推奨される修復方法の説明を提供します。 sp_dbcc_help_fault [fault_type]
sp_dbcc_patch_finishtime	アポットされた checkverify オペレーションと checkstorage オペレーションに関するレポートを生成します。 sp_dbcc_patch_finishtime dbname, opid [, optype [, seq [, finishtime]]]
sp_dbcc_recommendations	指定されたオペレーション ID または日付に該当する checkstorage オペレーションによってレポートされたフォールトを分析し、ターゲット/データベース内の指定されたオブジェクトに関して推奨される対処法のリストを生成します。 sp_dbcc_recommendations dbname [, "date" [, opid [, "objectname"]]]
sp_dbcc_runcheck	指定したデータベースについて dbcc checkstorage を実行してから、sp_dbcc_summaryreport または指定するレポートを実行します。 sp_dbcc_runcheck dbname [, user_proc]
sp_dbcc_statisticsreport	ターゲット・データベースにある指定したオブジェクトについて、アロケーション統計のレポートを生成します。 sp_dbcc_statisticsreport [dbname [, objectname [, date]]]
sp_dbcc_summaryreport	指定したデータベースについてサマリ・レポートを生成します。 sp_dbcc_summaryreport [dbname [, date [, op_name [, display_recommendations]]]]
sp_dbcc_updateconfig	ターゲット・データベースの設定情報を使用して、dbccdb にある dbcc_config テーブルを更新します。 sp_dbcc_updateconfig dbname, type, "str1" [, "str2"]

## システム・テーブル

以下は、Adaptive Server のシステム・テーブルの概要です。詳細については、『リファレンス・マニュアル：テーブル』を参照してください。テーブル、カラム、テーブル間の関係を視覚的に確認するには、『システム・テーブル・ダイアグラム』ポスターを参照してください。

syblicenseslog	<i>master</i> データベースのみ – 24 時間間隔で Adaptive Server で使用されるライセンスの最大数の更新ごとに 1 つのローが含まれています。カラム：status、logtime、maxlicenses
sysalternates	すべてのデータベース – 現在使用しているデータベースのユーザにマップされた (またはエイリアスを割り当てた) Adaptive Server のユーザごとに 1 つのローが含まれています。カラム：suid、altsuid
sysaltusages	スクラッチ・データベース – アーカイブ・データベースのページ番号を、データベース・ダンプとそのストライプ内、または変更済みページ・セクション内の実際のページにマップします。カラム：dbid、altsuid、lstart、start、size、vstart、vdevno、segmap
sysattributes	すべてのデータベース – オブジェクトのプロパティを定義します。カラム：class、attribute、object_type、object_cinfo、object_cinfo2、object、object_info1、object_info2、object_info3、int_value、char_value、text_value、image_value、comments
sysauditoptions	<i>sybsecurity</i> データベース – サーバワイドな監査オプションごとに 1 つのローを持ち、そのオプションの現在の設定を表します。カラム：num、val、minval、maxval、name、sval、comment
sysaudits_01 ~ sysaudits_08	<i>sybsecurity</i> データベース – 監査証跡が含まれています。カラム：event、eventmod、spid、eventtime、sequence、suid、dbid、objid、xactid、loginname、dbname、objname、objowner、extrainfo、nodeid、instanceid
syscharsets	<i>master</i> データベースのみ – Adaptive Server が使用する文字セットとソート順ごとに 1 つのローが含まれています。カラム：type、id、csid、status、name、description、definition、sortfile
syscolumns	すべてのデータベース – すべてのテーブルとビューのカラムごとに 1 つのロー、またはプロシージャのパラメータごとに 1 つのローが含まれています。カラム：id、number、colid、status、type、length、offset、usertype、cdefault、domain、name、printfmt、prec、scale、remote_type、remote_name、xstatus、xtype、xdbid、accessrule、status2、status3、computedcol、encrtype、enclen、encykeyid、encykeydb、encrdate
syscomments	すべてのデータベース – 各ビュー、ルール、デフォルト、トリガ、テーブル制約、パーティション、プロシージャ、計算カラム、関数ベースのインデックス・キー、コンパイル済みオブジェクトのその他のフォームに対するエントリが含まれています。カラム：id、number、colid、texttype、language、text、colid2、status、partitionid

---

sysconfigures	<i>master</i> データベースのみ – ユーザが設定できる設定パラメータごとに1つのローが含まれています。カラム：config、value、comment、status、name、parent、value2、value3、value4、instanceid
sysconstraints	すべてのデータベース – create table または alter table を使用して、新しい検査制約または参照制約を宣言すると、Adaptive Server によってローが sysconstraints テーブルに挿入されます。alter table を実行して制約を削除するまで、ローはそのまま残ります。カラム：colid、constrid、tableid、error、status、spare2
syscoordinations	<i>sysystemdb</i> データベースのみ – 分散トランザクション (リモート・パーティシパント) に関するリモート Adaptive Server とそれらのコーディネーション・ステータスについての情報が含まれています。カラム：participant、starttime、coordtype、owner、protocol、state、bootcount、dbid、logvers、spare、status、xactkey、gtrid、partdata、srvname、nodeid、instanceid
syscurconfigs	<i>master</i> データベースのみ – sysconfigures と同じように各設定パラメータのエントリが含まれていますが、その値はデフォルト値ではなく現在の値です。さらに、設定構造を説明する4つのエントリが含まれています。カラム：config、value、comment、status、value2、defvalue、minimum_value、maximum_value、memory_used、display_level、datatype、message_num、apf_percent、nodeid、instanceid、type
sysdatabases	<i>master</i> データベースのみ – Adaptive Server のデータベースごとに1つのローが含まれています。Adaptive Server をインストールすると、sysdatabases には、 <i>master</i> データベース、 <i>model</i> データベース、 <i>sysystemprocs</i> データベース、 <i>tempdb</i> データベースのエントリが含まれます。監査をインストールした場合は、 <i>sybsecurity</i> データベースのエントリも含まれます。カラム：name、dbid、suid、status、version、logptr、crdate、dumptrdate、status2、audflags、deftabaud、defwaud、defpraud、def_remote_type、def_remote_loc、status3、status4、audflags2、instanceid、durability
sysdepends	すべてのデータベース – プロシージャ、ビュー、またはトリガによって参照されるプロシージャ、ビュー、またはテーブルごとに1つのローが含まれています。カラム：id、number、depid、depnumber、status、selall、resultobj、readobj、columns
sysdevices	<i>master</i> データベースのみ – テープ・ダンプ・デバイス、ディスク・ダンプ・デバイス、データベース用ディスク、データベース用ディスク・パーティションごとに1つのローが含まれています。カラム：low、high、status、cntritype、name、phyname、mirrorname、vdevno、crdate、resizedate、status2、instanceid、uuid
sysencryptkeys	デフォルト・キーなど、データベースで作成される各キーには、データベース固有のシステム・カタログ sysencryptkeys にエントリがあります。カラム：id、ekalgorithm、type、status、eklen、value、uid、eksalt、ekpairid、pwdate、expdate、ekpwdwarn

