

SYBASE®

インストール・ガイド

**Adaptive Server® Enterprise
Cluster Edition 15.5**

Linux

ドキュメント ID：DC01102-01-1550-02

改訂：2010年6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。

Sybase の商標は、Sybase の商標リスト (<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。

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

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

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

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

目次

Adaptive Server® Enterprise Cluster Edition の概要	1
Adaptive Server Cluster Edition のオプション	2
インストールのワークフロー	3
表記規則	4
サーバ・コンポーネントの説明とディレクト リ・レイアウト	6
PC クライアント製品の説明とディレクトリ・ レイアウト	9
インストールのプランニング	13
ライセンスの取得	15
SPDC へのアクセス	16
新しいライセンス・サーバのインストール	19
電子メール通知の有効化と変更	22
サーバのシステム要件	22
プライベート相互接続技術を使用するためのシ ステム稼働条件	25
クライアントのシステム要件	26
サーバ・インストールのインストール前の作業	27
オペレーティング・システムの共有メモリ・パラメ ータの調整	29
SySAM ホスト ID の取得	30
サーバのインストール	33
GUI モードでのサーバのインストール	33
応答ファイルの作成	35
コマンド・ラインからのインストール	36
サイレント・モードでのサーバのインストール	36
クラスタ・サーバのアンインストール	37
既存の Adaptive Server のアンインストール	37
クラスタの作成と起動	39

基本設定手順の概要	39
共有インストールの概要	39
プライベート・インストールの概要	40
インストールを開始する前に	40
単一ノードでのシミュレートされたクラスタの 作成	41
環境の設定と Unified Agent の起動	42
Unified Agent の停止	42
クラスタの作成について	42
クラスタの作成のためのワークシート	43
ASE プラグインを使用したクラスタの作成	50
sybcluster を使用したクラスタの作成	60
オペレーティング・システムの起動時のクラスタの 起動	65
インストールの失敗後のクリーンアップ	66
補助サーバ	66
sybcluster を使用した Backup Server の設定	67
ASE プラグインを使用した Backup Server の設 定	68
Job Scheduler のインストール	69
sybcluster を使用した XP Server の設定	70
sybcluster を使用した Monitor Server の設定	71
インストール後の作業	73
システム管理者パスワードの設定	73
サンプル・データベースのインストール	73
サンプル・データベースのデフォルト・デバイ ス	74
データベース・スクリプトの実行	74
interpubs データベースのインストール	75
jpubs データベースのインストール	76
サンプル・データベースの管理	77
PC クライアントのインストール	79

クライアントのインストール	80
サイレント・モードでのクライアントのインス トール	81
インストール後の使用開始にあたって	82
PC クライアントのアンインストール	86
サーバのアップグレード	87
アップグレード前の作業	88
システム・テーブルの統計	90
アップグレード中のシステム・カタログの変更	90
システムとアップグレードの要件の確認	90
runserver ファイルのロケーションの確認	91
アップグレードに必要なプロシージャ・テキス ト	91
予約語	91
プライベート・インストールへのアップグレー ド	93
手動によるアップグレードの概要	97
手動による古いサーバのアップグレード	98
アップグレードに備えたデータベースとデバイスの 準備	102
デフォルトのデータベースのサイズを大きくす る	103
ASE プラグインを使用したサーバのアップグレード .	104
ASE プラグインを使用した古いサーバのアップ グレード	105
sybcluster を使用したサーバのアップグレード	107
アップグレードのためのサーバの確認	108
入力ファイルを使用した古いサーバのアップグ レード	110
対話形式での古いサーバのアップグレード	111
アップグレード後の作業	112
.jar ファイルと .xml ファイルの更新	113

失敗したアップグレードからのリカバリ	114
データベースでの Java のアップグレード	114
マイグレート	115
BCP を使用したバージョン 15.x 以前からのマ イグレート	116
高可用性設定サーバのマイグレート	116
ダウングレード	119
ダウングレードの基本手順	119
ダウングレード後の作業	121
ダウングレード前に 15.5 Cluster Edition の機能を使 用していた場合の追加手順	123
Job Scheduler のダウングレード	125
I/O フェンシング	127
I/O フェンシングの有効化	127
I/O フェンシングの設定	128
SySAM のトラブルシューティング	131
SySAM 関連エラーを検索する場所	131
SySAM のトラブルシューティング	132
Sybase サポート・センタに問い合わせ	139
サーバのトラブルシューティング	141
エラー・ログのロケーション	142
インストール・ユーティリティのログ	143
Sybase サーバのログ	143
インストール時に起こりがちな問題の解決	144
設定に失敗した場合	147
失敗の後でのサーバの停止	147
リソース・ファイル・インストールのトラブル シューティング	148
Adaptive Server がアップグレード前の適格性テ ストに失敗した場合	148
失敗したインストールからのリカバリまたはフ ァイル作成後のリカバリ	148

失敗したアップグレードからのリカバリ	149
失敗の原因がわかっている場合	151
dbcc upgrade_object を使用したコンパイル済みオブ ジェクトのアップグレード	151
コンパイル済みオブジェクトがアップグレード されているか調べる方法	152
コンパイル済みオブジェクトにおける運用前の エラー検出	153
dbcc upgrade_object の使用	156
アップグレードにおけるデータベース・ダンプ の使用	158
手動によるクラスタの設定と管理	161
作業を始める前に	161
環境の設定	161
ロー・デバイスの作成	161
手動によるクラスタの設定	162
クラスタ入力ファイルの作成	162
interfaces ファイルの設定	166
マスタ・デバイスとクォーラム・デバイスの構 築	167
sybssystemprocs の初期化と作成	168
システム・ストアド・プロシージャのインスト ール	168
runserver ファイルの作成	169
ローカル・システム・テンポラリ・データベー スの設定	169
クラスタへの自動クラスタ継承の設定	170
クラスタの起動	171
システム障害後のクラスタの起動	172
設定後の作業	172
クラスタまたはインスタンスの停止	173
インスタンスの停止	173

クラスタの再設定	173
手動設定後の sybcluster とプラグインの有効化	174
sybcluster のサンプル・セッションの設定値	175
sybcluster のサンプル・セッション	179
入力ファイルを使用したクラスタの設定	182
Linux でのロー・パーティション	183
ロー・パーティションの選択	183
ロー・パーティションの作成	184
Red Hat のロー・デバイス管理	185
SuSE のロー・デバイス管理	186
サーバからのロー・デバイスへのアクセス	187
複写データベースを含むサーバのアップグレード	189
既存のサーバに複写データベースが含まれているか どうかを調べる	189
トランザクション処理と複写アクティビティの中断	190
プライマリ・データベースのトランザクション・ロ グの排出	191
RSSD トランザクション・ログの排出	192
セカンダリ・トランザクション・ポイントの無効化	193
sybssystemprocs の使用	195
Sybssystemprocs データベースのサイズを大きくする	195
システム・プロシージャ用のデバイス容量とデータ ベース容量を増やす	196
追加の説明や情報の入手	199
索引	201

Adaptive Server® Enterprise Cluster Edition の概要

ここでは、Adaptive Server® Enterprise Cluster Edition について説明します。

Cluster Edition では、プライベート・モードでインストールした場合も含め、インストールした Adaptive Server (複数可) を、共通のシステム・ビューを使用する共有ディスク・クラスタ環境として複数のノード上で稼働できます。各サーバは単独のノード上で稼働し、「サーバ・インスタンス」または「インスタンス」と呼ばれます。単一の設定ファイルを使用して全インスタンスの設定を決定するか (共有インストール)、インスタンスごとに別々の設定ファイルを使用できます (プライベート・インストール)。

Cluster Edition では、複数の物理的および論理的クラスタを使用して負荷をスケールリングできます。クラスタ内のインスタンスが停止すると、稼働中の 1 台または複数台のインスタンスが、停止したインスタンスの負荷を継承します。クラスタは、各クライアントが接続するインスタンスを決定します。特定のインスタンスが過負荷になると、クラスタはクライアントを他のより利便性のよいインスタンスにシフトして、負荷を均衡化します。

Cluster Edition には主に次のような利点があります。

- 可用性の向上 - 一部のクラスタ・メンバが故障した後でも、1 台でも正常に動作しているクラスタ・メンバがあれば、アプリケーションは引き続き稼働できます。
- シンプルな管理 - データがすべてのインスタンスで共有されているため、クラスタのメンバシップの変更に応じてデータのパーティションを再設定する必要がありません。

Cluster Edition は、分散アーキテクチャを可能にします。ノード間通信は、共有メモリではなくネットワーク間通信を通じて実行されます。ノード間メッセージングを最小化するアプリケーションを使用すると、Cluster Edition 環境で最適のパフォーマンスが得られます。

単一のシステムとしてアクセス可能なシステム

Cluster Edition は、単一のシステムとしてアクセス可能なシステムをサポートします。つまり、クラスタを構成する複数のインスタンスが、クライアントには単一のシステムとして表示されます。新しいクライアント・テクノロジーにより、クライアントは個々のインスタンスとの物理的な接続を維持しながら、クラスタに論理的に接続できます。この論理的な接続により、Adaptive Server はクライアントをクラスタ内のさまざまなインスタンスにリダイレクトし、高可用性フェールオーバー・データをクライアントに動的に提供できます。

作業負荷の管理

Cluster Edition ワークロード・マネージャは、ビジネス・アプリケーションのそれぞれが最も効率的に性能を発揮できるように、負荷管理とフェールオーバーをカスタマイズできます。論理クラスタは、作業環境の個別化を可能にします。

インストール・オプション

Cluster Edition では次のいずれかの設定を選択できます。

- 共有インストール - Network File System (NFS) またはクラスタ・ファイル・システムを使用して作成した共有ファイル・システムが必要です。共有インストールは、1つの \$SYBASE インストール・ディレクトリ、Adaptive Server ホーム・ディレクトリ、およびサーバ設定ファイルをサポートします。
- プライベート・インストール - 共有ファイルシステムを使用しません。プライベート・インストールは、インスタンスごとに、個別の \$SYBASE インストール・ディレクトリ、Adaptive Server ホーム・ディレクトリ、およびサーバ設定ファイルをサポートします。

Cluster Edition の詳細については、『Cluster ユーザーズ・ガイド』を参照してください。

Adaptive Server Cluster Edition のオプション

使用できるオプションは、次のとおりです。

表 1: オプション・パッケージ

オプション	使用できるプラットフォーム	説明
セキュリティ・サービスとディレクトリ・サービス	<ul style="list-style-type: none">• HP-UX 64• IBM AIX 64• Linux 64• Solaris 64	ライトウェイト・ディレクトリ・サービスおよび SSL と Kerberos を使用するネットワークベースの認証と暗号化を提供します。
パーティション	<ul style="list-style-type: none">• HP-UX 64• IBM AIX 64• Linux 64• Solaris 64	テーブル・ロー・データのセマンティック分割を有効にします。

オプション	使用できるプラットフォーム	説明
暗号化カラム	<ul style="list-style-type: none"> • HP-UX 64 • IBM AIX 64 • Linux 64 • Solaris 64 	セキュリティ・パラメータを増やし、データ型の追加に対応します。
Tivoli Storage Manager	<ul style="list-style-type: none"> • HP-UX 64 • IBM AIX 64 • Linux 64 • Solaris 64 	データベースが操作を IBM Tivoli Storage Manager にバックアップおよびリストアできるようにします。

Adaptive Server の各エディションとオプション機能は、SySAM のライセンスによってロック解除されます。詳細については、『Sybase® Software Asset Management Users Guide』を参照してください。

インストールのワークフロー

ここでは、インストールのワークフローとサーバ・インストールのための前提条件について説明します。

Cluster Edition では、ネットワークをクラスタ内のノード向けに設定する必要があります。

1. オペレーティング・システムのパッチをインストールして、オペレーティング・システムが、使用しているプラットフォームのバージョン・レベル、RAM、ネットワーク・プロトコルの稼働条件を満たしていることを確認します。
2. ソフトウェアをインストールする前に、前のバージョンの Cluster Edition を使用しているクラスタを停止しておきます。
3. すべてのインストール作業を行うための "sybase" アカウントをシステム上に作成します。このアカウントは、すべてのデバイスとファイルを所有し、クラスタで使用されるすべてのデバイスの読み込み／書き込みパーミッションを所有する必要があります。
この "sybase" アカウントを使用して、すべてのデバイスを作成します。
4. サーバの最初のインスタンスをインストールする予定のノードに、"sybase" ユーザとしてログインします。

すべてのファイルおよびディレクトリに対して、一貫した所有権と権限を保持するようにしてください。読み込み／書き込み／実行のパーミッションを持つ

Sybase システム管理者である 1 人のユーザが、インストール、アップグレード、設定のすべての作業を行ってください。

5. 1 つの物理ボックスのみに対してローカルなローカル・アカウントを使用している場合は、各マシンで sybase ユーザ・アカウントを作成し、同じユーザ ID を割り当てます。
6. Adaptive Server をインストールする場所を決定します。十分な空きディスク領域があることを確認します。ディレクトリのパス名にスペースが含まれないようにします。\$SYBASE の場所は、同じパスを使用してすべてのクラスタ・ノードからアクセスできる、共有ファイル・システムにする必要があります。
7. ホストに対して ping を実行して、ネットワーク・ソフトウェアが設定されていることを確認します。
8. 使用しているプラットフォームに応じて、オペレーティング・システムの共有メモリ・パラメータを調整します。これを行う方法については、「オペレーティング・システムの共有メモリ・パラメータの調整 (29 ページ)」を参照してください。
9. ホスト ID を取得します。「ホスト ID の取得 (30 ページ)」を参照してください。
10. 使用する SySAM モデルを決定します。「ライセンスの取得」を参照してください。
11. SPDC にアクセスします。「SPDC へのアクセス」を参照してください。
12. ライセンスを生成します。「ライセンスの生成ここでは、サブド・ライセンス・モデルまたはアンサブド・ライセンス・モデルを生成する方法について説明します。」を参照してください。
13. コマンド・ライン、GUI モード、またはサイレント・モードからインストーラを実行します。ウィザードに表示される、ライセンスとインストールに関する手順に従います。

表記規則

以下の表記規則が使用されています。

- サンプル・ウィンドウでは、表記されているとおりに入力する必要のあるコマンドを次の字体で示します。

```
this font
```

- サンプル・ウィンドウでは、インストール環境に応じた適切な値で置き換える必要のある語を次の字体で示します。

```
this font
```

- このマニュアルの本文では、ファイル名とディレクトリ名を次の字体で示します。

YusrYuYsybase

- プログラム、ユーティリティ、プロシージャ、コマンドの名前は次のように示します。

sqlupgrade

- Cシェルと Bourne シェルでコマンドが異なる場合は、両方を示します。Cシェルの初期化ファイルは `cshrc`、Bourne シェルの初期化ファイルは `.profile` と呼ばれます。Korn シェルなど、別のシェルを使用している場合、正しいコマンド構文については、使用しているシェル固有のマニュアルを参照してください。

表 2 : SQL の構文規則

キー	定義
command	コマンド名、コマンドのオプション名、ユーティリティ名、ユーティリティのフラグ、キーワードは太字の san-serif フォントで示す。
<i>variable</i>	変数(ユーザが入力する値を表す語)は斜体で表記する。
{ }	中カッコは、その中から必ず 1 つ以上のオプションを選択しなければならないことを意味する。コマンドには中カッコは入力しない。
[]	角カッコは、オプションを選択しても省略してもよいことを意味する。コマンドには角カッコは入力しない。
()	() はコマンドの一部として入力する。
	中カッコまたは角カッコの中の縦線で区切られたオプションのうち 1 つだけを選択できることを意味する。
,	中カッコまたは角カッコの中のカンマで区切られたオプションをいくつでも選択できることを意味する。複数のオプションを選択する場合には、オプションをカンマで区切る。

サーバ・コンポーネントの説明とディレクトリ・レイアウト

ここでは、サーバ・コンポーネントとディレクトリ・レイアウトについて説明します。

製品	説明
サーバ・インストール・パッケージ	ASE-15_0 ディレクトリにインストールされる。 <ul style="list-style-type: none">• Adaptive Server - データベース・サーバ。• Backup Server - すべてのデータベース・バックアップ (ダンプ) およびリストア (ロード) オペレーションを管理する Open Server ベースのアプリケーション。• XP Server - Adaptive Server 内から拡張ストアド・プロシージャ (ESP) を管理、実行する Open Server アプリケーション。• Job Scheduler - Adaptive Server 用のジョブ・スケジューラを提供する。Job Scheduler コンポーネントは ASE-15_0 / jobscheduler / の固有のディレクトリ内にある。

製品	説明
共有ディスク・クラスタの管理ツール	<ul style="list-style-type: none"> • Sybase Central™ 6.x - システム管理ツールで使用される Java ベースのフレームワーク。shared ディレクトリにインストールされる。 • Adaptive Server プラグイン - クラスタ設定を行い、完全な管理機能を提供する Sybase Central プラグイン。ASEP ディレクトリにインストールされる。 • sybcluster - \$SYBASE/\$SYBASE_ASE/bin ディレクトリにインストールされたクラスタを設定および管理するための、インタラクティブなコマンド・ライン・インタフェース。 • Job Scheduler テンプレートおよびユーティリティ - 時間効率のよい有益なジョブを作成し、スケジュール設定するためにデータベース管理者によって使用される事前に定義されたテンプレート。\$SYBASE/\$SYBASE_ASE/jobscheduler ディレクトリにインストールされる。 • Unified Agent サーバ - ASE プラグインと sybcluster からアクセスされる中間層管理アプリケーション・ロジック用のコンテナを提供する。UAF-2_5 ディレクトリにインストールされる。 • Agent Management Console - Unified Agent を管理する Sybase Central プラグイン。AMCP ディレクトリにインストールされる。 • Interactive SQL - クラスタを設定および管理するためのインタラクティブなコマンド・ライン・インタフェース。 \$SYBASE/\$SYBASE_OCS/bin ディレクトリにインストールされる。 • JRE - Java Runtime Environment の略語。Sybase Central のような Java ベースのプログラムを実行するためのランタイム Java 仮想マシン。shared ディレクトリにインストールされる。 • Cluster Edition 管理ユーティリティは、\$SYBASE/SDCADMIN-15_0 にインストールされる。

製品	説明
Software Developer Kit (SDK)	<p>OCS-15_0 ディレクトリにインストールされる。</p> <ul style="list-style-type: none"> • コネクティビティ： <ul style="list-style-type: none"> • Open Client™ (Client-Library、dblib) • Embedded SQL™/COBOL 15.0 • Adaptive Server 用 XA インタフェース <p>DataAccess ディレクトリにインストールされる。</p> <ul style="list-style-type: none"> • ODBC (Windows、Solaris SPARC、Solaris x64、Linux Power、HP-UX Itanium、IBM AIX) – ODBC ベースのアプリケーションから Adaptive Server への接続に使用されるドライバ。
言語モジュール	<p>\$SYBASE/locales と \$SYBASE_ASE/locales の両ディレクトリにインストールされる。システム・メッセージと日付/時刻のフォーマットを提供する。</p>
文字セット	<p>charsets ディレクトリにインストールされる。Adaptive Server で使用できる文字セットを提供する。</p>
照合順	<p>collate ディレクトリにインストールされる。Adaptive Server で使用できる照合順を提供する。</p>
Sybase ソフトウェア資産管理 (SySAM)	<p>SYSAM-2_0 ディレクトリにインストールされる。Adaptive Servers とオプション機能の資産管理を提供する。</p>
Java クライアント・ユーティリティ	<p>jutils-2_0 ディレクトリにインストールされる。次のツールを含む Adaptive Server ユーティリティ・プログラムのコレクション。</p> <ul style="list-style-type: none"> • ribo- クライアント・プログラムと Adaptive Server の間の TDS トラフィックをトレースするツール。
Java データベース・コネクティビティ (JDBC)	<p>jConnect-6_0_5 と jConnect-7_0 の各ディレクトリにインストールされる。Adaptive Server 用の JDBC (Java Database Connectivity) ドライバを提供する。</p>
アンインストーラ	<p>sybuninstall/ASESuite ディレクトリにインストールされる。</p>

PC クライアント製品の説明とディレクトリ・レイアウト

ここでは、PC クライアント製品とディレクトリ・レイアウトについて説明します。

製品	説明
Software Developer Kit (SDK)	<p>%SYBASE_OCS% ディレクトリにインストールされる。</p> <ul style="list-style-type: none"> • コネクティビティ： <ul style="list-style-type: none"> • Open Client (Client-Library、dlib) • Embedded SQL™/COBOL 15.0 • Adaptive Server 用 XA インタフェース <p>DataAccess ディレクトリにインストールされる。</p> <ul style="list-style-type: none"> • ODBC (Windows、Solaris SPARC、Solaris x64、Linux Power、HP-UX Itanium、IBM AIX) – ODBC ベースのアプリケーションから Adaptive Server への接続に使用されるドライバ。 • OLEDB (Windows のみ) - OLE DB ベースのアプリケーションから Adaptive Server への接続に使用されるプロバイダ。 • ADO.NET (Windows のみ) - .NET ベースのアプリケーションから Adaptive Server への接続に使用されるプロバイダ。
Windows Cluster Server Admin ユーティリティ	<p>Windows Cluster Administrator は GUI ツールで、Microsoft Cluster Server (MSCS) の管理に使用する。これを使用すると、グループ、リソース、およびクラスタ自体の情報を作成、修正、および表示できる。また、クラスタを管理する代替コマンドライン・ツール Cluster.exe もある。</p>

製品	説明
PC クライアント管理 ツール	<p>ASEPlugin.jar ファイルは、ASEP¥lib にありますが、ASE プラグインを使用して Sybase Central を起動するための scjview.exe 実行プログラムは、次の場所にあります。shared ¥Sybase Central 6.0.0¥win32.</p> <ul style="list-style-type: none"> • Sybase Central™ 6.x - システム管理ツールで使用される Java ベースのフレームワーク。Shared ディレクトリにインストールされる。 • Adaptive Server プラグイン - クラスタ設定を行い、完全な管理機能を提供する Sybase Central プラグイン。ASEP ディレクトリにインストールされる。 • sybcluster - %SYBASE%¥SDCADMIN-15_0¥bin ディレクトリにインストールされたクラスタを設定および管理するための、インタラクティブなコマンド・ライン・インタフェース。 • Job Scheduler テンプレートおよびユーティリティ - 時間効率のよい有益なジョブを作成し、スケジュール設定するためにデータベース管理者によって使用される定義済みテンプレート。%SYBASE%¥%SYBASE_ASE%¥jobscheduler ディレクトリにインストールされる。 • Agent Management Console - Unified Agent を管理する Sybase Central プラグイン。AMCP ディレクトリにインストールされる。 • Interactive SQL - クラスタを設定および管理するためのインタラクティブなコマンド・ライン・インタフェース。 %SYBASE%¥%SYBASE_ASE%¥bin ディレクトリにインストールされる。 • JRE - Java Runtime Environment の略語。Sybase Central のような Java ベースのプログラムを実行するためのランタイム Java 仮想マシン。%SYBASE%¥Shared¥JRE-6_0_6_32BIT ディレクトリにインストールされる。 • 管理ユーティリティは %SYBASE%¥SDCADMIN-15_0 にインストールされます。 • jutils-2_0 - 次のツールなど、Adaptive Server ユーティリティ・プログラムのコレクション。 <ul style="list-style-type: none"> • jisql - JDBC ベースの GUI isql ツール。

製品	説明
	<ul style="list-style-type: none"> • ribo - クライアント・プログラムと Adaptive Server の間の TDS トラフィックをトレースするツール。
言語モジュール	%SYBASE%\locales ディレクトリにインストールされる。システム・メッセージと日付/時刻のフォーマットを提供する。
文字セット	charsets ディレクトリにインストールされる。Adaptive Server で使用できる文字セットを提供する。
jConnect™	jConnect-6_0_5 と jConnect-7_0 の各ディレクトリにインストールされる。Adaptive Server 用の JDBC (Java Database Connectivity) ドライバを提供する。
アンインストーラ	sybuninstall/PCClient ディレクトリにインストールされる。

個別にインストールできる PC クライアント製品

PC-Client CD には、本製品とは別にインストールされる追加の製品がいくつか含まれています。

- InfoMaker® - 開発者とエンド・ユーザ用の、個人的なデータ・アクセス、管理、およびレポート用ツール。InfoMaker を使用すると、InfoMaker プレゼンテーション・レベルのレポートや高機能なクエリを作成できます。デスクトップ生産性スイートだけでなく、クライアントまたはサーバのビジネス・アプリケーションや、開発ツールを補完します。
- PowerDesigner® Physical Architect - データベースの設計、生成、保守、リバースエンジニアリング、データベース構築マニュアルなどのデータ・モデリング用のツール。
- PowerTransfer - サポートされているデータベースから Adaptive Server にテーブル・データを転送する PowerDesigner 用のプラグイン。

インストールのプランニング

ここでは、インストール手順のプランニングについて説明します。

注意：バージョン 15.5 Cluster Edition で、CPU ごとおよびチップごとのライセンス・タイプで使用されるライセンス数に変更されました。Adaptive Server は設定とは無関係に、マシン上のコア (ライセンス・タイプによってはチップ) の数と同じライセンス数をチェックします。この動作はライセンス契約における変更点ではなく、以前のバージョンの Adaptive Server の問題点を修正したものです。以前のバージョンでは、Adaptive Server が CPU ごとまたは CPU チップごとにライセンス供与された場合、**max online engines** 設定パラメータがマシン上の CPU 数より低いと、要求されるライセンス数が削減されていました。

注意：運用システムで Infiniband、Interconnect を使用する場合のハードウェア稼働条件については、『ユーザーズ・ガイド』を参照してください。Sybase では複数のノードでの稼働時におけるファイル・システム・デバイスには対応していません。

注意：Symantec の Storage Foundation for Sybase Cluster Edition でクラスタを実行する方法については、『Cluster ユーザーズ・ガイド』の「第 11 章 Veritas Cluster Server と Cluster Edition の使用」を参照してください。

Cluster Edition のデータベース・デバイスは、SCSI PGR (SCSI-3 Persistent Group Reservations) のサポートが必須です。Cluster Edition は、SCSI PGR を使用して、クラスタ・メンバシップの変更時のデータの一貫性を保証します。Sybase では、SCSI PGR をサポートしないディスク・サブシステム上のデータの一貫性を保証できません (このような設定は、潜在的なデータ破壊に耐性のあるテスト環境および開発環境に対してサポートされます)。

1. インストーラの実行対象とするノード上に \$HOME ディレクトリを作成します。
2. すべてのノードで同じバージョンのオペレーティング・システムが実行されていることを確認します。
プロセッサの数とメモリの量はノード間で変えることはできますが、オペレーティング・システムのバージョンは同一にする必要があります。
3. クォーラムが専用のデバイス上にあることを確認します。
4. Adaptive Server プラグインまたは **sybcluster** を使用して、共有デバイス上にローカル・システム・テンポラリ・データベースを作成します。クラスタの初期起動時と、それ以降のクラスタにインスタンスを追加した場合は常に、これを行う必要があります。

どのインスタンスにおいても、ローカル・システム・テンポラリ・データベースの作成または削除は可能ですが、アクセスできるのは所有インスタンスからのみです。

5. クォーラム・デバイスを含むすべてのデータベース・デバイスがロー・パーティションに配置されていることを確認します。Network File System (NFS) は使用しないでください。

警告！ クラスタに対してファイル・システム・デバイスを使用しない - Cluster Edition は、ファイル・システムで稼働するように設計されていません。複数のノードにノンクラスタード・ファイル・システムをマウントすると、直後に障害が発生し、クラスタおよびクラスタのデータベースがすべて失われます。このような理由により、Sybase では複数のノードでの稼働時におけるファイル・システム・デバイスには対応していません。

6. 各ノードから同じアクセス・パスを使用してロー・パーティションにアクセスできることを確認します。Sybase では、デバイスにストレージ・エリア・ネットワーク (SAN) を接続することをおすすめします。

注意： ローカル・ユーザ・テンポラリ・データベースは、共有領域を必要とせず、プライベート・デバイスとして作成されたローカル・ファイル・システムを使用できます。この点で、共有ストレージを必要とするローカル・システム・テンポラリ・データベースとは異なります。

テスト環境では、単一のノードまたはマシンを使用して、クラスタ設定内で Cluster Edition の複数のインスタンスを稼働してください。ただし、その場合、データベース・デバイスとしてローカル・ファイル・システム (非 NFS) または SAN Storage を使用する必要があります。

7. ノード内のクロックの同期を維持するために、すべてのハードウェア・ノードが Network Time Protocol (NTP) またはこれに類似するメカニズムを使用していることを確認します。
8. 共有インストールを使用している場合は、Adaptive Server Enterprise のソフトウェアおよび設定ファイル (\$SYBASE ディレクトリ、interfaces ファイルなど) はすべて、クラスタ内の各ノードから同じアクセス・パスを使用してアクセスできる Network File System (NFS) またはクラスタ・ファイル・システム (CFS または GFS) にインストールされている必要があります。クラスタ・ファイル・システムのサポートされるバージョンについては、次の項で説明します。プライベート・インストールを使用している場合は、各ノードがクラスタ・ファイル・システム上で独自のインストールを持つことが必要です。
9. 高速ネットワーク相互接続 (ギガビット・イーサネットなど) によって、クラスタに参加するすべてのハードウェア・ノードを接続するローカル・ネットワーク接続が形成されていることを確認します。
10. Sybase では、クラスタ内の各ノードに 2 つのネットワーク・インタフェースを物理的に離して構成することをおすすめします。

- プライマリ・ネットワーク-クラスタ相互接続トラフィック用。
- セカンダリ・ネットワーク-クラスタ相互接続トラフィック用。

プライマリとセカンダリ・ネットワークは、物理的に離す必要があり、これらが必要とされるのはセキュリティ、フォールト・トレランス、およびパフォーマンス上の理由によります。フォールト・トレランスに対しては、ネットワーク障害の場合にもクラスタが存続するように、2枚のネットワーク・カードを別々のファブリックに装着する必要があります。

11. プライベート相互接続ファブリックには、クラスタに参加していないマシンへのリンクを含めることはできません (すなわち、すべてのクラスタ・ノードは、プライマリ相互接続を同じスイッチに接続し、そのスイッチを他のスイッチまたはルータに接続することはできません)。

ライセンスの取得

単純なインストールの場合は、ここに示す手順に従って SySAM ライセンスを取得し、インストールします。

前提条件

- 製品をインストールする前に、Sybase 製品ダウンロード・センタ (SPDC) からライセンス・ファイルを取得しておきます。
- 使用する SySAM ライセンス・モデルを決定します。
 - アンサーブド・ライセンス・モデル-ライセンス・ファイルから直接ライセンスを取得します。アンサーブド・ライセンスを使用するには、Sybase 製品のインストール先と同じマシンにライセンスを保存します。
 - サーブド・ライセンス・モデル-ライセンス・サーバを使用して、複数マシンに対するライセンスの割り当てを管理します。

サーブド・ライセンスを使用するには、ライセンス・サーバが必要です。ライセンス・サーバは同じマシンで実行することも、オペレーティング・システムとアーキテクチャの異なる別のマシンで実行することもできます。

既存のライセンス・サーバを使用するには、そのサーバのホスト名とポート番号を把握する必要があります。

注意： 1台のマシンで実行できる SySAM ライセンス・サーバは1インスタンスのみです。既に SySAM 1.0 ライセンス・サーバを実行しているマシンで SySAM 2.0 ライセンス・サーバをセットアップするには、古いライセンス・サーバを SySAM 2 にマイグレートする必要があります。マイグレートされたライセンス・サーバは、SySAM 1.0 に対する製品と SySAM 2 に対する製品の両方にライセンスを提供できます。

以下の詳細については、『Sybase ソフトウェア資産管理ユーザーズ・ガイド』を参照してください。

- サード・ライセンスとアンサーブド・ライセンス。使用環境に適したライセンス・モデルを特定する方法。
- ライセンス・サーバ・ホストの選択。
- ライセンス・サーバの管理。
- ライセンス・サーバのマイグレート。
- 3 サーバ冗長性、高可用性、仮想化などを使用した複雑なインストールの場合は、ライセンス・モデルを正しく設定する方法について、『Sybase ソフトウェア資産管理ユーザーズ・ガイド』を参照してください。

SPDC へのアクセス

Sybase 製品を購入したら、オンライン Web ポータルの Sybase 製品ダウンロード・センタ (SPDC) からその製品をダウンロードしてライセンスを生成します。

1. Sybase 製品を注文すると、SPDC サイトの URL とユーザ名およびパスワードが記載された電子メールが届きます。

注意： Sybase 認定販売店から Sybase ソフトウェアをご購入いただいた場合は、電子メール・メッセージではなく Web キーが送付されます。

2. URL をクリックし、要求画面が表示されたら、指定のユーザ名とパスワードを入力します。ログイン名は、通常は電子メール・アドレスです。パスワードを忘れた場合は、SPDC ログイン・ページのパスワード検索機能を使用します。

サード・ライセンスの生成

ライセンス・サーバと必要なライセンスをセットアップしてから、Sybase 製品をインストールします。

前提条件

ホスト ID を決定してから、ライセンスを生成します。「SySAM ホスト ID の取得 (30 ページ)」を参照してください。

- 問題がある場合は、SySAM のトラブルシューティングに関するトピックを参照してください。

手順

1. SPDC サイトの [welcome] 画面から、ライセンスを生成する製品を含む製品ファミリを選択します。

2. Sybase 製品のリストから、適切な製品エディションとバージョンを選択します。使用しているオペレーティング・システムに対応する製品を選択したことを確認してください。
ライセンス契約が表示されます。
3. ライセンス契約の画面が表示されたら、国または地域を選択して該当するライセンス テキストを表示し、契約に目を通して、[同意する] をクリックします。

注意： ライセンス契約は、初めて製品をダウンロードしようとしたときのみ表示されます。一度ライセンス契約に同意すると、以後その製品をダウンロードするときにはライセンス契約は表示されません。

4. 製品のダウンロード・ページで、[ライセンス・キー] をクリックします。
ライセンス情報のページが表示されます。
5. 生成するライセンスを選択します。
会社の発注書で製品名、エディション、およびライセンスの種類を比較することで、正しいライセンスを生成していることを確認してください。

注意： お使いの製品にサーバ・ライセンスとクライアント・ライセンスの両方が必要な場合は、サーバ・ライセンスを先に生成することをおすすめします。

6. [選択して生成] をクリックします。
7. [サブド・ライセンス] を選択し、[次へ] をクリックします。
8. 生成するライセンスの数を入力し、[次へ] をクリックします。最大 10 台のマシンのライセンスを生成できます。
9. ライセンス・サーバのホスト ID と、オプションのホスト名およびポート番号を入力します。
 - a) 3 ノード冗長クラスタを使用している場合を除いて、ポート番号の指定は不要です。有効なポート番号は 0 ~ 64000 の未使用の番号です。UNIX では、1024 未満は権限付きのポート番号になるため、1024 より大きいポート番号を選択してください。TCP/IP ポート番号を指定しない場合は、27000 ~ 27009 の範囲内のデフォルト・ポートのいずれかが使用されます。
 - b) 3 ノード冗長クラスタのライセンスを生成する場合は、ホスト ID、ホスト名、およびクラスタの各ノードに対するポート番号を指定します。この設定では、ポート番号は必須であり、27000 ~ 27009 の範囲外にする必要があります。
 - c) 3 冗長サーバのためにサーバを設定する場合は、冗長サーバの情報を指定します。
10. [生成] をクリックします。

11. サーバ・ライセンスしか生成しておらず、クライアント・ライセンスも製品に必要な場合は、前述の手順5～10を繰り返してクライアント・ライセンスを生成します。
12. 製品に必要なライセンスがすべて生成されたら、次の操作を行うことができます。
 - [ホストのすべてのライセンスをダウンロード] ボタンをクリックしてライセンス・ファイルをダウンロードする。
 - [印刷用ページ] ボタンをクリックしてライセンスを印刷する。
 - ライセンス情報のページに戻って、追加のライセンスを生成する。
13. ライセンスを生成した後で、拡張子 .lic を付けたライセンスをライセンス・サーバのインストール・ディレクトリ licenses に保存します。

アンサーブド・ライセンスの生成

ここでは、アンサーブド・ライセンスを生成する方法について説明します。

前提条件

ホスト ID を決定してから、ライセンスを生成します。「SySAM ホスト ID の取得 (30 ページ)」を参照してください。

- 問題がある場合は、トラブルシューティングに関するトピックを参照してください。

手順

1. SPDC サイトの [welcome] 画面から、ライセンスを生成する製品を含む製品ファミリを選択します。
2. Sybase 製品のリストから、適切な製品エディションとバージョンを選択します。使用しているオペレーティング・システムに対応する製品を選択したことを確認してください。
ライセンス契約が表示されます。
3. ライセンス契約の画面が表示されたら、国または地域を選択して該当するライセンス テキストを表示し、契約に目を通して、[同意する] をクリックします。

注意： ライセンス契約は、初めて製品をダウンロードしようとしたときのみ表示されます。一度ライセンス契約に同意すると、以後その製品をダウンロードするときにはライセンス契約は表示されません。

4. 製品のダウンロード・ページで、[ライセンス・キー] をクリックします。
ライセンス情報のページが表示されます。
5. 生成するライセンスを選択します。

会社の発注書で製品名、エディション、およびライセンスの種類を比較することで、正しいライセンスを生成していることを確認してください。

注意： お使いの製品にサーバ・ライセンスとクライアント・ライセンスの両方が必要な場合は、サーバ・ライセンスを先に生成することをおすすめします。

6. [選択して生成] をクリックします。
7. [アンサーブド・ライセンス] を選択し、[次へ] をクリックします。
8. ライセンスの対象マシンの数を選択して、[次へ] をクリックします。最大 10 台のマシンのライセンスを生成できます。
9. ライセンスを生成する各マシンのホスト ID とオプションのホスト名を入力します。
一部のライセンス・タイプでは、そのライセンスを使用するマシンの CPU やライセンスの数を入力する必要があります。
10. [生成] をクリックします。
11. サーバ・ライセンスしか生成しておらず、クライアント・ライセンスも製品に必要な場合は、前述の手順 5～10 を繰り返してクライアント・ライセンスを生成します。
12. 製品に必要なライセンスがすべて生成されたら、次の操作を行うことができます。
 - [ホストのすべてのライセンスをダウンロード] ボタンをクリックしてライセンス・ファイルをダウンロードする。
 - [印刷用ページ] ボタンをクリックしてライセンスを印刷する。
 - ライセンス情報のページに戻って、追加のライセンスを生成する。
13. ライセンスを生成した後で、拡張子 `.lic` を付けたライセンスをライセンス・サーバのインストール・ディレクトリ `licenses` に保存します。