sysengines	<i>master</i> データベースのみ – 現在オンラインになっている Adaptive Server エンジンごとに1つのローが含まれています。カラム: engine, osprocid, osprocname, status, affinityid, cur_kpid, last_kpid, idle_1, idle_2, idle_3, idle_4, starttime, nodeid, instanceid
sysgams	すべてのデータベース – <i>sysgams</i> には、データベースのグローバル・アロケーション・マップが格納されています。
sysindexes	すべてのデータベース – クラスタード・インデックスごとに1つのロー、ノンクラスタード・インデックスごとに1つのロー、クラスタード・インデックスのないテーブルごとに1つのロー、text カラムや image カラムのあるテーブルごとに1つのローが含まれています。カラム: name, id, indid, doampg, ioampg, oampgtrips, status3, status2, ipgtrips, first, root, distribution, usagecnt, segment, status, maxrowsperpage, minlen, maxlen, maxirow, keycnt, keys1, keys2, soid, csid, base_partition, fill_factor, res_page_gap, exp_rowsize, keys3, identitygap, crdate, partitiontype, conditionid
sysinstances	インスタンスの状態をレポートする偽のテーブルです。sysinstances は、クラスタ設定で定義されているインスタンスごとに1つのローを含んでいます。カラム: id, name, state, hostname, starttime, connections_active, engines_online
sysjars	すべてのデータベース – データベースに必要な各 Java アーカイブ・ファイルごとに1つのローが含まれています。カラム: jid, jstatus, jname, jbinary
syskeys	すべてのデータベース – プライマリ・キー、外部キー、または共通キーごとに1つのローが含まれています。カラム: id, type, depid, keycnt, size, key1 ~ key8, depkey1 ~ depkey8, spare1
syslanguages	<i>master</i> データベースのみ – Adaptive Server の言語ごとに1つのローが含まれています。カラム: langid, dateformat, datefirst, upgrade, name, alias, months, shortmonths, days
syslisteners	<i>master</i> データベースのみ – 現在の Adaptive Server と接続できるネットワーク・プロトコルごとに1つのローが含まれています。カラム: net_type, address_info, spare, nodeid, instanceid
syslocks	<i>master</i> データベースのみ – アクティブ・ロックに関する情報を格納し、ユーザがクエリを実行したときに動的に構築されます。カラム: id, dbid, page, type, spid, class, fid, context, row, loid, partitionid, nodeid, instanceid
sysloginroles	<i>master</i> データベースのみ – システム役割を持つサーバ・ログインのインスタンスごとに1つのローが含まれています。カラム: suid, srid, status
syslogins	<i>master</i> データベースのみ – 有効な Adaptive Server ユーザ・アカウントごとに1つのローが含まれています。カラム: suid, status, accdate, totcpu, totio, spacelimit, timelimit, resultlimit, dbname, name, password, language, pwdate, audflags, fullname, srvname, logincount, procid, lastlogindate, crdate, locksuid, lockreason, lockdate

---

syslogs	すべてのデータベース – トランザクション・ログが含まれています。リカバリとロールフォワードを行うために Adaptive Server によって使用されます。ユーザにとっては役に立つものではありません。カラム：xactid、op
syslogshold	<i>master</i> データベースのみ – 各データベースの最も古いアクティブ・トランザクション ( 存在する場合 ) と、トランザクション・ログに対する Replication Server のトランケーション・ポイント ( 存在する場合 ) についての情報が含まれていますが、このテーブルは、通常のテーブルではありません。ユーザがクエリを実行するときに動的に構築されます。カラム：dbid、reserved、spid、page、xactid、masterxactid、starttime、name、xloid
sysmessages	<i>master</i> データベースのみ – Adaptive Server が返すシステム・エラーまたは警告ごとに 1 つのローが含まれています。カラム：error、severity、dlevel、description、langid、sqlstate
sysmonitors	<i>master</i> データベースのみ – モニタ・カウンタごとに 1 つのローが含まれています。カラム：field_name、group_name、field_id、value、description、nodeid、instanceid
sysobjects	すべてのデータベース – テーブル、ビュー、ストアド・プロシージャ、拡張ストアド・プロシージャ、ログ、ルール、デフォルト、トリガ、検査制約、参照制約、計算カラム、関数ベースのインデックス・キー、テンポラリ・オブジェクト (tempdb のみ)、コンパイル済みオブジェクトのその他のフォームごとに 1 つのローが含まれています。また、オブジェクトの type が N のときには、各パーティション条件 ID ごとに 1 つのローが含まれています。カラム：name、id、uid、type、userstat、sysstat、indexdel、schemacnt、sysstat2、systat3、crdate、expdate、deltrig、instrig、updtrig、seltrig、ckfirst、cache、audflags、objspare、versionts、loginame、identburnmax、spacestate、erlchgt
sysoptions	すべてのデータベース – sp_options が問い合わせる偽のテーブルです。カラム：spid、name、category、currentsetting、defaultsetting、scope
syspartitionkeys	すべてのデータベース – テーブルのハッシュ、範囲、リストによる分割のパーティション・キーごとに 1 つのローが含まれています。すべてのカラムは null ではありません。カラム：indid、id、colid、position
syspartitions	すべてのデータベース – データ・パーティションごとに 1 つのロー、インデックス・パーティションごとに 1 つのローが含まれています。カラム：name、indid、id、partitionid、segment、status、datoampage、indoampage、firstpage、rootpage、data_partitionid、crdate、cdataptname
sysprocedures	すべてのデータベース – 各ビュー、デフォルト、ルール、トリガ、プロシージャ、宣言デフォルト、分割条件、検査制約、計算カラム、関数ベースのインデックス・キー、コンパイル済みオブジェクトのその他のフォームに対するエントリが含まれています。カラム：type、qp_setting、id、sequence、status、number、version

sysprocesses	<i>master</i> データベースのみ – Adaptive Server のプロセスについての情報が含まれていますが、このテーブルは、通常のテーブルではありません。カラム：spid, kpid, enginenum, status, suid, hostname, program_name, hostprocess, clientport, cmd, cpu, physical_io, memusage, blocked, dbid, uid, gid, tran_name, time_blocked, network_pktsz, fid, execclass, priority, affinity, id, stmtnum, llnum, origsuid, block_xloid, clientname, clienthostname, clientappliance, sys_id, ses_id, loggedindatetimestamp, ipaddr, nodeid, instanceid, pad, lcid
sysprotects	すべてのデータベース – ユーザ、グループ、役割に付与または取り消されたパーミッションの情報が含まれています。カラム：id, uid, action, protecttype, columns, grantor
sysquerymetrics	すべてのデータベース – 永続的なデータからの個々のクエリに使用される集合履歴クエリ処理の測定基準を示します。カラム：uid, gid, hashkey, id, sequence, exec_min, exec_max, exec_avg, elap_min, elap_max, elap_avg, lio_min, lio_max, lio_avg, pio_min, pio_max, pio_avg, cnt, abort_cnt, qtext
sysqueryplans	すべてのデータベース – 各概略クエリ・プランに対して複数のローが含まれています。データロー・ロックを使用します。カラム：uid, dbid, qdate, sprocid, hashkey2, key1, key2, key3, gid, hashkey, id, type, sequence, status, text
sysreferences	すべてのデータベース – テーブルまたはカラムに宣言された参照整合性制約ごとに1つのローが含まれています。カラム：indexid, constrid, tableid, reftabid, keycnt, status, frgnbid, pmrydbid, spare2, fokey1 ~ fokey16, refkey1 ~ refkey16, frgnbname, pmrydbname
sysremotelogins	<i>master</i> データベースのみ – Adaptive Server でリモート・プロシージャ・コールを実行できるリモート・ユーザごとに1つのローが含まれています。カラム：remoteserverid, remoteusername, suid, status
sysresourcelimits	<i>master</i> データベースのみ – Adaptive Server で定義されるリソース制限ごとに1つのローが含まれています。カラム：name, appname, rangeid, limitid, enforced, action, limitvalue, scope, spare
sysroles	すべてのデータベース – サーバの役割IDをローカルの役割IDにマップします。カラム：id, lrid, type, status
syssecmechs	<i>master</i> データベースのみ – Adaptive Server で利用できる各セキュリティ・メカニズムでサポートされているセキュリティ・サービスに関する情報が含まれています。カラム：sec_mech_name, available_service
syssegments	すべてのデータベース – セグメント(ディスクの一部分の集まり)ごとに1つのローが含まれています。カラム：segment, name, status
syssservers	<i>master</i> データベースのみ – この Adaptive Server がリモート・プロシージャ・コールを実行できるリモートの Adaptive Server, Backup Server, または Open Server ごとに1つのローが含まれています。カラム：srvid, srvstatus, srvstatus2, srvstat2, srvname, srvnetname, srvclass, srvsecmech, srvcost