新しいライセンス・サーバのインストール

新しいライセンス・サーバをインストールする前に、お使いのプラットフォーム用の正しいオペレーティング・システム・パッチでシステムが更新されていることを確認してください。

1. SySAM ユーティリティとライセンス・サーバを別々にインストールするには、次の手順に従います。

- a) Java Runtime Environment 1.6 を `$PATH` に追加します。
C シェルの場合は、次のように入力します。

```
setenv PATH <JRE 1.6>/bin:$PATH
```

Bourne シェルの場合は、次のように入力します。

```
set PATH=<JRE 1.6>/bin:$PATH; export PATH
```

- b) CD ドライブのコマンド・プロンプトから、SySAM ユーティリティ・インストーラ・プログラムを起動します。次のように入力します。

```
./archives/sysam_util/setup.bin
```

- c) CD ドライブのコマンド・プロンプトから、SySAM ライセンス・サーバ・インストーラ・プログラムを起動します。次のように入力します。

```
./archives/sysam_server/setup.bin
```

2. [Welcome] ウィンドウが表示されます。[次へ] をクリックします。
3. ドロップダウン・リストから適切な国を選択します。
4. ライセンス契約を受け入れます。
5. インストール先のディレクトリを入力または選択します。
6. [Typical Installation Set] を選択します。
7. [インストール] をクリックしてインストールを進めます。
8. パネルが表示され、インストールが成功したかどうか報告されます。
9. ライセンス・サーバを起動します。

注意： \$SYBASE/SYSAM-2_0/licenses にある licenses ディレクトリに少なくとも 1 つの有効なサブド・ライセンスが存在しないと、ライセンス・サーバは起動できません。

10. サブキャパシティ・ライセンスを使用する予定がある場合は、**sysamcap** ユーティリティの設定方法について『SySAM ユーザーズ・ガイド』を参照してください。
11. ライセンス・サーバを実行しているマシンのホスト ID を取得します。ホスト ID の生成方法については、「ホスト ID の取得」を参照してください。
12. Sybase 製品ダウンロード・センタ (SPDC) にアクセスします。
13. インストールする製品のライセンスを生成します。ライセンスの生成方法については、「SPDC でのライセンスの生成」のトピックを参照してください。
14. コマンド・プロンプトからインストール・ディレクトリに移動し、環境変数を読み込みます。
 - C シェル

```
source SYBASE.csh
```
 - Bourne シェル

```
./SYBASE.sh
```
15. ネットワーク・ライセンス・サーバ・マシンの \$SYBASE/SYSAM-2_0/licenses ディレクトリにあるライセンス・ファイルをコピーします。
16. ライセンス・サーバをリフレッシュまたは再起動します。

1. 次のディレクトリに変更します。

```
$SYBASE/SYSAM-2_0/bin
```

2. ライセンス・サーバが起動済みでない場合は、次のように入力して起動します。

```
sysam start
```

ライセンス・サーバが起動済みの場合は、ライセンス・サーバが新しいライセンス・ファイルを読み込むために `reread` コマンドを使用します。

```
sysam reread
```

17. 次のように入力して、ライセンス・デーモンが実行していることを検証します。

```
sysam status
```

サーバが実行している場合の出力は、次のようになります。

```
lmutil - Copyright (c) 1989-2006 Macrovision Europe
Ltd. and/or Macrovision Corporation. All Rights
Reserved. Flexible License Manager status on Wed
5/24/2006 15:04
```

```
License server status: 27000@mysysamserver
License file(s) on keyserver:
/opt/sybase/SYSAM_0/licenses/mysysamserver_603020315.lic
```

18. SySAM ライセンス・サーバのインストーラによって IPv4 と IPv4/IPv6 デュアル・スタックの両方のライセンス・サーバ・バイナリがインストールされ、IPv4 専用バイナリの使用が設定されます。次の SySAM スクリプトを使用すると、適切なバージョンのライセンス・サーバを設定できます。

```
sysam configure [IPv6|IPv4]
```

たとえば、次のコマンドは IPv4/IPv6 デュアル・スタック・バイナリの使用を設定します。

```
sysam configure IPv6
```

19. ライセンス・サーバ・ホストで IPv6 を有効にしている場合は、IPv4 バージョンのライセンス・サーバを使用できません。この場合は、次の操作を行うことができます。

- ライセンス・サーバ・ホストで IPv6 TCP/IP プロトコルを無効にします。
- IPv6 を有効にしていない別の Windows ホストを使用します。
- ライセンス・サーバに Unix ホストを使用します。Unix 上のライセンス・サーバは、IPv4 と IPv6 の両方のクライアント・ホストに対してライセンス処理を実行できます。
- ライセンス・サーバに Windows ホストを使用する必要があり、ライセンス・サーバに対して IPv4 と IPv6 の両方のクライアントがある場合、IPv4

ネットワーク用と IPv6 ネットワーク用の 2 つのライセンス・サーバを設定してください。

Windows ホストで IPv6 バージョンのライセンス・サーバを使用している場合は、IPv6 プロトコルを介してのみこのライセンス・サーバにアクセスできます。IPv4 専用のネットワーク・スタックを持つホストは、この IPv6 ライセンス・サーバからライセンスを取得できません。

Windows 7.0、Vista、または Windows 2008 の Windows サービスから SySAM サーバを起動する場合は、追加の手順なしでサーバが正常に起動します。

電子メール通知の有効化と変更

sp_lmconfig を使用して電子メール通知を設定します。現在の通知ステータスと構成を表示するには、**sp_lmconfig** コマンドをパラメータなしで実行します。

電子メール通知を有効にするには、**smtp host**、**smtp port**、**email sender**、**email recipients**、**email severity** の各プロパティを設定します。

電子メールのトリガとなる重大度を変更するには、**email severity** プロパティを "ERROR"、"WARNING"、"INFORMATIONAL"、または "NONE" に設定します。

電子メール受信者のリストを変更するには、**email recipients** プロパティをカンマ区切りの電子メール・アドレスのリストに設定します。電子メール・アドレスには、SMTP メッセージを受信できるものすべてを含めることができます。

サーバのシステム要件

Adaptive Server をインストールする前に、システムが最新のパッチで更新され、システム要件が満たされていることを確認します。使用しているオペレーティング・システムで下に示されているバージョンより前のパッチは使用しないでください。ただし、オペレーティング・システムのベンダが推奨する新しいパッチは、このリストにない場合でも適用してください。

現在インストールされているすべてのパッチをリストし、オペレーティング・システムのバージョン・レベルを表示するには、次のように入力します。

- Linux の場合：

```
rpm -q -a
```

注意： Java 仮想マシン (JVM) および関連した Adaptive Server サポートでは、起動に 250MB 以上の仮想メモリ領域が必要です。個々の Java 実行条件によっては、ここに示すよりも多くのメモリが必要となる場合があります。Java に十分な仮想メモリ領域があり、Adaptive Server と Java の両方のタスクが正常に共存できるように

するため、Java の実行時にメモリ・パラメータを調整する必要が生じることがあります。

特に、Adaptive Server の合計メモリが 1.5GB (一部の Enterprise サーバでは 2.5GB) より大きい値に設定されているシステムでは、問題が発生する可能性があります。その場合は、Adaptive Server の合計メモリを少なくする必要があります。

表 3 : Linux の Adaptive Server の仕様

ハードウェアとメモリ	
プロセッサ	AMD Opteron プロセッサまたは EM64T サポート付き Intel Xeon
Adaptive Server に必要な RAM の最小容量	128MB
デフォルトのユーザ・スタック・サイズ	86KB
追加ユーザ 1 人あたりに必要な RAM の最小容量	約 233KB

表 4 : Linux のオペレーティング・システム要件

ハードウェア	オペレーティング・システム	更新	優先的に使用される RAM
x86_64 プロセッサ (AMD Opteron または EM64T 対応 Intel Xeon)	RedHat Enterprise Linux 4.0 <ul style="list-style-type: none"> Kernel : 2.6.9-55.EL Glibc : 2.3.4-2.36 e2fsprogs-1.35-12.4.EL4 (libuuid.so.1 向け) libaio-0.3.105-2 libaudit-libs-1.0.14-1.EL4 libgcc-3.4.6-3 libstdc++3.4.6-3 (libstdc++.so.6 向け) libpam.so.0 	更新 : 5	1G 以上

ハードウェア	オペレーティング・システム	更新	優先的に使用される RAM
x86_64 プロセッサ (EM64T 対応 Intel Xeon)	RedHat Enterprise Linux 5.0 <ul style="list-style-type: none"> kernel-2.6.18-53.el5 glibc-2.5-18 e2fsprogs-lib-1.39-8.el5 (libuuid.so.1 向け) libaio-0.3.106-3.2 audit-1.5.5-7.el5 libgcc-4.1.2-14.el5 libstdc++-4.1.2-14.el5 libpam.so.0.81.5 libnsl-2.5 	更新 : 1	1G
x86_64 プロセッサ (AMD Opteron または EM64T 対応 Intel Xeon)	SuSE Linux Enterprise Server 9 <ul style="list-style-type: none"> kernel-smp-2.6.5-7.244 glibc-2.3.3-98.61 e2fsprogs-1.38-4.9 libaio-0.3.102-1.2 libgcc-3.3.3-43.41 libstdc++-3.3.3-44.41 	更新 : 3	1G
x86_64 プロセッサ (AMD Opteron または EM64T 対応 Intel Xeon)	SuSE Linux Enterprise Server 10 <ul style="list-style-type: none"> kernel-smp-2.6.16.46-0.12 glibc-2.4-31.30 e2fsprogs-1.38-25.21 libaio-0.3.104-14.2 libgcc-3.3.3-43.41 libstdc++-3.3.3-44.41 compat-libstdc++-5.0.7-22.2 	更新 : 1	1G

表 5 : Linux の最小ディスク領域要件

プラット フォーム	標準インス トール	デフォルト・ データベース	管理機能	必要な合計ディス ク領域
Linux x86-64	794MB	150MB	25MB	969MB

表 6 : Linux のクラスタ・ファイル・システム

オペレーティング・システム	製品
RHEL 4、5	Global File System 6.1
SuSE Enterprise 9、10	OCFS2

プライベート相互接続技術を使用するためのシステム稼働条件

Cluster Edition では、プライベート相互接続の際に UDP ネットワーク・プロトコルしか使用できません。TCP ネットワーク・プロトコルは使用しないでください。

プライベート相互接続とは、ノード間通信を可能にする物理的接続のことであり、共有ディスク・クラスタ・インストールの必須コンポーネントです。プライベート相互接続は、イーサネットでは単純なクロスオーバー・ケーブル、特別な専用通信プロトコルでは複雑な専用相互接続で実現できます。設定するノード数が 3 以上の場合、クラスタ内のノード間の高速通信を可能にするスイッチが必要です。

競合が原因で発生するトラフィック量を処理するには、スケーラブルな相互接続テクノロジーを使用してノードを接続します。トラフィック量は、インスタンス間の更新と送信の量に正比例します。使用できる最大の帯域幅を使用して、遅延が最低になる相互接続を実装することをおすすめします。

Sybase では、Linux 環境で 1GB イーサネットの相互接続帯域幅を使用することをおすすめします。

Cluster Edition は、最新の相互接続基準に対応しています。Sybase では、利用可能な相互接続を調査して、サイトに最も有効な相互接続を探することをおすすめします。

Cluster Edition では、Infiniband in IP over IB (internet protocol over Infiniband) モードをサポートしています。サーバは標準的な IP インタフェースを使用して Infiniband 相互接続と通信します。このモードが最も簡単に設定できます。

クライアントのシステム要件

ここでは、PC クライアントのシステム要件について説明します。

表 7: PC クライアントのオペレーティング・システム要件

製品	ハードウェア	オペレーティング・システム	更新	推奨される RAM 最小容量
PC クライアント	P4 1.0GHz	WinXP Pro SP2、Win2003 Server SP1、WinVista Business Edition		512MB

注意： ODBC、OLE DB、または ADO.NET の各ドライバを使用している場合は、Microsoft .NET Framework 2.0 Service Pack 1 がシステムにインストールされている必要があります。このサービス・パックがインストールされていないと、インストールの途中でエラーが発生し、ドライバが正しく機能しなくなります。

このサービス・パックがインストールされているかどうかを調べるには、次の手順に従います。

- コントロールパネルから [プログラムの追加と削除] を開きます。[現在インストールされているプログラム] に ".NET Framework" が表示されている場合は、システムを更新する必要はありません。

サーバ・インストールのインストール前の作業

サーバをインストールする前に、次の作業を行ってください。

1. インストールを開始する前に、テンポラリ・ディレクトリに 1 ギガバイト以上の空き領域があることを確認します。
2. 各ノードに適切なオペレーティング・システムのパッチをインストールします。
3. Linux システム上にインストールしており、環境変数 LD_ASSUME_KERNEL が設定されている場合、インストール前にこの変数の設定を解除します。
4. ソフトウェアのインストールを開始する前に、旧バージョンの Cluster Edition を使用しているすべてのクラスタをシャットダウンします。
5. インストーラの起動で問題が発生する場合は、Java Runtime Environment (JRE) バージョン 6 に対する必要なオペレーティング・システム・パッチが適用されていることを確認してください。必要なオペレーティング・システムのパッチについては、Sun Java Web サイトを参照してください。
6. すべてのインストール作業を行うための "sybase" アカウントをシステム上に作成します。
 - a) このアカウントは、すべてのデバイスとファイルを所有し、クラスタで使用されるすべてのデバイスの読み込み／書き込みパーミッションを所有している必要があります。
 - b) この "sybase" アカウントを使用して、すべてのデバイスを作成します。
 - c) クラスタ内のすべてのノードから全ディスク・デバイスにアクセスできることを確認します。
 - d) クラスタの起動に使用されるアカウントに、すべてのディスク・デバイスに対する読み込みと書き込みのパーミッションがあることを確認します。
 - e) クラスタと SCSI 汎用ドライバに、設定されたデータベース・デバイスに対応する /dev/sg* ファイルへの書き込みパーミッションがあることを確認します。
SCSI ドライバには、IO フェンシングで使用される SCSI-3 PGR コマンド用の /dev/sg* ファイルに対する書き込みアクセス権が必要です。
7. 現在のシェルに適切な継承可能な権限があることを確認します。

```
sudo lssecattr -p $$
487528 eprivs= mprivs= iprivs=PV_KER_RAS lprivs=PV_ROOT
uprivs=
> $SYBASE/ASE-15_0/bin/iofenceutil /dev/rhdisk2
/dev/rhdisk2
```

- a) フェンス機能に対応していない場合、現在のシェルに適切な継承可能な権限を付与します。

```
sudo setseccattr -p iprivs=+PV_KER_RAS $$  
Then restart UAF
```

8. **ls -l** コマンドを使用して、パスやファイルのパーミッションを検証します。
9. **dd** ユーティリティを使用して、Sybase アカウントがデバイスに対して読み込みおよび書き込み可能であることを確認します。
10. パーミッションを変更する必要がある場合は、**chmod** または **chown** を使用して `/dev/sg*` ファイルに対する書き込みのパーミッションを訂正します。マシンをリブートした後でのみこれらのファイルに対するアクセス・パーミッションを `root` に変更できます。書き込みのパーミッションは、サーバを起動する前に訂正する必要があります。
11. "sybase" ユーザとして、使用しているマシンにログインします。すべてのファイルおよびディレクトリに対して、一貫した所有権と権限を保持するようにしてください。読み込み/書き込み/実行のパーミッションを持つ Sybase システム管理者である 1 人のユーザが、インストール、アップグレード、設定のすべての作業を行ってください。
12. オープンな管理権限がない場合は、インストール先ディレクトリを作成してから `InstallAnywhere` を実行する必要があります。
13. 1 つの物理ボックスのみに対してローカルなローカル・アカウントを使用している場合、各マシンで sybase ユーザを作成し、必ず同じユーザ ID を割り当てます。
14. Adaptive Server ソフトウェアがインストールされる場所を決定します。
 - a) 使用できるディスク領域とテンポラリ・ディレクトリ領域が十分にあることを確認します。
 - b) ディレクトリのパス名にスペースが含まれないようにします。
 - c) これが共有インストールの場合、`$SYBASE` の場所は、どのクラスタ・ノードからも同じパスを使ってアクセスできる共有ファイル・システム上に設定する必要があります。
 - d) これがプライベート・インストールの場合は、クラスタ内の各ノードに Adaptive Server ソフトウェアをインストールします。
 - e) クラスタ内のインスタンスごとに専用の `$SYBASE` ディレクトリが必要です。

プライベート・インストール・モードでは、Network File System (NFS) とクラスタ・ファイル・システムのいずれも使用されません。
15. オペレーティング・システムが、使用しているプラットフォームのバージョン・レベル、RAM、ネットワーク・プロトコルの稼働条件を満たしていることを確認します。
16. ネットワーク・ソフトウェアが設定されていることを確認します。Cluster Edition は、クラスタに含める予定のノード向けにネットワークを構成すること

を暗黙的に要求します。接続に問題がある場合、またはネットワーク設定を確認する場合は、ホストに対して ping を実行します。

17. 使用しているプラットフォームに応じて、オペレーティング・システムの共有メモリ・パラメータを調整します。これを行う方法については、「オペレーティング・システムの共有メモリ・パラメータの調整 (29 ページ)」を参照してください。

オペレーティング・システムの共有メモリ・パラメータの調整

Adaptive Server が単一セグメントとしてラージ・メモリを取得できない場合、またはセグメント不足のために Backup Server ストライプに失敗する場合、共有メモリ・セグメントを調整します。

バックアップ (**dump**) とリカバリ (**load**) に使用するデバイスの数とタイプによっては、オペレーティング・システム設定ファイルの共有メモリ・セグメント・パラメータを調整し、同時実行型 Backup Server プロセスに対応できるようにする必要があります。プロセスの接続機構に使用できるデフォルトの共有メモリ・セグメント数は 6 です。

sp_configure による再設定によって追加のメモリが必要になる場合、Adaptive Server は起動後に共有メモリ・セグメントを割り付けます。この追加セグメントを考慮して、**allocate max shared memory** 設定パラメータを使用して使用可能な最大メモリを Adaptive Server に割り付けます。詳細については、『システム管理ガイド』を参照してください。

1. オペレーティング・システムの共有メモリ・パラメータを確認、調整するには、**sysctl(8)** メソッドを使用します。

現在の共有メモリ・サイズを確認するには、次のように入力します。

```
# /sbin/sysctl kernel.shmmax
```

実行時の共有メモリ・サイズを調整するには、次ように入力します。1GB の共有メモリの場合、*nnn* は 1073741824 になります。

```
# /sbin/sysctl -w kernel.shmmax=nnn
```

注意： 以前のリリースの Linux の中には、オペレーティング・システムの共有メモリがデフォルトで 32 MB のものがあります。Adaptive Server には、2K ページを使用するデフォルトのサーバで最低 64MB が必要です。Adaptive Server の **total memory** を増やす予定がある場合は、さらに多くの共有メモリが必要です。

2. システムを起動するたびにこの値を確実に適用するには、`/etc/sysctl.conf` ファイルを編集します。
3. Linux RHEL Update 4.0 以降では、Adaptive Server バージョン 15.x を複数エンジンで実行する場合はセキュリティ機能 Exec-Shield を無効にする必要があります。Exec-Shield を無効にする手順は、次のとおりです。

- a) `/etc/sysctl.conf`: 次の行を追加します。

```
kernel.exec-shield=0
kernel.exec-shield-randomize=0
```

RHEL Update 5.0 の場合は次の行を追加します。

```
kernel.exec-shield=0
kernel.randomize_va_space=0
```

- b) 有効にするアクションのスーパーユーザ (root) として、次のように入力します。

```
/sbin/sysctl -P
```

詳細については、Red Hat の Web サイトを参照してください。

4. スタック・サイズを **unlimited** に設定して、共有メモリの接続問題を回避します。これを設定するには、Bourne シェルでは **ulimit -s unlimited**、C シェルでは **limit stacksize unlimited** を使用します。
5. SHMMAX パラメータの調整後に、サーバの再起動に失敗する場合、別のカーネル・パラメータ SHMALL の値を大きくする必要があります。このパラメータは、割り付け可能な共有メモリの最大容量を指定します。この値は `/etc/sysctl.conf` ファイルで修正できます。
shmalls を大きくする構文は、次のとおりです。

```
# /sbin/sysctl -w kernel.shmall=nnn
```

SySAM ホスト ID の取得

ここでは、SySAM ホスト ID を取得する方法について説明します。

1. `$$SYBASE/SYSAM-2_0/bin`, ディレクトリに変更します。`$$SYBASE` は Sybase のインストール・ディレクトリです。
2. サード・ライセンスのホスト ID を取得するには、次のように入力します。

注意：一部のプラットフォームでは、ホスト ID はネットワーク・アダプタ・アドレスから派生します。複数のネットワーク・アダプタがある場合、`lmutil lmhostid` はそれぞれのネットワーク・アダプタに対して1つのホスト ID を返します。出力は次のようになります。

```
The FLEXlm host ID of this machine
is "0013023c8251 0015c507ea90"
Only use ONE from the list of hostids.
```

これらのホスト ID から 1 つを選択します。主要な有線の Ethernet アダプタに関連付けられた値を使用することをおすすめします。内部ループバック・アダプタに関連付けられた値は、使用しないでください。lmutil lmhostid の出力で使用するホスト ID を決定できない場合は、適切なネイティブ・オペレーティング・システムのコマンドを使用して詳細情報を取得してください。SPDC に関するよくある質問のトピック「ホスト ID の概要」を参照してください。

3. SPDC で後で使用できるように、ホスト ID を記録しておきます。

- ホスト ID はプラットフォーム固有のものですが、およそ次のようになります。
 - 00400516E525 (Linux プラットフォームの場合)
 - 170a3472 (Solaris プラットフォームの場合)
 - fb491d9 (AIX プラットフォームの場合)
 - ID_STRING= 972179e8-7b2b-11d7-bd55-4a651849415a (HP Itanium の場合)

サーバ・インストールのインストール前の作業

サーバのインストール

システムに作業用 Adaptive Server がある場合、Cluster Edition を同じ \$SYBASE ディレクトリにインストールしないでください。同じディレクトリに他の製品をインストールしなければならない場合、Cluster Edition を最後にインストールしてください。

警告！ Adaptive Server 15.5 ではインストールに InstallAnywhere を使用しますが、それ以前のバージョンの Adaptive Server ではその他の Sybase 製品と同様にインストールに InstallShield Multiplatform を使用します。この2種類のインストーラは競合し、必須ファイルが上書きされてもメッセージは表示されません。したがって、これらの異なるインストーラを使用して同じディレクトリに製品をインストールしないでください。

警告！ Cluster Edition を以前のバージョンの同じコンポーネントと同じディレクトリにインストールすると、古い方のバージョンが上書きされます。Cluster Edition の後に他の製品をインストールすると、1つ以上の製品が正しく動作しないことがあります。

プライベート・インストールを実行する場合は、クラスタ内の各ノードの Adaptive Server を専用のディレクトリにインストールすることをおすすめします。

InstallAnywhere では 508 の機能がサポートされますが、制限事項があります。次の制限事項に注意してください。

- [インストール・セットを選択します] 画面 - インストーラで「ホット・キー」を使用してインストールの種類を選択できません。フォーカスを変更するには [Tab] キー、選択するには スペース・キーを使用します。
- [製品フィーチャーの選択] 画面 - 機能の選択にはマウスを使用します。キーボードは使用できません。

GUI モードでのサーバのインストール

ここでは、サーバ・コンポーネントを GUI モードでインストールする方法について説明します。

Cluster Edition をプライベート・インストール・モードでインストールする場合、クラスタ内の各インスタンス用のインストーラを使用してください。これにより、環境変数やソフト・リンクなどが各インスタンスに対して正しく設定されません。

Sybase では、製品をシステム管理者としてインストールすることをおすすめします。ただし、root パーミッションがなくてもインストーラを実行できます。インストーラは、必要に応じて対象ディレクトリを作成し、選択したコンポーネントをすべてそのディレクトリにインストールします。インストール作業の最後に、製品のインストール状態を確認できます。製品を使用するために、さらに追加の設定が必要な場合もあります。

インストーラの終了後に SYBASE.csh スクリプト・ファイルを実行して、Adaptive Server 製品に必要な環境変数を読み込みます。

1. インストーラを実行する前に、すべてのプログラムを停止します。
2. Adaptive Server CD を挿入するか、Sybase 製品ダウンロード センタ (SPDC) から Adaptive Server インストール・イメージをダウンロードして解凍します。
3. インストーラを起動します。CD ドライブを使用する場合は、次のように入力します。

```
/cdrom/setup.bin
```

4. インストール・ディレクトリを選択します。[次へ] をクリックして、次に示すデフォルトを受け入れます。

```
/opt/sybase
```

5. 既存のサーバを更新している場合は、この手順は表示されません。実行するインストールの種類を選択します。:

- 標準
- フル
- カスタム

6. [カスタム] を選択した場合、[Products and Features Selection] ウィンドウが表示されます。製品名の横のボックスをチェックして、インストールする製品を選択します。一部の機能は他の機能と依存関係があります。このため、他の項目を先に解除しないと、項目の選択を解除できないことがあります。

7. 既存のサーバを更新している場合は、この手順は表示されません。Adaptive Server Suite のインストールの種類を選択します。選択できる種類はプラットフォームごとに異なります。

- Licensed copy of Adaptive Server Suite : Adaptive Server のライセンス済みコピーがある場合は、これを選択します。
- Evaluation Edition of Adaptive Server Suite : Adaptive Server を評価するには、これを選択します。評価版を選択した場合、ソフトウェアは最初のインストール日から 30 日間動作します。

8. ドロップダウン・リストから該当する国を選択し、ライセンス条件を読み、[同意する] をクリックします。[次へ] をクリックします。

9. ここでライセンス・キーを入力します。ライセンス・キーがない場合、ライセンス・キーなしで続行するを選択し、[次へ]をクリックしてサマリ画面を表示します。ただし、サブド・ライセンスをインストールする場合は、[次へ]をクリックしてライセンス・サーバを設定します。
10. Cluster Edition のライセンス・タイプを選択します。

評価版をインストールすることを選択した場合、この手順は表示されません。

 - **CP** – CPU ライセンス
 - **DT** – 開発とテスト用のライセンス
 - **AC** – OEM アプリケーション配備 CPU ライセンス
 - **LP** – アプリケーション固有の CPU ライセンス
 - **不明** – ライセンスなし
11. 電子メールによる通知をサーバに設定すると、介入が必要なライセンス管理イベントが発生した際に、指定したユーザに通知が送信されます。次の情報を入力します。

評価版をインストールすることを選択した場合、この手順は表示されません。

 - SMTP サーバ・ホスト名
 - SMTP サーバのポート番号
 - 返信先の電子メール・アドレス
 - 受信者の電子メール・アドレス
 - 電子メール・メッセージをトリガするメッセージ重要度
12. [Product Selection Summary] ウィンドウに、選択した内容が表示されます。正しいインストール・タイプを選択し、プロセスの完了に十分なディスク領域があることを確認します。[次へ]をクリックします。
13. [Install Progress] ウィンドウにインストールの進行状況が表示されます。[Install Status] ウィンドウにインストール・プロセスの結果が表示されます。
14. ASE プラグインにパスワードを記憶させるかどうかを選択できます。[有効化]または[無効化]を選択します。[次へ]をクリックします。
15. 最後に[インストール完了]ウィンドウが表示されます。エラーが発生した場合は、『トラブルシューティングおよびエラー・メッセージ・ガイド』を参照してください。インストールが正常に終了したら、[Done]をクリックします。

応答ファイルの作成

サイレント（「無人」）インストールを実行するには、`InstallAnywhere` を実行し、指定したインストール設定が含まれる応答ファイルを指定します。

GUI モードまたはコンソール・モードでインストールするときに応答ファイルを作成するには、`-r` コマンド・ライン引数を指定します。`-r` 引数を指定することで、

インストール・ウィザードのプロンプトへの応答が記録され、InstallAnywhere ウィザードの終了時に応答ファイルが作成されます。応答ファイルは編集可能なテキスト・ファイルであり、後続のインストールで使用する前に応答を変更できます。

GUI インストール時に応答ファイルを生成するには、次のコマンドを実行します。

```
setup.bin -r responseFileName
```

-r responseFileName は省略可能です。指定した場合は、インストール・セッションが応答ファイルに記録されます。

responseFileName は、応答ファイル用に選択する次のようなファイル名の絶対パスです。

```
/tmp/responsefile.txt
```

注意： 指定したディレクトリ・パスがすでに存在している必要があります。

コマンド・ラインからのインストール

コンポーネントをコンソール・モードでインストールする手順は、`setup` コマンドを使用してコマンド・ラインからインストーラを実行する点と、テキストを入力してインストール・オプションを選択する点を除き、「GUI モードでのインストール」で説明した手順と同じです。ここでは、コマンド・ラインからインストールする方法について説明します。

1. コマンド・ラインで次のように入力します。

```
setup.bin -i console
```

インストール・プログラムが起動します。

2. インストール・プログラムが自動的に起動する場合は、[キャンセル] を選択して GUI インストールを停止し、端末またはコンソールからセットアップ・プログラムを起動します。

インストール作業の流れは通常の GUI インストールの場合と同じです。ただし、表示は端末ウィンドウに出力され、応答はキーボードを使用して入力します。

サイレント・モードでのサーバのインストール

インストーラを GUI モードで実行し、応答を応答ファイルに記録した後で、ファイルを編集して応答をカスタマイズします。次に、サイレント・モードでインス

トーラを実行するときに、この応答ファイルに基づいてインストーラからの質問に答えます。

前提条件

コンソール・インストールまたは GUI インストール時に次のコマンドを使用します。

```
setup.bin -r responseFileName
```

これにより、インストール応答ファイルが生成されます。
サイレント・モードでインストールするには、次のコマンドを実行します。

```
setup.bin -f responseFileName -i silent  
-DAGREE_TO_SYBASE_LICENSE=true -DRUN_SILENT=true
```

responseFileName には、選択したインストール・オプションを含むファイル名の絶対パスを入力します。

注意：サイレント・モードでのインストール時に、Sybase ライセンス契約に同意する必要があります。次のどちらかを実行します。

- 次のオプションをコマンド・ライン引数に含めます。

```
-DAGREE_TO_SYBASE_LICENSE=true -DRUN_SILENT=true
```

- 応答ファイルを編集して、次のプロパティを含めます。

```
AGREE_TO_SYBASE_LICENSE=true  
RUN_SILENT=true
```

GUI 画面がないことを除けば、InstallAnywhere の動作はすべて同じです。サイレント・モードのインストール結果は、GUI モードで同じ応答を行った場合とまったく同じになります。

クラスタ・サーバのアンインストール

アンインストール・プロセスでは、インストーラによって実行されたアクションが削除されます。インストール後に作成されたファイルやレジストリ・エントリは削除されません。インストール・プロセスの完了後に手動で削除する必要があります。

Cluster Edition をアンインストールするには、次のコマンドを実行します。

```
$$SYBASE/sybuninstall/ASESuite/uninstall
```

既存の Adaptive Server のアンインストール

既存のサーバを削除するには、次の手順に従います。

前提条件

サーバを停止してから、次の手順に従います。

手順

1. 既存の Adaptive Server を削除するには、\$SYBASE から、次のように入力します。

```
rm servername.*
```

2. ディレクトリ \$SYBASE/\$SYBASE_ASE/install に移動して、次のコマンドを実行します。

```
rm RUN_servername.*
```

```
rm servername.*
```

3. \$SYBASE/interfaces ファイルを編集して、Adaptive Server へのすべての参照を削除します。
4. データベース・デバイス用のオペレーティング・システム・ファイルがある場合は、すべて削除します。

クラスタの作成と起動

ここでは、クラスタ・サーバの設定方法と起動方法について説明します。これらの操作はソフトウェアをインストールしたあとで実行します。

Sybase では、Adaptive Server プラグインまたは **sybcluster** ユーティリティを使用して共有ディスク・クラスタを作成して管理することをおすすめします。ただし、クラスタは手動で設定して管理することもできます。その方法については、「Manually Setting Up and Managing the Cluster (161 ページ)」のトピックを参照してください。

基本設定手順の概要

ここでは、クラスタ・サーバを設定するための基本手順について説明します。

1. 「インストールを開始する前に (40 ページ)」を参照してください。
2. \$SYBASE 環境を設定し、クラスタの各ノードで Unified Agent を起動します。
3. 次のいずれかを使用して、クラスタを設定します。
 - Adaptive Server プラグイン
 - **sybcluster** ユーティリティ
4. オプション - 補助サーバ(Monitor Server、XP Server、Backup Server、Job Scheduler) を設定します。
5. インストールと起動に失敗する場合は、再インストールを行う前に、この項の最後に説明されている手順に従ってください。

共有インストールの概要

共有インストールでは、クラスタ内の Adaptive Server の各インスタンスによって次の項目が共有されます。

- すべてのデータベースおよびデータベース・デバイス (たとえば、すべてのインスタンスは同じ *master* データベースを共有します)。
- 共通の \$SYBASE ディレクトリ
- すべてのサーバ・バイナリ (dataserver など) およびすべてのスクリプト (installmaster など)
- クラスタ入力ファイルを含むすべての設定ファイル
- interfaces ファイル (サーバ検索に LDAP が使用されない場合)
- インスタンスとクラスタとの調整に使用されるクォーラム・デバイス

プライベート・インストールの概要

プライベート・インストールでは、クラスタ内の Adaptive Server の各インスタンスによって次のディレクトリとファイルが共有されます。

- すべてのデータベースおよびデータベース・デバイス (たとえば、すべてのインスタンスは同じ *master* データベースを共有)
- インスタンスとクラスタとの調整に使用されるクォーラム・デバイス

プライベート・インストールの設定では、各インスタンスによって次の項目が独自に管理されます。

- \$SYBASE ディレクトリ
- interfaces ファイル (サーバ検索に LDAP が使用されない場合)
- サーバ設定ファイル

注意： 各インスタンスは専用のサーバ設定ファイルを持ちますが、クォーラム・デバイスにはマスタ設定ファイルが保持されます。Adaptive Server は、マスタ設定ファイルを使用して、各インスタンスのサーバ設定ファイルを常に最新の状態に保ちます。

インストールを開始する前に

作業を始める前に、次の条件を決定します。

- クラスタ・インストールで共有 (NFS) インストールとプライベート・インストールのどちらがサポートされるか。
- クラスタがサーバ検索情報用に LDAP ファイル (プライベート・インストールの場合に推奨) と interfaces ファイルのどちらを使用するか。
- クラスタがサードパーティ製 JVM を使用してデータベースでの Java をサポートするかどうか。
- クラスタがクラスタード環境でアプリケーション・サーバを管理するために Veritas Cluster Server (VCS) をサポートするかどうか。
- クラスタが単一の Backup Server と複数の Backup Server のどちらを使用するか。
- \$SYBASE インストール・ディレクトリのロケーション。これがプライベート・インストールの場合は、インスタンスごとに別々の \$SYBASE インストール・ディレクトリを指定する必要があります。
- クラスタ名。
- インスタンスの数とインスタンス名。

- クラスタが実行されるネットワークの DNS (Domain Name Service)。Unified Agent と **sybcluster** ユーティリティは、各ノードの DNS エントリが正しく入力されていない場合、正しく機能しません。
- クラスタ内のエージェントの数。Sybase では、クラスタ内のインスタンスごとに 1 つのノードを使用し、インスタンスに複数のエンジンを使用することをおすすめします。
- 各データベース・デバイスで使用されるロー・デバイス。Cluster Edition の場合、すべてのデバイス、データベース・デバイス、およびクォラム・デバイスを、共有ディスク上にロー・デバイスとして作成する必要があります。ローカル・ユーザ・テンポラリ・データベースは、共有領域を必要とせず、プライベート・デバイスとして作成されたローカル・ファイル・システムを使用できます。ただし、クラスタ設定の間に作成するローカル・システム・テンポラリ・データベースは、共有ディスクのみ使用します。一時データへのローカル・ディスクの使用方法の詳細については、『Cluster ユーザーズ・ガイド』の「第 8 章 テンポラリ・データベースの使用」を参照してください。
- 各インスタンスで、プライベート相互接続を通じて他のインスタンスとメッセージを交換する場合に使用される一連のポート番号。他のアプリケーションで使用されていないユニークなポート番号を選択します。

注意： お使いのシステムに既存の標準値が設定されていない場合、Adaptive Server プラグインと **sybcluster** によってデフォルト値が指定されます。

- 各インスタンスのクエリまたは受信ポート番号。他のアプリケーションで使用されていないユニークなポート番号を選択します。
- 各ノードのプライベート・プライマリおよびセカンダリ・ネットワーク・カードの IP アドレスまたはネットワーク名。今回サポートされているのは、UDP ネットワーク・プロトコルのみです。

単一ノードでのシミュレートされたクラスタの作成

サーバの設定とインストールをテストします。

Sybase では、各インスタンスを別々のノード上で、すなわちインスタンスごとに 1 つのノードを設定することをおすすめします。ただし、テスト環境では、単一ノードにシミュレートされたクラスタを作成し、そのノード上ですべてのインスタンスを実行することができます。

1. 最高のパフォーマンスを得るためには、単一ノード上で実行されるすべてのインスタンスのエンジンの数が、同じノードの CPU の数を上回らないようにしてください。
2. **runnable process search count** パラメータのデフォルト設定は 3' です。Sybase では、このデフォルト設定を使用することをおすすめします。これは **sp_configure** ストアド・プロシージャを使用して設定できます。

環境の設定と Unified Agent の起動

Unified Agent を使用すると、クラスタを分散管理できます。\$SYBASE 環境変数を設定し、クラスタのインスタンスをホストする各ノードで Unified Agent を起動します。

Unified Agent のデフォルト・ポートは 9999 です。何らかの理由で、これ以外のポートでエージェントを起動する必要がある場合は、手順 3 を参照してください。

各インスタンスの \$SYBASE ディレクトリから、次の手順を実行します。

1. SYBASE.csh または SYBASE.sh 環境スクリプトを読み込みます。
2. Unified Agents を起動します。

```
$SYBASE/UAF-2_5/bin/uafstartup.sh &
```

注意： Unified Agent が \$SYBASE/UAF-2_5/nodes/<node_name>/log/agent.log を作成し、エージェント出力をそのファイルに送信します。

3. オプションで UAF ポート番号を指定するには、次のように指定します。
port_number は使用可能な任意のネットワーク・ポートです。

```
$SYBASE/UAF-2_5/bin/uafstartup.sh -port port_number
```

Unified Agent の停止

ここでは、Unified Agent を停止する方法について説明します。

特定のポート番号で実行されている Unified Agent を停止するには、次のように入力します。

```
$SYBASE/UAF-2_5/bin/uafshutdown.sh -port port_number
```

たとえば、ポート番号 12345 で開始した UAF エージェントを停止するには、次のように入力します。

```
$SYBASE/UAF-2_5/bin/uafshutdown.sh -port 12345
```

クラスタの作成について

ここでは、クラスタの作成方法について説明します。

注意： Adaptive Server プラグインと **sybcluster** の場合は、クラスタの作成時に run_server ファイルが生成されません。Adaptive Server プラグインまたは **sybcluster** を使用してクラスタを作成したあと、Adaptive Server プラグインまたは

sybcluster ユーティリティを使用して、このクラスタと各インスタンスを起動する必要があります。このクラスタは、run_server ファイルを使用してコマンド・ラインから起動できません。

Storage Foundation for Sybase Cluster Edition でのクラスタの作成

Storage Foundation for Sybase Cluster Edition を使用してクラスタを作成する前に、『Cluster ユーザーズ・ガイド』の「第 11 章 Veritas Cluster Server と Cluster Edition の使用」の内容を確認しておいてください。この章には、Veritas Storage Foundation ソフトウェアとオペレーティング・システムに対して実行する必要がある手順が記載されています。

クラスタの作成のためのワークシート

Adaptive Server プラグインまたは **sybcluster** を使用してクラスタを作成するために必要な情報は、このワークシートに記載されています。

表 8 : クラスタを作成するための設定値

パラメータ	デフォルト値	最小値
クラスタ名		
インスタンスの数	4	
エージェントの数	4	
設定タイプ		
プライベート・インストールの \$SYBASE ディレクトリを使用してクラスタを設定する	N	
ページ・サイズ		
ページ・サイズ (キロバイト単位) :	2KB	
マスタ・デバイス		
マスタ・デバイスのフル・パスと名前		
マスタ・デバイスのサイズ	30MB	
master データベースのサイズ	13MB	
Sybase システム・プロシージャ・デバイス		
sybssystemprocs デバイスのフル・パスと名前		

クラスタの作成と起動

パラメータ	デフォルト値	最小値
sybsystemprocs デバイスのサイズ (MB)	152	
sybsystemprocs データベースのサイズ (MB)	152	140
システム・データベース・デバイス		
システムのデータベース・デバイスのフル・パスと名前		
システムのデータベース・デバイスのサイズ (MB)	6MB	
システム・データベースのサイズ (MB)	6MB	
このクラスタはセカンダリ・ネットワークを保有する	Y	
クォーラム・デバイス		
クォーラム・デバイスのフル・パスと名前		
トレース・フラグ		
PCI デバイス (オプション)		
PCI データベース デバイスへのフル・パス		
PCI データベース・デバイスのサイズ	24MB	
PCI データベースのサイズ	24MB	
ファイルのロケーションを確認 (共有インストールの場合)		
ホーム・ディレクトリのロケーション	\$SYBASE	
環境変数スクリプトのフル・パス	\$SYBASE/ SYBASE.sh	
\$SYBASE_ASE へのパス	ASE-15_0	
interfaces ファイル・ディレクトリへのパス	\$SYBASE	
dataserver 設定ファイルへのパス	\$SYBASE/ cluster_name.cf g	
インスタンス情報 (共有インストールの場合)		
ノード名		

パラメータ	デフォルト値	最小値
インスタンス名 1		
instance1 のクエリ・ポート番号		
instance1 のプライマリ・プロトコル・アドレス		
instance1 のセカンダリ・プロトコル・アドレス		
インスタンス情報 (プライベート・インストールの場合)		
ノード名		
インスタンス名 1		
instance1 の \$SYBASE インストール・ディレクトリのフル・パス		
instance1 の環境シェル・スクリプトへのフル・パス		
instance1 の Adaptive Server ホーム・ディレクトリ		
instance1 のサーバ設定ファイルへのフル・パス		
ローカル・システム・テンポラリ・データベース		
ローカル・システム・テンポラリ・データベース・デバイス名 (ローカル・システム・テンポラリ・データベースの Adaptive Server データベース・デバイスの名前を入力)。ローカル・システム・テンポラリ・データベースのデバイスは必ず共有ディスクとする。		
ローカル・システム・テンポラリ・データベース・デバイスへのパス		
ローカル・システム・テンポラリ・データベース・デバイスのサイズ (MB)		
ローカル・システム・テンポラリ・データベース名		

クラスタの作成と起動

パラメータ	デフォルト値	最小値
ローカル・システム・テンポラリ・データベースのサイズ (MB)		
インスタンス情報 (共有インストールの場合)		
ノード名		
インスタンス名 2		
instance2 のクエリ・ポート番号		
instance2 のプライマリ・プロトコル・アドレス		
instance2 のセカンダリ・プロトコル・アドレス		
インスタンス情報 (プライベート・インストールの場合)		
ノード名		
インスタンス名 2		
instance2 の \$SYBASE インストール・ディレクトリのフル・パス		
instance2 の環境シェル・スクリプトへのフル・パス		
instance2 の Adaptive Server ホーム・ディレクトリ		
instance2 のサーバ設定ファイルへのフル・パス		
ローカル・システム・テンポラリ・データベース		
ローカル・システム・テンポラリ・データベース・デバイス名 (ローカル・システム・テンポラリ・データベースの Adaptive Server データベース・デバイスの名前を入力)。ローカル・システム・テンポラリ・データベースのデバイスは必ず共有ディスクとする。		

パラメータ	デフォルト値	最小値
ローカル・システム・テンポラリ・データベース・デバイスへのパス		
ローカル・システム・テンポラリ・データベース・デバイスのサイズ (MB)		
ローカル・システム・テンポラリ・データベース名		
ローカル・システム・テンポラリ・データベースのサイズ (MB)		
インスタンス情報 (共有インストールの場合)		
ノード名		
インスタンス名 3		
instance3 のクエリ・ポート番号		
instance3 のプライマリ・プロトコル・アドレス		
instance3 のセカンダリ・プロトコル・アドレス		
インスタンス情報 (プライベート・インストールの場合)		
ノード名		
インスタンス名 3		
instance3 の \$SYBASE インストール・ディレクトリのフル・パス		
instance3 の環境シェル・スクリプトへのフル・パス		
instance3 の Adaptive Server ホーム・ディレクトリ		
instance3 のサーバ設定ファイルへのフル・パス		
ローカル・システム・テンポラリ・データベース		

クラスタの作成と起動

パラメータ	デフォルト値	最小値
ローカル・システム・テンポラリ・データベース・デバイス名(ローカル・システム・テンポラリ・データベースの Adaptive Server データベース・デバイスの名前を入力)。ローカル・システム・テンポラリ・データベースのデバイスは必ず共有ディスクとする。		
ローカル・システム・テンポラリ・データベース・デバイスへのパス		
ローカル・システム・テンポラリ・データベース・デバイスのサイズ (MB)		
ローカル・システム・テンポラリ・データベース名		
ローカル・システム・テンポラリ・データベースのサイズ (MB)		
インスタンス情報 (共有インストールの場合)		
ノード名		
インスタンス名 4		
instance4 のクエリ・ポート番号		
instance4 のプライマリ・プロトコル・アドレス		
instance4 のセカンダリ・プロトコル・アドレス		
インスタンス情報 (プライベート・インストールの場合)		
ノード名		
インスタンス名 4		
instance4 の \$SYBASE インストール・ディレクトリのフル・パス		
instance4 の環境シェル・スクリプトへのフル・パス		

パラメータ	デフォルト値	最小値
instance4 の Adaptive Server ホーム・ディレクトリ		
instance4 のサーバ設定ファイルへのフル・パス		
ローカル・システム・テンポラリ・データベース		
ローカル・システム・テンポラリ・データベース・デバイス名(ローカル・システム・テンポラリ・データベースの Adaptive Server データベース・デバイスの名前を入力)。ローカル・システム・テンポラリ・データベースのデバイスは必ず共有ディスクとする。		
ローカル・システム・テンポラリ・データベース・デバイスへのパス		
ローカル・システム・テンポラリ・データベース・デバイスのサイズ (MB)		
ローカル・システム・テンポラリ・データベース名		
ローカル・システム・テンポラリ・データベースのサイズ (MB)		
設定情報をファイルに保存する	Y	
Veritas Cluster Server サポート (オプション - VCS がサポートされているシステムでしか表示されない)		
Cluster Edition サーバを VCS と統合する	Y	
クラスタ内の interfaces ファイルへのパス <u>注意： LDAP がサポートされていない場合、このプロンプトは表示されません。</u>		

パラメータ	デフォルト値	最小値
このデバイス(マスタ・デバイス、システム・プロシージャ・デバイス、およびシステム・データベース・デバイス)が Veritas Cluster File System または Veritas Volume Manager によって管理されているかどうかをチェックする	Y	
各クラスタ・デバイス(マスタ・デバイス、システム・プロシージャ・デバイス、およびシステム・データベース・デバイス)の IO フェンシング機能をチェックする	Y	

ASE プラグインを使用したクラスタの作成

Adaptive Server プラグインでは、クラスタの作成と起動のプロセスをステップ・バイ・ステップ方式で進めるウィザード機能を利用できます。

Adaptive Server プラグインを使用してクラスタを設定および管理する方法については、『Cluster ユーザーズ・ガイド』の「Adaptive Server プラグインによるクラスタの管理」を参照してください。Adaptive Server プラグインおよび **sybcluster** はアップグレードをサポートしています。

1. Adaptive Server プラグインを登録します。
2. Sybase Central を起動します。
3. 左側のウィンドウ枠で、[Adaptive Server Enterprise] アイコンを選択します。
4. 右側のウィンドウ枠の [ユーティリティ] タブを選択します。
5. [クラスタの作成] をダブルクリックするか右クリックし、[オープン] を選択して [クラスタの作成] ウィザードを起動します。または、[ツール] - [Adaptive Server Enterprise] - [クラスタの作成] を選択して、[クラスタの作成] ウィザードを起動することもできます。
6. クラスタの名前を入力します。
7. ドロップダウン・リストからプラットフォームを選択します。
8. 共有インストールとプライベート・インストールのどちらを作成するかを選択します。
9. このクラスタに使用する予定のインスタンスの最大数を選択します。[次へ] をクリックします。

10. 検出の設定 - クラスタで使用するノードの検出に Adaptive Server プラグインで使用される設定を変更できます。詳細については、『Cluster ユーザーズ・ガイド』の「クラスタの管理」を参照してください。
11. クラスタに参加するホストを選択します。[Specify Node] をクリックして、ホストのリストに表示しないノードを指定することもできます。
12. Unified Agent のユーザ名とパスワードを入力します。[次へ] をクリックします。
13. クォーラム・デバイス・パス - クォーラム・デバイスへのフル・パスを入力します。[次へ] をクリックします。
14. プライベート・インストールを実行している場合、クラスタ内のインスタンスを指定する段階までこの手順はスキップされます。共有インストールを実行している場合は、次の情報の入力が必要されます。
 - Sybase ホーム - Adaptive Server リリース・ディレクトリ (\$SYBASE の値)。
 - Adaptive Server Enterprise ホーム - ASE-15_0 ディレクトリへのフル・パス (\$SYBASE/\$SYBASE_ASE の値)。
 - Interfaces ディレクトリ - interfaces ファイルを格納したディレクトリへのパス。
 - クラスタ設定ファイル - クラスタ設定ファイルへのフル・パス。
 - Sybase 環境スクリプト - 環境変数の設定に使用するスクリプトへのフル・パス。
 - 起動パラメータ - クラスタの起動パラメータをリストします。
 - トレース・フラグ - 起動時に使用するトレース・フラグをリストします。[次へ] をクリックします。
15. マスタ・デバイス情報を入力します。
 - マスタ・デバイス・パス - マスタ・デバイスへのフル・パス。
 - デバイス・サイズ - メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) のいずれかを単位とするサイズ。
 - データベース・サイズ - メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) のいずれかを単位とするサイズ。
 - ページ・サイズ - インタフェースのページ・サイズ、2K、4K、8K、または 16K (デフォルトは 2K)。[次へ] をクリックします。
16. 指定したデバイスに SCSI-3 PGR 機能が用意されていない場合は、処理を続行するかどうかを確認するプロンプトが表示されます。[いいえ] を選択した場合、次の画面には移動しません。
17. システム・プロシージャ (*sybssystemprocs*) 情報を入力します。
 - システム・プロシージャ・デバイス - *sybssystemprocs* デバイスへのフル・パス。

- デバイス・サイズ–メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) のいずれかを単位とするサイズ。
- データベース・サイズ–メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) のいずれかを単位とするサイズ。

[次へ] をクリックします。

18. 指定したデバイスに SCSI-3 PGR 機能が用意されていない場合は、処理を続行するかどうかを確認するプロンプトが表示されます。[いいえ] を選択した場合、次の画面には移動しません。

19. システム・データベース (*sybssystemdb*) 情報を入力します。

- システム・データベース・デバイス - *sybssystemdb* デバイスへのフル・パス。
- デバイス・サイズ–メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) のいずれかを単位とするサイズ。
- データベース・サイズ–メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) のいずれかを単位とするサイズ。

指定したデバイスに SCSI-3 PGR 機能が用意されていない場合は、処理を続行するかどうかを確認するプロンプトが表示されます。[いいえ] を選択した場合、次の画面には移動しません。[次へ] をクリックします。

20. チェックボックスをオンにして、PCI デバイスを入力します。PCI デバイスへのフル・パスを入力し、デバイスとデータベースのサイズを入力します。[次へ] をクリックします。PCI デバイスの詳細については、『Adaptive Server Enterprise における Java』を参照してください。

21. 指定したデバイスに SCSI-3 PGR 機能が用意されていない場合は、処理を続行するかどうかを確認するプロンプトが表示されます。[いいえ] を選択した場合、次の画面には移動しません。

22. 次の内容を確認するか、適切な値に変更します。

- ネットワーク・プロトコル–UDP ネットワーク・プロトコルを選択します。
- 開始ポート–クラスタのネットワーク相互接続に使用する開始ポート番号を選択します。このポート番号が使用されていないことを確認します。
- 使用可能なセカンダリ・ネットワークがある場合、このボックスを選択します。

23. [ポートの検証] をクリックして、ポートが使用中ではないことを確認し、この後 [次へ] をクリックします。

24. クラスタのインスタンスを指定します。[追加] をクリックして、リストにインスタンスを追加します。

25. [サーバ・インスタンスの定義] ダイアログで、クラスタを追加する各インスタンスの [一般] タブ内に次の内容を入力します。

- インスタンス名–インスタンスの名前。

- ノード – ノードを選択します。
 - クエリ・ポート - interfaces ファイルのクエリ・ポート・エン트리用のポート番号を選択します。
[ポートの検証] を選択して、このポートが使用可能であることを確認します。
 - ログ・ファイル - このインスタンス用のエラー・ログのフル・パス。
- a) [詳細設定] タブで次のように入力します。
- プライマリ – ドロップダウンからアドレスを選択します。
 - セカンダリ – セカンダリ・アドレスを使用可能に指定している場合は有効になっています。セカンダリ接続用のホスト名またはプロトコル・アドレスを選択します。
 - 起動パラメータ – インスタンスの起動パラメータをリストします。
 - トレース・フラグ – 起動時にインスタンスに使用するトレース・フラグをリストします。
- b) [Private configuration] タブが使用可能な場合は、タブに次の情報を入力します。
- Sybase ホーム - Adaptive Server リリース・ディレクトリ (\$SYBASE の値)。
 - Adaptive Server Enterprise ホーム - ASE-15_0 ディレクトリへのフル・パス (\$SYBASE/\$SYBASE_ASE の値)。
 - Interfaces ディレクトリ - interfaces ファイルを格納したディレクトリへのパス。
 - Sybase 環境スクリプト - 環境変数の設定に使用するスクリプトへのフル・パス。
 - クラスタ設定ファイル - クラスタ設定ファイルへのフル・パス。
- c) [次へ] をクリックして、クラスタ・インスタンス・プロパティを受け入れます。
26. IO フェンシングをサポートするノードに複数のデバイスがある場合は、選択内容を確認するよう求めるプロンプトが表示されます。
27. ローカル・システム・テンポラリ・データベースとデバイスを指定します。ローカル・システム・テンポラリ・データベースは、共有ディスクに作成する必要があります。プラグインに、ローカル・テンポラリ・データベースのデフォルト値がリストされます。デフォルトを使用するか、テーブル・セルを選択して値を変更します。
28. デバイスをさらに追加する場合は、[デバイスの作成] をクリックします。次を入力します。
- デバイス名 – デバイスの名前。
 - デバイス・パス – データベース・デバイスへのフル・パス。

- デバイス・サイズ-メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) のいずれかを単位とするサイズ。

[OK] をクリックしてから、[次へ] をクリックします。

29. [Summary] ページの情報が正しいか確認します。ウィザードの終了後にクラスタを自動的に起動するには、クラスタの作成時に適切なボックスを選択します。このオプションが選択されていない場合、[クラスタの作成] ウィザードの終了後もクラスタは停止した状態が続きます。

30. [次へ] をクリックして、このクラスタを作成します。

Adaptive Server プラグインでクラスタが作成されるときに、サーバ・メッセージを示した別のウィンドウが表示されます。

Sybase Central の起動と停止

ここでは、Sybase Central の起動方法と停止方法について説明します。

Sybase Central を起動するには、次の手順に従います。

- UNIX プラットフォームでは `$SYBASE/shared/sybcentral600/scjview.sh` を実行します。
- Windows プラットフォームでは [スタート] メニューからショートカットを選択します。

Sybase Central を終了するには、[ファイル]-[終了] を選択します。 >

Adaptive Server プラグインの登録

Adaptive Server プラグインは、サーバ・インストール環境の一部として Sybase Central に登録されます。ただし、Adaptive Server プラグインが適切に登録されない場合は、Adaptive Server プラグインを手動で登録することもできます。

1. UNIX の場合 `$SYBASE/ASEP/bin/registerASEP` を実行します。
2. Windows の場合、`%SYBASE%¥ASEP¥bin¥registerASEP.bat` を実行します。
3. Adaptive Server プラグインを手動で登録するには、次の手順に従います。
 - a) [ツール]-[プラグイン]-[登録] を選択します。 > 登録ウィザードが表示されます。
 - b) [プラグイン登録ファイルの指定によって、プラグインを登録します] を選択します。
 - c) [参照] をクリックします。`$SYBASE/ASEP/bin` (Windows の場合は `%SYBASE%¥ASEP¥bin`) に移動して、`ASEPlugin.jpr` を選択します。ウィザード内の手順に従って、Adaptive Server プラグインを登録します。

クラスタへのインスタンスの追加

インスタンスを追加するには、**max instances** パラメータが現在のインスタンス数を上回る必要があり、インスタンスを作成するノード上でクラスタでサポートされるエージェントが稼働している必要があります。Unified Agent (UA) のホスト名とポート番号を入手する必要があります。

注意： I/O フェンシングでは、各インスタンスが別々のノードにあることが必要となります。そのため、既にインスタンスを持つノードに、さらにインスタンスを追加することはできません。ポップアップ・ダイアログで [はい] を選択すると、処理を続行できます。

1. 左側のウィンドウ枠の [サーバ・インスタンス] フォルダを開き、右側のウィンドウ枠にサーバ・インスタンスとオプションを表示します。
2. [クラスタ・サーバ・インスタンスの追加] を選択し、クラスタ・サーバ・インスタンスの追加ウィザードを開きます。
3. ウィザードの手順に従い、クラスタにインスタンスを追加します。
4. [完了] をクリックします。[サーバ・インスタンス] ビューの下に、新しいインスタンスがリストされます。

論理クラスタの追加

論理クラスタは、負荷、フェールオーバー、クライアント・アプリケーションのクラスタへのアクセスを管理する場合に使用される物理共有ディスク内の、1つまたは複数のインスタンスを抽象的に表したものです。

各論理クラスタは、一連のインスタンスで実行され、フェールオーバー用の一連のインスタンスを保有できます。ルート指定ルールにより、着信接続はクライアントから指定されるアプリケーション、ユーザ・ログイン、またはサーバ・エイリアスに基づいて、特定の論理クラスタに送られます。論理クラスタをバインド接続に制限したり、認証された接続から論理クラスタへのアクセスを許可したりするルールもあります。

1. ツールバーから [論理クラスタ] を選択するか、[負荷管理] [論理クラスタ] フォルダを右クリックし [UI-New Logical Cluster] を選択します。[論理クラスタ] ウィザードが起動します。
2. 論理クラスタに名前を付けます。この論理クラスタで実行されるジョブを表す名前を選択します。たとえば、SalesLC など。
[次へ] をクリックします。
3. 論理クラスタを構成するインスタンスを選択します。
 - a) [追加] を選択して、使用可能なインスタンスのリストを表示します。[論理クラスタにインスタンスの追加] ウィンドウに、インスタンス名、ID、状態

(オンラインまたはオフライン) がリストされます。これらのインスタンスは、「ベース」インスタンスとも呼ばれます。

- b) 追加するインスタンスを選択します。Ctrl キーを押しながら、複数のインスタンスを選択します。**[OK]** をクリックします。
- c) **[次へ]** をクリックします。

このリストからインスタンスを削除するには、インスタンス名を強調表示して **[削除]** を選択します。

4. フェールオーバー・サーバ・インスタンスを追加します。

これらのインスタンスでは、1 つまたは複数のベース・インスタンスに障害が発生した場合に、論理クラスタが実行されます。物理クラスタのインスタンスは、フェールオーバー・リソースとして使用できます。負荷管理機能により、リソースをグループ化し、フェールオーバーの順序と優先度を指定できます。

- a) **[追加]** を選択して、使用可能なフェールオーバー・インスタンスのリストを表示します。
- b) **[論理クラスタにフェールオーバー・インスタンスの追加]** ウィンドウが表示され、ここで次のように選択できます。
 - フェールオーバー・グループ-フェールオーバー・グループを使用して、フェールオーバー・インスタンスの使用される優先度と順序を指定できます。低い番号のグループが先に使用されます。これらのフェールオーバー・インスタンスを関連づける、フェールオーバー・グループの番号を選択します。**[OK]** をクリックします。
 - インスタンス - **[論理クラスタにフェールオーバー・インスタンスの追加]** ウィンドウに、フェールオーバー・インスタンスに使用できるインスタンスがリストされます。インスタンスを選択します。Ctrl キーを押しながら、複数のインスタンスを選択します。**[OK]** をクリックしてから、**[次へ]** をクリックします。

5. ルート指定されたアプリケーション、ログイン、エイリアスを追加します。ルート指定ルールにより、特定のアプリケーション、ログイン、およびエイリアスから接続する論理クラスタを指定できます。ルート指定の詳細については、『Cluster ユーザーズ・ガイド』を参照してください。

[ルート指定されたアプリケーション、ログイン、エイリアス] ウィンドウには、現在定義されているルートの名前とタイプが表示されます。ルートを追加するには、次の手順に従います。

- アプリケーション - **[アプリケーション・ルートの追加]** を選択します。**[アプリケーション・ルート]** ウィンドウでアプリケーションの名前を指定します。**[OK]** をクリックします。
- ログイン - **[ログイン・ルートの追加]** を選択します。**[新しいログイン・バインド]** ウィンドウのリストからログイン名を選択します (複数の名前を選択するには、**[Ctrl]** キーを押しながら選択します)。**[OK]** をクリックします。

- エイリアス - [エイリアス・ルート追加] を選択します。[エイリアス・ルート] ウィンドウでエイリアス名を指定します。[OK] をクリックします。

リストされたルートを削除するには、ルート名を選択し、[ルートの削除] をクリックします。[次へ] をクリックします。

6. 論理クラスタのオプションを指定します (これらのオプションの詳細については、『Cluster ユーザーズ・ガイド』の「負荷の管理」の章を参照してください)。
 - システム・ビュー-インスタンス・ビューとは、**sp_who**、**sp_lock**、およびモニタリング・テーブルなどの監視ツールと情報ツールに、稼働環境であるインスタンスの情報のみが表示されることを意味します。クラスタ・ビューは、そのようなツールにクラスタ内のすべてのインスタンスの情報が表示されることです。
 - 論理クラスタの自動起動 - このオプションを選択し、クラスタの起動時に論理クラスタを起動するかどうかを決定します。
 - フェールオーバー・モード - ドロップダウン・リストから「インスタンス」または「グループ」を選択し、インスタンスをグループと個別のどちらの形でオンラインに移行するかを指定します。
 - すべてに失敗 - 任意のインスタンスをフェールオーバー・リソースに使用するか、特定のインスタンスのみをフェールオーバー・リソースに使用するかを指定します。
 - 下方ルーティング・モード - ルート指定ルールで使用される論理クラスタが利用できない場合、クライアント接続をどのようにルート指定するかを指定します。オプションは次のとおりです。
 - **system** - ルート指定不可能な接続をシステム論理クラスタに送信します。
 - **open** - ルート指定不可能な接続をオープン論理クラスタに送信します。
 - **disconnect** - ルート指定不可能な接続を切断します。
 詳細については、『Cluster ユーザーズ・ガイド』を参照してください。
 - 論理クラスタの役割 - このオプションを選択して、この論理クラスタがオープンな役割を引き受けるかどうかを指定します。すなわち、明示的なルート指定ルールにより論理クラスタにルート指定されていない接続はすべて、現在のオープンな論理クラスタにルート指定されます。新しくクラスタを作成すると、システム論理クラスタは自動的にオープンな論理クラスタに指定されます。オープンな役割は、別の論理クラスタに再度割り当てることができます。ただし、各物理クラスタに存在できるオープンな論理クラスタは、1つのみです。
- a) [次へ] をクリックします。

7. [完了] を選択して、論理クラスタを構築します。

クラスタへの接続

Sybase Central を開始すると、メイン・ウィンドウが開き、ASE プラグインが以前接続されていたクラスタとインスタンスのアイコンのリストと共に表示されます。クラスタが稼働中の場合、クラスタ名の横のアイコン内に、緑色の三角が表示されます。

注意： サーバに接続できなかつたり、左ウィンドウ枠のツリー・ビューにサーバが表示されなかつたりする場合には、ツールバーの [ツール] - [接続] で表示される [接続] プロファイルに interfaces ファイルに記載されたサーバに関連付けられたポート番号があるかどうかを確認してください。 > Windows 上で PC クライアントを使用している場合には、サーバに関連付けられたポート番号を %SYBASE%\¥ini¥sql.ini ファイルで確認します。

リスト内の稼働中のクラスタに接続する場合、クラスタ名を右クリックし、[選択] を選択するのが最も早い方法です。ASE プラグインでは、以前の接続データを使用して接続が行われます。クラスタがツリー・ビューに表示されない場合は、Server Discovery を使用して探すか、クラスタのホストとポート、ログイン名、およびパスワード情報を指定します。いずれの方法も [Sybase Central] ウィンドウの最上部付近にあるツールバーの [接続] アイコンをクリックして開始します。

要求される接続情報を入手している場合、[接続] ウィンドウの該当するフィールドにこれを入力します。クラスタのホストとポート番号、またはクラスタ・ノードがわからない場合、ログイン名とパスワードを入力して、[Find] をクリックします。Unified Agent はクラスタを検索し、利用可能なクラスタをリストします。検索中のクラスタがこのリストに表示されない場合は、検出設定を変更する必要があります。

1. [ツール] - [接続] を選択します。 >
複数の登録された Sybase Central プラグインを実行している場合、ASE プラグインを選択します。
2. インスタンスとの接続に使用するログイン名を入力します。
3. ログインのパスワードを入力します。
4. interfaces ファイルのエントリが表示される [サーバ名] リストから、クラスタ名を選択するか、クラスタ・ノードのホストとポートを入力します。PC クライアントを使用する Windows の場合、[サーバ名] リストには、sql.ini ファイルの内容が入力されています。
5. (オプション) クラスタ内のインスタンスのホストとポートを指定します。
6. [OK] をクリックします。

サーバ検索設定の変更

ここでは、検出設定を変更する方法について説明します。

1. [ツール] - [Adaptive Server Enterprise] - [ユーザ各自の設定] - [Server Discovery] に移動します。
2. [サーバ検出を有効化] をオンにします。
3. **[追加]** を選択します。
4. スクロール・メニューから、検出方法を選択します。
 - JINI - 変更への高い適応能力を備えたネットワーク中心型サービスの開発を可能にするオープン・アーキテクチャ。JINI は標準の検索サービスで検出を実行します。
 - UDP (ユーザ・データグラム・プロトコル) - アプリケーション・プログラムが、最小限のプロトコル・メカニズムで他のプログラムにメッセージを送信するためのプロシージャを提供するネットワーク・プロトコル。

注意： UDP が使用されると、Sybase Central が実行中のサブネットと同じサブネットのサーバのみが検索されます。

5. 前の手順で JINI を選択している場合：
 - a) JINI サーバのホストを選択します。
 - b) デフォルトのホストとポートを選択するか、新しいホストとポートを入力します。
6. **[OK]** をクリックします。
7. 検出フィルタを追加または編集するには、**[フィルタ]** をクリックします。Server Discovery で検索に使用されるのは、選択されたフィルタだけです。フィルタを指定するには、次の手順に従います。
 - a) **[追加]** をクリックします。
 - b) **[フィルタを有効にする]** を選択します。
 - c) フィルタするターゲット、ホスト、名前、OS、プラットフォーム、ポート、リリース・タイプ、ステータス、バージョンを選択します。
 - d) 以下の条件を選択します。含む、含まない、等しい、等しくない、始まる、終わる。
 - e) サーバ検出フィルタの基準となる条件文字列を入力します。
 - f) **[OK]** をクリックします。
8. システムで現在稼働中のクラスタを検出するように、ASE プラグインを設定します。次のオプションを使用します。
 - **削除** - リストから検出サービスを削除します。
 - **編集** - 現在の検出サービスの設定を編集します。

- 上 – 選択した検出サービスをリストの上に移動します。
- 下 – 選択した検出サービスをリストの下に移動します。

sybcluster を使用したクラスタの作成

sybcluster を使用してクラスタを設定する前に、「インストールを開始する前に」のトピックと「環境の設定と Unified Agent の起動」のトピックの指示に従ってください。

「sybcluster のサンプル・セッション (179 ページ)」のすべてのプロンプトへの応答を含む完全な **sybcluster** セッションを確認してください。

注意： **sybcluster** を使用して共有ディスク・クラスタを作成して設定します。論理クラスタを作成するには、Adaptive Server プラグインを使用するか、『Cluster ユーザーズ・ガイド』の「負荷の管理」の章で説明されているシステム・ストアド・プロシージャを使用します。

クラスタの設定

ここでは、クラスタの設定方法について説明します。

1. **sybcluster** を起動します。次に例を示します。

```
sybcluster -U uafadmin -P -C mycluster  
-F "blade1,blade2,blade3"
```

sybcluster の構文と使用方法の詳細については、『Cluster ユーザーズ・ガイド』を参照してください。

2. 次のように入力します。

```
create cluster
```

必要な情報を 1 パラメータずつ指定するよう求めるプロンプトが **sybcluster** によって表示されます。デフォルト値がある場合は、デフォルト値が **sybcluster** のコマンド・プロンプトに表示されます。デフォルト値を使用する場合は、**[Enter]** キーを押します。デフォルト値を使用しない場合は、正しい値を入力し、**[Enter]** キーを押します。

3. **sybcluster** から次の情報の入力が必要されます。詳細については、「クラスタの作成のためのワークシート (43 ページ)」の項を参照してください。

- コマンド・ラインでデフォルト・クラスタを設定していない場合に作成するクラスタの名前。
- クラスタに作成するインスタンスの最大数。
- クラスタ内のエージェントの数。

sybcluster によって、利用可能なホスト・マシンのリストが表示されます。このリストには、Unified Agent が動作するように設定されているノードのうち、**sybcluster** コマンド・ラインの **-F** パラメータと **-d** パラメータで

指定されるすべてのノードが示されます。このリストからエージェントを選択します。

注意： 単一ノード上ですべてのインスタンスをホストすることで、シミュレートされたクラスタを作成できます。最高のパフォーマンスを得るためには、単一ノード上で実行されるすべてのインスタンスのエンジンの総数が、ホスト・マシン上の CPU またはコアの数を上回らないようにしてください。

また、**runnable process search count** パラメータのデフォルト設定は '3' です。Sybase では、この環境ではこのデフォルト設定を使用することをおすすめします。これは **sp_configure** ストアド・プロシージャを使用して設定できます。

- クラスタ・エージェントを表す番号。sybcluster では、クラスタ内の他のエージェントにこの番号から始まる番号を割り当てます。
- プライベート・インストール・モードを使用してクラスタを設定するかどうか。デフォルト値は「いいえ」(N) です。クラスタは共有インストール用に設定されます。
- クォーラム・デバイスのフル・パス。例： /dev/raw/raw11。
- トレース・フラグ。
- master データベースのページ・サイズ (KB)。
- マスタ・デバイスのフル・パス。例： /dev/raw/raw12。
- マスタ・デバイスのサイズ。
- master データベースのサイズ。
- システム・プロシージャ・データベース・デバイス *sysprocsdev* へのフル・パス。例： /dev/raw/raw13。
- システム・プロシージャ・データベース・デバイスのサイズ。
- システム・プロシージャ・データベースのサイズ。
- Sybase システムのデータベース・デバイス *systemdbdev* へのフル・パス。例： /dev/raw/raw14。
- システムのデータベース・デバイスのサイズ。
- システム・データベースのサイズ。
- データベースで Java をサポートするためにプラグ可能コンポーネント・インタフェース (PCI) を有効にするかどうか。[N]
"Y" と入力した場合は、次の情報を入力します。
 - PCI データベース・デバイスへのパス。たとえば、"/dev/raw/raw20" と入力します。
 - PCI データベース・デバイスのサイズ (24MB)。
 - PCI データベースのサイズ (24MB)。

- このクラスタにセカンダリ・ネットワークがあるかどうか。(Y)。
 - セカンダリ・ネットワークに "Y" と入力した場合、**sybcluster** は、インスタンスがメッセージを交換するために要求されるポート番号の範囲の開始ポート番号として、デフォルトのポート番号 15100 を使用し、デフォルト値から始まるその数のポートを予約します。

注意： Adaptive Server は、各相互接続に複数のソケットを使用します。各インスタンスに要求されるポートの数は、インスタンスの最大数の 5 倍です。

- セカンダリ・ネットワークに "N" と入力した場合、**sybcluster** は開始ポート番号の指定を要求し、要求される追加ポートの数を計算し、そのポート数を予約します。デフォルト値は 15100 です。
- クラスタがプライベート・インストールを使用している場合は、手順 4 に進んでください。クラスタが共有インストールを使用している場合は、**sybcluster** から次の情報の入力を求められます。
 - \$SYBASE ホーム・ディレクトリ。例： /remote/var/sybase。
 - ".sh" 環境シェル・スクリプトへのフル・パス。例： /remote/var/sybase/SYBASE.sh。
 - Adaptive Server ホーム・ディレクトリ。例： /remote/var/sybase/ASE-15_0。
- **注意：** interfaces ファイルに、クラスタとインスタンスの情報がまだ指定されていないことを確認してください。**sybcluster** は、"クラスタの作成" プロセスの間にこの情報を追加します。

interfaces ファイルを含むディレクトリ。例： /remote/var/sybase。**sybcluster** は、設定中に現在のクラスタおよびインスタンスの情報を追加します。

- dataserver 設定ファイルへのフル・パス。例： /remote/var/sybase/mycluster.cfg。**sybcluster** は、設定中にこのファイルを検出できなければなりません。
4. **sybcluster** にノード名が表示され、各インスタンスの値を 1 ノードずつ指定するように要求されます。
- インスタンス名。
 - クラスタで共有設定がサポートされている場合は、**sybcluster** から次の情報の指定が要求されます。
 - インスタンスのクエリ・ポート番号。この番号が利用可能であり、別のアプリケーションで使用されていないことを確認します。
 - インスタンスのプライマリ・プロトコル・アドレス。例： 10.0.1.1。

- インスタンスのセカンダリ・プロトコル・アドレス (セカンダリ・ネットワークに Y と回答している場合)。例：10.0.1.2。
- クラスタでプライベート設定がサポートされている場合は、**sybcluster** から次の情報の指定が要求されます。
 - \$SYBASE ホーム・ディレクトリ
 - 環境シェル・スクリプトのパス
 - Adaptive Server ホーム・ディレクトリ
 - サーバ設定ファイルへのフル・パス