---

sysessions	<i>master</i> データベースのみ – フェールオーバ・プロパティを使用して Adaptive Server に接続するクライアントごとに 1 つのローが含まれています。カラム : <code>sys_id</code> , <code>ses_id</code> , <code>state</code> , <code>spare</code> , <code>status</code> , <code>dbid</code> , <code>name</code> , <code>nodeid</code> , <code>instanceid</code> , <code>ses_data</code>
syslices	すべてのデータベース – スライスされたテーブルのスライス ( ページ・チェーン ) ごとに 1 つのローが含まれています。 <code>syslices</code> は、Adaptive Server がプロセスをアップグレードする間だけ使用されます。カラム : <code>state</code> , <code>id</code> , <code>partitionid</code> , <code>firstpage</code> , <code>controlpage</code> , <code>spare</code>
sysrrroles	<i>master</i> データベースのみ – システム定義役割またはユーザ定義役割ごとに 1 つのローが含まれています。カラム : <code>srid</code> , <code>name</code> , <code>password</code> , <code>pwdate</code> , <code>status</code> , <code>logincount</code>
sysstatistics	すべてのデータベース – ユーザ・テーブル上のインデックス・カラムとパーティションごとに 1 つ以上のローが含まれています。インデックスされていないカラムに対してローを含むこともあります。カラム : <code>statid</code> , <code>id</code> , <code>sequence</code> , <code>moddate</code> , <code>formatid</code> , <code>usedcount</code> , <code>colidarray</code> , <code>c0...c79</code> , <code>indid</code> , <code>ststatus</code> , <code>partitionid</code> , <code>spare2</code> , <code>spare3</code>
systabstats	すべてのデータベース – クラスタード・インデックスごとに 1 つのロー、ノンクラスタード・インデックスごとに 1 つのロー、クラスタード・インデックスのないテーブルごとに 1 つのロー、パーティションごとに 1 つのローが含まれています。カラム : <code>indid</code> , <code>id</code> , <code>activestatid</code> , <code>indexheight</code> , <code>leafcnt</code> , <code>pagecnt</code> , <code>rowcnt</code> , <code>forwrowcnt</code> , <code>delrowcnt</code> , <code>dpagecrnt</code> , <code>ipagecrnt</code> , <code>drowcrnt</code> , <code>oamapgcnt</code> , <code>extent0pgcnt</code> , <code>datarowsize</code> , <code>leafrowsize</code> , <code>status</code> , <code>plljoindegree</code> , <code>spare2</code> , <code>rslastoam</code> , <code>rslastpage</code> , <code>frlastoam</code> , <code>frlastpage</code> , <code>conopt_thld</code> , <code>plldegree</code> , <code>emptypgcnt</code> , <code>spare4</code> , <code>partitionid</code> , <code>spare5</code> , <code>statmoddate</code> , <code>unusedpgcnt</code> , <code>oampagecnt</code>
systhresholds	すべてのデータベース – データベースに定義されたスレッシュホールドごとに 1 つのローが含まれています。カラム : <code>segment</code> , <code>free_space</code> , <code>status</code> , <code>proc_name</code> , <code>suid</code> , <code>currauth</code>
systimeranges	<i>master</i> データベースのみ – リソース制限をアクティブにする期間を制御するために Adaptive Server が使用する名前付き時間範囲が格納されます。カラム : <code>name</code> , <code>id</code> , <code>startday</code> , <code>endday</code> , <code>starttime</code> , <code>endtime</code>
sysrtransactions	<i>master</i> データベースのみ – Adaptive Server のトランザクションについての情報が含まれていますが、このテーブルは、通常のテーブルではありません。カラム : <code>xactkey</code> , <code>starttime</code> , <code>failover</code> , <code>type</code> , <code>coordinator</code> , <code>state</code> , <code>connection</code> , <code>status</code> , <code>status2</code> , <code>spid</code> , <code>masterdbid</code> , <code>loid</code> , <code>namelen</code> , <code>xactname</code> , <code>srvname</code> , <code>nodeid</code> , <code>instanceid</code>
systypes	すべてのデータベース <i>s</i> – システム定義データ型およびユーザ定義データ型ごとに 1 つのローが含まれています。ドメイン ( ルールで定義される ) とデフォルトが存在する場合は、それらも含まれています。カラム : <code>uid</code> , <code>usertype</code> , <code>variable</code> , <code>allownulls</code> , <code>type</code> , <code>length</code> , <code>tdefault</code> , <code>domain</code> , <code>name</code> , <code>printfmt</code> , <code>prec</code> , <code>scale</code> , <code>ident</code> , <code>hierarchy</code> , <code>xtypeid</code> , <code>xdbid</code> , <code>accessrule</code>

---

sysusages	<i>master</i> データベースのみ – データベースに割り当てられたディスク割り当ての集まりごとに1つのローが含まれています。各データベースは、指定された数のデータベース (論理) ページ番号を持っています。カラム: <code>dbid</code> , <code>segmap</code> , <code>lstart</code> , <code>size</code> , <code>vstart</code> , <code>pad</code> , <code>unreservedpgs</code> , <code>crdate</code> , <code>vdevno</code>
sysusermessages	すべてのデータベース – Adaptive Server によって返されるユーザ定義メッセージごとに1つのローが含まれています。カラム: <code>error</code> , <code>uid</code> , <code>description</code> , <code>langid</code> , <code>dlevel</code>
sysusers	すべてのデータベース – データベースに許可されたユーザごとに1つのロー、およびグループまたは役割ごとに1つのローが含まれています。カラム: <code>suid</code> , <code>uid</code> , <code>gid</code> , <code>name</code> , <code>environ</code>
sysxtypes	すべてのデータベース – 拡張機能 Java-SQL データ型ごとに1つのローが含まれています。カラム: <code>xtid</code> , <code>xtstatus</code> , <code>xtmetatype</code> , <code>xtcontainer</code> , <code>xtname</code> , <code>xtsource</code> , <code>xtbinaryinrow</code> , <code>xtbinaryoffrow</code>

## DBCC テーブル

以下は、Adaptive Server の `dbcc` テーブルの概要です。詳細については、『リファレンス・マニュアル・テーブル』を参照してください。

<code>dbcc_config</code>	現在実行中の、または最後に完了した <code>dbcc checkstorage</code> オペレーションを記述します。
<code>dbcc_counters</code>	<code>dbcc checkstorage</code> によって実行された分析の結果を格納します。
<code>dbcc_exclusions</code>	<code>checkverify</code> による処理から除外されるフォールト、テーブル、フォールトとテーブルの組み合わせと、 <code>sp_dbcc_faultreport</code> を介して報告されるフォールトを格納します。
<code>dbcc_fault_params</code>	<code>dbcc_faults</code> テーブルに入力されたフォールトに関する補足説明を提供します。
<code>dbcc_faults</code>	<code>dbcc checkstorage</code> によって検出された各フォールトの説明を提供します。
<code>dbcc_operation_log</code>	<code>dbcc checkstorage</code> オペレーションの使用を記録します。
<code>dbcc_operation_results</code>	<code>dbcc_operation_log</code> テーブルに記録されたオペレーションに関する補足説明を提供します。
<code>dbcc_types</code>	<code>dbcc checkstorage</code> によって使用されるデータ型の定義を提供します。

---

## モニタリング・テーブル

以下は、Adaptive Server のモニタリング・テーブルの概要です。詳細については、『リファレンス・マニュアル：テーブル』を参照してください。テーブル、カラム、テーブル間の関係を視覚的に確認するには、『モニタリング・テーブル・ダイアグラム』ポスターを参照してください。

monCachedObject	現在データ・キャッシュに格納されているすべてのテーブル、パーティション、およびページを持つインデックスについての統計情報を保存します。カラム：CacheID、InstanceID、DBID、IndexID、PartitionID、CachedKB、CacheName、ObjectID、DBName、OwnerUserID、OwnerName、ObjectName、PartitionName、ObjectTypeTotalSizeKB、ProcessesAccessing
monCachePool	すべてのデータ・キャッシュに割り付けられたすべてのプールについての統計情報を保存します。カラム：CacheID、InstanceID、IOBufferSize、AllocatedKB、PhysicalReads、Stalls、PagesTouched、PagesRead、BuffersToMRU、BuffersToLRU、CacheName
monCachedProcedures	プロシージャ・キャッシュに現在格納されているすべてのストアド・プロシージャ、トリガ、コンパイル済みプランについての統計情報を保存します。カラム：ObjectID、InstanceID、OwnerUID、DBID、PlanID、MemUsageKB、CompileDate、ObjectName、ObjectType、OwnerName、DBName、RequestCnt、TempdbRemapCnt、AvgTempdbRemapTime
monCachedStatement	ステートメント・キャッシュに関する詳細なモニタリング情報を保存します。これには、以前に文を実行したときに使用されたリソース、文の実行頻度、特定のプランに対して有効になっている設定、文の同時使用数などに関する情報が含まれます。カラム：SSQLID、HashKey、UserID、SUserID、DBID、DBName、CachedDate、LastUsedDate、CurrentUsageCount、StatementSize、MaxUsageCount、SessionSettings、ParallelDegree、QuotedIdentifier、TransactionIsolationLevel、TransactionMode、SAAuthorization、SystemCatalogUpdates、ExecutionMetrics、MetricsCount、MaxElapsedTime、MinElapsedTime、AvgElapsedTime、MaxLIO、MinLIO、AvgLIO、MaxPIO、MinPIO、AvgPIO、NumRecompilesPlanFlushes、NumRecompilesSchemaChanges、MaxPlanSize、MinPlanSize、LastRecompiledDate、UseCount、HasAutoParams
monCIPC	クラスタ環境特有のテーブル - 現在のインスタンス、またはすべてのインスタンスを基準にした、クラスタ内での総メッセージングの合計の数字を示します。カラム：InstanceID、ReceiveCount、TransmitCount、Multicast、Synchronous、ReceiveSoftError、ReceiveHardError、TransmitsSoftError、TransmitHardError、Retransmits、Switches、FailedSwitches、RegularBuffersInUse、FreeRegularBuffers、MaxRegularBuffersInUse、LargeBuffersInUse、FreeLargeBuffers、MaxLargeBuffersInUse

monCIPCEndpoints	クラスタ環境特有のテーブル – クラスタ・インスタンス内にある各サブシステムのトラフィック・データを含む詳細な一覧を示します。カラム：InstanceID、ReceiveCount、TransmitCount、ReceiveBytes、TransmitBytes、ReceiveQ、MaxReceiveQ、DoneQ、MaxDoneQ、MaxRecvQTime、AvgRecvQTime、EndPoint
monCIPCLinks	クラスタ環境特有のテーブル – クラスタ内のインスタンスの間のリンクの状態をモニタします。カラム：InstanceID、LocalInterface、RemoteInterface、PassiveState、PassiveStateAge、ActiveState、ActiveStateAge
monCIPCMesh	クラスタ環境特有のテーブル – インスタンスごとにその接続網、つまり現在のインスタンスから同じクラスタ内にある他のすべてのインスタンスへの接続の合計の数字を示します。カラム：InstanceID、FarInstanceID、Received、Dropped、Transmitted、Resent、Retry、ControlRx、ControlTx、SendQ、MaxSendQ、SentQ、MaxSentQ、MaxSendQTime、AvgSendQTime、Mesh、MinRTT、MaxRTT、AverageRTT
monCLMObjectActivity	クラスタ環境特有のテーブル – クラスタ・ロック情報を収集します。カラム：InstanceID、DBID、Object_PartitionID、LockRequests、LocalMaster、Waited、Granted、RWConflictWaited、AvgRWConflictWaitTime、MaxRWConflictWaitTime、WWConflictWaited、AvgWWConflictWaitTime、MaxWWConflictWaitTime、ClusterMsgWaits、AvgClusterMsgWaitTime、MaxClusterMsgWaitTime、DowngradeReqRecv、DowngradeReqRecvWithNoBlocker、ClusterDeadlock、Locktype
monClusterCache-Manager	クラスタ環境特有のテーブル – 各インスタンスで実行されているクラスタ・キャッシュ・マネージャ・デーモンに関する診断情報を保存します。monClusterCacheManager は、クラスタ全体にわたる情報をインスタンス単位でレポートします。カラム：InstanceID、RequestsQueued、RequestsRequeued、RequestsServiced、DiskWrites、SleepCount、DaemonName、TransfersInitiated、Downgrades、Releases、AvgServiceTime、MaxQSize
monCMSFailover	クラスタ環境特有のテーブル – クラスタのメンバシップ・サービス (CMS) によるエラーの検出、新しいクラスタ・ビューの取得、ハートビートの再同期、エラー・イベントの通知、およびエラー・イベントの完了が行われる時刻をそれぞれ追跡します。カラム：InstanceID、FailedInstanceID、FailDetectTime、InitViewTime、FinalViewTime、ResynchHBTime、NotifyFailTime、EventdoneTime
monDataCache	Adaptive Server データ・キャッシュに関する統計情報を保存します。カラム：CacheID、InstanceID、RelaxedReplacement、BufferPools、CacheSearches、PhysicalReads、LogicalReads、PhysicalWrites、Stalls、CachePartitions、CacheName