注意：サーバ設定ファイルのパスは、どのインスタンスでも同じ場合とインスタンスごとに異なる場合があります。

サーバ検索に `interfaces` ファイルを使用している場合、**sybcluster** から次の情報の指定が要求されます。

- interface ファイルのクエリ・ポート番号
- プライマリ・プロトコル・アドレス
- セカンダリ・プロトコル・アドレス
- ローカル・システム・テンポラリ・データベース・デバイス。ローカル・システム・テンポラリ (LST) データベースの Adaptive Server データベース・デバイスの名前。ローカル・システム・テンポラリ・データベースは、共有ディスクに作成する必要があります。
- LST デバイス・パス。例：/dev/raw/raw15。
- LST デバイス・サイズ。

注意：同じデバイスにすべての LST データベースを配置している場合、デバイス・サイズにはすべての LST データベースに対応する十分な大きさが必要です。

- LST データベース名。
 - LST データベース・サイズ。
5. **sybcluster** から、別のインスタンスを追加するかどうか尋ねられます。"Y" と入力すると、**sybcluster** は次のインスタンスに手順 4 を繰り返します。
 6. **sybcluster** から、入力したクラスタ設定を保存するかどうか尋ねられます。

"Y" と入力すると、**sybcluster** は XML ファイルに設定を保存します。このファイルは次のように、**sybcluster** コマンドを使用して編集および再生できます。

```
create cluster cluster_name file file_name
```

7. **sybcluster** から、クラスタを作成するかどうか尋ねられます。

"Y" と入力すると、**sybcluster** は入力した情報を使用してクラスタを設定します。これには数分間必要です。

8. クラスタを作成する間、**sybcluster** によって次の操作が行われます。
 - a) VCS がシステムでサポートされているかどうかを確認します。サポートされている場合は、**sybcluster** から次の情報の指定が求められます。
 - クラスタ・データベースを VCS と統合するかどうか [Y]。"Y" と入力した場合は、**sybcluster** から次の情報の指定が求められます。
 - LDAP がサポートされていない場合は、各インスタンスの `interfaces` ファイルへのパス。
 - master データベース、システム・プロシージャ・デバイス、およびシステム・データベース・デバイスが VCS Volume Manager または Veritas Cluster File System によって管理されているかどうか [Y]。"Y" と入力した場合は、**sybcluster** から各デバイスのステータスが報告され、処理を続行するかどうかを尋ねられます。
 - b) "N" と入力した場合、LDAP がサポートされていない場合は、**sybcluster** から各インスタンスの `interfaces` ファイルのパスの指定を求められます。
 - c) 各クラスタ・デバイスの IO フェンシング対応状況をチェックするかどうかを確認します。"Y" と入力した場合、**sybcluster** はすべてのデバイスを確認し、各デバイスの IO フェンシング機能を報告します。

クラスタ設定の確認

クラスタ設定が稼働中であり、接続できることを確認します。

1. **sybcluster** が実行されていない場合、起動します。次のように入力します。

```
sybcluster -U uafadmin -P -C cluster_name -F "node_name[:port_num]
[,node_name[:port_num]]..."
```

この文では、デフォルト・クラスタと、クラスタ内の各ノード上の Unified Agent が特定されます。**sybcluster** コマンド・ラインでこの情報を入力しない場合、次の手順で入力できます。詳細については、『Clusters ユーザーズ・ガイド』を参照してください。

2. クラスタに接続します。次のように入力します。

```
connect
```

3. クラスタを起動します。次のように入力します。

```
start cluster
```

4. クラスタが稼働していることを確認します。次のように入力します。

```
show cluster status
```

5. クラスタ設定を確認します。次のように入力します。

```
show cluster config
```


sybcluster を使用したクラスタの起動と停止

ここでは、**sybcluster** を使用したクラスタの起動方法と停止方法について説明します。

1. **sybcluster** が実行されていない場合、起動します。次のように入力します。

```
sybcluster -U uafadmin -P -C cluster_name -F "node_name[:port_num]
[,node_name[:port_num]]..."
```

たとえば、"blade1"、"blade2"、"blade3" で "mycluster" を起動するには、次のように入力します。

```
sybcluster -U uafadmin -P -C mycluster
-F "blade1,blade2,blade3"
```

2. クラスタに接続します。次のように入力します。

```
connect
```

3. クラスタを起動します。次のように入力します。

```
start cluster
```

4. クラスタを停止するには、次のように入力します。

```
shutdown cluster
```

Adaptive Server は、すべての文とトランザクションが終了するまで待機してからクラスタを停止します。

オペレーティング・システムの起動時のクラスタの起動

オペレーティング・システムの起動時に、自動的に Adaptive Server クラスタが起動するようにホスト・システムのオペレーティング・システムを設定できます。これを行うには、ホスト・システムで Unified Agent を起動し、その後、**sybcluster** コマンドを実行してシェル・スクリプトによりクラスタを起動します。

1. UAF を起動します。
2. UAF が正常に起動していることを確認します。
3. **sybcluster** を実行し、システム上でインスタンスを起動するコマンドを含んだコマンド・ファイルを **sybcluster** に渡します。
4. コマンド・ファイルは、**sybcluster -i** コマンド・ライン・パラメータを使用して渡され、次のように入力されます。

```
connect to asece15
start instance asece1
quit
```

5. **sybcluster** コマンド・ラインは、次のように入力されます。**asece1_startup** は、上記のコマンド・ファイルです。

```
sybcluster -U uafadmin -P -F host1:9999,host2:9999 -i  
asecel_startup
```

インストールの失敗後のクリーンアップ

クラスタの設定中に予期しないエラーが発生すると、一部のファイルまたはオペレーティング・システムのプロセスが残っている場合があります。クラスタの作成を再度試みる前に、これらを削除してください。

1. **srvbuildres** または **dataserver** プロセスが稼働中であれば、終了します。
2. すべてのノードの UAF エージェントを停止します。
3. `$SYBASE_UA/nodes/*/plugins/<cluster_name>` を削除します。
4. クラスタのエントリ、またはクラスタ・インスタンスを `interfaces` ファイルから削除します。
5. 前回の試行後に `<cluster name>.cfg` ファイルが存在している場合は、これを削除します。
6. UAF エージェントを再起動します。

補助サーバ

sybcluster を使用した補助サーバの設定方法について説明します。Sybase Central を使用して補助サーバの Backup Server、XP Server、Monitor Server を設定することもできます。詳細については、ヘルプのマニュアルを参照してください。ヘルプにアクセスするには、Sybase Central のツールバーにある **[ヘルプ]** をクリックします。

Cluster Edition バージョン 15.5 以降では、クラスタは次の方式のいずれかで複数の Backup Server を使用できます。

- 専用方式 - 各インスタンスは特定の Backup Server に割り当てられます。
- ラウンドロビン方式 - **dump** または **load** コマンドを使用するとき、Cluster Edition が Backup Server の使用状況に合わせてインスタンスをグループ内の Backup Server に割り当てます。
- SYB_BACKUP という名前の 1 台の Backup Server

詳細については、『Cluster ユーザーズ・ガイド』の「第 5 章 クラスタ環境での Backup Server の使用」を参照してください。

共有ディスク・クラスタ環境の Backup Server は、クラスタのすべてのノードにわたって単一の Backup Server として使用するか、クラスタ内の単一のインスタンスにのみ作用できます。Backup Server はクラスタの任意のノードで設定できます。

また現在のノードが停止したときに Backup Server が起動するホストとポート番号を追加指定できます。

dump コマンドと **load** コマンドを、クラスタ内の任意のノードから実行します。コマンドを実行すると、ローカル・インスタンスによって **dump** と **load** が処理され、クラスタの Backup Server にそれらがルート指定されます。インスタンスは、`interfaces` ファイルに指定された順序で Backup Server への接続を試みます。クラスタ内の任意のノードで Backup Server が実行されていない場合、Backup Server がそのノードで実行するように `interfaces` ファイルで設定されていれば、**dump** コマンドまたは **load** コマンドを発行したインスタンスによって Backup Server が起動されます。複数の Backup Server が設定されている場合、設定が専用モードまたはラウンドロビン・モードのいずれであるかに基づいて Backup Server が割り当てられます。

sybcluster ユーティリティを使用して Backup Server をインストールおよび起動できます。このユーティリティでは、必要な情報の入力が必要とされ、入力後にインストールが自動的に実行されます。

Backup Server バイナリ (`$SYBASE/$SYBASE_ASE/bin/backupserver`) は、Adaptive Server バージョン 15.5 Cluster Edition のインストール時にインストールされます。

sybcluster を使用した Backup Server の設定

ここでは、**sybcluster** を使用した Backup Server の設定方法について説明します。

1. **sybcluster** を起動します。

たとえば、**sybcluster** を起動してノード "blade1"、"blade2"、および "blade3" 上の Unified Agents を指定するには、次のように入力します。

```
sybcluster -U uafadmin -P -F
"blade1:1234,blade2:2345,blade3:3456"
```

注意： 選択したポートが利用可能であるかどうかを確認するには、Unified Agent がクラスタ内のすべてのノードで稼働している必要があります。

2. クラスタに接続します。たとえば、"mycluster" に接続するには、次のように入力します。

```
connect to mycluster
```

3. クラスタが稼働していない場合、起動します。次のように入力します。

```
start cluster
```

4. クラスタに単一または複数の Backup Server を作成するには、次のように入力します。

```
create backupserver
```

5. **sybcluster** によって次のプロンプトが表示されます。

Do you want to create multiple Backup Servers?

- a) 「はい」を意味する "Y" を入力した場合は、**sybcluster** から次の情報の指定が求められます。
 - 複数の Backup Server に対するルーティング・ポリシー。選択肢は次のとおりです。
 - 1 – 専用
 - 2 – ラウンドロビン
 - 各 Backup Server の名前。デフォルト値は "*cluster_name_BS*" です。たとえば、"*mycluster_BS*" のようになります。
 - 各 Backup Server ログ・ファイルへのパス。
 - 各 Backup Server の受信ポート。
- b) 「いいえ」を意味する "N" を入力した場合は、ホスト上の Backup Server を 1 つずつ設定するよう **sybcluster** から求められます。
 - Backup Server 名。デフォルト値は "*cluster_name_BS*" です。たとえば、"*mycluster_BS*" のようになります。
 - クラスタ内の各ノードの Backup Server 受信ポート。

Backup Server がどのノードでも起動できるように、すべてのノードに Backup Server を設定することをおすすめします。Backup Server がすべてのノードに設定されていない場合、Backup Server が実行されていないと、Adaptive Server は Backup Server を起動できません。このような状態は、Backup Server が設定されていないノードで **dump** コマンドが開始される場合に起こります。

また、後でノードに対して Backup Server を追加または削除するオプションもあります。

ASE プラグインを使用した Backup Server の設定

ここでは、プラグインを使用した Backup Server の設定方法について説明します。

1. 「補助サーバ」を選択します。右ウィンドウ枠に、[Backup Server]、[XP Server]、[Monitor Server] の各タブが表示されます。
2. 設定する補助サーバのタブを選択します。
3. 左パネルで [Auxiliary Server] フォルダを右クリックします。
4. [Backup Server の設定]、[XP Server の設定]、または [Monitor Server の設定] の見出しを選択します。
5. ウィザードに表示される指示に従います。

Job Scheduler のインストール

クラスタ内のすべてのインスタンスは、単一の Job Scheduler を共有します。Job Scheduler が稼働しているインスタンスで障害が発生した場合に別のノードにフェールオーバーできるように Job Scheduler を設定します。

1. クラスタ内のすべてのインスタンスがアクセス可能な共有ロー・デバイス上に、90MB 以上のサイズのデバイス *sybmgmtdev* を作成します。
2. `installjsdb` スクリプトを実行します。

```
isql -Usa -Psa_password -Sservername
-i $SYBASE/$SYBASE_ASE/scripts/installjsdb
```

注意：パスに `isql` 実行ファイル(`$SYBASE/$SYBASE_OCS/bin`)のロケーションを含むディレクトリを作成している必要があります。

`installjsdb` スクリプトは、*sybmgmtdb* データベースを検索します。データベースが存在する場合は、Job Scheduler のテーブルとストアド・プロシージャを作成します。存在しない場合、スクリプトは *sybmgmtdb* データベース、テーブル、およびストアド・プロシージャを作成する *sybmgmtdev* デバイスを検索します。

注意：`installjsdb` スクリプトで *sybmgmtdev* デバイスと *sybmgmtdb* データベースのいずれも見つからない場合は、*master* デバイス上に *sybmgmtdb* データベースが作成されます。Sybase では、ディスクに障害が発生した場合、簡単に復旧できるように、*master* デバイスから *sybmgmtdb* データベースを削除することを強くおすすめします。

3. 適宜、`dscpc`、`dsedit`、またはテキスト・エディタを使用して、`interfaces` ファイルに JSAGENT のディレクトリ・サービスのエントリを作成します。Sybase では、このエントリに "`clustername_JSAGENT`" の名前を付けることをおすすめします。

高可用性フェールオーバーを有効にするために、JSAGENT エントリにクラスタ内の各ノードのマスタ・ローとクエリ・ローを含める必要があります。たとえば、ノードを2つ含むクラスタ "`mycluster`" に JSAGENT エントリを追加するには、次のような構文を指定します。

```
mycluster_JSAGENT
master tcp /dev/tcp node_name1 17780
query tcp /dev/tcp node_name1 17780
master tcp /dev/tcp node_name2 16780
query tcp /dev/tcp node_name2 16780
```

ノード名は、UNIX プロンプトで実行された `uname -n` コマンドで返される名前と一致する必要があります。たとえば、ホスト "`myxml1`" では、`uname -n` は値 "`myxml1.sybase.com`" を返します。ホスト "`myxml2`" では、`uname -n` は値

"myxml2.sybase.com" を返します。JSAGENT の正しいエントリは、次のようになります。

```
mycluster_JSAGENT
  master tcp /dev/tcp myxml1.sybase.com 17780
  query tcp /dev/tcp myxml1.sybase.com 17780
  master tcp /dev/tcp myxml2.sybase.com 16780
  query tcp /dev/tcp myxml2.sybase.com 16780
```

注意：必ず現在使用されていないポートを指定してください。

ディレクトリ・サービスの詳細については、『システム管理ガイド』の第1巻を参照してください。

4. **sp_addserver**, を使用して、クラスタの *syservers* テーブルにエントリを作成します。次に例を示します。

```
sp_addserver SYB_JSAGENT, null, mycluster_JSAGENT
```

sp_addserver の詳細については、『リファレンス・マニュアル：コマンド』を参照してください。

5. 次のように Job Scheduler を有効にします。

```
sp_configure "enable job scheduler", 1
```

6. Job Scheduler を起動するには、サーバを再起動するか、以下を実行します。

```
use sybmgmtdb
go
sp_js_wakeup "start_js", 1
go
```

7. Job Scheduler が稼働しているインスタンスを判断するには、グローバル変数 *@@jsinstanceid* を照会します。

```
select @@jsinstanceid
go
```

sybcluster を使用した XP Server の設定

クラスタ内の各インスタンスに XP Server を設定する場合は、設定手順を実行する **sybcluster create xpserver** コマンドを使用します。

1. **sybcluster** を起動します。

たとえば、**sybcluster** を起動してノード “blade1”、“blade2”、および “blade3” 上の Unified Agents を指定するには、次のように入力します。

```
sybcluster -U uafadmin -P -F
"blade1:1234,blade2:2345,blade3:3456"
```

2. クラスタに接続します。たとえば、“mycluster” に接続するには、次のように入力します。

```
connect to mycluster
```

3. クラスタを起動します。次のように入力します。

```
start cluster
```

4. XP Server を設定するには、次のように入力します。

```
create xpserver
```

Adaptive Server から、各インスタンスの XP Server のポート番号の指定が要求されます。xp server ユーティリティの詳細については、『ユーティリティ・ガイド』を参照してください。

sybcluster を使用した Monitor Server の設定

クラスタ内の各インスタンスの Monitor Server を設定するには、**sybcluster create monitorserver** コマンドを使用します。

1. **sybcluster** を起動します。たとえば、**sybcluster** を起動して、ノード "blade1"、"blade2"、"blade3" で Unified Agent を指定するには、次のように入力します。

```
sybcluster -U uafadmin -P -F  
"blade1:1234,blade2:2345,blade3:3456"
```

2. クラスタに接続します。たとえば、“mycluster”に接続するには、次のように入力します。

```
connect to mycluster
```

3. クラスタを起動します。

```
start cluster
```

4. Monitor Server を設定するには、次のように入力します。

```
create monitorserver
```

5. 各インスタンスについて、Adaptive Server から次の情報の指定が要求されます。
 - 各インスタンスの Monitor Server ログ・ファイルのロケーション
 - Monitor Server のポート番号
 - ユーザ名
 - パスワード

インストール後の作業

ここでは、サーバをインストールした後に行う作業について説明します。

システム管理者パスワードの設定

Sybase ソフトウェアをインストールすると、“sa” と呼ばれるユーザ・アカウントが、Sybase システム管理者用に作成されます。“sa” を使用してログインしたユーザは、*master* データベースを含む Adaptive Server 上のすべてのデータベースを、フル・アクセスで使用できます。

新しくインストールした直後は、“sa” アカウントにはパスワードが設定されていません。パスワードのデフォルト値が NULL のためです。運用環境では、Sybase システム管理者は必ずデフォルト以外のパスワードを使用してください。

Sybase システム管理者は、Adaptive Server に “sa” としてログインし、次のように、**sp_password** を使用してパスワードを設定してください。

```
$SYBASE/$SYBASE_OCS/bin/isql -Usa -P -Sserver_name
1> sp_password null, new_password
2> go
```

各パラメータの意味は、次のとおりです。

- **null**- デフォルトのパスワードです。
- **new_password** - “sa” アカウントに割り当てるパスワードです。

セキュリティを最大限に確保するため、文字と数字を組み合わせた 6 文字以上のパスワードを作成することをおすすめします。

サンプル・データベースのインストール

サンプル・データベースには、架空のビジネスの情報が保存されています。この架空データを使用して、Sybase 製品の学習ができます。学習中に、重要なデータに影響を与える心配もありません。

- **installpubs2** - 出版社でのオペレーションを表すデータを含む、*pubs2* サンプル・データベースをインストールします。サーバ接続のテストや Transact-SQL の学習に、このデータベースを使用します。Adaptive Server マニュアルにある例のほとんどが、*pubs2* データベースへの問い合わせです。
- **installpubs3** - 参照の整合性を使用する *pubs2* の更新されたバージョンである、*pubs3* サンプル データベースをインストールします。また、テーブルも

*pubs2*で使用されているテーブルと若干異なります。Adaptive Serverのマニュアルでは、例の中で *pubs3* データベースも使用しています。

- `installpix2 - pubs2` データベースとともに使用する *image* データをインストールします。

注意： *image* データを含むフルの *pubs2* データベースをインストールするには、マスタ・デバイスのサイズは少なくとも 30MB が必要です。`installpubs2` を実行したあとで、`installpix2` スクリプトを実行します。

サンプル・データベースのデフォルト・デバイス

`$$SYBASE/$$SYBASE_ASE/scripts` ディレクトリには、*us_english* のサンプル・データベース、その他の言語のサンプル・データベース、英語の *pubs2* サンプル・データベースに関連する *image* データが格納されています。

マスタ・デバイスのステータスを変更したり、別のデフォルト・デバイスを指定したりするとき、`sp_diskdefault` を使用していない場合は、スクリプトによってマスタ・デバイス上にサンプル・データベースがインストールされます。ただし、本来はシステム・テーブルに使用すべき貴重な領域が使用されるので、この設定はおすすめできません。サンプル・データベースごとに、データベース・デバイス上に、2K サーバでは 3MB の領域、4K、8K、16K のサーバでは 3MB の倍数の領域が必要です。

マスタ・デバイスにサンプル・データベースをインストールしないようにするには、次のいずれかの手順に従います。

- `sp_diskdefault` を使用して、デフォルトのデバイスをマスタ・デバイス以外に指定します。`sp_diskdefault` の詳細については、『リファレンス・マニュアル』を参照してください。
- サンプル・データベースごとにインストール・スクリプトを修正して、異なるデバイスを指定します。

データベース・スクリプトの実行

ここでは、データベース・スクリプトを実行する方法について説明します。

1. サーバ・インスタンスを起動します。
2. *pubs2* データベースと *pubs3* データベースを格納するデバイスのタイプ (ロー・パーティション、論理ボリューム、オペレーティング・システム・ファイルなど) とロケーションを決定します。この情報はあとで必要になります。
3. オリジナル `installpubs2` スクリプトと `installpubs3` スクリプトのコピーを作成します。編集したスクリプトに問題が発生したときのために、コピーにアクセスできるようにしておいてください。

4. 必要に応じてテキスト・エディタを使用してスクリプトを編集し、マスタ・デバイス以外のデフォルト・デバイスを指定するか、**sp_diskdefault** を使用します。

isql を使用してインスタンスにログインします。\$SYBASE/\$SYBASE_ASE/scripts ディレクトリから、次のスクリプトを実行します。

```
isql -Usa -P*****-Sserver_name -iscript_name
```

各パラメータの意味は、次のとおりです。

- *server_name* - データベースの宛先サーバを示します。
- *script_name* - 実行するスクリプトのファイル名です。

たとえば、*pubs2* を VIOLIN というサーバにインストールするには、次のように入力します。

```
isql -Usa -P*****-SVIOLIN ¥
-i $SYBASE/$SYBASE_ASE/scripts/installpubs2
```

5. *pubs2* と組み合わせて使用するすべての *image* データ (*pubs3* では *image* データを使用しません) をインストールするには、次のように入力します。

```
isql -Usa -Ppassword -Sservername ¥
-i $SYBASE/$SYBASE_ASE/scripts/installpix2
```

注意： *image* データでは、PICT、TIFF、Sun raster ファイル・フォーマットのそれぞれに2つずつ、計6つのピクチャが用意されているため、10MB の領域が必要になります。*image* データ型の使用時やテスト時のみ *installpix2* スクリプトを実行してください。Sybase では *image* データを表示するツールを用意していません。イメージをデータベースから抽出したら、適切なウィンドウ・グラフィック・ツールを使用してそのイメージを表示してください。

これらのスクリプト実行の詳細については、『設定ガイド』を参照してください。

interpubs データベースのインストール

interpubs は、*pubs2* データベースに似たデータベースであり、フランス語とドイツ語のデータが格納されています。

interpubs8 ビット文字を含んでいて、ISO 8859-1 (*iso_1*)、ISO 8859-15 (*iso_15*)、Roman8、または Roman9 (HP-UX 用) 文字セットを使用した、Adaptive Server インストール環境で使用できます。フランス語とドイツ語を正しく表示するには、8 ビット文字を表示するように端末を設定してください。

1. *iso_1*、*iso_15*、Roman8、Roman9 または UTF-8 が、デフォルト文字セットか追加文字セットとしてインストールされていることを確認してください。

2. *interpubs* データベースを格納するデバイスのタイプ (ロー・パーティション、論理ボリューム、オペレーティング・システム・ファイルなど) とロケーションを決定します。この情報はあとで必要になります。
3. オリジナル *installintpubs* スクリプトのコピーを作成します。編集したスクリプトに問題が発生したときは、このコピーを使用します。
4. 必要に応じてテキスト・エディタを使用してスクリプトを編集し、マスタデバイス以外のデフォルト・デバイスを指定するか、**sp_diskdefault** を使用します。
5. **-J** フラグを使ってスクリプトを実行し、データベースが正しい文字セットでインストールされたことを確認します。

```
isql -Usa -Ppassword -Sservername -Jiso_1 ¥  
-i $SYBASE/$SYBASE_ASE/scripts/iso_1/installintpubs
```

jpubs データベースのインストール

ここで *Jpubs* データベースをインストールする方法について説明します。

注意： 使用しているサーバに日本語モジュールをインストールした場合、スクリプト・ファイルには *jpubs* データベースをインストールするための *installjpubs* スクリプトが含まれています。*jpubs* は *pubs2* と同じようなデータベースで、日本語データが含まれています。*installjpubs* は、EUC-JIS (eucjis)、UTF-8 (utf8)、またはシフト JIS (sjis) の文字セットのいずれかを使用します。

1. 端末を 8 ビット文字表示に設定します。
2. EUC-JIS、シフト JIS、または UTF-8 の文字セットが、Adaptive Server のデフォルト文字セットまたは追加文字セットとしてインストールされていることを確認します。
3. *jpubs* データベースを格納するデバイスのタイプ (ロー・パーティション、論理ボリューム、オペレーティング・システム・ファイルなど) とロケーションを決定します。この情報はあとで必要になります。
4. オリジナル *installjpubs* スクリプトのコピーを作成します。編集したスクリプトに問題が発生したときのために、コピーにアクセスできるようにしておいてください。
5. 必要に応じてテキスト・エディタを使用してスクリプトを編集し、マスタ・デバイス以外のデフォルト・デバイスを指定するか、**sp_diskdefault** を使用します。
6. **-J** フラグを使って *installjpubs* スクリプトを実行し、データベースが正しい文字セットでインストールされたことを確認します。

```
isql -Usa -Ppassword -Sservername -Jeucjis ¥  
-i $SYBASE/$SYBASE_ASE/scripts/eucjis/installjpubs
```

```
isql -Usa -Ppassword -Sservername -Jeucjis ¥  
-i %SYBASE%¥%SYBASE_ASE%¥scripts¥eucjis¥installjpubs
```

または

```
isql -Usa -Ppassword -Sservername -Jsjis ¥  
-i $SYBASE/$SYBASE_ASE/scripts/sjis/installjpubs
```

```
isql -Usa -Ppassword -Sservername -Jsjis ¥  
-i %SYBASE%¥%SYBASE_ASE%¥scripts¥sjis¥installjpubs
```

isql の **-J** オプションの詳細については、『ユーティリティ・ガイド』を参照してください。

サンプル・データベースの管理

運用システムのユーザ・データベースから "guest" ユーザ・オプションを削除することをおすすめします。

サンプル・データベースには guest ユーザ・オプションが登録されていて、guest ユーザとしてアクセスすれば、認可されたユーザであれば誰でもそのデータベースにアクセスできます。guest ユーザには、ユーザ・テーブルの選択や挿入、更新、削除など幅広い権限が与えられています。guest ユーザの詳細と guest パーミッションの一覧については、を参照してください。

1. 十分な領域があれば、各新規ユーザにサンプル・データベースのクリーン・コピーを与えて、他のユーザが行った変更による混乱を避けてください。
2. 領域に問題がある場合は、**begin transaction** コマンドを実行してからサンプル・データベースを更新するように、ユーザに指示します。
3. こうすると、サンプル・データベースの更新が終わった後で、**rollback transaction** コマンドを実行して変更を元に戻すことができます。

PC クライアントのインストール

PC クライアントのインストーラには、Adaptive Server にアクセスしてクエリを実行したりサーバを管理したりするために使用できるクライアント・アプリケーションとユーティリティが含まれています。また、Sybase Open Client/ Open Server™ Software Developers Kit も含まれています。これを使用してサーバと ODBC、OLE DB、および ADO.NET の各クライアントにアクセスするアプリケーションを開発することができます。

Sybase Central、Open Client などのクライアント・アプリケーションを使用して Adaptive Server にアクセスします。

Sybase Central を使用すると、クライアント接続を介して Adaptive Server に接続できます。Sybase Central を使用して、リモート・サーバ、ローカル・サーバ(ローカル・マシンにインストールされたサーバ)、複数のサーバに接続できます。ネットワーク接続の設定については、「インストール後の使用開始にあたって」の項を参照してください。

Sybase PC クライアント CD には、次の内容で構成される Windows プラットフォーム用 Software Developer's Kit (SDK) が含まれています。

- Embedded SQL/C (ESQL/C)
- Embedded SQL/Cobol (ESQL/Cobol)
- Monitor Client Library
- ASE 分散トランザクション管理用 XA インタフェース・ライブラリ
- Additional Connectivity Language Modules
- Open Client (CT-Library、DB-Library)
- Agent Management Console
- Microsoft Cluster Server Resource Type for ASE
- Interactive SQL
- Sybase Central
- SySAM プラグイン
- ASE ADO.NET Data Provider
- Sybase 製 ASE OLE DB プロバイダ
- Sybase 製 ASE ODBC ドライバ
- ASE プラグイン
- QPTune
- jConnect™ for JDBC
- jConnect 6.0.5 for JDBC

- jConnect 7.0 for JDBC
- SySAM ライセンス・ユーティリティ
- DBISQL
- SDC 管理ユーティリティ

クライアントのインストール

PC Client CD には、いくつかのコンポーネントが含まれ、それぞれのインストーラがパッケージされています。PC Client CD を挿入すると、メニュー・プログラムが自動的に起動します。メニュー・プログラムでは、CD からインストールできるコンポーネントのリストが表示されます。1 回に 1 つのコンポーネントをインストールできます。

表 9 : Windows での PC クライアントの最小システム要件

Windows OS	ハードウェア	最小 RAM
WinXP Pro SP2、Win2003 Server SP1、WinVista Business Edition	P4 1.0 GHz	512MB

1. 使用しているコンピュータに各製品用に十分なディスク領域があることを確認します。
2. コンポーネントをアンロードする場合は、管理者権限を持つアカウントを使用してログインします。
3. 開いているアプリケーションやユーティリティを閉じて、メモリとシステム・リソースを解放します。
4. Windows の [スタート] メニューから [ファイル名を指定して実行] を選択してインストール・プログラムを起動し、次のように入力します(ここで X は PC クライアントのインストール・ファイルをコピーしたディレクトリです)。

```
X:¥autorun.exe
```

5. [Client Components 15.5] を選択します。
6. インストーラが起動し、[Welcome] ウィンドウが表示されます。[次へ] をクリックします。
7. Sybase のライセンス契約が表示されたら、国を選択して [同意する] をクリックしてから、[次へ] をクリックします。
8. ディレクトリ・パスを入力し、[次へ] をクリックします。

- a) 以前のバージョンのサーバが格納されているディレクトリを指定した場合は、[インストールの更新を選択します] パネルが表示されます。このパネルでは、更新する機能を選択します。
9. 既存のサーバを更新している場合は、この手順は表示されません。実行するインストールの種類を選択します。
- [標準インストール] – 多くのユーザに必要なデフォルトのコンポーネントをインストールします。
 - [フル・インストール] – CD に収められたコンポーネントをすべてインストールします。
 - [カスタム・インストール] – インストールするコンポーネントを選択できます。選択したコンポーネントのインストールにその他のコンポーネントが必要な場合は、そのコンポーネントは自動的にインストールされます。
10. カスタム・インストールを選択すると、次にインストールするコンポーネントを指定します。
- 標準インストールでインストールされるコンポーネントは自動的に選択されます。このリストを使用して、コンポーネントの選択や選択の解除ができます。
11. [次へ] をクリックします。
- [Summary] ウィンドウには、インストーラでインストールされるすべてのコンポーネント、必要なディスク領域、使用可能なディスク領域が表示されます。
- 対象ディレクトリに十分な空き領域がない場合は、使用可能領域が赤く表示されます。[Previous] をクリックして前のウィンドウに戻って選択を変更するか、[キャンセル] をクリックしてインストーラを終了する必要があります。
12. [次へ] をクリックします。インストーラによって、コンポーネントがインストールされ、処理の進行状況が表示されます。

Adaptive Server へのクライアント・ネットワーク接続を設定するには、『Adaptive Server Enterprise 設定ガイド Windows 版』を参照してください。

サイレント・モードでのクライアントのインストール

インストーラを GUI モードで実行し、応答を応答ファイルに記録した後で、ファイルを編集して応答をカスタマイズします。

サイレント・モードでインストールするには、次のコマンドを実行します。

```
setupConsole.exe -f responseFileName -i silent
-DAGREE_TO_SYBASE_LICENSE=true
```

responseFileName には、選択したインストール・オプションを含むファイル名の絶対パスを入力します。

注意：サイレント・モードでのインストール時に、Sybase ライセンス契約に同意する必要があります。次のどちらかを実行します。

- 次のオプションをコマンド・ライン引数に含める。

```
-DAGREE_TO_SYBASE_LICENSE=true
```

- 応答ファイルを編集して、次のプロパティを含める。

```
AGREE_TO_SYBASE_LICENSE=true
```

GUI 画面がないことを除けば、InstallAnywhere の動作はすべて同じです。サイレント・モードのインストール結果は、GUI モードで同じ応答を行った場合とまったく同じになります。

警告！ Sybase では、サイレント・インストールの実行時に、フォアグラウンドで実行される `setupConsole.exe` 実行可能ファイルを使用することをおすすめします。通常の `setup.exe` 実行可能ファイルはバックグラウンドで実行されるため、インストールが異常終了したという印象をユーザに与え、サイレント・インストールを使用して再度インストールが試行される結果になります。複数のインストールを同時に実行すると、Windows レジストリが破壊され、オペレーティング・システムを再起動できなくなることがあります。

インストール後の使用開始にあたって

Adaptive Server、Sybase Central、Java Runtime Environment をインストールしたら、インストール状態とネットワーク接続をテストしてください。Adaptive Server は、他の Adaptive Server、Open Server アプリケーション (Backup Server など)、ネットワーク上のクライアント・ソフトウェアと通信します。クライアントは 1 つ以上のサーバと通信でき、サーバはリモート・プロシージャ・コールによって別のサーバと通信できます。

Sybase 製品間で対話するには、他の製品がネットワーク上のどこにあるかを各製品が認識する必要があります。この情報は、`interfaces` ファイル (Windows の場合) または LDAP (Lightweight Directory Access Protocol) サーバに格納されています。

1. [スタート]-[プログラム]-[Sybase]-[Sybase Central 6.0.0] を選択します。
2. Adaptive Server ログイン・ウィンドウを表示するには、Sybase Central のメニュー・バーから [ツール]-[接続] を選択します。
3. デフォルトのユーザ ID "sa" を使用して、パスワードなしでログインします。パスワードを変更した場合は、新しいパスワードを使用します。
4. 接続先の Adaptive Server を選択します。
5. サーバとの接続を切断するには、[ツール]-[切断] を選択するか、接続しているサーバのアイコンを右クリックして [切断] を選択します。
6. Sybase Central を終了します。

テスト環境の作成

テスト環境を作成するには、単一ノードにシミュレートされたクラスタを作成し、そのノード上ですべてのインスタンスを実行します。

Sybase では、各インスタンスを別々のノード上で、すなわちインスタンスごとに 1 つのノードを設定することをおすすめします。ただし、テスト環境では、単一ノードにシミュレートされたクラスタを作成し、そのノード上ですべてのインスタンスを実行することができます。最高のパフォーマンスを得るためには、単一ノード上で実行されるすべてのインスタンスのエンジンの数が、同じノードの CPU の数を上回らないようにしてください。

runnable process search count パラメータのデフォルト設定は '3' です。Sybase では、このデフォルト設定を使用することをおすすめします。これは **sp_configure** ストアド・プロシージャを使用して設定できます。

LDAP 用の libtcl.cfg の設定

libtcl.cfg ファイルを使用して、LDAP サーバ名、ポート番号、DIT ベース、ユーザ名、LDAP サーバへの接続に使用するパスワードを指定します。

32 ビット LDAP ドライバでの最も簡単なフォームで、libtcl.cfg ファイルは次のようなフォーマットになります。

```
[DIRECTORY]
ldap=libsybdldap.dll
```

LDAP サーバが libtcl.cfg ファイルで指定されている場合、サーバ情報は LDAP サーバからのみアクセスできます。Adaptive Server は interfaces ファイルを無視します。起動時に **-i** オプションを使用する Open Client/Open Server アプリケーションは、libtcl.cfg ファイルを上書きして interfaces ファイルを使用します。

ディレクトリ・サービスの使い方

詳細については、お使いのプラットフォームの設定ガイドを参照してください。

1. ディレクトリ・サービスを使用するように libtcl.cfg ファイルを設定します。標準的な ASCII テキスト・エディタを使用して、次のように修正します。
 - [DIRECTORY] エントリの下に libtcl.cfg ファイルにある LDAP URL 行の行頭から、コメント・マーカのセミコロン (;) を削除します。
 - [DIRECTORY] エントリに LDAP URL を追加します。サポートされている LDAP URL 値については、『設定ガイド』を参照してください。

警告！ LDAP URL は 1 行に書いてください。

```
ldap=libsybdldap.dll  
ldap://host:port/ditbase??scope??  
bindname=username?password
```

次に例を示します。

```
[DIRECTORY]  
ldap=libsybdldap.dll  
ldap://huey:11389/dc=sybase,dc=com??one??  
bindname=cn=Manager,dc=sybase,dc=com?secret
```

注意： Windows x64 では、.dll ファイルは libsybdldap64.dll と呼ばれます。

2. 必要なサード・パーティ・ライブラリが、適切な環境変数で指定されていることを確認します。Netscape LDAP SDK ライブラリは %SYBASE%\%SYBASE_OCS%\%d11 にあります。

Windows の PATH 環境変数に、このディレクトリを指定する必要があります。

3. **dsedit** を使用して、ディレクトリ・サービスにサーバを追加します。
 - a) Windows の [スタート] メニューから [プログラム] - [Sybase] - [コネクティブティ] - [Open Client ディレクトリ・サービス・エディタ] を選択します。
 - b) サーバの一覧から **[LDAP]** を選択して、**[OK]** をクリックします。
 - c) **[Add New Server Entry]** をクリックします。
 - d) 次を入力します。
 - サーバ名 - 必須。
 - セキュリティ・メカニズム - オプションです。セキュリティ・メカニズム OID の一覧は、%SYBASE%\ini\objectid.dat にあります。
 - e) **[Add New Network Transport]** をクリックします。
 1. トランスポート・タイプを選択します。
 2. ホスト名を入力します。
 3. ポート番号を入力します。
 - f) **[OK]** を 2 回クリックして、**dsedit** ユーティリティを終了します。

interfaces ファイルへのサーバの追加

sql.ini ファイルを使用して Sybase サーバにアクセスするには、各サーバがクライアント・コンピュータの sql.ini ファイルに記載されている必要があります。

注意： Windows Directory Replication を使用して、sql.ini を複数のロケーションに複製できます。詳細については、Microsoft のマニュアルを参照してください。レジストリに格納されているディレクトリ・サービスを使用することもできます。

クライアント・コンピュータ上での **dsedit** の使用については、『Open Client/Server 設定ガイド デスクトップ・プラットフォーム版』および『Open Client/Server