---

monDBRecovery	<p>クラスタ環境特有のテーブル – クラスタ内のすべてのインスタンスからのロー、およびリカバリに必要な各データベースのローが含まれています。カラム：DBID、InstanceID、MaxOpenXacts、MaxPFTSEntries、Buckets、LogBTotPages、LogBTotAPFWaited、LogBTotIO、AnlTotRec、AnlPhase1Recs、AnlPhase1RedoRecs、AnlPhase2Recs、AnlPhase2RedoRecs、AnlTotPages、AnlTotAPFWaited、AnlTotIO、RedoOps、RedoOpsNotRedonePFTS、RedoOpsRedonePFTS、RedoOpsRedoneTS、RedoOpsNotRedoneTS、RedoLogTotPages、RedoLogTotAPFWaited、RedoLogTotIO、RedoRecTotPage、RedoRecTotAPFWaited、RedoRecTotIO、UndoRecsUndone、UndoLogTotPages、UndoLogTotAPFWaited、UndoLogTotIO、UndoRecTotPages、UndoRecTotAPFWaited、UndoRedTotIO、DBName、FailedInstanceID、Command、RecType、LobBStartTime、LogBEndTime、AnlStartTime、AnlEndTime、RedoStartTime、RedoEndTime、UndoStartTime、UndoEndTime</p>
monDBRecoveryLRTypes	<p>クラスタ環境特有のテーブル – リカバリ中に参照されたログ・レコードを追跡します。最低でも1つのログ・レコードがリカバリによって参照された、各ログ・レコード・タイプ用のローが含まれています。カラム：DBID、InstanceID、NumRecs、LogRecType</p>
monDeadLock	<p>デッドロックに関する情報を提供します。deadlock pipe max messages を使用すると、返されるメッセージの最大数を調整できます。カラム：DeadLockID、VictimKPID、InstanceID、ResolveTime、ObjectDBID、PageNumber、RowNumber、HeldFamilyId、HeldSPID、HeldKPID、HeldProcDBID、HeldProcedureID、HeldBatchID、HeldContextID、HeldLineNumber、WaitFamilyId、WaitSPID、WaitKPID、WaitTime、ObjectName、HeldUserName、HeldAppName、HeldTranName、HeldLockType、HeldCommand、WaitUserName、WaitLockType、HeldSourceCodeID、WaitSourceCodeID</p>
monDeviceIO	<p>データベース・デバイス上のアクティビティに関連する統計情報を返します。カラム：InstanceID、Reads、APFReads、Writes、DevSemaphoreRequests、DevSemaphoreWaits、IOTime、LogicalName、PhysicalName</p>
monEngine	<p>Adaptive Server エンジンについての統計情報を提供します。カラム：EngineNumber、InstanceID、CurrentKPID、PreviousKPID、CPUTime、SystemCPUTime、UserCPUTime、IOCPUTime、IdleCPUTime、Yields、Connections、DiskIOChecks、DiskIOPolled、DiskIOCompleted、MaxOutstandingIOs、ProcessesAffinitied、ContextSwitches、HkgcMaxQSize、HkgcPendingItems、HkgcHWMItems、HkgcOverflows、Status、Starttime、StopTime、AffinitiedToCPU、OSPID</p>
monErrorLog	<p>Adaptive Server エラー・ログの最新のエラー・メッセージを返します。カラム：SPID、InstanceID、KPID、FamilyID、EngineNumber、ErrorNumber、Severity、State、Time、ErrorMessage</p>
monFailoverRecovery	<p>クラスタ環境特有のテーブル – クラスタ・ロック・マネージャ (CLM)、データベース・リカバリ、クラスタ・メンバシップ・サービス (CMS) の各モジュール用に、集計されたフェールオーバー・リカバリ診断情報が含まれています。カラム：InstanceID、ModuleName、FailedInstanceID、StartTime、EndTime</p>

monIOQueue	デバイス I/O の統計を、各デバイス上の通常のデータベースとテンポラリ・データベースのデータ I/O とログ I/O として表示します。カラム：InstanceID、IOs、IOTime、LogicalName、IOType
monLicense	Adaptive Server によって現在チェックアウトされているすべてのライセンスを表示したリストを返します。カラム：InstanceID、Quantity、Name、Edition、Type、Version、Status、LicenseExpiry、GraceExpiry、LicenseID、Filter、Attributes
monLocks	付与されたロックと未処理のロック要求のリストを返します。 カラム：SPID、InstanceID、KPID、DBID、ParentSPID、LockID、Context、DBName、ObjectID、LockState、LockType、LockLevel、WaitTime、PageNumber、RowNumber、BlockedBy、BlockedState、SourceCodeID
monLogicalCluster	クラスタ環境特有のテーブル - 現在システム上に設定されている論理クラスタに関する情報が表示されます。カラム：LCID、Attributes、ActiveConnections、BaseInstances、ActiveBaseInstances、FailoverInstances、ActiveFailoverInstances、Name、State、DownRoutingMode、FailoverMode、StartupMode、SystemView、Roles、LoadProfile、ActionnRelease、Gather
monLogicalClusterAction	クラスタ環境特有のテーブル - 起動からこれらのアクションの解放まで、ローカル・クラスタに対するすべての管理アクションを示します。カラム：Handle、State、LCID、LogicalClusterName、Action、FromInstances、ToInstances、InstancesWaiting、WaitType、StartTime、Deadline、CompleteTime、ConnectionsRemaining、NonMigConnections、NonHAConnections
monLogicalClusterInstance	クラスタ環境特有のテーブル - インスタンスと論理クラスタ間の多対多の関係に関する情報が表示されます。カラム：LCID、LogicalClusterName、InstanceID、InstanceName、Type、FailoverGroup、State、ActiveConnections、NonMigConnections、NonHAConnections、LoadScore
monLogicalClusterRoute	クラスタ環境特有のテーブル - 設定されているルートに関する情報 (アプリケーション、ログイン、エイリアス・バインディング) が表示されます。カラム：LCID、LogicalClusterName、RouteType、RouteKey
monNetworkIO	Adaptive Server とクライアント接続間でのすべての通信のネットワーク I/O 統計を返します。カラム：InstanceID、PacketsSent、PacketsReceived、BytesSent、BytesReceived
monOpenDatabases	サーバのメタデータ・キャッシュ内に現在あるデータベースについてステータスおよび統計情報を提供します。カラム：DBID、InstanceID、BackupInProgress、LastBackupFailed、TransactionLogFull、AppendLogRequests、AppendLogWaits、DBName、BackupStartTime、SuspendedProcesses、QuiesceTag、LastCheckpointTime、LastTranLogDumpTime

---

monOpenObjectActivity	開いているすべてのテーブルおよびインデックスについての統計情報を提供します。カラム: DBID、ObjectID、IndexID、InstanceID、DBName、ObjectName、LogicalReads、PhysicalReads、APFReads、PagesRead、PhysicalWrites、PagesWritten、RowsInserted、RowsDeleted、RowsUpdated、Operations、LockRequests、LockWaits、OptSelectCount、LastOptSelectDate、UsedCount、LastUsedDate、HkgcRequests、HkgcPending、HkgcOverflows、PhysicalLocks、PhysicalLocksRetained、PhysicalLocksRetainWaited、PhysicalLocksDeadlocks、PhysicalLocksWaited、PhysicalLocksPageTransfer、TransferReqWaited、AvgPhysicalLockWaitTime、AvgTransferReqWaitTime、TotalServiceRequests、PhysicalLocksDowngraded、PagesTransferred、ClusterPageWrites、AvgServiceTime、AvgTimeWaitedOnLocalUsers、AvgTransferSendWaitTime、AvgIOServiceTime、AvgDowngradeServiceTime
monOpenPartitionActivity	サーバ上の各オープン・パーティションの使用に関する情報を提供します。カラム: DBID、ObjectID、IndexID、PartitionID、InstanceID、DBName、ObjectName、PartitionName、LogicalReads、PhysicalReads、APFReads、PagesRead、PhysicalWrites、PagesWritten、RowsInserted、RowsDeleted、RowsUpdated、OptSelectCount、LastOptSelectDate、UsedCount、LastUsedDate、HkgcRequests、HkgcPending、HkgcOverflows、PhysicalLocks、PhysicalLocksRetained、PhysicalLocksRetainWaited、PhysicalLocksDeadlocks、PhysicalLocksWaited、PhysicalLocksPageTransfer、TransferReqWaited、AvgPhysicalLockWaitTime、AvgTransferReqWaitTime、TotalServiceRequests、PhysicalLocksDowngraded、PagesTransferred、ClusterPageWrites、AvgServiceTime、AvgTimeWaitedOnLocalUsers、AvgTransferSendWaitTime、AvgIOServiceTime、AvgDowngradeServiceTime
monPCIBridge	Java PCI Bridge についての情報が含まれています。カラム: InstanceID、Status、ConfiguredSlots、ActiveSlots、ConfiguredPCIMemoryKB、UsedPCIMemoryKB
monPCIEngine	PCI Bridge とそのプラグインのエンジン情報を表示します。カラム: InstanceID、Engine、Status、PLBStatus、NumberOfActiveThreads、PLBRequests、PLBwakeUpRequests
monPCISlots	PCI Bridge 内の各スロットにバインドされているプラグインについての情報が含まれています。カラム: InstanceID、Slot、Status、Modulename、engine
monPCM	クラスタ環境特有のテーブル - クラスタ内のピア調整モジュール (PCM) クライアント・アクティビティを追跡し、各 PCM クライアントのローを含みます。カラム: InstanceID、Sent、Fragments_sent、Fragments_received、Received、Reply、Unicast、Multicast、Sync、Async、MinBytes、AvgBytes、MaxBytes、MinDialog、AvgDialog、MaxDialog、Dialog、MinTimeSyncApi、AvgTimeSyncApi、MaxTimeSyncApi、MinTimeAsyncApi、AvgTimeAsyncApi、MaxTimeAsyncApi、MinTimeCIPCMsgAlloc、AvgTimeCIPCMsgAlloc、MaxTimeCIPCMsgAlloc、MinTimeCIPCMsgSendCB、AvgTimeCIPCMsgSendCB、MaxTimeCIPCMsgSendCB、MinTimeCIPCUnicastmsg、AvgTimeCIPCUnicastmsg、MaxTimeCIPCUnicastmsg、MinTimeCIPCMsgMulticastmsg、AvgTimeCIPCMsgMulticastmsg、MaxTimeCIPCMsgMulticastmsg、MinTimeClientRecvCB、AvgTimeClientRecvCB、MaxTimeClientRecvCB、ModuleName

monProcedureCache	Adaptive Server プロシージャ・キャッシュに関する統計情報を返します。カラム：Requests、Loads、Writes、Stalls、InstanceID
monProcedureCache-MemoryUsage	プロシージャ・キャッシュ・アロケータごとに 1 つのローが含まれています。カラム：InstanceID、AllocatorID、ModuleID、Active、HWM、ChunkHWM、AllocatorName、NumReuseCaused
monProcedureCache-ModuleUsage	プロシージャ・キャッシュからメモリを割り付けるモジュールごとに 1 つのローが含まれています。カラム：InstanceID、ModuleID、Active、HWM、NumPagesReused、ModuleName
monProcess	現在実行中または待機中のプロセスについての詳細な統計情報を提供します。カラム：SPID、InstanceID、KPID、ServerUserID、BatchID、ContextID、LineNumber、SecondsConnected、DBID、EngineNumber、Priority、FamilyID、Login、Application、Command、NumChildren、SecondsWaiting、WaitEventID、BlockingSPID、BlockingXLOID、DBName、EngineGroupName、ExecutionClass、MasterTransactionID
monProcessActivity	プロセス・アクティビティについての詳細な統計情報を提供します。カラム：SPID、InstanceID、KPID、ServerUserID、CPUTime、WaitTime、PhysicalReads、LogicalReads、PagesRead、PhysicalWrites、PagesWritten、MemUsageKB、LocksHeld、TableAccesses、IndexAccesses、WorkTables、TempDbObjects、ULCBytesWritten、ULCFlushes、ULCFlushFull、ULCMaxUsage、ULCCurrentUsage、Transactions、Commits、Rollbacks
monProcessLookup	サーバ上の各プロセスに関する識別情報を提供します。カラム：SPID、InstanceID、KPID、Login、Application、ClientHost、ClientIP、ClientOSPID
monProcessMigration	クラスタ環境特有のテーブル - 現在マイグレートしている接続に関する情報を表示します。カラム：SPID、KPID、LogicalCluster、Instance、MigrationLogicalCluster、MigrationInstance、Command
monProcessNetIO	各プロセスのネットワーク I/O アクティビティ情報を提供します。カラム：SPID、InstanceID、KPID、NetworkPacketSize、PacketSent、PacketsReceived、BytesSent、BytesRecieved、NetworkEngineNumber
monProcessObject	プロセスが現在アクセスしているオブジェクトについての統計情報を提供します。カラム：SPID、InstanceID、KPID、DBID、ObjectID、PartitionID、IndexID、OwnerUserID、LogicalReads、PhysicalReads、PhysicalAPFReads、DBName、ObjectName、PartitionName、ObjectType、PartitionSize
monProcessProcedures	プロセスによって実行されているすべてのプロシージャのリストを返します。カラム：SPID、InstanceID、KPID、DBID、OwnerUID、ObjectID、PlanID、MemUsageKB、CompileDate、ContextID、LineNumber、DBName、OwnerName、ObjectName、ObjectType
monProcessSQLText	プロセスが現在実行している SQL テキストを提供します。カラム：SPID、InstanceID、KPID、ServerUserID、BatchID、LineNumber、SequenceInLine、SQLText

---

monProcessStatement	現在実行中の文についての情報を提供します。カラム：SPID、 InstanceID、 KPID、 DBID、 ProcedureID、 PlanID、 BatchID、 ContextID、 LineNumber、 CPUTime、 WaitTime、 MemUsageKB、 PhysicalReads、 LogicalReads、 PagesModified、 PacketsSent、 PacketsReceived、 NetworkPacketSize、 PlansAltered、 RowsAffected、 DBName、 StartTime
monProcessWaits	サーバ上で現在のプロセスが待機しているすべての待機イベントのリストを提供します。カラム：SPID、 InstanceID、 KPID、 ServerUserID、 WaitEventID、 Waits、 WaitTime
monProcessWorkerThread	現在設定されている各ワーカー・プロセスのアクティビティの統計情報を提供します。カラム：SPID、 InstanceID、 KPID、 ThreadsActive、 MaxParallelDegree、 MaxScanParallelDegree、 ParallelQueries、 PlansAltered、 FamilyID
monState	Adaptive Server 全体のステータスについての情報を提供します。カラム： InstanceID、 LockWaitThreshold、 LockWaits、 DaysRunning、 CheckPoints、 NumDeadlocks、 Diagnostic Dumps、 Connections、 MaxRecovery、 Transactions、 StartDate、 CountersCleared
monStatementCache	文のキャッシュについての統計情報を提供します。カラム： InstanceID、 TotalSizeKB、 UsedSizeKB、 NumStatements、 NumSearches、 HitCount、 NumInserts、 NumRemovals、 NumRecompilesSchemaChanges、 NumRecompilesPlanFlushes
monSysLoad	クラスタ環境特有のテーブル エンジン単位の統計情報の傾向を提供します。カラム： InstanceID、 EngineNumber、 SteadyState、 Avg_1min、 Avg_5min、 Avg_15min、 Max_1min、 Max_5min、 Max_15min、 Max_1min_Time、 Max_5min_Time、 Max_15min_Time、 Statistic、 Sample、 Peak、 Peak_time、 StatisticID
monSysPlanText	最近実行されたクエリのクエリ・プラン履歴を提供します。カラム： PlanID、 InstanceID、 SPID、 KPID、 BatchID、 ContextID、 SequenceNumber、 DBID、 ProcedureID、 DBName、 PlanText
monSysSQLText	前回実行された SQL テキスト、または現在実行中の SQL テキストを提供します。カラム： SPID、 InstanceID、 KPID、 ServerUserID、 BatchID、 SequenceInBatch、 SQLText
monSysStatement	サーバ上で最近実行された文の履歴を提供します。カラム： SPID、 InstanceID、 KPID、 DBID、 ProcedureID、 PlanID、 BatchID、 ContextID、 LineNumber、 CpuTime、 WaitTime、 MemUsageKB、 PhysicalReads、 LogicalReads、 PagesModified、 PacketsSent、 PacketsReceived、 NetworkPacketSize、 PlansAltered、 RowsAffected、 ErrorStatus、 HashKey、 SsqlId、 ProcNestLevel、 StatementNumber、 DBName、 StartTime、 EndTime
monSysWaits	プロセスが待機しているイベントに関するサーバワイドの統計情報を提供します。カラム： InstanceID、 WaitEventID、 WaitTime、 Waits