プログラマーズ・ガイド補足 デスクトップ・プラットフォーム版』を参照してください。

1. Windows の[スタート]メニューから[プログラム]-[Sybase]-[コネクティビティ]-[Open Client ディレクトリ・サービス・エディタ]を選択します。
2. 最初のウィンドウで [OK] をクリックして、[Interfaces ドライバ] ウィンドウを開きます。
3. メニューから [サーバ・オブジェクト] を選択し、[追加] をクリックします。
4. [Input Server Name] ボックスに、エントリを作成するサーバ名を入力します。
[OK] をクリックします。
5. [属性] カラムで、先ほど追加したサーバ・アドレス・ローをダブルクリックします。
6. [Protocol Network Address] で [追加] をクリックします。
7. [ネットワーク・アドレス] ボックスに、サーバ・ポート番号とともにサーバ名または IP アドレスを入力できます。次に例を示します。

```
machine_name, 4100
```

各パラメータの意味は、次のとおりです。

- *machine_name* はコンピュータの名前
- 4100 はクライアントからの応答を「受信」するためにサーバが使用するポート番号

注意： Windows では、TCP 方式の接続情報も使用できます。

```
[ SERVER ]
MASTER=NLWNSCK, huey, 2222
QUERY=NLWNSCK, huey, 2222
```

または

```
[ SERVER ]
MASTER=TCP, heuy, 2222
QUERY=TCP, heuy, 2222
```

推奨するフォーマットは、TCP と *host_name* および *port_number* をスペースで区切るフォーマットです。これは、すべてのプラットフォームでサポートされているためです。interfaces ファイルは、標準的な ASCII テキスト・エディタを使用して編集できます。

8. マシンの IP アドレスを探すには、UNIX ボックスでは **ypmatch**、Windows では **ipconfig** を使用します。

PC クライアントのアンインストール

インストーラが行ったアクションは、アンインストール・プロセスで削除されません。インストール後に作成されたレジストリ・エントリやファイルは削除されないため、インストール・プロセスの完了後に手動で削除する必要があります。

指示に従ってコンポーネントの一部または全部をアンインストールします。

Windows マシンから PC クライアントをアンインストールするには、次を実行します。

```
%SYBASE%\sybuninstall¥PCClient¥uninstall.exe
```

[コントロールパネル]の[プログラムの変更と削除]を使用して PC クライアントを削除することもできます。

サーバのアップグレード

Adaptive Server 15.5 Noncluster Edition から 15.5 Cluster Edition にアップグレードするには、このマニュアルで説明されているアップグレード手順を使用します。アップグレードの互換性については、このトピックを参照してください。

次の製品から Cluster Edition にアップグレードすることができます。

- 12.5 ～ 12.5.4 ESD #8
- 15.0 ～ 15.5

次の製品からこのバージョンの Cluster Edition にアップグレードすることができます。

- 15.0.1 Cluster Edition ～ 15.0.1 Cluster Edition ESD #4
- 15.0.3 Cluster Edition
- ESD を含む 12.5.4 および 15.5 ノンクラスタード・サーバを含む 15.0.x バージョン

注意： 15.5 Cluster Edition にアップグレードしたら、15.x 以前のノンクラスタード・サーバにダウングレードすることはできません。15.0.1 Cluster Edition ESD #4、Cluster Edition 15.0.3、Cluster Edition 15.5 から 15.0.1 Cluster Edition にダウングレードすることはできません。

同じページ・サイズ間のアップグレードだけがサポートされます。**sybmigrate** を使用して、スキーマを再作成し、別のページ・サイズにデータをロードします。

Adaptive Server プラグインまたは **sybcluster** ユーティリティを使用すると、アップグレード前の手順およびアップグレードの手順の多くが自動的に実行されます。サーバを手動でアップグレードすることもできます。

- Adaptive Server プラグインを使用するには、「Upgrading the Server With ASE Plug-in」のトピックを参照してください。
- **sybcluster** を使用してサーバをアップグレードする方法については、「**sybcluster** を使用したサーバのアップグレード」の項を参照してください。
- 手動でサーバをアップグレードするには、このマニュアルのあとの項に記載されている手順に従います。

マイグレーションの詳細については、「マイグレート」の項を参照してください。

注意： Adaptive Server 15.5 Cluster Edition で共有インストール・モードからプライベート・インストール・モードにアップグレードする方法については、『リリース・ノート』のアップグレードに関する説明を参照してください。

アップグレード前の作業

アップグレードを実行するには、"SA" 権限を持っている必要があります。

アップグレードを確実に成功させるためには、アップグレード前の作業の一覧をよく読んで、必要に応じて実行してください。古いサーバの設定によっては、アップグレード前の作業を一部省略できます。

注意：アップグレード・プロセスによって *sybsecurity* データベース内の *sysaudits* テーブルが変更されます。Sybase では、アップグレードを行う前に監査データをアーカイブし、これらのテーブルをトランケートすることをおすすめします。これによって、*sybsecurity* データベース内での領域不足によってアップグレードが失敗する可能性を低くします。

1. Cluster Edition をシステムの新しいロケーションにインストールします。
2. システムとアップグレードの要件を確認します。「システムとアップグレードの要件の確認」の項を参照してください。
3. RUN_server ファイルのロケーションを確認します。「runserver ファイルのロケーションの確認」の項を参照してください。
4. Adaptive Server をアップグレードする場合は、前にインストールしたバージョンのサーバを起動しておく必要があります。Backup Server、Historical Server、Monitor Server、XP Server をアップグレードする場合は、これらのサーバを起動しておかないでください。
5. アップグレードには *syscomments* テーブルのストアド・プロシージャ・テキストが必要です。「アップグレードに必要なプロシージャ・テキスト」の項を参照してください。
6. 引用符付き識別子を使用して予約語を解決します。「予約語」の項を参照してください。
7. ユーザがログオフしていることを確認します。
8. データベースの整合性をチェックします。
9. データベースをバックアップします。
10. トランザクション・ログをダンプします。
11. master データベースが "sa" ユーザのデフォルト・データベースになっていることを確認します。
12. アップグレードに備えてデータベースとデバイスを準備します。次の手順に従います。
 - a) 監査を無効にします。

- b) "enable Job Scheduler" が off になっていることを確認して、Job Scheduler を無効にします。
- c) 監査データをアーカイブし、監査テーブルをトランケートします。
- d) ディスク・ミラーリングを無効にします。

注意： Cluster Edition バージョン 15.5 では、ディスク・ミラーリングがサポートされていません。

- e) \$SYBASE 環境変数が、アンロードした新しい Adaptive Server ソフトウェア・ファイルのロケーションを指していることを確認します。
13. プライベート・インストールにアップグレードする場合、「プライベート・インストールへのアップグレード (93 ページ)」を参照してください。
 14. バージョン 12.5.4、15.0.2、または 15.5 のノンクラスタード・サーバからアップグレードしており、それより下位のバージョンのサーバでアーカイブ・データベースへのアクセスが適用されている場合は、アップグレードする前に関連機能を無効にしておきます。
 15. **sybssystemdb** にキャッシュ・バインドがある 12.5.3 インストール環境からアップグレードしている場合、ユーザ定義キャッシュにバインドされた **sybssystemdb** のキャッシュ・バインドを削除してから、**preupgrade** を実行してください。

これを行わないで **preupgrade** を実行すると、次のエラーが表示されます。

```
Current process... infected with 11
```

. このエラーが表示された場合、キャッシュ・バインドを削除してから、**preupgrade** ユーティリティを再度実行します。

16. \$SYBASE ディレクトリが古い \$SYBASE ディレクトリと同じでない場合、Adaptive Server の以前のバージョンから対応する Cluster Edition インストールに次のファイルをコピーします。

- \$SYBASE/interfaces
- \$SYBASE/<servername>.cfg
- \$SYBASE/\$SYBASE_OCS/config/libtcl.cfg

17. データベース機能の Java が有効に設定されている場合は、**sybpcidb** データベースを作成します。**sybpcidb** データベースには、プラグ可能コンポーネント・インタフェース (PCI: Pluggable Component Interface) とすべてのプラグ可能コンポーネント・アダプタ (PCA: Pluggable Component Adapter) コンポーネントの設定情報が格納されます。このデータベースは **installpcidb** スクリプトが使用します。次に例を示します。

```
1> disk init
2> name = "sybpcidb_dev",
3> physname = "/dev/raw/raw20",
4> size = '24MB'
5> go
```

```
1> create database sybpcidb on sybpcidb_dev = 24
2> go
```

注意： `installpci` スクリプトは、クラスタ内の最初のノードに対してのみ実行する必要があり、他のノードに対しては実行しません。また、PCI デバイスは最初のノードにしか作成しないでください。単一インスタンスの PCI 設定がクラスタのノード間で共有されます。

システム・テーブルの統計

サーバを Cluster Edition にアップグレードした場合、`index` および `テーブル・レベル` の統計が不正確になります。

次のシステム・テーブルで `update index statistics` を実行します。

- `sysobjects`
- `sysindexes`
- `syscolumns`
- `systypes`
- `syslogins`
- `sysusers`

アップグレード中のシステム・カタログの変更

Cluster Edition には、いくつかの新しいシステム・カテゴリと既存のカタログに対する変更が導入されています。

Cluster Edition のカタログ・アップグレードは既存のアプリケーションに影響を与える場合があります。アップグレードしている場合は、『Cluster ユーザーズ・ガイド』の「システムの変更点」の章で、影響を受けるカタログの完全なリストを参照してください。

システムとアップグレードの要件の確認

ここでは、システムとアップグレードの要件を確認する方法について説明します。

1. アップグレードする Sybase 製品が搭載されたコンピュータが、システム稼働条件の項に記載されている稼働条件を満たしていることを確認します。
2. サーバのバージョンが Cluster Edition にアップグレード可能かどうか確認します。
3. アンロードした Adaptive Server が、前の Adaptive Server をインストールしてあるディレクトリと異なるディレクトリに入っていることを確認します。ただし、以前にインストールした製品を上書きする場合は、次の手順に従います。
 - a) 最新のバックアップから以前のサーバ環境をリストアします。

- b) Adaptive Server の製品ファイルを別のディレクトリに再インストールします。
 - c) アップグレードを継続します。
4. オペレーティング・システムのリリースが適切であり、Cluster Editionに必要なオペレーティング・システムのパッチがすべてそろっていることを確認します。

runserver ファイルのロケーションの確認

runserver ファイルの名前とロケーションを確認します。現在使用しているサーバの runserver ファイルが、\$SYBASE/\$SYBASE_ASE/install/RUN_server_name にあることを確認します。

ファイルの名前が RUN_servername のままになっていることを確認します。*servername* は、古いサーバの名前です。*servername* は interfaces ファイル内の名前と完全に一致する必要があります。SYBASE サーバ用の RUN_servername ファイルは、RUN_SYBASE になります。現在の Adaptive Server の RUN_servername ファイルに名前が付いている場合は、アップグレード・プロセス中にその名前を変更する必要があります。

アップグレードに必要なプロシージャ・テキスト

syscomments からテキストを削除済みの場合は、ストアド・プロシージャを削除して作り直してから、再びテキストを追加します。Sybase では、テキストを削除するのではなく、**sp_hidetext** ストアド・プロシージャを使用してテキストを隠すことをおすすめします。

予約語

予約語とは SQL 構文の要素で、コマンドの一部として使用されると特別な意味を持つものです。

コマンド構文の一部である単語は、二重引用符で囲まないかぎり、Transact-SQL で識別子として使用することはできません。Adaptive Server をアップグレードして、ユーザ・データベース内の識別子が新しい予約語と一致すると、その識別子を使用するクエリ、ストアド・プロシージャ、またはアプリケーションを実行したときにエラーが発生することがあります。

注意：アップグレードを実行する前に、**sp_renamedb** を使用して予約語と同じ名前のユーザ・データベースの名前を変更しておく必要があります。

オブジェクト名の競合があっても、アップグレード・プロセスの完了が妨げられることはありません。ただし重複しているオブジェクト名を参照するアプリケーションは、アップグレード後は動作しません。

オブジェクト名を変更した場合は、そのオブジェクトを参照しているアプリケーションとストアド・プロシージャも変更してください。

予約語のリストについては、『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。

予約語の競合への対処

sp_dboption を使用して、データベースをシングルユーザ・モードに設定してから、**sp_renamedb** を実行して新しい名前を指定します。

これらのプロシージャの詳細については、『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。

1. その他の識別子が予約語になっている場合は、次の方法で変更します。

- **sp_rename** を使用して、アップグレード前またはアップグレード後にオブジェクト名を変更する。
- 引用符付き識別子を使用する。
- 識別子を角カッコで囲む。次に例を示します。

```
create table [table] ( [int] int, [another int] int )
```

2. *master* データベースとそれぞれのユーザ・データベースで **sp_checkreswords** を実行して、競合する識別子の名前と場所を表示します。

sp_rename と **sp_checkreswords** の詳細や、予約語の競合を避ける方法については、『リファレンス・マニュアル』を参照してください。

引用符付き識別子の使用

予約語の競合を避けるには、サーバ上のすべてのユーザが、予約語が含まれているすべてのストアド・プロシージャとクエリで **quoted_identifier** オプションを呼び出す必要があります。

予約語が含まれるプロシージャとクエリで **set** コマンドの **quoted_identifier** オプションを呼び出すには、予約語になっている識別子を二重引用符で囲みます。**set quoted_identifier** オプションは、Adaptive Server に対して、二重引用符で囲まれた文字列をすべて識別子として処理するように指示します。

予約語チェックの実行

古い Adaptive Serve で予約語チェックを実行します。

1. `installupgrade` の Cluster Edition バージョンをインストールします (\$SYBASE と \$SYBASE_ASE は Cluster Edition の値です)。

```
isql -Usa -Ppassword -Sserver_name  
-i$SYBASE/$SYBASE_ASE/scripts/installupgrade
```

2. `usage.sql` の Cluster Edition バージョンをインストールします。

```
isql -Usa -Ppassword -Sserver_name
-i$SYBASE/$SYBASE_ASE/upgrade/usage.sql
```

- 古い Adaptive Server にログインし、すべてのデータベースに対して **sp_checkreswords** を実行します。次に例を示します。

```
use sybssystemprocs
go
sp_checkreswords
go
```

- 予約語チェックでエラーが見つかった場合は修正します。

プライベート・インストールへのアップグレード

共有インストールからプライベート・インストールにアップグレードします。

Adaptive Server Cluster Edition バージョン 15.5 では、クラスタを "共有" インストールまたは "プライベート" インストールとして設定できます。『Cluster ユーザーズ・ガイド』の「第 1 章 Cluster Edition の概要」を参照してください。

Adaptive Server の対称型マルチプロセッシング (SMP) バージョンを Cluster Edition のプライベート・インストールにアップグレードする作業は、手動で実行する必要があります。最初に、Adaptive Server を Cluster Edition の共有インストールにアップグレードします。次に、次の手順に従ってプライベート・インストールに切り替えます。プライベート・インストールはバージョン 15.0.3 から導入されたので、それより前のバージョンの Adaptive Server Cluster Edition で作成されたクラスタ・インスタンスは、共有インストールとして自動的に引き継がれます。SMP Adaptive Server を共有ディスク クラスタにアップグレードする方法については、使用しているプラットフォーム用の『インストール・ガイド』を参照してください。

注意： Adaptive Server Cluster Edition 15.5 のインストール ロケーションを決定する際には、このノード用のプライベート・インストールをインストールするロケーションを選択します。このロケーションはクラスタに参加している他のノードからアクセスできる必要はありません。

共有インストール・モードからプライベート・インストール・モードへの変更

- クラスタに参加している各ノードに独自の \$SYBASE 環境変数があることを確認します。通常は、プライベート・インストールはローカル・ファイル・システム上で実行されます。これは、クラスタに参加している他のノードがこのインストール環境にアクセスする必要がなくなるからです。
- クラスタに参加している各ノードに Cluster Edition をインストールします。既存のインストール環境が要件を満たしている場合には、それらのノードの 1 つを設定して使用できます。そうでなければ、この処理の最後に既存のインストール環境を破棄できます。既存のインストール環境が、たとえばノードによって使用されている NFS ファイル・システム上にある場合には、その環境を

破棄してローカル・ファイル・システムに新たにインストールすることもできます。各ノードに Cluster Edition をインストールする方法については、使用しているプラットフォーム用の『インストール・ガイド』を参照してください。

3. 各ノードで、クラスタと UAF エージェントを停止します。
4. クラスタに参加しているノードの 1 つで、使用しているシェルに応じて SYBASE.csh または SYBASE.sh を読み込んで、環境を設定します。SYBASE のインストール・ロケーションが共有インストールおよびプライベート・インストールとは異なる場合、共有インストール領域から環境を設定します。
5. クォーラム・デバイスから現在のクラスタ・クォーラム設定を抽出します。次に例を示します。

```
% $SYBASE/$SYBASE_ASE/bin/qrmutil
--extract-config=mycluster_shared.cfg
--quorum-dev=/dev/raw/raw50m41

Executing command 'extract cluster configuration', argument
'mycluster_shared.cfg'...

Extracted input file 'mycluster_shared.cfg'

Command 'extract cluster configuration', argument
'mycluster_shared.cfg' succeeded.

qrmutil execution completed.
```

6. 新しいクラスタ設定ファイルを作成して、必要な情報を更新します。
 - a) 抽出された設定ファイルのコピーを作成してから、その新しいファイルを編集して必要な設定を変更します。たとえば、次のとおりです。cp mycluster_shared.cfg mycluster_private.cfg
 - b) 新しい設定ファイルを編集します。**[cluster]** セクションで次のように変更します。

変更前：

```
installation mode = shared
```

変更後：

```
installation mode = private
```

- c) **[instance]** セクションで次のようにします。
 1. 設定ファイルとインタフェースのエントリを **[cluster]** セクションから **[instance]** セクションに移動します。
 2. SYBASE インストール・ロケーションが共有インストールからプライベート・インストールに変更された場合、エラー・ログ、設定ファイル、およびインタフェース・パス・ロケーションのパスを調整します。
 3. 設定ファイルにインスタンスが複数ある場合、各インスタンスで次の操作を行います。次に例を示します。

```
% cat mycluster_private.cfg
```

```
# All input files must begin with a comment
[cluster]
name = mycluster
max instances = 4
primary protocol = udp
secondary protocol = udp
master device = /dev/raw/rawlg2
traceflags =
additional run parameters =
installation mode = private
membership mode =

[management nodes]
hostname = nuno1
hostname = nuno2

[instance]
name = mycluster_instance1
id = 1
node = nuno1
primary address = nuno1
primary port start = 15100
secondary address = nuno1
secondary port start = 15181
errorlog = /mysybase1/mycluster_inst1.log
config file = /mysybase1/mycluster.cfg
interfaces path = /mysybase1
traceflags =
additional run parameters =

[instance]
name = mycluster_instance2
id = 2
node = nuno2
primary address = nuno2
primary port start = 15100
secondary address = nuno2
secondary port start = 15181
errorlog = /mysybase2/mycluster_inst2.log
config file = /mysybase2/mycluster.cfg
interfaces path = /mysybase2
traceflags =
additional run parameters =
```

7. 更新されたクラスタ設定ファイルをクラスタ・クォーラム・デバイスにロードします。次に例を示します。

```
% $SYBASE/$SYBASE_ASE/bin/qrmutil
--quorum-dev=/dev/raw/raw50m41
--cluster-input=mycluster_private.cfg
Loaded a new quorum configuration.
qrmutil execution completed.
```

8. 次のような条件があるとします。

- クラスタにノードが複数あるかまたは SYBASE インストール・ロケーションを変更した。この場合には、Adaptive Server 設定ファイル (通常の名前は

`servername.cfg`) とインタフェース・ファイルを元の共有インストール・クラスタからプライベート・インストール・クラスタの各インスタンスについて対応するインタフェース・パスと設定ファイルのロケーションにコピーします。これらのロケーションは、更新されたクラスタ設定ファイルの **[instance]** セクションにあります。

- クラスタにあるノードまたはインスタンスは1つのみであり、SYBASE インストール・ディレクトリは変更されていない。この場合、UAF エージェント設定情報を更新します。エージェントのプラグイン XML ファイルは `$SYBASE/UAF-2_5/nodes/[machine_name]/plugins/[cluster_name]/agent-plugin.xml` にあります。

その中で、置換前は次のとおりです。

```
<set-property property="ase.installation.mode"
value="shared" />
```

置換後は次のとおりです。

```
<set-property property="ase.installation.mode"
value="private" />
```

9. プライベート・インストール・ディレクトリを使用してクラスタの各ノードで UAF エージェントを再起動します。 `$SYBASE` ディレクトリから **UAF-2_5/bin/uafstartup.sh** と入力します。
10. クラスタにノードが複数あるかまたは SYBASE インストール・ロケーションが変更された場合、各ノードに UAF エージェント・プラグインを展開します。
 - a) **sybcluster** を起動します。たとえば、次のように入力します。

```
sybcluster -U uafadmin -P -C mycluster
-F "bladel1, blade2,blade2"
```

- b) プラグインを各ノードに個別に展開します。たとえば、次のように入力します。

```
deploy plugin agent "bladel1"
deploy plugin agent "blade2"
deploy plugin agent "blade3"
```

sybcluster と Adaptive Server プラグインの構文と使用方法の詳細については、『Cluster ユーザーズ・ガイド』の「**sybcluster** ユーティリティ」を参照してください。

11. これで、共有インストールからプライベート・インストールにアップグレードできました。クラスタを起動するには **start cluster** コマンド、ノードを個別に起動するには **start instance <instance name>** コマンドを使用できます。

いずれかのコマンドを発行すると、**sybcluster** によって、クォーラム・デバイス上のクラスタ ID がマスタ・デバイスと一致しないことを示すエラー・メッセージを含む、次のメッセージが表示されることがあります。

```
INFO - Starting the cluster mycluster instance
mycluster_instancel using the operating system command:
```



```
/mysybase1/ASE-15_0/bin/dataserver --quorum_dev= /dev/raw/
raw50m41 --instance_name= mycluster_instance1
INFO - 01:00:00000:00000:2009/06/07 23:09:35.46 kernel Quorum
UUID: 00000000-0000-0000-0000-000000000000
INFO - 01:00:00000:00000:2009/06/07 23:09:35.46 kernel Master
UUID: 91f058aa-bc57-408d-854d-4c240883a6c9
INFO - 01:00:00000:00000:2009/06/07 23:09:35.46 kernel Unique
cluster id on quorum device does not match master device. You may
be using the wrong master device. If this is the correct master,
pass 'create-cluster-id' on the command line to pair the devices.
```

この場合、同じコマンドを再発行してください。ただし、メッセージに示されているように、**create-cluster-id**を追加してマスタ・デバイスをペアにしてノードを手動で起動します。たとえば、次のコマンドを発行します。

```
/mysybase1/ASE-15_0/bin/dataserver --quorum_dev= /dev/raw/raw50m41
--instance_name= mycluster_instance1--create-cluster-id
```

これで、コマンドを実行してもエラー・メッセージが表示されません。

- このクラスタに新しいノードを追加するには、Sybase Central の Adaptive Server プラグインか **sybcluster** ツールを使用できます。『Clusters ユーザーズ・ガイド』を参照してください。

これで、共有インストールからプライベート・インストールにアップグレードできました。このクラスタに新しいノードを追加するには、Sybase Central の Adaptive Server プラグインか **sybcluster** ツールを使用できます。『Clusters ユーザーズ・ガイド』を参照してください。

手動によるアップグレードの概要

手動での Adaptive Server のアップグレードは、次の6つのプロセスで構成されます

- アップグレード前の作業を実行します。
- データベースをバックアップします。
- 新しいサーバを独自のインストール・ディレクトリにインストールします。

注意： アップグレードを実行するには、古いインストール環境と新しいインストール環境の両方が必要です。

- Cluster Edition リリース・ロケーションから **preupgrade** ユーティリティを選択します。このユーティリティを古いサーバに対して実行し、基本となるスキーマを更新して各構造体を新しいサーバに合わせます。

注意： Adaptive Server の新しいバージョンでは、それぞれ新しいパラメータ、コマンド、予約語などを使用するさまざまな機能が用意されています。そのため、**preupgrade** を使用して古いサーバでアップグレードの準備をします。た

だし、15.0.1 Cluster Edition または 15.0.3 Cluster Edition から 15.5 Cluster Edition にアップグレードする場合は、この手順を省略できます。

注意： 設定パラメータがデフォルトに設定されていない場合は、通知警告が発行されます。この場合は、特に何も実行する必要はありません。

5. アップグレードの作業を実行します。
6. アップグレード後の作業を実行します。

手動による古いサーバのアップグレード

アップグレードが完了するまで、単一インスタンスで Cluster Edition を起動してください。次の手順に従って、古いサーバを手動でアップグレードします。

1. すべての古いデータベースをバックアップします。
2. 以前のバージョンの Adaptive Server を起動します。
 - a) 古い \$SYBASE ディレクトリに移動します。

```
cd $SYBASE
```
 - b) SYBASE.sh (Bourne シェル) または SYBASE.csh (C シェル) のソースを指定します。

```
source SYBASE.csh
```
 - c) runserver ファイルを実行します。

```
$SYBASE/$SYBASE_ASE/install/RUN_server_name
```
3. 別のウィンドウで、新しい \$SYBASE ディレクトリに移動します。
4. SYBASE.sh (Bourne シェル) または SYBASE.csh (C シェル) のソースを新しい \$SYBASE ディレクトリで指定します。

```
source SYBASE.csh
```
5. \$SYBASE/\$SYBASE_ASE/upgrade にある **preupgrade** ユーティリティを使用して、古いサーバでアップグレード前のテストを実行します。\$SYBASE と \$SYBASE_ASE は、Cluster Edition の値です。アップグレードが完了するまでデフォルト・パケット・サイズを 512 から 2048 に変更しないでください。

注意： アップグレード前に、**default network packet size** が 2048 に設定されると、パケット・サイズとして 2048 バイトを使用するよう **preupgrade** に通知する方法がないため、12.5.x サーバでアップグレード前の作業を完了するためにログインできません。

- a) 以下を実行します。

```
$SYBASE/$SYBASE_ASE/upgrade/preupgrade -Sserver_name -  
Username -Ppassword -I $OLD_SYBASE/interfaces
```

各パラメータの意味は、次のとおりです。

- **\$SYBASE_ASE** – Adaptive Server の Cluster Edition です。
 - -U オプションが無視された場合、-P オプションがシステム管理者のパスワードに使用されます。
- b) アップグレード前のテストで出力されたエラーをすべて修正します。エラーがなくなるまで、**preupgrade** を再実行します。
 - c) 必要に応じて古い Adaptive Server を再起動します。
6. 古い Adaptive Server で予約語チェックを実行します。
 - a) `installupgrade:` の Cluster Edition バージョンをインストールします。


```
isql -Usa -Ppassword -Sserver_name
-i$SYBASE/$SYBASE_ASE/scripts/installupgrade
```
 - b) `usage.sql` の Cluster Edition バージョンをインストールします。


```
isql -Usa -Ppassword -Sserver_name
-i$SYBASE/$SYBASE_ASE/upgrade/usage.sql
```
 - c) 古い Adaptive Server にログインし、すべてのデータベースに対して **sp_checkreswords** を実行します。


```
1> use sybssystemprocs
2> go
1> sp_checkreswords
2> go
```
 - d) 予約語チェックで見つかったエラーを修正します。
 7. 古い Adaptive Server を停止します。
 8. 古い \$SYBASE ディレクトリから古い Adaptive Server の設定ファイル `mycluster.cfg` を新しい \$SYBASE ディレクトリにコピーします。
 9. 15.0.1 Cluster Edition または 15.0.3 Cluster Edition から 15.5 Cluster Edition サーバにアップグレードする場合は、この手順を省略します。ノンクラスタード・サーバからアップグレードしている場合には、この手順を完了してください。クラスタ入力ファイルを作成します。たとえば、以下のような `mycluster.inp` というファイルを作成します。

```
#all input files must begin with a comment

[cluster]
name = mycluster
max instances = 2
master device = /dev/raw/raw101
config file = /sybase/server_name.cfg
interfaces path = /sybase/
traceflags =
primary protocol = udp
secondary protocol = udp

[management nodes]
hostname = blade1
hostname = blade2
```

```
[instance]
id = 1
name = server_name
node = blade1
primary address = blade1
primary port start = 38456
secondary address = blade1
secondary port start = 38466
errorlog = /sybase/install/server_name.log
config file = /sybase/server_name.cfg
interfaces path = /sybase/
traceflags =
additional run parameters =

[instance]
id = 2
name = server_name_ns2
node = blade2
primary address = blade2
primary port start = 38556
secondary address = blade2
secondary port start = 38566
errorlog = /sybase/install/server_name_ns2.log
config file = /sybase/server_name.cfg
interfaces path = /sybase/
traceflags =
additional run parameters =
```

この入力ファイルに必要な値の例については、「クラスタ入力ファイルの作成」のトピックを参照してください。

注意：最初のインスタンスの *server_name* は、アップグレードする古いサーバの名前である必要があります。

10. 15.0.1 Cluster Edition または 15.0.3 Cluster Edition から 15.5 Cluster Edition サーバにアップグレードする場合は、この手順を省略します。ノンクラスタード・サーバからアップグレードしている場合には、この手順を完了してください。クラスタ入力ファイルのインスタンスごとに *interfaces* ファイルにエントリを追加します (手順 9 で説明)。詳細については、「*interfaces* ファイルの設定」のトピックを参照してください。
11. ノンクラスタード・サーバからアップグレードしている場合には、この手順を完了してください。15.0.1 Cluster Edition または 15.0.3 Cluster Edition から 15.5 Cluster Edition サーバにアップグレードする場合は、元のクォーラム・デバイスとクラスタ入力ファイルを使用し、**--buildquorum=force** を指定してクォーラムを再構築し、既存のクォーラムを上書きします。クォーラム・デバイス用に使用するロー・デバイスを決定します。Cluster Edition のバージョンの場合は、共有ディスク上のロー・デバイスを使用します。ファイル・システムのデバイスは使用しないでください。

12. クォーラム・デバイスを作成し、古いマスタ・デバイスを使って新しいインスタンスを起動します。

```

$SYBASE/$SYBASE_ASE/bin/dataserver¥
--instance=server_name¥
--cluster-input=mycluster.inp¥
--quorum-dev=/dev/raw/raw102
--buildquorum
-M$SYBASE

```

注意： `--instance` パラメータによって示される `server_name` は、アップグレードするサーバの名前である必要があります。 `interfaces` ファイルにはこのインスタンス用のエントリが含まれている必要があります。 `-M` などの追加オプションは、データサーバではクォーラムからこれらの値を読み込まないため、`RUN_FILE` で指定する必要があります。データサーバの詳細については、『Cluster ユーザーズ・ガイド』を参照してください。

13. `upgrade` ユーティリティを実行します。 `instance_name` は、アップグレードしているサーバと同じ名前を持つクラスタ内の最初のインスタンスです。

```

$SYBASE/$SYBASE_ASE/upgrade/upgrade
-S instance_name -Ppassword

```

14. 15.0.1 Cluster Edition または 15.0.3 Cluster Edition から 15.5 Cluster Edition サーバにアップグレードする場合は、この手順を省略します。インスタンスにログインします。クラスタ内のインスタンスごとにローカル・システム・テンポラリ・データベース・デバイスとローカル・システム・テンポラリ・データベースを作成します。構文は次のとおりです。

```

create system temporary database database_name
for instance instance_name on device_name = size

```

詳細については、「ローカル・システム・テンポラリ・データベースの設定」のトピックを参照してください。

15. インスタンスを停止します。 `isql` を使用してインスタンスにログインし、次のコマンドを発行します。

```

shutdown instance_name

```

16. クラスタを再起動します。

```

$SYBASE/$SYBASE_ASE/bin/dataserver ¥
--instance=server_name¥
--quorum-dev=/dev/raw/raw102¥
-M$SYBASE

```

17. Cluster Edition にログインし、すべてのデータベースに対して `sp_checkreswords` を実行します。たとえば、インスタンスにログインして次のコマンドを実行します。

```
1> use sybserverprocs
2> go
1> sp_checkreswords
2> go
```

18. 予約語チェックでエラーが見つかった場合は修正します。
19. 古い `run_server` ファイルを新しいディレクトリにコピーして、修正します。このファイルを正しい `$SYBASE` ディレクトリ内のバイナリを指すように編集する必要があります。
 - a) この引数を `run_server` ファイルに追加します。 **--quorum-dev=<path to the quorum device>**
 - b) これで情報はクォラム・デバイスに保存されたため、次のオプションを削除します。
 - **-c**
 - **-i**
 - **-e**詳細については、「`runserver` ファイルの作成」のトピックを参照してください。

20. クラスタ内の各インスタンスを起動します。

```
cd $SYBASE/$SYBASE_ASE/install
startserver -fRUN_server_name
```

21. システム・プロシージャをインストールします。

```
isql -Usa -Ppassword -Sserver_name
-i$SYBASE/$SYBASE_ASE/scripts/installmaster
```

22. Adaptive Server に監査が含まれる場合は、`installsecurity` を実行します。

```
isql -Usa -P password -S server_name
-i$SYBASE/$SYBASE_ASE/scripts/installsecurity
```

23. `installcommit` を実行します。

```
isql -Usa -Ppassword -Sserver_name
-i$SYBASE/$SYBASE_ASE/scripts/installcommit
```

アップグレードに備えたデータベースとデバイスの準備

`preupgrade` を実行した場合は、システム・カタログがスキャンされ、アップグレードに必要な追加領域が計算されます。

`preupgrade` を実行しなかった場合は、システム・カタログとデータベースに追加する必要がある空き領域を手動で計算してください。一般的に、アップグレードするカタログ中で最も大きなもののサイズを2倍し、その値に約10%の領域を追加して計算します。いずれかのデータベースにストアド・プロシージャまたはコンパイル済みオブジェクトが多数ある場合、`syscomments` カタログと `sysprocedures` カタログには、より多くの空き領域が必要になります。

注意： **preupgrade** ユーティリティを使用して、アップグレードに必要な空き領域を調べることをおすすめします。

1. **sybssystemdb** データベースの作成

すべてのサーバに **sybssystemdb** データベースがあることが必要です。Adaptive Server は、このデータベースを使用してトランザクションの追跡とリカバリを行います。さらに、2 フェーズ・コミットと分散トランザクション管理 (DTM) を使用するアプリケーションにも使用します。

2. データベースがアップグレードするのに十分な領域がない場合は、**alter database** を使用してサイズを大きくします。**preupgrade** を使用してデータベース・サイズを判別します。

sybssystemdb を使用しており、データベースが十分な大きさである場合、変更は必要ありません。

3. **sybssystemdb** データベースを使用しておらず、さらに 2 フェーズ・コミットを使用していない場合は、最低でも 4MB の **sybssystemdb** データベースを作成します。必要なサイズは、それぞれのページ・サイズによって異なります。

4. **sybssystemprocs** データベースが十分な大きさであることを確認します。

アップグレードするには、**sybssystemprocs** のサイズを 140MB より大きくします。または、既存の **sybssystemprocs** データベース、およびアップグレードの対象となる最大サイズのカatalog を格納できる十分な空き領域を確保し、アップグレードの変更を記録するため 10% の領域をさらに追加します。ユーザ定義のストアド・プロシージャを追加する場合は、さらに多くの領域が必要です。内部データ構造を追加する場合は 140MB が適切ですが、この値では、大きなサイズのユーザ定義システム・プロシージャが作成される可能性が考慮されていません。

sybprocsdev は、このデバイスのデフォルト名です。ただし、Adaptive Server では度々 **sybssystemprocs** デバイスとして参照されます。

5. データベース機能で Java を有効にするには、PCI を有効にしてから **sybpcidb** データベースを設定します。

sqlupgrade[res] の場合は、古いサーバの **enable java** 設定が検出されます。**preupgrade** が 1 に設定されている場合は、**sybpcidb** デバイスとデータベースを作成してから **upgrade** を実行するように要求されます。

デフォルトのデータベースのサイズを大きくする

sp_helpdb または Sybase Central を使用して、**master**、**model**、**sybssystemprocs**、**tempdb** の各データベースの現在のサイズを算定できます。

master、**tempdb**、**model** の各データベースのサイズを大きくするには、次の手順に従います。

1. **isql** で **alter database** を使用して、*master* データベースのサイズを大きくします。次に例を示します。

```
1> use master
2> go
1> alter database master on master=x
2> go
```

この例では、*master* データベースのサイズが大きくなります。*x* は既存のデータベースに追加する領域のメガバイト数です。

2. この手順を繰り返して、*tempdb* データベースと *model* データベースのサイズも大きくします。次に例を示します。

```
1> alter database tempdb on master=x
2> go
1> alter database model on master=x
2> go
```

注意： *model* データベースを *tempdb* データベースよりも大きくすることはできません。最初に *tempdb* のサイズを大きくしてから、*model* データベースのサイズを大きくしてください。

3. 各データベースのサイズを確認します。次のように入力します。
database_name はチェックするシステム・データベースの名前です。

```
1> sp_helpdb database_name
2> go
```

ASE プラグインを使用したサーバのアップグレード

ASE プラグインにログインする方法については、「環境の設定と Unified Agent の起動」の項から始まる説明を参照してください。

アップグレードを行うには、次のことが前提となります。

- 既存のサーバの **dataserver** は、クラスタ UAF エージェントが実行されている同じシステム上にある必要があります。
- UAF エージェントが実行されているシステムから、Cluster Edition のインストール・ファイルおよび既存のサーバに到達できる必要があります。

これはつまり、古いサーバが、Cluster Edition の UAF エージェントが実行されているのと同じシステムで実行され、新しくアップグレードされたクラスタもそこに配置される必要があるということです。

注意： サーバに接続できない、または左側のウィンドウ枠のツリー・ビュー内での表示に問題がある場合は、[ツール]-[接続] の下のツールバーの [Connection profile] 内に、**interfaces** ファイル (PC-Client の `sql.ini`) のサーバに関連づけられたポート番号が記載されているかどうかを確認します。