monSysWorkerThread	ワーカー・スレッドの設定と実行に関するサーバワイドの統計情報を返します。カラム：InstanceID、ThreadsActive、TotalWorkerThreads、HighWater、ParallelQueries、PlansAltered、WorkerMemory、TotalWorkerMemory、WorkerMemoryHWM、MaxParallelDegree、MaxScanParallelDegree
monTableColumns	各モニタリング・テーブルのすべてのカラムについて説明します。カラム：TableID、ColumnID、TypeID、Precision、Scale、Length、Indicators、TableName、ColumnName、TypeName、Description
monTableParameters	モニタリング・テーブルのクエリ・パフォーマンスを最適化するために使用する、モニタリング・テーブル内のすべてのカラムについての説明を提供します。カラム：TableID、ParameterID、TypeID、Precision、Scale、Length、TableName、ParameterName、TypeName、Description
monTables	すべてのモニタリング・テーブルについての説明を提供します。カラム：TableID、Columns、Parameters、Indicators、Size、TableName、Description
monTableTransfer	MonTableTransfer は、Adaptive Server のアクティブなメモリに存在するテーブルの転送履歴情報を提供します。カラム：InstanceID、DBID、TableID、TableName、SequenceID、TrackingID、PercentDone、BeginTime、EndTime、EndCode、TransferFloor、TransferCeiling、RowsSent、BytesSent、Format
monTempdbActivity	クラスタ環境特有のテーブル - インスタンスが tempdb 設定モードで開始されたときの、オープンしているローカル・テンポラリ・データベースすべての統計情報を、グローバル・システムの tempdb も含めて提供します。カラム：DBID InstanceID、DBName、AppendLogRequest、AppendLogWaits、LogicalReads、PhysicalReads、APFReads、PagesRead、PhysicalWrites、PagesWritten、LockRequests、LockWaits、CatLockRequests、CatLockWaits、AssignedCnt、SharableTabCnt
monWaitClassInfo	すべての待機クラスのテキストによる説明を提供します (たとえば、ディスクの読み込み完了を待機する場合)。カラム：WaitClassID、Description
monWaitEventInfo	Adaptive Server 内でプロセスが待機させられる可能性のあるあらゆる状況について、テキストによる説明を提供します。カラム：WaitEventID、WaitClassID、Description
monWorkload	クラスタ環境特有のテーブル - 各インスタンスにおける各論理クラスタの負荷スコアをそのロード・プロファイルに従って表示します。カラム：LCID、InstanceID、LoadProfileID、LoadScore、ConnectionsScore、CpuScore、RunQueueScore、IoLoadScore、EngineScore、UserScore、LogicalClusterName、InstanceName、LoadProfileName
monWorkloadPreview	クラスタ環境特有のテーブル - ロード・プロファイルがプロファイルを有効にしないでどれだけ負荷スコアに影響を及ぼすかの見積もりを提供します。カラム：InstanceID、LoadProfileID、LoadScore、ConnectionScore、CpuScore、RunQueueScore、IoLoadScore、EngineScore、UserScore、InstanceName、LoadProfileName

---

monWorkloadProfile	クラスタ環境特有のテーブル – 現在設定されている作業負荷プロファイルを表示します。カラム：ProfileID、ConnectionsWeight、CpuWeight、RunQueueWeight、IoLoadWeight、EngineWeight、UserWeight、LoginThreshold、DynamicThreshold、Hysteresis、Name、Type
monWorkloadRaw	クラスタ環境特有のテーブル – 各インスタンスについてのロー作業負荷統計情報を提供します。このモニタリング・テーブルの問い合わせには、mon_role の役割は必要ありません。カラム：InstanceID、ConnectionsRaw、CpuRaw、RunQueueRaw、IoLoadRaw、EngineRaw、UserRaw、InstanceName

## ユーティリティ

以下は、Adaptive Server ユーティリティの構文と概要です。詳細については、『ユーティリティ・ガイド』を参照してください。

backupserver 実行形式の Backup Server プログラムです。

```

backupserver
  [-C server_connections]
  [-S b_servername]
  [-l interfaces_file]
  [-e error_log_file]
  [-M sybmultipbuf_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

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。

```

bcp [[database_name.]owner.]table_name [:partition_id | slice_number ] |
partition partition_name {in | out} datafile
  [-f formatfile]
  [-e errfile]
  [-d discardfileprefix]
  [-F firstrow]
  [-L lastrow]
  [-b batchsize]
  [-m maxerrors]
  [-n]
  [-c]
  [-t field_terminator]
  [-r row_terminator]
  [-U username]
  [-P password]
  [-l interfaces_file]

```

```

[-S server]
[-a display_charset]
[-z language ]
[-A packet_size]
[-J client_charset ]
[-T text_or_image_size]
[-E]
[-g id_start_value]
[-N]
[-W]
[-X]
[-M LabelName LabelValue]
[-labeled]
[-K keytab_file]
[-R remote_server_principal]
[-C]
[-V [security_options]]
[-Z security_mechanism]
[-Q]
[-Y]
[-y sybase_directory]
[-x trusted.txt_file]
[--maxconn maximum_connections]
[--show-fi]
[--hide-vcc]
[--colpasswd [[database_name.[owner].table_name.]column_name
[password]]]
[--keypasswd [[database_name.[owner].]key_name [password]]]

```

または

```
bcp -v
```

certauth

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。

```

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

certpk12

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。

```

certpk12
{-O Pkcs12_file | -I Pkcs12_file}
[-C Cert_file]
[-K Key_file]
[-P key_password]
[-E Pkcs12_password]

```

- または certpk12 -v

---

certreq	<p>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。</p> <pre>certreq [-F <i>input_file</i>] [-R <i>request_filename</i>] [-K <i>PK_filename</i>] [-P <i>password</i>]</pre> <ul style="list-style-type: none"> <li>• または certreq -v</li> </ul>
charset	<p>UNIXプラットフォームのみ – Adaptive Server に文字セットとソート順ファイルをロードします。\$SYBASE/\$SYBASE_ASE/bin にあります。</p> <pre>charset [-P <i>password</i>] [-S <i>server</i>] [-l<i>interface</i>] <i>sort_order</i> [ <i>charset</i> ]</pre> <ul style="list-style-type: none"> <li>• または charset -v</li> </ul>
cobpre	<p>COBOL のプリコンパイラです。</p>
cpre	<p>C のプリコンパイラです。</p>
dataserver	<p>UNIXプラットフォームのみ – 実行形式の Adaptive Server プログラムです。</p> <pre>dataserver -u, --admin-name=<i>sa/sso_name</i> --buildquorum=[<i>force</i>] -a, --caps-file=<i>filename</i> -F, --cluster-input=<i>filename</i> --cluster-takeover -L, --conn-config-file=<i>filename</i> --create-cluster-id [=quorum] -D, --default-db-size=<i>size_spec</i> -e, --error-log=<i>filename</i> -G, --event-log-server=<i>logserv_name</i> -f, --forcebuild -H, --ha-server -h, --help={0 1 2 3}[,<i>display_width</i>] --instance=<i>instance_name</i> -y, --key-password=[<i>key_password</i>] -K, --keytab-file=<i>filename</i> -N, --license-prop-file=<i>filename</i> -z, --logical-page-size=<i>page_size</i> -Z, --master-db-size=<i>size_spec</i> -d, --master-dev=<i>master_device_name</i> -b, --master-dev-size=[<i>size_spec</i>] -r, --master-mirror=<i>filename</i> -m, --masterrecover -g, --no-event-logging -Q, --quorum-dev=<i>quorum_dev</i> -q, --recover-quiet -w, --rewrite-db=<i>database_name</i> -p, --sa-name={SSO_login_account   sso_role   sa_role} -k, --server-principal=<i>s_principal</i> -M, --shared-mem-dir=<i>directory_name</i> -X, --sybmon -T, --trace=<i>trace_flag</i> -v, --version</pre> <p>または</p> <pre>dataserver -v</pre>

ddlgen

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 ベースのツールです。

```
ddlgen
-Ulogin
-P password
-S[[ssl:]server | host_name :port_number]
[-interfaces_file]
[-Tobject_type]
[-Nobject_name]
[-Ddbname]
[-Xextended_object_type]
[-Output_file]
[-Error_file]
[-Lprogress_log_file]
[-J client_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
```