Cluster Edition をインストールした後、Adaptive Server プラグインを起動し、左側のウィンドウ枠に古いサーバ・アイコンが表示されていることを確認します。Cluster Server に接続する場合と同じ方法で古いサーバに接続します。詳細については、「クラスタに接続」の項を参照してください。

ASE プラグインを使用した古いサーバのアップグレード

ここでは、ASE プラグインを使用してサーバをアップグレードする方法について説明します。

環境変数を設定して、Unified Agent を起動します。この方法については、「環境の設定と Unified Agent の起動」の項を参照してください。

1. アップグレードを実行する前に、古いサーバをバックアップします。
2. 左のウィンドウ枠のツリー・ビューで古いサーバ・アイコンを右クリックし、[アップグレード] を選択します。ウィザードが開きます。
3. [次へ] をクリックして、処理を続けます。サーバ名を変更するには、クラスタ名のフィールドに新しい名前を入力します。デフォルト名は、古いサーバと同じ名前です。
4. クラスタ・サーバに含まれる最大インスタンス数を選択します。[次へ] をクリックします。アップグレード・ウィザードではクラスタ名を検証し、使用可能なホストをチェックします。
5. 使用可能なホストの一覧から適切なホストを選択します。そのホスト用のエージェントが表示されない場合は、[検出設定] ボタンを使用してプロトコルの設定を確認します。
6. 選択した UAF エージェントの管理者のユーザ名とパスワードを入力します。[次へ] をクリックします。
7. 古いサーバのリリース・ディレクトリのフル・パスを入力します。古いサーバの Adaptive Server ディレクトリと OCS ディレクトリが正しいことを確認します。[次へ] をクリックします。
8. たとえば /dev/raw/raw21 などの、クォーラム・デバイスへのフル・パスと名前を入力します。[次へ] をクリックします。

注意：クォーラムにロー・デバイスを使用することをおすすめします。ファイル・デバイスでクォーラムを実行することは可能ですが、サポートされていません。

9. クラスタの新しいリリース領域へのパスが正しいことを確認します。パラメータとトレースフラグを入力するオプションがあります。パラメータとトレースフラグの詳細については、『Cluster ユーザーズ・ガイド』を参照してください。[次へ] をクリックします。
10. 相互接続ポートの範囲のデフォルトのポート番号を使用します。これらをインスタンスのクエリ・ポートと間違えないでください。後者の詳細については、

『Cluster ユーザーズ・ガイド』の「第 19 章 ネットワークを介する通信の設定」を参照してください。

別のポート番号を選択した場合は、入力したポート番号から始めて、その範囲内のポートが検証されます。使用可能なポートがない場合、ウィザードは範囲にあるすべての使用不可のポートの一覧を表示します。

- a) セカンダリ・ネットワークが使用可能な場合は、チェック・ボックスをオンにします。
- b) [次へ] をクリックします。

11. アップグレード・プロセスで作成されたクラスタのインスタンスの名前を入力します。クラスタが作成されると、クラスタの最大インスタンス数として指定した数までインスタンスを追加できます。インスタンスの設定ページでは各インスタンスごとに開始パラメータとトレースフラグを指定できます。

クラスタの開始パラメータとトレースフラグを指定するには、クラスタの設定ページでそれらを入力します。デフォルトのインスタンス・ログ・ファイル用のパスを確定するか、新しいパスを入力します。[次へ] をクリックします。

注意： アップグレード完了後にさらにインスタンスを追加する方法については、「クラスタへのインスタンスの追加」の項を参照してください。

12. ローカル・システム・テンポラリ・データベース名とサイズを指定するか、デフォルトを受け入れます。あとで追加する予定のインスタンス用のものを含め、すべてのローカル・システム・テンポラリ・データベースを作成する場合、デバイスはすべてのローカル・システム・テンポラリ・データベースを格納できる大きさである必要があります。[次へ] をクリックします。

ローカル・システムの *tempdb* ページで [デバイスの作成] ボタンをクリックし、[デバイスの作成] ダイアログにパス、名前、およびサイズを入力すると、ローカル・システム・テンポラリ・データベースを配置するデバイスを作成できます。

ローカル・システム・テンポラリ・データベースは、共有ディスクにのみ作成できます。テンポラリ・データへのローカル・ディスクの使用の詳細については、『Cluster ユーザーズ・ガイド』の「第 8 章 テンポラリ・データベースの使用」を参照してください。

13. アップグレードが正しく行われたあとでサーバを起動したい場合は、サマリ・ページでチェック・ボックスをオンにします。[次へ] をクリックします。

preupgrade チェックが開始されます。これには時間がかかる場合があります。サーバのアップグレードの準備が整っていない場合、**preupgrade** によってエラーが返されます。アップグレードを正常に実行するには、ウィザードの **preupgrade** チェックで特定されたパラメータに従って、エラーを修正する必要があります。

注意：ウィザードでは設定オプションやデータベースのサイズなどを変更でき、ウィザードを閉じなくても、修正が必要なほとんどのエラーを修正できません。

クラスタ固有の設定パラメータとストアド・プロシージャの詳細については、『Cluster ユーザーズ・ガイド』を参照してください。報告されたエラーの修正については、既存の Adaptive Server 用の『インストール・ガイド』の Adaptive Server バージョンを参照してください。これらのマニュアルは、製品マニュアル・ダウンロード・サイトから入手できます。

14. 報告されたエラーを修正した後、[リトライ]をクリックして再び **preupgrade** チェックを実行します。エラーがない場合、アップグレード・ウィザードは古いサーバを Cluster Edition サーバにアップグレードします。アップグレードが完了すると、左側のウィンドウ枠のツリー・ビューには、古いサーバのアイコンに代わって新しいサーバ用のアイコンが表示されます。

sybcluster を使用したサーバのアップグレード

ここでは、**sybcluster** ユーティリティを使用したアップグレード・プロセス、および古いサーバをアップグレードする前に完了しておく必要のある作業について説明します。Cluster Edition への手動アップグレードを実行することもできます。この方法については、「手動でのアップグレード」の項を参照してください。

sybcluster は、これを使用してクラスタを作成したり管理したりすることのできるコマンド・ライン・ユーティリティです。**sybcluster** では、UAF (Unified Agent Framework) を使用してクラスタ内の各ノードの Unified Agent に「プラグイン」します。Unified Agent では、**sybcluster** コマンドを処理し、これによってユーザはクラスタを管理できます。**sybcluster** と UAF の詳細については、『Cluster ユーザーズ・ガイド』を参照してください。

アップグレードを開始する前に、次の操作を実行します。

- 古いデータベースをバックアップします。
- Cluster Edition をインストールします。アップグレードの対象となるノンクラスタード Adaptive Server と Adaptive Server Cluster Edition バージョン 15.5 のインストール・ディレクトリが同じマシンにあることを確認してください。これは、アップグレード・マシンです。
- アップグレードに必要な情報のリストについては、「クラスタの作成と起動」のトピックと「クラスタの作成のためのワークシート」を参照してください。サーバをアップグレードする前に作業を完了します。
 1. 「アップグレードのためのサーバの確認」のトピックを読んで、サーバでアップグレードの準備ができていることを確認します。

2. サーバを Adaptive Server Cluster Edition バージョン 15.5 にアップグレードする方法については、「Upgrading Your Old Server With **sybcluster**」のトピックを参照してください。
3. サーバを手動でアップグレードしたか、**sybcluster** を使用してアップグレードしたかにかかわらず、アップグレード後の作業を実行します。

アップグレードのためのサーバの確認

この手順では、サーバのアップグレード・プロセスの準備が整っていることを確認します。実際にアップグレード手順を実行するわけではありません。

1. **sybcluster** を起動します。たとえば、次のように入力します。

```
sybcluster -U uafadmin -P -F "mynode:8999"
```

各パラメータの意味は、次のとおりです。

- **-U** - Unified Agent のログイン名です。"uafadmin" はデフォルト値です。
- **-P** - Unified Agent のパスワードです。"uafadmin" のデフォルト・パスワードは、NULL または空白です。
- **-F** - Adaptive Server が実行されているノードと Unified Agent の受信ポートを指定します。デフォルト値は "9999" です。

この例では、ノード名は "mynode"、UAF 受信ポートは "8999" です。

注意：アップグレードを実行するノードを指定する必要があります。

2. 検証を開始します。 **sybcluster** コマンド・ラインで次のように入力します。

```
upgrade server server_name checkonly
```

次に例を示します。

```
upgrade server myserver checkonly
```

3. **upgrade server checkonly** の指示に従ってください。角カッコ内にデフォルト値を示しています。プロンプトは次のとおりです。
 - a) クラスタ [**server_name**] の名前を入力します。サーバ **server_name** 用の既存の Sybase インストール・ディレクトリを入力します。
 - b) Sybase インストール・ディレクトリへの完全パスを入力します。たとえば、"/myserver/sybase15" と入力します。
 - c) サーバ **server_name** 用の ASE インストールを含むサブディレクトリの名前を入力します。たとえば、"ASE-15_0" と入力します。
 - d) サーバ **server_name** 用の OCS インストールを含むサブディレクトリの名前を入力します。たとえば、"OCS-15_0" と入力します。
 - e) サーバ **server_name** での sa ログインの名前を入力します。たとえば、"sa" と入力します。

- f) パスワードを入力します。sa ログイン用のパスワードを入力します。
- g) 最大インスタンス数を入力します。[4]
- h) このクラスタに参加するノード(ハードウェア・ホスト)の数。[1]
- i) クラスタ・ノード 1 を示す数字を入力します。[1]
- j) クォーラム・ディスクのフル・パスを入力します。
- k) トレース・フラグを入力します。
- l) 既存のマスタ・デバイスのフル・パスを入力します。
- m) このクラスタにセカンダリ・ネットワークがあるかどうか。[Y]
- n) この範囲が適用される開始ポート番号を入力します。[15100]
- o) Sybase ホーム・ディレクトリを入力します。

注意：手順 o～s について、**sybcluster** ではデフォルトのロケーションが提供されています。

- p) 環境シェル・スクリプトのパスを入力します。
- q) ASE ホーム・ディレクトリを入力します。
- r) **interfaces** ディレクトリを入力します。
- s) データ・サーバ設定ファイルのパスを入力します。
- t) このクラスタの最初のインスタンスの名前を入力します。

注意：アップグレード・プロセスでは、単一のインスタンスを持つクラスタが作成されます。

- u) *instance_name* の **interfaces** ファイル・クエリ・ポート番号を入力します。

注意：インスタンスのクエリ・ポート番号が古いノンクラスタード・サーバのクエリ・ポート番号と同じであることを確認します。そうでないと、アップグレードに失敗することがあります。

- v) ローカル・システム・テンポラリー・データベース・デバイス名を入力します。ローカル・システム・テンポラリー・データベースは共有ディスクにのみ作成できます。テンポラリー・データに対してローカル・ディスクを使用する方法の詳細については、『Cluster Edition ユーザ・ガイド』の「第 6 章 テンポラリー・データベースの使用」を参照してください。
- w) ローカル・システム・テンポラリー・データベース・デバイスのパスを入力します。
- x) ローカル・システム・テンポラリー・データベースのデバイスのサイズ (MB) を入力します。[100]
- y) ローカル・システム・テンポラリー・データベース名を入力します。例：
[*cluster_name_tdb_1*]
- z) ローカル・システム・テンポラリー・データベースのサイズ (MB) を入力します。[100]
- aa) この設定情報をファイルに保存するかどうか。[Y]

注意： 検証が成功すると、このファイルを入力として使用してサーバをアップグレードできます。また、必要に応じてそれ以降の多数のアップグレードでも入力として使用して、サーバを検証できます。たとえば、次のように入力します。

```
upgrade server server_name file /data/myserver_ce.xml checkonly
```

ab) クラスタ作成情報の保存先ファイルの名前を入力します。例： /data/myserver_ce.xml

ac) 今すぐクラスタを作成するかどうか。[Y]

"Y" と入力すると、**sybcluster** は検証プロセスを開始します。プロセスの進行を示す情報メッセージが表示されます。アップグレード手順を実行するわけではありません。

4. 検証プロセスが成功すると、サーバをアップグレードできます。

5. **sybcluster** によってエラー・メッセージが表示された場合は、サーバをアップグレードする前に、チェック結果のすべてのエラー・メッセージを解決します。

入力ファイルを使用した古いサーバのアップグレード

検証手順が正しく完了したら、入力ファイルを使用してサーバをアップグレードできます。

1. **sybcluster** を起動します。たとえば、次のように入力します。

```
sybcluster -U uafadmin -P -F "mynode:8999"
```

2. アップグレードを開始します。たとえば、次のように入力します。

```
upgrade server myserver file /data/myserver_ce.xml
```

この例では、/data/myserver_ce.xml は、検証手順のときに作成した設定ファイルです。

注意： 設定ファイルは、クラスタを作成するのに **sybcluster** によって使用されるファイルと類似しています。「クラスタの作成と起動」のトピックを参照してください。

3. **sybcluster** によって次のプロンプトが表示されます。

a) サーバ *server_name* での sa ログインの名前を入力します。[sa] [Enter] キーを押して、デフォルトを受け入れるか、ターゲット・サーバで sa 権限を持つ他のユーザの名前を入力します。

b) sa ログイン用のパスワードを入力します。

sybcluster はアップグレード手順を開始します。アップグレードの進行状況やアップグレードの完了を確認する情報メッセージが表示されます。

対話形式での古いサーバのアップグレード

ここでは、対話形式で古いサーバをアップグレードする方法について説明します。

1. sybcluster を起動します。次のように入力します。

```
sybcluster -U uafadmin -P -F "node_name:port_number"
```

2. アップグレードを開始します。次のように入力します。

```
upgrade server server_name
```

3. **upgrade server** の指示に従ってください。角カッコ内にデフォルト値を示しています。プロンプトは次のとおりです。
 - a) クラスタの名前を入力します。[*server_name*]
 - b) Sybase インストール・ディレクトリへの完全パスを使用して、サーバ *server_name* 用の既存の Sybase インストール・ディレクトリを入力します。たとえば、"/myserver/sybase15" と入力します。
 - c) サーバ *server_name* 用の ASE インストールを含むサブディレクトリの名前を入力します。たとえば、"ASE-15_0" と入力します。
 - d) サーバ *server_name* 用の OCS インストールを含むサブディレクトリの名前を入力します。たとえば、"OCS-15_0" と入力します。
 - e) サーバ *server_name* での sa ログインの名前を入力します。
 - f) sa ログイン用のパスワードを入力します。
 - g) 最大インスタンス数を入力します。[4]
 - h) このクラスタに参加するノード (ハードウェア・ホスト) の数。[1]
 - i) クラスタ・ノード 1 を示す数字を入力します。[1]
 - j) クォーラム・ディスクのフル・パスを入力します。
 - k) トレース・フラグを入力します。
 - l) 既存のマスタ・デバイスのフル・パスを入力します。
 - m) このクラスタにセカンダリ・ネットワークがあるかどうか。[Y]
 - n) この範囲が適用される開始ポート番号を入力します。[15100]
 - o) SYBASE ホーム・ディレクトリを入力します。[デフォルト・ロケーション]
 - p) 環境シェル・スクリプトのパスを入力します。[デフォルト・ロケーション]
 - q) ASE ホーム・ディレクトリを入力します。[デフォルト・ロケーション]
 - r) interfaces ディレクトリを入力します。[デフォルト・ロケーション]
 - s) データ・サーバ設定ファイルのパスを入力します。[デフォルト・ロケーション]
 - t) このクラスタの最初のインスタンスの名前を入力します。

注意： 単一のインスタンスを持つクラスタを作成し、**add instance** コマンドを使用して後からインスタンスを追加します。

- u) *instance_name* の interfaces ファイル・クエリ・ポート番号を入力します。
- v) ローカル・システム・テンポラリ・データベース・デバイス名を入力します。
- w) ローカル・システム・テンポラリ・データベース・デバイスのパスを入力します。
- x) ローカル・システム・テンポラリ・データベースのデバイスのサイズ (MB) を入力します。[100]
- y) ローカル・システム・テンポラリ・データベース名を入力します。たとえば、[*cluster_name_tdb_1*] のように入力します。
- z) ローカル・システム・テンポラリ・データベースのサイズ (MB) を入力します。[100]
- aa) この設定情報をファイルに保存するかどうか。 [Y]
- ab) クラスタ作成情報の保存先ファイルの名前を入力します。
- ac) 今すぐクラスタを作成するかどうか。 [Y]

"Y" と入力した場合、**sybcluster** はサーバをアップグレードします。プロセスの進行に応じて情報メッセージが表示されます。

アップグレード後の作業

Cluster Edition にアップグレードしたあとは、以下の手順を実行します。

注意： Cluster Edition にアップグレードした後で、ユーザ・テーブルに対して **update statistics** を実行する必要はありません。アップグレード・プロセスによって、既存の統計値が変更されることはありません。バージョン 15.x からアップグレードする場合は、統計値を使用可能にするためにサーバを再起動する必要があります。

警告！ **installpcidb** を実行すると、**sybpcidb** データベースとそのすべてのテーブルおよびデータが削除され、再作成されます。設定はデフォルトの状態に戻ります。以前にプラグ可能コンポーネント・インタフェース (PCI) の設定を変更している場合は、**installpcidb** を実行したあとで、同じ変更を再び加える必要があります。

1. Adaptive Server Cluster Edition バージョン 15.0.1 から 15.5 Cluster Edition にアップグレードした後に、ユーザ作成ノードごとにプロシージャを 1 回実行して、**jar** ファイルと **.xml** ファイルを更新します。その方法については、「**jar** ファイルと **.xml** ファイルの更新」のトピックを参照してください。

.jar ファイルと .xml ファイルの更新 (113 ページ)

2. Adaptive Server の機能をリストアします。
3. 監査を再び有効にします。

.jar ファイルと .xml ファイルの更新

ここでは、アップグレード前の作業を行う前に .jar ファイルと .xml ファイルを更新する方法について説明します。

アップグレード・プロセスが完了した後、クラスタを管理するため Unified Agent を設定することをおすすめします。これによって、**sybcluster** ユーティリティと Adaptive Server プラグインを使用してクラスタを管理できるようになります。

1. Unified Agent を起動します。「環境の設定と Unified Agent の起動」のトピックを参照してください。
2. **sybcluster** ユーティリティを実行します。「sybcluster を使用したクラスタ・サーバの作成」のトピックを参照してください。
3. **sybcluster deploy plugin** コマンドを使用して、クラスタ用の Unified Agent を設定します。

.xml ファイルが低いバージョン番号 \$LOWVERSION_SYBASE からのものである場合、低いバージョンのインストール・ロケーションを参照してください。 .xml ファイルが高いバージョン番号 \$HIGHVERSION_SYBASE からのものである場合、高いバージョンのインストール・ロケーションを参照してください。

4. バージョンを比較し、低いバージョンの agent-plugin.xml のユニークなプロパティを高いバージョンの agent-plugin.xml のプロパティに追加します。

```
$LOWVERSION_SYBASE/UAF-2_5/nodes/<Host Name>/plugins/<Cluster OR InstanceName>/agent-plugin.xml with $HIGHVERSION_SYBASE/UAF-2_5/nodes/<Host Name>/plugins/<Cluster OR InstanceName>/agent-plugin.xml,
```

たとえば、次のプロパティを低いバージョンの xml ファイルから高いバージョンの xml ファイルにコピーします。

```
<set-property property="ase.database.devices"
value="[1,sysprocsdev,/work/river/ase1501ce/data/
sybsystemprocs.dat,150,135;0,master,/work/river/ase1501ce/data/
master.dat,80,60;2,systemdbdev,/work/river/ase1501ce/data/
sybsystemdb.dat,12,12;3,lstdb1,/work/river/ase1501ce/data/
lstdb1.dat,50,50]" />
```

5. この手順をクラスタ内のすべてのノードに対して繰り返します。
6. UAF エージェントを停止します。

失敗したアップグレードからのリカバリ

アップグレード・プロセス中に、カタログ変更が行われたためにトランザクション・ログがいっぱいになることがあります。このような場合は **isql** を使用して新しいサーバにログインし、次のコマンドを発行します。

アップグレード・プロセス中に、カタログ変更が行われたためにトランザクション・ログがいっぱいになることがあります。このような場合は **isql** を使用して新しいサーバにログインし、次のコマンドを発行します。

```
dump tran dbname with no_log
```

これによってログ領域が解放され、アップグレード・プロセスを続行できるようになります。

アップグレード前に見積もったディスク領域が、アップグレード・プロセスのデータ・コピー・フェーズで不足する場合があります。この場合は、エラーが発生し、アップグレードの *system* セグメントで領域が不足していることが通知されます。アップグレード・プロセスは停止し、必要な領域が空くまで待機します。**isql** を使用して新しいサーバにログインし、**alter database** を使用してデータベースのサイズを大きくします。

```
alter database dbname on device_name = "2m"
```

注意： Adaptive Server では、**alter database** コマンドを使用して単位指定子 "m" または "M" を使用し、変更するデータベースのサイズを指定できます。

データベースでの Java のアップグレード

プラグ可能コンポーネント・インタフェース (PCI) の設定値は、クラスタ内のすべてのノードによって使用されます。作業ディレクトリを使用してノード間でファイルを共有するには、同じパスを使ってすべてのノードに対して表示される共有ディレクトリがファイル・システム内にあることが必要となります。

ノードごとに別々のローカル・テンポラリ・ディレクトリ (通常、Unix では /tmp、Windows では tmp) があります。

1. **installpci** スクリプトは、クラスタ内の最初のノードに対してのみ実行する必要があります。他のノードに対しては実行しません。単一インスタンスの PCI 設定がクラスタのノード間で共有されます。
2. **installpcidb** スクリプトを実行する前に、**sybpcidb** データベースを作成しておく必要があります。このデータベースの場所とサイズは選択できます。デー

データベースの名前が **sybpcidb** であれば、**installpcidb** スクリプトは正しく機能します。次に例を示します。

```
1> disk init
2> name = "sybpcidb_dev",
3> physname = "/dev/raw/raw20",
4> size = '24MB'
5> go
1> create database sybpcidb on sybpcidb_dev = 24
2> go
```

3. **sybpcidb** データベースが作成されたら、**installpcidb** スクリプトを実行して、プラグ可能コンポーネント・インタフェース (PCI) とプラグ可能コンポーネント・アダプタ (PCA) の設定プロパティを格納するテーブルを作成し、値を入力します。ストアド・プロシージャとテーブルが **sybpcidb** データベースに作成されます。**isql** を使用して、このリリースに含まれている **installpcidb** スクリプトを実行します。出力をオペレーティング・システム・ファイルに保存します。次に例を示します。

```
isql -U<sa_password> -P<sa_password> -S<server_name>
-i$SYBASE/$SYBASE_ASE/scripts/installpcidb -o<output_file>
```

4. データベース機能で Java を有効にするには、次の 2 つの設定パラメータを有効にします。

```
1> sp_configure 'enable pci', 1
2> go
1> sp_configure 'enable java', 1
2> go
```

5. これらのパラメータを有効にするために、'**max memory**' を増やす必要がある場合があります。パラメータを有効にした後に、変更を有効にするためにサーバを再起動します。PCI ブリッジのメモリ・プールの最大サイズを '**pci memory size**' 設定パラメータで設定できます。詳細については、『Adaptive Server Enterprise における Java』(バージョン 15.5) を参照してください。

マイグレート

dump と **load** を使用して Cluster Edition にマイグレートする手順については、使用しているプラットフォーム用の Adaptive Server 15.5 のインストール・ガイドを参照してください。

1. 元のサーバに含まれているすべてのデータベース上で **dbcc** チェックを実行し、正しく実行されるかどうか確認します。
2. Cluster Edition サーバを新しいディレクトリに作成します。
3. Cluster Edition サーバと一致するようにデバイスおよびデータベースを作成します。*sysusages* マッピングが正しいことを確認します。

注意： *sybssystemprocs* データベースが使用できる領域を、10% 多くします。

4. 元のサーバからデータベースをダンプします。
5. Cluster Edition サーバへデータベースをロードします。
6. 分割されたテーブルがある場合は分割情報を更新します。
7. Cluster Edition サーバ上で **dbcc** チェックを実行し、正しく実行されるかどうか確認します。

BCP を使用したバージョン 15.x 以前からのマイグレート

bcp ユーティリティを使用して任意のバージョンの Adaptive Server から Cluster Edition にマイグレートできます。

手順の詳細については、『Adaptive Server 15.5 マイグレーション・ガイド』および各プラットフォームごとの『インストール・ガイド』を参照ください。

デバイス、データベース、テーブル、ルール、ストアド・プロシージャ、トリガ、ビューを作成する DDL スクリプトがある場合は、**bcp** コマンドを使用して、古いデータベースから新しいデータベースへデータを移動できます。

DDL スクリプトがない場合は、**ddlgen** ユーティリティを使用してアップグレードする Adaptive Server のスキーマを再作成します。構文および使用法の詳細については、『ユーティリティ・ガイド』を参照してください。

1. データベース上で **dbcc** チェックを実行して、データの整合性を確認します。
2. **bcp** を使用して、データベース内の全テーブルからすべてのデータを抽出します。
3. 新しい Cluster Edition サーバを新しいディレクトリに作成します。
4. デバイス、データベース、テーブルを作成します。
5. **bcp** を使用してデータをテーブルにバルク・コピーします。
6. すべてのビュー、トリガ、ストアド・プロシージャを再作成します。
7. Cluster Edition サーバ上で **dbcc** チェックを実行し、正しく実行されるかどうか確認します。

高可用性設定サーバのマイグレート

Sybase によってアクティブ/アクティブまたはアクティブ/パッシブの高可用性設定になっている Adaptive Server からマイグレートする場合は、Cluster Edition にアップグレードする前に以下を実行します。

高可用性 Adaptive Server の詳細については、『高可用性システムにおける Sybase フェールオーバーの使用』を参照してください。

1. 適切なクラスタ・サブシステム・コマンドを使用して、各ノードで Adaptive Server に関連付けられているリソースの監視を停止します。
2. Adaptive Server がアクティブ/アクティブ高可用性設定になっている場合、高可用性コンパニオン関係を削除します。

- a) 非対称型設定の場合は、セカンダリ・コンパニオンでこのコマンドを発行します。

```
sp_companion primary_companion_name, "drop"
```

- b) 対称型設定の場合は、プライマリ・コンパニオンとセカンダリ・コンパニオンでこのコマンドを発行します。

```
sp_companion companion_name, "drop"
```

3. (アクティブ/アクティブ設定を実行している場合) 両方のコンパニオンでこのコマンドを発行し、両方ともシングル・サーバ・モードであることを確認します。

```
sp_companion
```

シングル・サーバ・モードである場合、それぞれのコンパニオンは次のメッセージを発行します。

```
Server 'server_name' is currently in 'Symmetric normal' mode.
```

4. リソース・グループなど、高可用性用に作成されたクラスタ・サブシステム内のリソースを削除します。これらのリソースは、Cluster Edition では必要ありません。
5. コンパニオン・サーバで設定オプション **enable HA** を無効にします。コンパニオンにログインし、次のコマンドを発行します。

```
sp_configure "enable HA", 0
```

アクティブ/アクティブ設定では、両方のコンパニオンに対してこれを実行する必要があります。

6. 次の項に説明されている手順に従って、Adaptive Server を Cluster Edition にアップグレードします。

サーバのアップグレード

ダウングレード

ここでは、Adaptive Server 15.5 Cluster Edition から前のリリースの Adaptive Server Cluster Edition にダウングレードする方法について説明します。15.x、12.5.x、それより前のリリースへのダウングレードはサポートされていません。15.0.1 Cluster Edition ESD バージョンまたは 15.0.3 Cluster Edition バージョンへのダウングレードについて、以下に説明します。

- 新しいシステム・プロシージャ **sp_downgrade** を使用したダウングレードの基本手順
- ダウングレード後の作業
- 15.5 Cluster Edition の機能を使用した場合のダウングレードに関する情報
- Adaptive Server バージョン 15.5 Cluster Edition に戻す

ダウングレードの基本手順

ここでは、Adaptive Server 15.5 Cluster Edition から 15.0.1 Cluster Edition、15.0.1 Cluster Edition ESD、および 15.0.3 Cluster Edition にダウングレードするための基本手順について説明します。15.5 Cluster Edition の機能を使用した場合は、追加の手順が必要となります。「ダウングレード前に 15.5 Cluster Edition の機能を使用していた場合の追加手順」のトピックに示す手順に従ってください。

Adaptive Server 15.5 Cluster Edition では、ダウングレード・プロセスを効率化する **sp_downgrade** と呼ばれるプロシージャが用意されています。このプロシージャを使用する場合は、"**sybase_ts_role**" が存在する必要があります。このプロシージャを実行するには、"**sa_role**" または "**sso_role**" パーミッションが必要となります。このシステム・プロシージャとその構文の詳細については、『Adaptive Server 15.0.2 新機能ガイド』の「システムの変更点」の章内の「新しいシステム・プロシージャ」の項を参照してください。

1. プラグ可能コンポーネント・インタフェース (PCI) 機能を使用したことがある場合に、15.0.1 Cluster Edition または 15.0.1 Cluster Edition ESD のバージョンにダウングレードするには、**sybpcidb** を削除して、古いサーバ・バイナリを再起動します。
2. バージョン 15.0.3 Cluster Edition にダウングレードする場合は、**installpcidb** を再実行して、古いサーバ・バイナリを再起動します。
3. **dataserver -m** オプションを使用して 15.5 Cluster Edition サーバをシングル・ユーザ・モードで起動し、実行中のインスタンスが 1 つのみになるようにします。これは、ダウングレード手順の間に、他のユーザが Adaptive Server にアク

セスできないようにするためです。シングル・ユーザ・モードでのサーバの起動の詳細については、『ユーティリティ・ガイド』を参照してください。

4. **sp_downgrade** をマスタ・データベースから実行して、Adaptive Server 15.5 Cluster Edition のダウングレードの準備が整っていることを確認します。

```
sp_downgrade 'prepare', @toversion=<'version'>  
, @override = 1
```

"version" の値は、"15.0.1.5"、"15015"、"15.0.3.5" または "15035" として記述できます。ただし、入力したバージョンは、ダウングレードするバージョンと一致させる必要があります。**sp_downgrade 'prepare'** は、Adaptive Server 15.5 Cluster Edition のダウングレードの準備ができていかどうかを検証します。つまり、ダウングレードを完了するために手動で変更を加える必要がある場合は、**sp_downgrade 'prepare'** を実行すると、これらの変更に関するメッセージが出力されます。レポートされたエラーを訂正し、表示された警告が及ぼす影響について理解してからでないと、ダウングレードの手順を先に進めることはできません。**sp_downgrade 'prepare'** でレポートされた問題の修正方法の詳細については、「ダウングレード前に 15.0.3 Cluster Edition の機能を使用していた場合の追加手順」のトピックを参照してください。

5. **sp_downgrade 'prepare'** からエラーがレポートされなくなるまで、手順3を繰り返します。以上でダウングレードの準備が完了しました。
6. **sp_downgrade** をマスタ・データベースから実行して、ダウングレード・プロセスを開始します。

```
sp_downgrade 'downgrade', @toversion = '15015', @override = 1
```

バージョン番号は "15.0.1.5"、"15015"、"15.0.3.5" または "15035" でなければなりません。上記のコマンドが正常に完了した後、15.5 Cluster Edition サーバ上でアクティビティを行うことはできません。15.5 Cluster Edition サーバをただちに停止してください。**sp_downgrade 'downgrade'** によって、sa およびその他のログイン・アカウントのパスワードがリセットされ、アカウントがロックされていることが報告される場合があります。この場合、ダウングレード後に **sp_downgrade** によって表示される新しいパスワードが sa のパスワードとして使用されます。その他の影響を受けたログイン・アカウントは、ダウングレードが完了した後でロック解除する必要があります。

7. チェックポイントを実行し、古いサーバを停止します。
8. **RUN_SERVER** ファイルを、ダウングレードする予定のバージョンのリリース領域にコピーします。ダウングレードする予定のバージョンのリリース領域からの **dataserver** バイナリを使用するように、**RUN_SERVER** ファイルを変更します。**RUN_SERVER** ファイルを変更する場合、**-m** オプションを削除します。または、サーバが **single user** モードで稼働していないことを確認します。
9. 次のコマンドを使用してクォーラム・デバイスを抽出します。


```
qrmutil --quorum-dev=<dev> --extract-config=quorum.out
```

- a) Cluster Edition バージョン 15.0.1 または Cluster Edition 15.0.1 ESD にダウングレードする場合は、quorum.out の次の 2 行をコメントにします。

```
#installation mode=shared
#membership mode=
```

10. 次の行を **dataserver** パラメータに追加することで、作成した古いクォーラム・デバイス・バックアップを使用して新しいクォーラム・デバイスを再構築します。

```
--buildquorum=force --cluster-input=quorum.out
```

11. 修正した RUN_SERVER ファイルを使用して、古いサーバを再起動します。

注意： ダウングレードしたサーバを起動するときに 15.5 の設定ファイルを使用する場合は、新しいオプションを使用すると、「不明なパラメータ」というメッセージが表示されます。不明なオプションが報告されるのは、サーバを初めて起動したときだけです。15.0.1 Cluster Edition サーバにダウングレードする場合、ユーザ・パスワードがリセットされ、コンソールに出力されることがあります。 **sp_downgrade 'downgrade', <version>** の出力をファイルに保存して、パスワードの紛失を防ぎます。古いパスワードは使用できません。パスワードが紛失した場合は、**-psa** を使用してダウングレード後のサーバを再起動し、sa パスワードを再生成する必要があります。

12. 新しいインストールと以前にアップグレードした 15.x インストールのどちらから起動しているかに応じて、以前に保存された restore_msgs.ebf、または何も保存されていないならば、15.0.1 Cluster Edition または 15.0.3 Cluster Edition から instmsgs.ebf を実行し、その後で Cluster Edition の **installmaster** を実行します。
13. クラスタ内のすべてのインスタンスを再起動します。

ダウングレード後の作業

ダウングレードの基本手順を実行した後も、15.5 Cluster Edition へのアップグレードによる変更が一部残っています。

Sybase では、Adaptive Server の各バージョンについて、コンポーネント統合サービスを通じた旧リリースとの接続を確認しています。コンポーネント統合サービスはテストされ、以前のバージョンの Adaptive Server に接続できることが保証されています。

Sybase としては、以前のバージョンの Adaptive Server 上で実行されているコンポーネント統合サービスが後のバージョンに接続できることを保証しません。以前のバージョンの Adaptive Server によってプロキシ・テーブルが後のバージョン

にマップされており、かつ以前のバージョンでは使用できない機能がリモート・テーブルで使用されている場合、エラーが発生することがあります。

たとえば、ローカル・サーバとリモート・サーバがいずれも Adaptive Server バージョン 15.0.1 Cluster Edition を稼働している場合、それぞれをアップグレードするときには必ずローカル・サーバから先にアップグレードしてください。いずれかをアップグレードしない場合、ローカル・サーバを最初にアップグレードしてください。

1. ダウングレードしたサーバで `installmaster`、`installcommit`、`installsecurity`、`installhasvss`、`installmsgsvss` を実行した後、変更されたシステム・ストア・プロシージャはすべて以前の形式に戻ります。Adaptive Server に導入された新しいストア・プロシージャは削除されません。このようなストア・プロシージャを古いバイナリに対して実行しようとすると、予期しない結果が生じます。
2. 以前の 15.0.1 Cluster Edition または 15.0.3 Cluster Edition バージョンから Adaptive Server 15.5 Cluster Edition にアップグレードする間、`restore_msgs.ebf` ファイルが作成されます。ここでは、このスクリプトを実行して、アップグレード元のバージョンにメッセージをリストアする必要があります。

```
isql -Usa -P<sa password> -S<server name> -i<restore_msgs.ebf>
```
3. ストアド・プロシージャ、トリガ、ビューのディスク上構造には、以前のバージョンの Adaptive Server によって認識されない文識別トークン、データ型、オブジェクト参照が含まれていることがあります。ダウングレードしたりリリースよりも後のリリースで Adaptive Server に導入された機能を使用するコンパイル済みオブジェクトは、すべて削除する必要があります。
4. 15.5 Cluster Edition より前のバイナリに戻した後で、次の作業を行います。
 - a) アップグレード・プロセスの間に、**'update all statistics'** を `syslogins` で実行した場合、**delete statistics** を `syslogins` で実行し、統計を再作成する必要があります。
 - b) 15.0.1 Cluster Edition または 15.0.3 Cluster Edition の `installmaster` を実行すると、**spt_values** が削除されてから再作成されるため、このテーブルから新しいすべてのタイプが削除されます。
5. 15.5 Cluster Edition に新しい設定パラメータが追加されました。ダウングレードしたサーバが、15.5 Cluster Edition 設定ファイルを使用して起動される場合、サーバを初めて起動するときこの新しいオプションによって「不明なパラメータ」というメッセージが表示され、不明なオプションが報告されます。設定ファイルは不明なオプションを使用しないで再作成されるので、これらのメッセージは無視してかまいません。15.0.1 Cluster Edition または 15.0.3 Cluster Edition バージョンの `installmaster` を実行すると、`sysurconfigs` に存在していない設定パラメータの `sysconfigures` ローが削除され、15.5 Cluster Edition の設定パ

ラメータが削除されます。 **installmaster** を実行した後にサーバを起動すると、エラー・メッセージは表示されません。

6. Adaptive Server 15.5 Cluster Edition では、キーワード **decrypt_default**, **xmltable**, **path**, が追加されているため、これらの名前を使用して識別子を作成できなくなっています。これらの名前を使用した場合は、アプリケーションを変更する必要があります。詳細については、「ダウングレード前に 15.5 Cluster Edition の機能を使用していた場合の追加手順」のトピックを参照してください。

ダウングレード前に 15.5 Cluster Edition の機能を使用していた場合の追加手順

15.5 Cluster Edition の機能を使用した後でロールバックする場合は、追加の手順が必要となることがあります。これらの手順の中には、15.0.1 Cluster Edition、15.0.1 Cluster Edition ESD、15.0.3 Cluster Edition にダウンロードする前に完了しておく必要のあるものと、ダウンロード後にただちに実行するものがあります。手動による変更が必要な機能については、以下に示すドキュメントを参照してください。

エラーのある項目を修正するには、次の手順に従います。

一般に、戻すバージョンの Adaptive Server でその機能がすでに使用可能であった場合は、追加の手順は必要ありません。ただし、詳細情報を Sybase の Web サイトの製品マニュアル・ページ (<http://sybooks.sybase.com/nav/base.do>) で入手できます。そこでは、使用する機能の詳細について、Adaptive Server Enterprise のマニュアル・セットを検索できます。

1. リアルタイム・メッセージング機能を使用した場合 - リアルタイム・メッセージング機能に関するメッセージング組み込み機能を使用するストアド・プロシージャ、ビュー、トリガをすべて削除します。リアルタイム・メッセージングの詳細については、『Real Time Data Services Messaging Users Guide』を参照してください。
2. 新しいソート順を使用した場合 - 中国語または日本語の文字セット用の新しい nocase sortorder が Adaptive サーバのデフォルトの sortorder として設定されている場合は、15.0.1 Cluster Edition または Cluster Edition ESD リリースにダウングレードする前に、そのリリースと互換性のある sortorder に切り替える必要があります。sortorder を切り替えるとは、すべてのユーザ・インデックスを再構築することを指します。サーバのデフォルトの sortorder を変更する方法の詳細については、『システム管理ガイド』の「第9章 文字セット、ソート順、言語の設定」を参照してください。新しい nocase sortorder の使用中に **sp_downgrade** が呼び出された場合は、次のエラー・メッセージが表示され、ダウングレード・プロセスが中断されます。

```
Cannot downgrade to '%1!' server, which does not support
server's current default sortorder.
```

3. ネイティブ XML を使用した場合 - Adaptive Server の XML サービス機能に、新たに `xmltable` 関数が加わりました。15.5 Cluster Edition で `xmltable` 関数を使用してビューやストアド・プロシージャを作成した後に 15.0.1 Cluster Edition に戻した場合は、エラーが返されます。
4. `instead of` トリガを使用した場合 - `instead of` トリガは、システム・カタログに格納されたオブジェクトです。これらのオブジェクトは、ダウングレード前に削除します。15.0.1 Cluster Edition または 15.0.3 Cluster Edition サーバが起動されると、削除されなかった `instead of` トリガはシステム・カタログに残りますが、実行されるわけではありません。
5. SQL ユーザ定義関数を使用した場合 - SQL ユーザ定義関数は、システム・カタログに格納されたオブジェクトです。ダウングレード前に削除しなかった SQL ユーザ定義関数は、ダウングレード後もカタログに残ります。15.0.1 Cluster Edition または 15.0.3 Cluster Edition バージョンの SQL ユーザ定義関数を削除または実行しようとする、誤解を招くようなエラー・メッセージが生成されます。
6. 15.5 の自動データベース拡張プロシージャを `installdbextend` を使用してインストールした後に、1 つ以上のデータベース・セグメントにスレッショルド・プロシージャを適用した場合は、15.0.1 Cluster Edition または 15.0.3 Cluster Edition へのダウングレード後に、ログ・セグメントに適用されたスレッショルドが正しく機能しないことがあります。セグメントに存在している可能性のある自動拡張スレッショルドをダウングレード前にクリアするには、次のコマンドを実行します。

```
sp_dbextend 'clear', 'threshold'
```

または、既存のルールを変更したり、スレッショルドをクリアしたりすることなく、ダウングレードする前に自動拡張機能全体をサーバ・レベルで無効にすることもできます。 `sa_role` コマンドを使用して、以下を実行します。

```
use master
go
sp_dbextend 'disable', 'database', 'server-wide'
go
```

これにより、スレッショルド・プロシージャが実行時に起動されても、何も処理が行われなくなります。

注意： ポリシーとスレッショルドはすべてそのままにして、機能全体をサーバ・レベルで無効にしてからダウングレードすることをおすすめします。このようにすれば、15.0.2 以降のバージョンに戻したときに自動拡張を再度有効にすることが簡単になります。

7. ダウングレードにおける複写の問題がある場合-ダウングレードするサーバで、暗号化データが格納されているデータベースに対して複写が有効になっている場合は、ダウングレード手順を開始する前に次のいずれかを実行します。

- a) プライマリ・データベースのトランザクション・ログのすべての複写データが、スタンバイ・データベースまたは複写データベースに正常に転送されたことを確認します。確認のプロセスはアプリケーションによって異なります。
- b) 次に示すコマンドを使用して、プライマリ・データベースのトランザクション・ログをトランケートし、Replication Server® でのそのデータベースの RS ロケータを 0 に設定します。プライマリ・データベースで次のコマンドを実行します。

```
sp_stop_rep_agent primary_dbname
dbcc settrunc ('ltm', 'ignore')
dump tran primary_dbname with truncate_only
dbcc settrunc ('ltm', 'valid')
```

Replication Server を停止します。Replication Server の RSSD で次のコマンドを実行します。

```
rs_zeroltm primary_servername, primary_dbname
```

Job Scheduler のダウングレード

Adaptive Server バージョン 15.0.1 Cluster Edition、15.0.1 Cluster Edition ESD、または 15.0.3 Cluster Edition にダウングレードする場合は、Job Scheduler をダウングレードするときに、下位バージョンから **installjsdb** スクリプトを実行します。

1. Adaptive Server をダウングレードする前に、Job Scheduler を無効にします。
2. Adaptive Server をダウングレードします。詳細については、『インストール・ガイド』を参照してください。
3. **installjsdb** スクリプトは、最終的なダウングレード・バージョンから実行します。

```
isql -Usa -Psa_password -Sservername -i $SYBASE_ASE/scripts/
installjsdb
```

注意： **isql** 実行ファイルのロケーションを含むディレクトリ (\$SYBASE_OCS/bin) がパスに含まれている必要があります。

4. 次のように Job Scheduler を有効にします。
5. Job Scheduler を起動するには、サーバを再起動するか、以下を実行します。

```
sp_configure "enable job scheduler", 1
```

```
use sybmgmtdb
go
```

ダウングレード

```
sp_js_wakeup "start_js",1  
go
```

I/O フェンシング

Sybase では、I/O フェンシングが有効にされていない場合のデータ整合性は保証できません。I/O フェンシングを使用しないと、データ損失またはその他のデータ破壊がまれに発生する場合があります。このリスクを受け入れることができるテスト環境と開発環境でない限り、I/O フェンシングを有効にして配備してください。

共有ディスク・クラスタは、協調していないインスタンスの存在を検出してクラスタから削除できます。ただし、めったにないことですが、協調していないインスタンスがクラスタの一部ではなくなっても、そのインスタンスが共有ディスクに書き込むのをクラスタが阻止できない場合があります。たとえば、インスタンスがクラスタから削除されてもリソースの解放や停止が行われていないと、共有ディスクに書き込める場合があります。I/O フェンシングを使用して、協調していないインスタンスがデータを書き込めないようにします。

Cluster Edition では、I/O フェンシングを提供するために SCSI-3 デバイスの SCSI-3 Persistent Group Reservation (PGR) 機能をサポートしています。PGR は SCSI-3 規格で、1 台のディスクが複数のホストによって共有される環境でディスクへの読み込み/書き込みアクセスを管理します。

SCSI-3 PGR 機能で提供されている I/O フェンシングは、パーティションではなく、デバイスに対してのみ作用します。たとえば、`/dev/sda1` と `/dev/sda2` は、デバイス `/dev/sda` のパーティションです。`/dev/sda1` にバインドされているロー・デバイスをターゲットとしたフェンシング動作は、`/dev/sda` のすべてのパーティションに影響するため、そのデバイス上のパーティションを使用しているすべてのファイル・システムまたはその他のアプリケーション (他の Adaptive Server を含む) にも影響します。そのため、そのデバイスはクラスタ・インスタンスによって排他的に使用する必要があります。

I/O フェンシングの有効化

I/O フェンシングを設定すると、I/O フェンシング機能を持つシステム全体のすべてのデバイスに影響します。

1. クラスタ内の各インスタンスを別のノード上で実行します。
2. データベース・デバイスを保持するすべてのストレージ・デバイス (ディスク) が SCSI-3 規格をサポートし、パーティション分割できないようにします。

注意： SCSI-3 PGR 機能は、物理 SCSI ディスク・デバイスまたはストレージ・エリア・ネットワーク (SAN) によってエクスポートされた仮想ディスク・デバイスにのみ使用できます。そのようなデバイスをオペレーティング・システ

ム・レベルでパーティション分割しても、パーティションごとに SCSI-3 PGR 機能は提供されません。言い換えれば、SCSI-3 PGR (つまり、フェンシングのサポート) 機能は、デバイス上のすべてのパーティションによって共有されません。Adaptive Server はデータベース・デバイスを個別にフェンスできません。

- クォーラムは、各々のデバイスに配置されている必要があります。Adaptive Server では、クォーラム・デバイス上にデータベース・デバイスを作成することはできません。
- また、クラスタの外部でデバイス・パーティションが使用されている場合、クラスタによって実行されるすべてのフェンシングは、外部アプリケーション用のその他のパーティションまたはそこに配置されているファイル・システムへのアクセスを拒否します。

3. I/O フェンシングは、特定のデバイス・ドライバをターゲットとしているデバイス・ドライバ API に基づいています。デバイス・ドライバは通常、フェンス・デバイスと呼ばれます。オペレーティング・システム・コマンドを使用して、クラスタを実行する各ノード上にフェンス・デバイスを作成します。**enable i/o fencing** 設定パラメータをオンにするには、次のように入力します。

```
sp_configure "enable i/o fencing", 1
```

4. SCSI-3 PGR 機能はプラットフォームに依存しており、Cluster Edition によって使用されるすべてのデバイスにこの機能を持たせる必要があります。詳細な構文と設定情報については、オペレーティング・システムのマニュアルを参照してください。クラスタ作成プロセスの一部として、Adaptive Server プラグインと **sybcluster** の両方を使用すると、各デバイスの IO フェンシングが有効であることを確認できます。次の **qrmutil** ユーティリティを実行することもできます。

```
qrmutil -Q<quorum path> --fence-capable=<device path>
```

I/O フェンシングの設定

ここでは、I/O フェンシングの設定方法について説明します。

1. Linux で I/O フェンシングを有効にするには、オペレーティング・システム・コマンドを使用して、共有ディスク・クラスタを実行する各ノード上に SCSI 汎用 (sg) ドライバをロードします。ドライバをロードすると、`/dev/sg*` ファイルが自動的に作成されます。
2. sg ドライバによってノード上に作成されるすべての `/dev/sg*` ファイルの各インスタンスに読み込み/書き込みパーミッションを付与します。

たとえば、ロー・パーティション `/dev/raw/raw1` 上のデータベース・デバイスが `/dev/sg3` にマップされている場合は、`/dev/sg3` ファイルに書き込みパーミッションを付与します。

3. 設定されたデータベース・デバイスに対応する `/dev/sg` ファイルへの書き込みアクセス権が Cluster Edition に付与されていることを確認します。また、フェンス・デバイスには、SCSI-3 PGR コマンドを含む `/dev/sg` ファイルへの書き込みアクセス権が必要です。フェンス・デバイスのパスは対応する物理デバイス・パスと異なる場合があります。たとえば、データベース・デバイスは `/dev/raw/raw1`、物理デバイスは `/dev/sda`、フェンス・デバイスは `/dev/sg0` となります。

注意： `sg` ドライバがシステム起動時に（またはユーザ・コマンドを使用して）ロードされるたびに、システムによって `/dev/sg*` ファイルが再作成されます。起動プロセスの一部としてこれらのファイルにパーミッションが適切に付与されていることを確認してください。付与されていない場合、システム管理者はシステムが起動するたびにパーミッションを手動で再設定します。

SySAM のトラブルシューティング

SySAM のエラーがタイムリーに解決されないと、Adaptive Server が機能しなくなる可能性があります。ここでは、SySAM 猶予期間について説明します。

SySAM FAQ のオンラインの『Sybase ソフトウェア資産管理 (SySAM) ユーザーズ・ガイド』に最新の情報が掲載されている場合があります。このインストール・ガイドの最新のトラブルシューティング情報については、製品マニュアル・サイトを参照してください。Sybase 製品が必要なライセンスを見つけられない場合、猶予期間が使用可能であれば、猶予モードの操作が続きます。**sp_lmconfig** コマンドの出力を使用して、製品のライセンス・ステータスを判別します。Status カラムは、猶予期間内に操作中のライセンスに対して `graced` と表示されます。

一般に、SySAM 関連の問題を解決するために 30 日の猶予期間があります。ライセンスが猶予期間に入ると、Sybase 製品のエラー・ログおよびオプションの電子メール・メッセージには、猶予期間の終了する日時が含まれます。また、**sp_lmconfig** コマンドを実行して期間の終了日を判別できます。期間の終了日まで問題が修正されない場合、Sybase 製品は、そのライセンスの対応する機能を無効にします。有効期限が切れたライセンスが ASE_CORE の場合、Adaptive Server は停止します。

Sybase 製品ログには、猶予モードで与えられたライセンスに関する情報も記載されます。重大度が「警告」のものに対する電子メール通知が設定されている場合、このイベントについて電子メール・メッセージが送信されます。詳細については、「電子メール通知の有効化と変更」のトピックを参照してください。Adaptive Server のエラー・ログおよび電子メール・メッセージの警告は、ライセンスが猶予モードの間は繰り返されます。

SySAM 関連エラーを検索する場所

SySAM 関連エラーはすべて Sybase 製品エラー・ログに出力され、プレフィックスとして "kernel SySAM." が付きます。次に例を示します。

```
<timestamp> kernel SySAM: Failed to obtain 1 license(s)
for ASE_CORE feature from license file(s) or server(s).
```

エラー・ログの保存場所については、「エラー・ログのロケーション」の項を参照してください。

電子メール警告が設定されている場合、受信者は SySAM イベントが発生するたびに電子メールを受け取ります。

Sybase 製品での SySAM ライセンスの現在のステータスは、`sp_lmconfig` コマンドを使っていつでも確認できます。出力には、現在の SySAM 設定と各ライセンスのステータスが示されます。

サブド・ライセンス・モデルを使用している場合、log ディレクトリにあるライセンス・サーバのデバッグ・ログには、ライセンスのチェックアウトに関するすべての問題が記録されます。

SySAM のトラブルシューティング

ここでは、SySAM のトラブルシューティング方法について説明します。

表 10 : SySAM のトラブルシューティング

説明	処理
Adaptive Server のインストール中にライセンスの種類を検出できない	<p>Adaptive Server インストール・プログラムは、最も一般的に使用されるライセンスの種類のみを表示する。古いライセンスまたは新しいライセンスの種類を使用していると、使用しているライセンスがインストール・プログラムでリストに表示されない場合がある。</p> <p>必要なライセンスの種類が見つからない場合、ドロップダウン・リストで [不明] または [その他] を選択し、インストールを完了する。インストールが完了した後、<code>sp_lmconfig</code> を使用してライセンスの種類を変更する。<code>sp_lmconfig</code> の使用方法については、『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。</p> <p>注意： ライセンスの種類を「不明」または「その他」に設定すると、Adaptive Server は使用できる最初のライセンスを採用します。複数の種類のライセンスを保有している場合、Adaptive Server は不正なライセンスを選択する場合があります。ライセンス・サーバに複数の種類のライセンスを保有している場合、正しいライセンスの種類を設定します。</p>

説明	処理
<p>インストール・プログラムが、適切なライセンスが見つからないという警告を発行した</p>	<p>サブド・ライセンス・モデルを選択すると、インストール・プログラムは、選択したエディションとライセンスの種類を確認する。適切なライセンスがない場合は、警告メッセージが表示される。</p> <p>新規の Adaptive Server インストールを実行している、またはバージョン 12.5.x 以前の既存の Adaptive Server をアップグレードしている場合は、インストールを続行する。Adaptive Server には、ライセンスの問題を解決するために 30 日の猶予期間がある。インストールが完了したら、ライセンス・チェックアウトの失敗をデバッグする。詳細については、この表の「Adaptive Server がライセンスをチェックアウトできず、代わりに猶予ライセンスを起動する」を参照。</p> <p>Adaptive Server バージョン 15.0 以降にアップグレードする場合、適切なライセンスを保有していない場合はアップグレード後に Adaptive Server を起動できない場合がある。インストール・プログラムを完了し、既存の Adaptive Server がライセンスをチェックアウトできるかどうか確認する。チェックアウトできない場合、この問題をトラブルシューティングしてから、アップグレードを続行する。</p> <p>既存の Adaptive Server がライセンスをチェックアウトできる場合は、使用しているライセンスが、適用する更新の使用を認可していることを確認する。これは、日付ベース・バージョンのライセンス・ファイルと、この更新がリリースされた日付を使用して判別可能。更新がリリースされた日付以降の日付ベース・バージョンのライセンスを保有する必要がある。Adaptive Server には、サポートが変更された後に、ライセンスを更新するための猶予期間がある。</p> <p>ライセンスの日付ベース・バージョンは、sp_lmconfig の出力の <i>Version</i> カラムで、Adaptive Server エラー・ログのライセンス・チェックアウト・メッセージを参照するか、ライセンス自体を調査することで判別可能。詳細については、『Sybase ソフトウェア資産管理 ユーザーズ・ガイド』の「付録 A ライセンス・ファイルの内容」を参照。</p> <p>Adaptive Server のリリース日は、バージョン文字列、カバー・レター、SPDC ダウンロード・リンクに表示される日付で判別可能。この日付は、通知されたエラー・メッセージにも表示される。</p> <p>更新がリリースされた日のサポート料金を支払っている場合、更新されたライセンスを SPDC からダウンロードできる。サポート料金が支払われていない場合、更新のインストールは認可されない。</p> <p>SySAM での日付ベース・バージョン管理方法については、『Sybase ソフトウェア資産管理ユーザーズ・ガイド』の「付録 B」を参照。ライセンスの更新方法については、『Sybase ソフトウェア資産管理 ユーザーズ・ガイド』の「第 3 章 ライセンスの取得と使用」を参照。</p>

説明	処理
<p>ライセンス・サーバの実行可能プログラムとスクリプトがインストールされない</p>	<p>そのライセンス・サーバの <code>sysam</code> ラップ・スクリプトおよび実行可能プログラム (<code>lmgrd</code> と <code>SYBASE</code>) が <code>SYSAM-2_0¥bin</code> ディレクトリにインストールされていないか見つからない。</p> <p>ライセンス・サーバ・コンポーネントは、デフォルトではインストールするように選択されていない。ライセンス・サーバをインストールするには、カスタム・インストール・オプションを終了し、ライセンス・サーバ・コンポーネントを選択する。</p> <p>または、スタンドアロン SySAM ライセンス・サーバのインストール・プログラムを SPDC からダウンロードできる。</p>
<p>ライセンス・サーバが起動しない</p>	<p>ライセンス・サーバが起動に失敗する最も一般的な原因は、サブド・ライセンスがサーバに存在しないことである。ライセンス・サーバを起動するには、<code>licenses</code> ディレクトリに少なくとも 1 つのサブド・ライセンスが存在する必要がある。新しいライセンス・サーバをインストールする場合、<code>licenses</code> ディレクトリにはサブド・ライセンスが存在しないので、SPDC からのライセンスをアクティブにして、それを <code>licenses</code> ディレクトリにコピーするまでライセンス・サーバは起動されない。</p> <p>ライセンス・サーバが起動しないその他の理由は以下のとおり。</p> <ul style="list-style-type: none"> • SPDC からアクティブにされたライセンスは、アンサーブド・ライセンスだった。ライセンス・ファイルを検査する必要がある。サブド・ライセンスは、必ず <code>SERVER</code> ヘッダで開始する。ライセンスの先頭に <code>SERVER</code> で開始する行がない場合は、ライセンス・サーバで無効なアンサーブド・ライセンスを使用している。 • ライセンス用に特別なポート番号を使用している場合、そのポート番号はすでに使用されている可能性がある。"<code>netstat -a</code>" を使用して、そのポート番号が空いていることを確認する。空いていない場合は、そのポートを解放するか、ライセンス・サーバ用に別のポート番号を使用する。 • <code>SERVER</code> ヘッダに記録されているホスト名が実際のホスト名に一致していない。ホスト名は、<code>SERVER</code> キーワードの横に記録される。一致しない場合、ライセンス・ファイルのホスト名を訂正するか、それを、すべてのホスト名と使用できるキーワード "<code>this_host</code>" に設定する。 • コピーされたライセンスは、別のマシンに対してアクティブにされた可能性がある。行に記録されたホスト ID を調査する。ホスト ID は、<code>SERVER</code> ヘッダのホスト名の横に記録される。これは、ライセンス・サーバが実行されているマシンのホスト ID と一致する必要がある。

説明	処理
<p>ライセンス・サーバがライセンス・ファイルを認識しない</p>	<p>ライセンス・サーバがライセンスの処理を拒否する場合、次の理由による可能性があります。</p> <ul style="list-style-type: none"> • ライセンスが別のマシンに対してアクティブにされたか、不正なホスト ID を使用してアクティブにされた。そのマシンのホスト ID を持つライセンスに記録されたホスト ID を調査する必要がある。それらが一致しない場合、ライセンスを正しいホスト ID でチェックインして再生成する。 • ライセンスが変更されている。ライセンス内の情報は、指定されたもの以外変更できない。他の情報を変更すると、ライセンスは拒否される。ライセンスが紙面に印刷されたものから入力されて作成された場合、ライセンスの入力にエラーがないかを確認する。アクティブにされたライセンスの新しいコピーを SPDC からダウンロードすることもできる。 <p>プラットフォームのホスト ID がネットワーク・アダプタ ID をベースにしている場合、有効な NIC に関連づけられている ID を使用したことを確認する。ループバック・アダプタに関連づけられた ID は有効にならない。リムーバブル・ネットワーク・アダプタに関連づけられた ID が使用される場合、そのアダプタがコンピュータに接続されていることを確認する。</p>
<p>Adaptive Server が起動せず、ライセンス・チェックアウトに失敗するエラーが発生した</p>	<p>Adaptive Server が必要なライセンスをチェックアウトできない場合、ライセンスを猶予モードで発行できるかどうかを判別する。猶予期間が指定できない場合、ライセンスは認可されない。Adaptive Server のベース・ライセンス (ASE_CORE) が猶予モードで、問題が猶予期間の終了までに修正されないと、Adaptive Server は起動に失敗する。</p> <p>詳細については、この表の「Adaptive Server がライセンスをチェックアウトできず、代わりに猶予ライセンスを起動する」を参照。猶予期間に発行されたライセンスを解決するためのトラブルシューティングのヒントは、この問題の解決にも該当する。</p>

説明	処理
<p>Adaptive Server がライセンスをチェックアウトできず、代わりに猶予ライセンスを起動する</p>	<p>sysam diag feature_name コマンドを SYSAM-2_0¥bin ディレクトリから実行する。ここで、<i>feature_name</i> は Adaptive Server 用の SySAM 機能名または猶予モードのオプション機能。機能名は Adaptive Server のエラー・ログとオプションの電子メール通知に出力される。</p> <p>diag コマンドによってチェックアウトできるライセンスが表示されない場合、これは次のいずれかの理由による。</p> <ul style="list-style-type: none"> • サブド・モデルを使用している場合： <ul style="list-style-type: none"> • ライセンス・サーバが実行されていること、および使用しているマシンからアクセスできることを確認する。"sysam status" を使用して、ライセンス・サーバにアクセスできることを確認する。ライセンス・サーバにアクセスできない場合は、ライセンス・サーバが実行しているかどうかを確認する。 • ライセンス・サーバが実行している場合は、"sysam status -f feature_name" を使用して、ライセンス・サーバが指定された機能のライセンスを提供しているかを判別する。提供していない場合は、SPDC から正しいライセンスを取得する。 • ライセンス・サーバ上のすべてのライセンスが使用中。"sysam status -f feature_name" が使用できるライセンスはないことを示す場合、追加のライセンスを取得するか、Adaptive Server の既存のインスタンスを停止する。 • アンサーブド・モデルを使用している場合： <ul style="list-style-type: none"> • 指定された機能のライセンスがローカルの licenses ディレクトリにコピーされていない可能性がある。SPDC から正しいライセンスを取得し、それをローカルの licenses ディレクトリにコピーする。 • ライセンスが別のマシンに対してアクティブにされたか、不正なホスト ID を使用してアクティブにされた。ライセンスを正しいホスト ID でチェックインして再びアクティブにする。 • 使用できるライセンスは別のオペレーティング・システムまたはアーキテクチャのライセンスである。SPDC から正しいプラットフォームのライセンスを取得する。 • 端末サーバ環境で実行している場合、アンサーブド・ライセンスは使用できない。端末サーバ環境用のサブド・ライセンスを設定する。 • 使用できるライセンスがこのバージョンの Adaptive Server の使用を認可していない可能性がある。日付ベースのバージョン管理に関する情報については、『Sybase ソフトウェア資産管理ユーザーズ・ガイド』の「付録 B 製品更新版、EBF、およびサポート更新のインストール」を参照。ライセンスの更新方法については、

説明	処理
	<p>『Sybase ソフトウェア資産管理ユーザーズ・ガイド』の「第3章 ライセンスの取得と使用」の「Sybase ライセンスの更新」の項を参照。</p>
<p>Adaptive Server がライセンスをチェックアウトできず、代わりに猶予ライセンスを起動する(続き)</p>	<p>diag コマンドによって、指定されたライセンスがチェックアウトできることが示される場合、Adaptive Server がライセンスをチェックアウトできない理由は次のとおり。</p> <ul style="list-style-type: none"> • 製品エディションまたはライセンスの種類が要件に一致しない。edition および license type 設定パラメータが設定されている場合、Adaptive Server はこれらの設定に一致するライセンスのみを使用する。 <ol style="list-style-type: none"> 1. sp_lmconfig を実行して設定値を確認する。 2. ライセンスの VENDOR_STRING 属性をチェックして、一致するライセンスが存在することを確認する。 一致するライセンスが存在しない場合は、SPDC から適切なライセンスを取得するか、Adaptive Server の設定を調整する。 猶予期間が過ぎているために Adaptive Server が起動できない場合、SYBASE ディレクトリの下の ASE-15_0¥sysam ディレクトリにある servername.properties ファイルで確認する。これらの値は、PE= および LT= で始まる2行に格納されている。これらの行は、設定がある場合のみ存在する。 • 詳細については、この表の「ライセンスが存在しているにも関わらず Adaptive Server はオプション機能のライセンスを見つけれない」を参照。
<p>問題が修正されたあとも Adaptive Server は猶予ライセンスを表示する</p>	<p>Adaptive Server は定期的にライセンス・チェックを行うが、ライセンスのステータスは即時に更新されない。ステータスは、次のハートビート・サイクルが完了したあとにのみ更新される。これには数時間かかる場合がある。</p>

説明	処理
<p>ライセンスが存在しているにも関わらず Adaptive Server はオプション機能のライセンスを見つけない</p>	<p>"sysam diag feature_name" を実行して、オプション機能のライセンスが存在し、Adaptive Server が実行されているマシンからチェックアウトできることを確認する。機能が存在しているが Adaptive Server からチェックアウトできない場合、この原因は次のいずれかである。</p> <ul style="list-style-type: none"> • そのオプション機能は他のエディション用の機能である。 • アクティブ・フラグまたはスタンバイ・フラグが、ベースの Adaptive Server およびオプション機能に一致しない。 <p>sp_lmconfig を実行して、エディション、アクティブおよびスタンバイの設定値を判別する。エディション、アクティブおよびスタンバイの値は、<i>Property Name / Property Value</i> のテーブルに表示される。エディション値は PE ローに、アクティブおよびスタンバイ値は AS ローに表示される。</p> <p>Adaptive Server の PE および AS の値を、オプション機能ライセンスの <i>VENDOR_STRING</i> および <i>ISSUER</i> フィールドに記録されている同じ値と比較する。これらは一致する必要がある。ASE_HA のような一部の機能はスタンバイでは提供されないため、この機能のライセンスには "AS=A S" フラグがない。このフィールドは、アクティブおよびスタンバイ機能のないライセンスを検索するときは考慮されない。</p>
<p>Adaptive Server が、目的のエディションまたはライセンスの種類を起動しない</p>	<p>edition および license type 設定パラメータが設定されていない場合、Adaptive Server は使用できる最初の ASE_CORE ライセンスを使用する。異なるエディションおよびライセンスの種類を持つ複数の ASE_CORE ライセンスがある場合、最初に使用可能になるライセンスは、ライセンス・ファイル・ディレクトリのソート順、そのライセンス・サーバで使用できるライセンスなどの多くの要素に依存する。</p> <p>認可されるエディションおよびライセンスの種類は確認できないため、edition および license type 設定パラメータの指定が推奨される。これらの値が設定されていると、Adaptive Server は必ずその設定で起動される。一致するライセンスが使用できない場合、Adaptive Server は、ライセンスの問題を解決できるように猶予モード (可能であれば) で起動される。</p>

説明	処理
Adaptive Server が正しいライセンスを選択しない	<p>Adaptive Server がライセンスを検索する場合は、次のロケーションを検索する。</p> <ul style="list-style-type: none"> Windows プラットフォームの registry に保存された値、または UNIX プラットフォームの .flexlmrc ファイル。"lmpath -status" を使用して保存された値を表示する。 SYBASE_LICENSE_FILE と LM_LICENSE_FILE 環境変数の値セット。 SYSAM-2_0 ディレクトリの下に licenses ディレクトリにあるすべてのライセンス・ファイル (.lic 拡張子を持つすべてのファイル)。 <p>適切なライセンスが見つかるまで、前述のすべてのロケーションが、指定された順番で検索される。ディレクトリが指定されると、そのディレクトリのライセンス・ファイルがディレクトリのソート順でロードされる。</p> <p>機能名、バージョン、エディション、ライセンスの種類フィルタが最初に一致するライセンスが使用される。このライセンスは、目的のものではない可能性がある。前述のロケーションを調整することで、検索されるライセンスを制御する。</p>

Sybase サポート・センタに問い合わせ

SySAM 関連の問題について、Sybase 製品の保守契約を結んでいるサポート・センタに電話する場合、次の情報を用意してください。

- Adaptive Server のエラー・ログ。
 - `$SYBASE/ASE-15_0/install/<servername>.log`
- Adaptive Server が起動しない場合は、サーバのプロパティ・ファイル。
 - `$SYBASE/ASE-15_0/sysam/<servername>.properties`*servername* は、起動しない Adaptive Server の名前と対応させる必要があります。
- Adaptive Server を実行しているマシンの SYSAM-2_0¥licenses ディレクトリに保存されたライセンス・ファイル。
- SYBASE_LICENSE_FILE および LM_LICENSE_FILE 環境変数の値セット。
- `lmutil lmpath -status` コマンドの出力。lmutil は、SYSAM-2_0¥bin フォルダの下に bin ディレクトリにあります。
- サブド・ライセンス・モデルを使用している場合：
 - ライセンス・サーバの `$SYBASE¥SYSAM-2_0¥licenses` ディレクトリに保存されたライセンス。

SySAM のトラブルシューティング

- \$SYBASE¥SYSAM-2_0¥log ディレクトリのライセンス・サーバ・ログ・ファイル。

サーバのトラブルシューティング

Sybase サーバ製品では、エラーには次の 2 種類があります。

- インストール、アップグレード、セットアップ・ユーティリティによって生成されるエラー。
- サーバ (Adaptive Server、Backup Server など) によって生成されるエラー。

エラーの原因を特定するには、まず使用しているユーティリティのログ・ファイルの中から、問題が発生したときにユーティリティが実行していたタスクを探します。次にサーバのエラー・ログをチェックします。詳細については、「エラー・ログのロケーション」の項を参照してください。

表「UNIX プラットフォームのトラブルシューティングのガイドライン」に、初めてインストールまたはアップグレードするときに発生する可能性がある、一般的な問題の原因と解決法を示します。引き続き問題が発生する場合は、インストールまたはアップグレードを再試行してください。

インストール・プログラムまたは `srvbuild` が予期せず終了した場合や、問題を解決できない場合は、『トラブルシューティング&エラー・メッセージ・ガイド』を参照してください。

表 11 : UNIX プラットフォームのトラブルシューティングのガイドライン

問題	考えられる原因と解決法
インストール・プログラムが Adaptive Server を起動できない。	<p>Adaptive Server 起動の失敗は通常、RAM またはディスクの使用可能領域の不足によって発生する。</p> <p>必要な RAM 容量の条件を満たしているか確認する。RAM 容量の条件を満たしていたら、すべてのアプリケーションを削除し、その後、すべてのアプリケーションをハード・ドライブにもう一度インストールして、インストール・プロセスを再開する。</p> <p>Adaptive Server のインストール後、ディスク・ドライブの空きディスク領域は 25MB 必要である。Adaptive Server が共有メモリ・ファイルを作成するには、約 18MB 必要である。</p> <p>システム管理者としてログインしているか確認する。Adaptive Server を起動するには、システム管理者としてログインする必要がある。</p> <p>Monitor Server を停止してから Adaptive Server を再起動する。</p> <p>注意： SySAM ライセンスが見つからない場合や猶予期間が過ぎている場合、Adaptive Server は起動しません。Adaptive Server のエラー・ログでライセンス・チェックアウトに失敗した理由を調べ、問題を解決してください。</p>
Adaptive Server のアップグレード後、 srvbuild を使用できない。	サーバのアップグレードを開始すると、他のタスクに対して同じ srvbuild セッションを使用できない場合がある。 srvbuild を終了し、再起動する。
アップグレードした Adaptive Server にインストール・プログラムが接続できない。	サーバのアップグレードを開始すると、他のタスクに対して同じ srvbuild セッションを使用できない場合がある。 srvbuild を終了し、再起動する。
インストール・プログラムが予約語の競合を検出した。	「失敗の後での Adaptive Server の停止」のトピックを参照。
アップグレードに失敗する。	「失敗したアップグレードからのリカバリ」の項を参照。

エラー・ログのロケーション

エラー・ログ内の情報は、エラー・メッセージが表示された理由と、考えられる解決法を知るのに役立ちます。

インストール・ユーティリティのログ

エラー・ログ内の情報は、エラー・メッセージが表示された理由と、考えられる解決法を知るのに役立ちます。ここでは、インストール、アップグレード、セットアップ・ユーティリティに関するデフォルトのエラー・ログのロケーションを示します。

表 12：インストール・ユーティリティのエラー・ログのロケーション

ユーティリティ	エラー・ログのロケーション
InstallAnywhere	\$SYBASE/log/ASE_Suite.log
srvbuildres	\$SYBASE/\$SYBASE_ASE/init/logs/srvbuildMMDD.NNN 各パラメータの意味は、次のとおりです。 <ul style="list-style-type: none"> • <i>MM</i>は月です。 • <i>DD</i>は日です。 • <i>NNN</i>は srvbuild セッションを示す 3 桁の数字です。
sqlupgraderes	<ul style="list-style-type: none"> • \$SYBASE/\$SYBASE_ASE/init/logs/sqlupgradeMMDD.NNN • \$SYBASE/\$SYBASE_ASE/upgrade/upgrade.NNN (アップグレード・プロセスで作成されたテンポラリ・ファイル)

Sybase サーバのログ

エラー・ログ内の情報は、エラー・メッセージが表示された理由と、考えられる解決法を知るのに役立ちます。ここでは、Sybase サーバごとのデフォルトのエラー・ログのロケーションを示します。

表 13：UNIX 用 Sybase サーバのエラー・ログのロケーション

Server	デフォルトのエラー・ログ・パスとファイル名
Adaptive Server	\$SYBASE/\$SYBASE_ASE/install/servername.log
Backup Server	\$SYBASE/\$SYBASE_ASE/install/servername.log
Monitor Server	\$SYBASE/\$SYBASE_ASE/install/servername.log
XP Server	\$SYBASE/\$SYBASE_ASE/install/servername.log

インストール時に起こりがちな問題の解決

この項で説明されていない問題が発生した場合は、『トラブルシューティングおよびエラー・メッセージ・ガイド』を参照してください。

表 14：インストール時に起こりがちな問題の解決

問題	考えられる原因と解決法
X-Window を使用できない	<p>セットアップ・ユーティリティと設定ユーティリティが正しく表示されない場合、モニタの解像度の調整が必要な場合がある。</p> <p>フォント・サイズを小さくするには、次の UNIX コマンドを発行する。</p> <pre>% cd \$SYBASE/ASE-15_0 % chmod +w xappdefaults % cd xappdefaults % chmod +w * % foreach i(*) ? cat \$i sed -e "s/140/100/g" sed -e "s/^#D/D/g" sed -e "s/^#S/S/g" > p ? mv p \$i ? end %</pre> <p>フォントを小さくすると、インストール・ユーティリティによって使用されるウィンドウ領域は約 25% 縮小される。</p>
ドライブから CD または DVD を取り出せない	<p>ドライブから CD を取り出せない場合、次の項目を確認する。</p> <ul style="list-style-type: none"> • UNIX 端末ウィンドウで CD ドライブ・パスが現在のディレクトリ (pwd) であるかどうか。現在のディレクトリになったら、別のディレクトリに移動する (cd)。 • sybhelp プロセスの場合。プロセスが存在していたら、UNIX の kill コマンドを使って強制終了する。