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。

```
defncopy
[-X]
[-a display_charset]
[-interfaces_file]
[-J [client_charset]]
[-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

dscp

UNIX プラットフォームのみ – UNIX プラットフォームのコマンド・ラインから、interfaces ファイルのサーバ・エントリを表示、編集できます。  
\$SYBASE/\$SYBASE\_OCS/bin にあります。

```
dscp [-p]
```

- または dscp -v

---

**dsedit** UNIXプラットフォーム – **dsedit** ユーティリティでは、UNIXプラットフォームで X11/Motif ベースの GUI を使用して、**interfaces** ファイルのサーバ・エントリを表示、編集できます。*Windows* – **dsedit.exe** ユーティリティを使用して **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。

**dsedit**

- または **dsedit -v**

**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。

**extractjava** (Windows では **extrjava**)

```
-j jar_name
-f file_name
[-S server_name]
[-U user_name]
[-P password]
[-D database_name]
[-l interfaces_file]
[-a display_charset]
[-J client_charset ]
[-z language ]
[-t timeout]
[-v]
```

- または **extractjava -v**

**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。

**installjava**

```
-f file_name
[-new | -update ]
[-j jar_name ]
[-S server_name]
[-U user_name]
[-P password]
[-D database_name]
[-l interfaces_file]
[-a display_charset]
[-J client_charset ]
[-z language ]
[-t timeout]
[-v]
```

- または **installjava -v**

**isql** Adaptive Server に対する対話型の SQL パーサです。

**isql** [-b] [-e] [-F] [-p] [-n] [-v] [-W] [-X] [-Y] [-Q]

```
[-a display_charset]
[-A packet_size]
[-c cmdend]
[-D database]
[-E editor]
[-h header]
[-H hostname]
[-i inputfile]
[-l interfaces_file]
[-J client_charset ]
[-K keytab_file]
[-l login_timeout]
[-m errorlevel]
```

		<pre> [-o <i>outputfile</i>] [-P <i>password</i>] [-R <i>remote_server_principal</i>] [-s <i>colseparator</i>] [-S <i>server_name</i>] [-t <i>timeout</i>] -U <i>username</i> [-V [<i>security_options</i>]] [-w <i>columnwidth</i>] [-z <i>locale_name</i>] [-Z <i>security_mechanism</i>] [--conceal] </pre>
langinstall	Adaptive Server に新しい言語をインストールします。	<pre> langinstall [-S <i>server</i>] [-U <i>user</i>] [-P <i>password</i>] [-R <i>release_number</i>] [-I <i>path</i>] <i>language</i> <i>character_set</i> </pre> <ul style="list-style-type: none"> <li>• または langinstall -v</li> </ul>
optdiag	オプション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。	<pre> optdiag [<i>binary</i>] [<i>simulate</i>] <i>statistics</i> { -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>] [-i <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>
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。	<pre> preupgrade [-v] [-h] [-N] [-p [<i>skip_sybprocs</i>]] [-D <i>database_name</i>] [-i <i>interfaces_file</i>] [-P <i>password</i>] [-S <i>server_name</i>] [-U <i>user_name</i>] [-X <i>option</i> [<i>option</i>]...] </pre>
pwdcrypt	<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。	<pre> pwdcrypt </pre>

---

qptune

紛失統計を修正し、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るようにします。

```
QPTune
[-U username]
[-P password]
[-S hostname:port/database]
[-A <action [start|collect(_full)|compare|fix|(start|collect|fix|undo_fix)_stats]>]
[-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)]
```

qrmutil

クラスタ・エディションのみ – クォーラム・デバイスのバックアップ、リストア、再設定を可能にします。qrmutilは \$SYBASE/\$SYBASE\_ASE/bin にあります。

```
--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]

```
- showserver *UNIX* プラットフォームのみ – 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

```
- sqldbgr *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。
- ```

sqldbgr
-U username
-P password
-S host:port

```
- sqlloc *UNIX* プラットフォームのみ – X11/Motif ベースの 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
[-S server]
[-U user]
[-P password]
[-s sybase dir]
[-I interfaces file]
[-r resource file]

```
- または *sqlloc -v*
- sqllocres *UNIX* プラットフォームのみ – 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
[-S server]
[-U user]
[-P password]
[-s sybase dir]
[-I interfaces file]
[-r resource file]

```
- または *sqllocres -v*
- sqlsrvr *Windows* プラットフォームのみ – 実行形式の Adaptive Server プログラムです。
- ```

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]
[-M shared_memory_repository_directory]
[-p sa_login_name]
[-r mirror_disk_name]
[-s server_name]
[-T trace_flag]
[-u sa/ssu_name]

```

---

	<pre> [-w master   model database] [-y password] [-z page_size [k   K]] </pre>
sqlupgrade	<p><i>UNIX</i> プラットフォームのみ – X11/Motif ベースの GUI を使用して、現在インストールされている Adaptive Server のバージョンを最新のリリースにアップグレードします。</p> <pre> sqlupgrade [-s sybase_dir] [-r resource_file] </pre> <ul style="list-style-type: none"> <li>• または <code>sqlupgrade -v</code></li> </ul>
sqlupgraderes	<p><i>UNIX</i> プラットフォームのみ – 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。</p> <pre> sqlupgraderes [-s sybase_dir] [-r resource_file] </pre> <ul style="list-style-type: none"> <li>• または <code>sqlupgraderes -v</code></li> </ul>
srvbuild	<p><i>UNIX</i> プラットフォームのみ – キー設定属性のデフォルトまたはユーザ定義の値を使用して、Adaptive Server、Backup Server、Monitor Server、または XP Server を新しく作成します。</p> <pre> srvbuild [-s sybase_dir] [-l interfaces_file] [-r resource_file] </pre> <ul style="list-style-type: none"> <li>• または <code>srvbuild -v</code></li> </ul>
srvbuildres	<p><i>UNIX</i> プラットフォームのみ – 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、Monitor Server、または XP Server を新しく作成します。</p> <pre> srvbuildres [-s sybase_dir] [-l interfaces_file] [-r resource_file] </pre> <ul style="list-style-type: none"> <li>• または <code>srvbuildres -v</code></li> </ul>
startserver	<p><i>UNIX</i> プラットフォームのみ – Adaptive Server または Backup Server を起動します。</p> <pre> startserver [[-f runserverfile] [-m]] ... </pre>

**sybcluster** クラスタ環境のみ – 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。

```
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 ]
```

**sybmigrate** Adaptive Server のページ・サイズを別のページ・サイズに変換し、プラットフォーム間でマイグレートします。

```
sybmigrate [-v ] [-h ] [-f ]
    [-D 1 | 2 | 3 | 4 ]
    [-l 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 ]
```

**sybtspasswd** ユーザのパスワードを記録または変更し、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。

```
sybtspasswd
```

**xpserver** XP サーバを手動で起動します。

```
xpserver -S XP_Server
xpserver
-SXP_Server
[-l interfaces_file ]
[-ppriority ]
[-sstack_size ]
[-u ]
[-v ]
[-x ]
```