問題	考えられる原因と解決法
<p>DISPLAY 環境変数が正しく設定されていない</p>	<p>このエラー・メッセージは、<i>remote</i> マシンの DISPLAY 環境変数が、インストーラを <i>local</i> マシンに正しく表示するよう設定されていないことを意味する。</p> <pre>The DISPLAY environment variable is not set correctly.</pre> <p>問題を解決するには、<i>remote</i> マシンの UNIX プロンプトで次のコマンドを入力する。ここで、<i>host_name</i> は、インストーラを表示するマシン (<i>local</i> マシン) の名前。C シェルでは、次のように入力する。</p> <pre>setenv DISPLAY host_name:0.0</pre> <p>Bourne シェルでは、次のように入力します。</p> <pre>DISPLAY=host_name:0.0; export DISPLAY</pre>
<p>クライアントからサーバへの接続が許可されない</p>	<p>インストーラを実行し、次のエラー・メッセージが表示された場合、作業中のローカル・マシンにユーザ・インタフェースを表示するためのパーミッションをリモート・マシンが持っていないことを意味する。</p> <pre>Xlib: connection to "host_name" refused by server Xlib: Client is not authorized to connect to Server xhost: unable to open display "host_name"</pre> <p>この問題を解決するには、次の手順に従います。</p> <ol style="list-style-type: none"> 1. <i>local</i> マシンの UNIX プロンプトで次のコマンドを入力します。ここで、<i>remote_machine</i> はインストーラを実行中のマシンです。 <pre>xhost +remote_machine</pre> <ol style="list-style-type: none"> 2. インストーラを再起動します。
<p>アドレスがすでに使われている</p>	<p>srvbuild からの次のメッセージは、入力したポート番号がすでに使用されていることを意味する。</p> <pre>kernel: ninit: bind, Address already in use</pre> <p>この問題を解決するには、別のポート番号を srvbuild ウィンドウに入力する。 netstat -a コマンドを使用して、使用中のポート番号のリストを出力する。</p>

問題	考えられる原因と解決法
Adaptive Server が起動に失敗する	<p>オペレーティング・システムの共有メモリが不足している場合がある。</p> <p>この問題を解決するには、「サーバのインストール」のトピックを参照して共有メモリ値を調整し、もう一度インストール・プロセスまたはアップグレード・プロセスを開始する。</p>
インストーラが起動しない	<p>インストーラが起動しない場合、デバッグ・モードを使用して再起動する。</p> <ol style="list-style-type: none"> 1. 環境変数 LAX_DEBUG を true に設定してから、インストーラを実行する。
XP Server を起動できない	<p>XP Server が xp_cmdshell やその他の拡張ストアード・プロシージャによって起動されるときに、次のようなメッセージが表示されることがある。</p> <pre>Msg 11018, Level 16, State 1: Procedure "xp_cmdshell", Line 2: XP Server must be up for ESP to execute. (return status = -6)</pre> <p>Adaptive Server の sys.servers テーブルに XP Server エントリがあることを確認する。Adaptive Server とは別の srvbuild セッションで XP Server を作成して、関連する Adaptive Server を指定しなかった場合、srvbuild は sys.servers テーブルを更新できない。XP サーバが interfaces ファイルまたは LDAP サーバに存在することを確認する。</p> <p>sp_addserver を使用してエントリを sys.servers テーブルに追加する。</p>
リソース・ファイル・インストールのトラブルシューティング	<p>srvbuild[res]、sqlloc[res]、または sqlupgrade[res] を使用して構築、設定、またはアップグレードの処理中に問題が発生した場合は、これらのユーティリティで Adaptive Server が正しく停止するための十分な時間を取れなかった可能性がある。</p> <p>SYBSHUTWAIT 環境変数を設定し、Adaptive Server が停止するまで待つようにユーティリティに強制する。たとえば、次のコマンドは、次のタスクに進む前に Adaptive Server が停止できるように 2 分間待つようユーティリティに強制する。</p> <pre>% setenv SYBSHUTWAIT 120</pre>

問題	考えられる原因と解決法
設定が失敗した	<p>システム共有ライブラリのロケーションを確認する。共有ライブラリ <code>libXt.a</code> と <code>libX11.a</code> は通常は <code>/usr/lpp/X11/lib/R5</code> に格納されている。 <code>libXm.a</code> ライブラリは通常は <code>/usr/lpp/X11/Motif1.2/lib</code> または <code>/usr/lpp/X11/R5/Motif1.2</code> にある。</p> <p>これらの共有ライブラリが上記以外のディレクトリにある場合は、<code>LIBPATH</code> 環境変数とそのディレクトリを示すように設定する。</p> <p>共有ライブラリ <code>libXt.so</code> と <code>libX11.so</code> は通常は <code>/usr/openwin/lib</code> に格納されている。共有ライブラリ <code>libsocket.so</code> は通常は <code>/usr/lib</code> に格納されている。</p> <p>オペレーティング・システムの共有ライブラリがこれ以外のディレクトリにある場合は、その場所を <code>LD_LIBRARY_PATH</code> 環境変数に設定する。</p>

設定に失敗した場合

システム共有ライブラリのロケーションを確認します。

共有ライブラリ `libXt.so` と `libX11.so` は通常、`/usr/openwin/lib` に格納されています。共有ライブラリ `libsocket.so` は、通常 `/usr/lib` にあります。

オペレーティング・システムの共有ライブラリがこれ以外のディレクトリにある場合は、その場所を `LD_LIBRARY_PATH` 環境変数に設定してください。

失敗の後でのサーバの停止

ここでは、失敗した後でサーバを停止する方法について説明します。

Adaptive Server の起動後にインストールまたはアップグレード・セッションが失敗した場合、次の手順に従って、**shutdown** コマンドを使用します。

1. "sa" としてログオンします。
2. **shutdown with nowait** コマンドを使用して Adaptive Server を停止します。
with nowait オプションを使用すると、現在実行している SQL 文の終了を待たず、ただちに Adaptive Server が停止します。

```
1> shutdown cluster with nowait
2> go
```

リソース・ファイル・インストールのトラブルシューティング

`srvbuild[res]`、`sqlloc[res]`、または `sqlupgrade[res]` を使用して Adaptive Server の構築、設定、またはアップグレードの処理中に問題が発生した場合は、これらのユーティリティで Adaptive Server が正しく停止するための十分な時間を取れなかった可能性があります。

`SYBSHUTWAIT` 環境変数を設定すると、Adaptive Server が停止するまで待つようにユーティリティに強制できます。たとえば、次のコマンドは、次のタスクに進む前に Adaptive Server が停止できるように2分間待つようユーティリティに強制します。

```
% setenv SYBSHUTWAIT 120
```

Adaptive Server がアップグレード前の適格性テストに失敗した場合

Adaptive Server がアップグレード前のテストに失敗した場合、`preupgrade` は次のメッセージを表示します。

```
Server SERVER_NAME failed preupgrade eligibility test.  
See log for more information.
```

1. [アップグレード] ウィンドウで [終了] を選択します。
2. `$$SYBASE/$$SYBASE_ASE/init/logs` ディレクトリに作成されたログ・ファイルを調べて、Adaptive Server がアップグレード前の適格性テストに失敗した理由を確認します。
3. 問題を解決したら、Adaptive Server を停止し、`sybcluster` または手動によるアップグレードを使用してアップグレード・セッションを完了します。

失敗したインストールからのリカバリまたはファイル作成後のリカバリ

インストールに失敗した場合、インストール・プログラムはエラー・メッセージを表示します。エラー・メッセージと Adaptive Server のエラー・ログを確認して、インストール失敗の原因を把握します。エラー・ログのデフォルトの保存場所については、「サーバのトラブルシューティング」の項を参照してください。

Adaptive Server の設定中にインストール・プログラムが終了した場合は、次の手順に従います。

1. Adaptive Server が生成したログ・ファイルの内容を確認します。エラー・ログのデフォルトの保存場所については、「サーバのトラブルシューティング」の項を参照してください。
2. 問題を解決するために推奨されている対処法を実行します。

3. マスタ・デバイスやシステム・プロシージャ・デバイス・ファイルなどのオペレーティング・システムのファイルをインストール・プログラムが作成した後でインストールに失敗した場合は、これらのファイルを削除します。
4. インストールしようとしている Adaptive Server をインストール・プログラムが起動した後でインストールに失敗した場合は、そのサーバを停止します。

失敗したアップグレードからのリカバリ

アップグレード・プロセスが失敗した場合、インストール・プログラムはエラー・メッセージを表示します。エラー・メッセージと Adaptive Server のエラー・ログを確認して、アップグレード失敗の原因を把握します。エラー・ログのデフォルトの保存場所については、「サーバのトラブルシューティング」の項を参照してください。

アップグレードに失敗すると、データベースのリストアが必要な場合があります。

アップグレードの失敗または失敗の原因によってデータベースが破損したと思われる場合は、バックアップからデータベースをリストアします。データベースのリストアについては、『システム管理ガイド』を参照してください。

データベースが破損した可能性がある場合は、**sqlupgrade** を終了しますが、バックアップからデータベースをリストアするまでアップグレード・セッションを再開しないでください。リストアが完了したら、アップグレードを再試行します。

アップグレードの再実行

サーバ・インストール環境のアップグレード中の失敗は、個々のデータベースをアップグレードする際の失敗、またはすべてのデータベースをアップグレードした後に設定変更を完了する際の失敗のいずれかに分類されます。

1. 個々のデータベースのアップグレードに失敗した場合は、アップグレードを手動でリトライできます。最初に、失敗の原因となった問題を解決します。アップグレード・ユーティリティの出力で問題を確認する必要があります。最も一般的な失敗の原因は、一部のリソースが不足することです。領域（データまたはログ）、ロック、補助スキャン記述子などが考えられます。**alter database** コマンドを使用してデータベースに領域を追加できます。リソースに関する他の失敗は、**sp_configure** ストアド・プロシージャを使用してサーバの設定を変更することで修正できることがあります。このトレース・フラグを設定すると、ユーザ "sa" はオフライン・データベースを使用して必要な変更を行い、アップグレード時の失敗を修正できます。
2. アップグレードの失敗でデータベースがオフラインのままになり、失敗はデータベースのデータ変更でのみ修正される場合、失敗したデータベースへは **isql** または同様のプログラムを使用してアクセスして、影響を受けたサーバにユーザ "sa" として接続し、次のコマンドを発行します。

```
dbcc traceon(990)
```

注意： このトレース・フラグへのアクセスは、ユーザ "sa" のみが認可されています。 "sa_role" を持つアカウントを使用しても十分ではありません。 "sa" のログインを無効にしてある場合は、それを再度有効にし、この方法でアクセスする必要があります。

- 失敗したアップグレードを再開するには、**online database** コマンドを使用します。

```
online database <failed_db_name>
```

サーバは、データベースのアップグレードに失敗した時点から再開します。

- 失敗が、すべてのデータベースのアップグレード後に発生した場合、または失敗がアップグレード・ユーティリティのクラッシュの原因になった場合は、ユーティリティを手動で再実行できます。最初に失敗の診断して修正してから、アップグレード・ユーティリティを実行します。

```
$_SYBASE/$_SYBASE_ASE/upgrade/upgrade
```

この方法で再開した場合、アップグレード・プロセスで「開始中」ではなく「検証中」と表示されますが、当初のアップグレードと完全に同じチェックが実行されます。

- データベースが正常にアップグレードされたことを確認するには、**online database** コマンドを使用してデータベースのアップグレード・ステータスを調べます。データベースをアップグレードする必要がある場合は、このコマンドによって実行します。このような手順で、指定のインストール時にすべてのデータベースを確認することもあります。

```
declare @dbname varchar(255)
select @dbname = min(name)
from master..sysdatabases
while @dbname is not null
begin
online database @dbname
select @dbname = min(name)
from master..sysdatabases
where name > @dbname
end
```

注意： サーバがリカバリできないようなアップグレードの失敗もあります。たとえば、システム・テーブルをバージョン 15.0 形式にアップグレードすることは、必要な変更の間のある時点で、失敗の影響を非常に受けやすいものです。そのような失敗を検出した場合は、失敗したデータベースをバックアップからリストアします。アップグレードが再び失敗しないようにするには、最初の失敗の原因となった問題を修正した後に、そのデータベースの **online database** コマンドを発行します。このような致命的な失敗は、前に説明したようにリソース不足が原因で発生し、最終的にはトランザクションのアポートを元に戻すのに失敗します。

失敗の原因がわかっている場合

エラー・ログまたはエラー・メッセージによって失敗の原因が明確に示された場合やデータベースが破損していないと思われる場合は、次の手順に従って、問題を解決してアップグレードをただちに再実行することができます。

1. 必要な対処法を実行して、問題を解決します。
たとえば、既存のデータベースに十分な領域がないためにアップグレードが失敗したことがエラー・ログに示されている場合は、**alter database** コマンドを使用して使用可能な領域を増やします。
2. Adaptive Server の停止が必要な場合があります。
サーバを停止すると、インストール・プログラムがサーバを起動してアップグレード・セッションを再実行できるようになります。
3. **sybcluster** を実行するか、手動によるアップグレードを再開します。
4. [Upgrade Adaptive Server] を選択してアップグレードを続行します。

アップグレード・プロセスがまた失敗し、失敗の原因を判断できない場合は、アップグレードが失敗した段階と場所をエラー・ログ・ファイルから確認して、Sybase 製品の保守契約を結んでいるサポート・センタまでお問い合わせください。

デフォルトでは、ログ・ファイルは \$SYBASE/\$SYBASE_ASE/install/errorlog にあります。

dbcc upgrade_object を使用したコンパイル済みオブジェクトのアップグレード

Adaptive Server version 11.9.3 から、コンパイル済みオブジェクトをそのソース・テキストに基づいてアップグレードするプロセスが導入されました。

コンパイル済みオブジェクトは、次に示すものです。

- 検査制約
- デフォルト
- ルール
- ストアド・プロシージャ (拡張ストアド・プロシージャを含む)
- トリガ
- ビュー

各コンパイル済みオブジェクトのソース・テキストは、手動で削除されていないかぎり *syscomments* テーブルに格納されます。サーバをアップグレードする場合は、ソース・テキストが *syscomments* に存在するかどうか、そのプロセスで検証されます。ただし、コンパイル済みオブジェクトは、それらが呼び出されるまで実際にはアップグレードされません。

たとえば、**list_proc** というユーザ定義のストアド・プロシージャがあるとし
ます。Adaptive Server 15.0 へアップグレードするときに、**list_proc** のソース・テキ
ストが存在するかどうか検証されます。アップグレード後、最初に **list_proc** が呼
び出されると、Adaptive Server はコンパイル済みオブジェクトである **list_proc** が
アップグレードされていないことを検出します。Adaptive Server は、*syscomments*
内のソース・テキストに基づいて **list_proc** を再コンパイルします。次いで、その
新しいコンパイル済みオブジェクトが実行されます。

アップグレードされたオブジェクトは、アップグレードの前に使用していたのと同
じオブジェクト ID とパーミッションを持ちます。

sp_hidetext を使用してソース・テキストが隠されているコンパイル済みオブ
ジェクトも、ソース・テキストが隠されていないオブジェクトと同様にアップグ
レードされます。**sp_hidetext** 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。

注意： 32 ビットのインストール環境をアップグレードして 64 ビットの Adaptive
Server を使用するようにすると、各データベースの *sysprocedures* テーブルに含ま
れている 64 ビットのコンパイル済みオブジェクトのサイズは、アップグレード後
に約 55% 大きくなります。正確なサイズは、アップグレード前のプロセスで計算
されます。この値に従って、アップグレードされるデータベースのサイズを大き
くしてください。

コンパイル済みオブジェクトが、アップグレードの成功後に呼び出されることを
確実にするために、**dbcc upgrade_object** コマンドを使用して手動でアップグ
レードできます。詳細については、「コンパイル済みオブジェクトにおける運用
前のエラー検出」を参照してください。

コンパイル済みオブジェクトがアップグレードされているか調べる方法

コンパイル済みオブジェクトがアップグレードされているか調べるには、次のい
ずれかを実行します。

sysprocedures.version カラムを確認します。オブジェクトがアップグレードされた
場合は、このカラムに数値 15000 が含まれます。

ポインタのサイズを同じバージョンの 64 ビット・ポインタにアップグレードする
場合は、*sysprocedures.status* カラムを調べます。このカラムは、オブジェクトが 64
ビット・ポインタを使用することを示す 0x2 という 16 進数ビット設定を含んでい
ます。ビットが設定されていない場合、オブジェクトは 32 ビット・オブジェクト
であり、アップグレードされていないことを意味します。

コンパイル済みオブジェクトにおける運用前のエラー検出

前のバージョンの Adaptive Server で行った変更が原因で、12.5.x 以降のバージョンで、コンパイル済みオブジェクトが異なる動作をする場合があります。

dbcc upgrade_object を使用することにより、次のようなエラーと発生する可能性のある問題点を見つけることができます。正しく動作させるには、これらに手動で変更を加える必要があります。

エラーと発生する可能性のある問題点を調べて、変更の必要がある部分を修正した後に、**dbcc upgrade_object** を使用することにより、サーバがオブジェクトを自動的にアップグレードするまで待たないでコンパイル済みオブジェクトを手動でアップグレードできます。詳細については、「**dbcc upgrade_object** の使用」の項を参照してください。

予約語エラー

dbcc upgrade_object がコンパイル済みオブジェクト内でオブジェクト名として使用される予約語を検出すると、エラーが表示され、そのオブジェクトはアップグレードされません。

エラーを修正するには、手動でオブジェクト名を変更するか、オブジェクト名を引用符で囲んで **set quoted identifiers on** コマンドを実行します。その後、コンパイル済みオブジェクトを削除して再作成します。

たとえば、データベース・ダンプを Adaptive Server 11.5 から Adaptive Server 15.0 にロードするとき、“lock” という単語を使用しているストアド・プロシージャがダンプに含まれているとします。そのストアド・プロシージャに対して **dbcc upgrade_object** を実行すると、バージョン 11.5 では予約語ではなかった“lock”がバージョン 11.9.2 では予約語になっているため、エラーになります。このエラー表示によって、ストアド・プロシージャおよび関連するテーブルを事前に変更してから運用環境で使用できます。

削除、トランケート、または破損したソース・テキスト

syscomments 内のソース・テキストが、削除、トランケート、または損傷している場合は、**dbcc upgrade_object** が構文エラーを表示することがあります。

ソース・テキストが隠されていない場合は、**sp_helptext** を使用してソース・テキストが完全なものかどうか調べることができます。トランケートまたはその他の破損が発生している場合は、コンパイル済みオブジェクトを削除して再作成します。

引用符付き識別子のエラー

ここでは、引用符付き識別子のエラーについて説明します。

dbcc upgrade_object は、次の場合に引用符付き識別子のエラーを返します。

- 11.9.2 より前のリリースで引用符付き識別子をアクティブにしてコンパイル済みオブジェクトが作成された (**set quoted identifiers on**)。
- 引用符付き識別子が、現在のセッションでアクティブではない (**set quoted identifiers off**)。

このエラーを避けるには、引用符付き識別子をアクティブにしてから **dbcc upgrade_object** を実行してください。引用符付き識別子がアクティブな場合は、二重引用符ではなく一重引用符で **dbcc upgrade_object** キーワードを囲む必要があります。

引用符付き識別子のエラーが発生する場合は、**set** コマンドを使用して **quoted identifiers** をアクティブにしてから、**dbcc upgrade_object** を実行してオブジェクトをアップグレードします。

バージョン 11.9.2 以降で作成されたコンパイル済みオブジェクトの場合は、アップグレード・プロセスが、引用符付き識別子を必要に応じて自動的にアクティブにしたり非アクティブにしたりします。

注意： 引用符付き識別子は、二重引用符で囲まれたリテラルと同じではありません。リテラルの場合は、アップグレードの前に特別なアクションを行う必要はありません。

テンポラリー・テーブルの参照

ストアド・プロシージャやトリガなどのコンパイル済みオブジェクトがテンポラリー・テーブル (**#temp table_name**) を参照する場合、それがオブジェクト本体の外に作成されるとアップグレードは失敗して、**dbcc upgrade_object** はエラーを返します。

このエラーを修正するには、コンパイル済みオブジェクトが必要とするのと同じテンポラリー・テーブルを作成してから、**dbcc upgrade_object** を再実行してください。コンパイル済みオブジェクトを自動的にアップグレードする場合は、この作業は必要はありません。

select * で発生する可能性がある問題

Adaptive Server 11.9.3 以降では、それより前のバージョンの Adaptive Server で作成されたストアド・プロシージャ、トリガ、またはビュー内の **select *** 句の結果が、期待どおりにならないことがあります。

この変更の詳細については、『リファレンス・マニュアル』を参照してください。

dbcc upgrade_object は、ストアド・プロシージャの最も外側のクエリ・ブロックに **select *** 句を検出するとエラーを返し、オブジェクトをアップグレードしません。

たとえば、次のようなストアド・プロシージャがあります。

```
create procedure myproc as
  select * from employees
go
```

```
create procedure yourproc as
  if exists (select * from employees)
    print "Found one!"
go
```

myproc が最も外側のクエリ・ブロックに **select *** 句を持った文を含んでいるので、**dbcc upgrade_object** は **myproc** に関してエラーを返します。このプロシージャはアップグレードされません。

dbcc upgrade_object は、**yourproc** に関してはエラーを返しません。これは、**select *** 句がサブクエリ内にあるからです。このプロシージャはアップグレードされます。

select * をビュー内で変更する必要があるかどうか調べる方法

dbcc upgrade_object がビュー内に **select *** が存在するとレポートする場合は、元のビューの *syscolumns* の出力をテーブルの出力と比較して、ビューが作成された以降にテーブルに対してカラムが追加または削除されていないかどうか調べてください。

1. たとえば、次の文があるとします。

```
create view all_emps as select * from employees
```

2. *all_emps* ビューをアップグレードする前に、次のクエリを使用して、元のビューのカラム数と更新後のテーブルのカラム数を調べます。

```
select name from syscolumns
  where id = object_id("all_emps")
select name from syscolumns
  where id = object_id("employees")
```

3. 2つのクエリの出力を比較します。
 - a) テーブルがビューより多数のカラムを含んでいて、**select *** 文のアップグレード前の結果を保持することが重要な場合は、**select *** 文を、特定のカラム名を指定する **select** 文に変更してください。
 - b) ビューが複数のテーブルから作成された場合は、ビューを構成するすべてのテーブルのカラムを調べて、必要に応じて **select** 文を書き換えてください。

警告！ **select *** 文をビューから実行しないでください。実行すると、ビューがアップグレードされて、*syscolumns* 内の元のカラム情報に関する情報が上書きされます。

4. ビュー内のカラム数と新しいテーブル内のカラム数との差を調べる別の方法は、ビューとビューを構成するテーブルの両方について **sp_help** を実行することです。

- a) この比較は、ビューに対してだけ実行でき、他のコンパイル済みオブジェクトに対しては実行できません。他のコンパイル済みオブジェクト内の **select * 文**の変更が必要かどうかを調べるには、各コンパイル済みオブジェクトのソース・テキストを調べてください。

dbcc upgrade_object の使用

ここでは、**dbcc upgrade_object** の構文、パラメータ、パーミッション、ログ・セグメントの増大、エラー・レポート、および使用例を示します。

dbcc upgrade_object を使用するための構文

ここでは、**dbcc upgrade_object** を使用するための構文を示します。

```
dbcc upgrade_object [( dbid | dbname  
    [, ['database.[owner].]compiled_object_name' |  
    'check' | 'default' | 'procedure' | 'rule' |  
    'trigger' | 'view'  
    [, 'force' ] ] ) ]
```

パラメータ

ここでは、パラメータについて説明します。

- **dbid** – データベース ID を指定します。**dbid** を指定しない場合は、現在のデータベース内のすべてのコンパイル済みオブジェクトがアップグレードされます。
- **dbname** – データベース名を指定します。**dbname** を指定しない場合は、現在のデータベース内のすべてのコンパイル済みオブジェクトがアップグレードされます。
- **compiled_object_name** – アップグレードする特定のコンパイル済みオブジェクトの名前です。完全に修飾された名前を使用する場合は、**dbname** と **database** が一致している必要があります、完全に修飾された名前を引用符で囲む必要があります。データベースが同じ名前の複数のコンパイル済みオブジェクトを含んでいる場合は、完全に修飾された名前を使用してください。そうしないと、同じ名前を持つすべてのオブジェクトが解析され、エラーがない場合はアップグレードされます。
- **check** – すべてのルールと検査制約をアップグレードします。参照整合性制約はコンパイル済みオブジェクトではないので、アップグレードの必要はありません。
- **default** – すべての宣言デフォルトと、**create default** コマンドで作成されたデフォルトをアップグレードします。
- **procedure** – すべてのストアド・プロシージャをアップグレードします。
- **rule** – すべてのルールと検査制約をアップグレードします。
- **trigger** – すべてのトリガをアップグレードします。
- **view** – すべてのビューをアップグレードします。

キーワード **check**、**default**、**procedure**、**rule**、**trigger**、**view** は、アップグレードされるコンパイル済みオブジェクトのクラスを指定します。クラスを指定すると、**dbcc upgrade_object** がエラーや問題が発生する可能性のある領域を検出しないかぎり、指定したデータベース内のそのクラスのすべてのオブジェクトがアップグレードされます。

- **force** – 指定したオブジェクトが **select * 句** を含んでいてもアップグレードしたい場合に指定します。**force** は、**select * 文** が予期しない結果を返すことがないという確信がないかぎり使用しないでください。**force** オプションは、予約語を含んでいるオブジェクト、トランケートまたは削除されたソース・テキストを含んでいるオブジェクト、存在しないテンポラリ・テーブルを参照するオブジェクト、引用符付き識別子の設定と合わないオブジェクトはアップグレードしません。これらのオブジェクトは、修正してからアップグレードする必要があります。

注意： **set quoted identifiers** が **on** の場合は、キーワードを一重引用符で囲んでください。**set quoted identifiers** が **off** の場合は、二重引用符または一重引用符のどちらでも使用できます。

パーミッション

ここでは、パーミッションについて説明します。

dbcc upgrade_object を実行できるのは、データベース所有者またはシステム管理者だけです。データベース所有者は、データベース内の自分のオブジェクトをアップグレードできます。

アップグレードの前後で、オブジェクトの所有者は変更されません。

ログ・セグメントのサイズを大きくする

特定のクラスのすべてのコンパイル済みオブジェクトを **dbcc upgrade_object** の1回の実行でアップグレードするように指定できます。たとえば、**trigger** キーワードを使用することによってすべてのトリガをアップグレードできます。

ただし、**dbcc** コマンドを1つだけ使用する場合でも、各オブジェクトのアップグレードは別のトランザクションに記録されます。古いローは *sysprocedures* から削除されて新しいローが書き込まれます。したがって、**dbcc upgrade_object** を数多くのコンパイル済みオブジェクトに対して実行すると、システムのログ領域を使い果たすことがあります。

1. このコマンドを実行しようとしているデータベース内のログ・セグメントのサイズを増やして、すべてのアップグレードのログを取るための十分な領域を確保してください。

エラーのレポート

ここでは、エラー報告機能について説明します。

システム管理者は、**dbcc traceon(3604)** を実行することで、**dbcc upgrade_object** からのすべての出力をウィンドウに送信できます。このコマンドは、エラー・メッセージの出力がエラー・ログをオーバーフローする可能性がある場合に使用することをおすすめします。

例

ここでは、**dbcc upgrade_object** の使用例を示します。

1. **例 1** – アクティブ・データベース内のすべてのコンパイル済みオブジェクトをアップグレードします。

```
dbcc upgrade_object
```

2. **例 2** – *listdb* データベース内のすべてのストアド・プロシージャをアップグレードします。 **procedure** が一重引用符で囲まれているのは、**set quoted identifiers** が **on** だからです。

```
dbcc upgrade_object(listdb, 'procedure')
```

3. **例 3** – *listdb* データベース内のすべてのルールと検査制約をアップグレードする。 **rule** が二重引用符で囲まれているのは、**set quoted identifiers** が **off** だからです。

```
dbcc upgrade_object(listdb, "rule")
```

4. **例 4** – *listdb* データベース内の **listdb** というすべてのストアド・プロシージャをアップグレードします。

```
dbcc upgrade_object(listdb, list_proc)
```

5. **Example 5** – ログイン "jkarrik" が所有しているストアド・プロシージャ **list_proc** をアップグレードします。

```
dbcc upgrade_object(listdb,  
"listdb.jkarrik.list_proc")
```

6. **例 6** – *dbname* の値が *master*、*database* の値が *listdb* であるため、エラーを返します。これらの値は一致している必要があります。

```
dbcc upgrade_object(master,  
"listdb.jkarrik.list_proc")
```

アップグレードにおけるデータベース・ダンプの使用

12.5 データベース・ダンプとトランザクション・ログをロードして、データベースをアップグレードできます。ただし、以下の点に注意してください。

1. アップグレードには、データをコピーするディスク領域と、アップグレード・プロセス中にシステム・テーブルへの変更のログを取るディスク領域が必要です。ダンプ内のソース・データベースが満杯に近くなると、領域が不足してアップグレード・プロセスが失敗する可能性があります。この問題の発生頻度

は低いはずですが、**alter database** を使用して領域不足エラーが発生しないように空き領域を増やすことができます。

- 古いダンプを再ロードしたら、新しいインストール環境からロードしたデータベース上で **sp_checkreswords** を実行し、予約語をチェックしてください。

データベース・ダンプ内のコンパイル済みオブジェクトのアップグレード

現在の Adaptive Server より前のバージョンで作成されたデータベース・ダンプをロードする場合は、ダンプをロードする前にアップグレード前の作業を実行する必要はありません。したがって、データベース・ダンプ内のコンパイル済みオブジェクトのソース・テキストが削除されていても何も通知されません。

sp_checksource の使用方法については、『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。

- データベース・ダンプのロードが終了したら、**sp_checksourc**e を実行してデータベース内のすべてのコンパイル済みオブジェクトについてソース・テキストが存在するか確認してください。
- 確認ができれば、コンパイル済みオブジェクトが実行されるときにアップグレードできます。または、発生する可能性のある問題を見つけるために **dbcc upgrade_object** を実行して、オブジェクトを手動でアップグレードできます。

手動によるクラスタの設定と管理

クラスタを手動で設定する場合は、Unified Agent を作成するまで、クラスタの管理に Adaptive Server プラグインも **sybcluster** も使用できません。「手動設定後の **sybcluster** とプラグインの有効化」の項を参照してください。

作業を始める前に

ここでは、クラスタを手動で設定する前に行う必要のある操作について説明します。

クラスタを設定する前に、次の操作を完了しておく必要があります。

- Sybase 環境変数の設定
- ロー・デバイスの作成

環境の設定

Sybase 環境を設定します。Sybase リリース・ディレクトリから、SYBASE.sh ファイルまたは SYBASE.csh ファイルを読み込みます。

環境変数スクリプトが保存されている \$SYBASE ディレクトリから、環境変数を読み込みます。次に例を示します。

```
. SYBASE.sh
```

または

```
source SYBASE.csh
```

ロー・デバイスの作成

各ロー・デバイスには、各ノードから同じパスを使用してアクセスできなければなりません。ロー・デバイスの設定の参照については、使用中のオペレーティング・システムのストレージ管理者に照会するか、マニュアルを参照してください。

ローカル・システム一時データベース・デバイスと、クォーラム・ディスク・デバイスは、Cluster Edition から加わった機能です。他の必須デバイスは、すべての Adaptive Server に要求されるものと同じです。

- マスタ・データベース・デバイス
- **sybstemprocs** データベース・デバイス
- システム・データベース・デバイス

- ローカル・システム・テンポラリ・データベース・デバイス (複数のデバイスを作成し、デバイスごとにローカル・システム・テンポラリ・データベースを1つ設定できます)。

注意： ローカル・システム・テンポラリ・データベースは、共有ディスクを使用する必要があります。

- クォーラム・ディスク・デバイス (20MB 以上)
- その他のすべてのデータベース・デバイス

手動によるクラスタの設定

Sybase では、すべてのインスタンスのエンジンの合計数が、CPU の数を超えないようにすることを強くおすすめしています。

- クラスタ入力ファイルを作成します。
- interfaces ファイルを設定します。
- マスタ・デバイスとクォーラム・デバイスを構築します。
- 各インスタンスに runserver ファイルを作成します。
- sybssystemprocs のデバイスを初期化し、sybssystemprocs データベースを作成します。
- システム・ストアド・プロシージャをインストールします。
- 各インスタンスに、ローカル・システム・テンポラリ・データベースを設定します。
- クラスタを起動します。

クラスタ入力ファイルの作成

クラスタを設定する前に、クラスタの名前、クラスタ内のインスタンスの数、interfaces ファイルを含むディレクトリへのパス、ログ・ファイル、およびクォーラム・ディスク・デバイス、その他の必須設定情報を指定するクラスタ入力ファイルを作成します。クラスタ入力ファイルに任意の名前を選択します (例: mycluster.inp)。

クラスタを設定すると、Adaptive Server はクラスタ入力ファイルから情報を読み込み、クォーラム・デバイスに安全に保存します。Adaptive Server は、以降、クォーラム・デバイスからクラスタ設定情報を取得します。

クラスタが初期化された後の設定情報の変更の詳細については、「クラスタの再設定」の項を参照してください。

注意： 各クラスタ入力ファイルを使って、クラスタを1つ設定できます。

クラスタ入力ファイルは、**sp_configure**に関連した Adaptive Server 設定値を保存するサーバ設定ファイルとは異なります。

以下に、クラスタ入力ファイルの構文を示します。

```
# all input files must begin with a comment
[cluster]
name = cluster name
max instances = number
master device = path to the master device
configuration file = common path to all server configuration files
primary protocol = udp | tcp | other
secondary protocol = udp | tcp | other
installation mode = shared | private
configuration file = Adaptive Server configuration file name
interfaces path = interfaces file path
traceflags = trace flag number, trace flag number, . . .
additional run parameters = any additional run parameters

[management nodes]
hostname = node_name
hostname = node_name
hostname = node_name
hostname = node_name

[instance]
id = instance ID
name = instance name
node = name of node on which this instance runs
primary address = primary interconnect address
primary port start = port number
secondary address = secondary interconnect address
secondary port start = port number
errorlog = file name
interfaces path = interfaces file path
config file = path to server configuration file for this instance
traceflags = trace flag number, trace flag number, . . .
additional run parameters = any additional run parameters

[instance]
id = instance ID
name = instance name
node = name of node on which this instance runs
primary address = primary interconnect address
primary port start = port number
secondary address = secondary interconnect address
secondary port start = port number
errorlog = file name
interfaces path = interfaces file path
configuration file = path to server configuration file for this
instance
traceflags = trace flag number, trace flag number, . . .
additional run parameters = any additional run parameters
```

各パラメータの意味は、次のとおりです。

- **name** - クラスタの名前。
- **max instances** - 使用中のバージョンの Adaptive Server でサポートされている、クラスタ内のインスタンスの最大数。使用中のバージョンの Adaptive Server でサポートされているインスタンスの最大数については、『リリース・ノート』を参照してください。
- **master device** - マスタ・デバイスへのパス。
- **primary protocol** - プライマリ相互接続に使用されるプロトコル。
- **secondary protocol** - セカンダリ相互接続に使用されるプロトコル。
- **installation mode** - 共有またはプライベート・インストール・モード
- **config file** - Adaptive Server 設定ファイルへのパス。このファイルは、個々のインスタンスによりこの設定が上書きされるまで、クラスタ内のすべてのインスタンスで使用されます。
すべての設定ファイルで同じパス名が共有されるプライベート・インストールでは、これが共通パスになります。
- **interfaces path** - interfaces ファイルへのパス。LDAP 環境を使用している場合は、このパラメータを省略します。この interfaces ファイルは、個々のインスタンスによりこの設定が上書きされるまで、すべてのインスタンスで使用されます。
- **traceflags** - インスタンスの起動時に使用されるトレースフラグのコンマ区切りリスト。
- **additional run parameters** - 起動時にインスタンスに渡される追加パラメータ。
- **hostname** - ノードの名前。この名前は、このノードで実行される hostname コマンドで返されるものと同じ名前になる必要があります。ノードごとに、ホスト名フィールドが1つ登録されている必要があります。そのノードは、管理ノード・セクションで1回だけ指定する必要があります。
- **name** - インスタンスの名前。
- **ID** - インスタンスの ID。
- **node** - このインスタンスが実行されるノードの名前。
- **primary address** - プライマリ相互接続のこのインスタンスのアドレス。
- **primary port start** - プライマリ相互接続の開始ポート番号。
- **secondary address** - セカンダリ相互接続のこのインスタンスのアドレス。相互接続セクションにセカンダリが定義されている場合、またはセカンダリ・プロトコルが指定されている場合に必要です。セカンダリが定義されていない場合は無視されます。
- **secondary port start** - セカンダリ相互接続の開始ポート番号。セカンダリ・アドレスまたはセカンダリ・プロトコルが指定されている場合に必要です。
- **error log** - このインスタンスのエラー・ログへのフル・パス。

- **config file** - Adaptive Server 設定ファイルへのパス。このファイルにより、クラスタ入力ファイルの **cluster** セクションに指定される設定ファイル・フィールドが上書きされます。
個々のサーバ設定ファイルへのパス名が同一でないプライベート・インストールの場合、これは現在のサーバ設定ファイルへのパスです。
- **interfaces path** - サーバ側 interfaces ファイルへのパス。このファイルにより、クラスタ入力ファイルの **cluster** セクションに指定される interfaces ファイル・フィールドが上書きされます。パスに interfaces ファイルの名前を含まないでください。LDAP を使用している場合は、このパラメータを省略します。
- **traceflags** - インスタンスの起動時に使用されるトレースフラグのコンマ区切りリスト。これらのパラメータは、クラスタ入力ファイルの **cluster** セクションで指定されるトレース・フラグの代わりに使用されるのではなく、併用されます。
- **additional run parameters** - 起動時にインスタンスに渡される追加パラメータ。
ソケット・ポート範囲を検出する式は、次のようになります。
 $start_port_number + (max_instances * 5) - 1$

注意： ADO.NET を使用しない場合、選択したポート番号が他のプロセスで使用されていないか確認します。

この例では、クラスタ入力ファイルで、ノード "blade1" に "ase1"、ノード "blade2" に "ase2" の 2 つのインスタンスを持つ、"mycluster" という名前のクラスタが定義されています。プライベート相互接続のアドレスは、192.169.0.1 と 192.169.0.2 です。サーバ設定ファイルの名前は、mycluster.cfg です。最大インスタンスは 2 です。"ase1" は 15015 で始まるポート範囲、"ase2" は 16015 で始まるポート範囲を使用します。次のコマンドによって mycluster クラスタに情報が追加されます。

```
#input for a 2 node / 2 instance cluster
[cluster]
name = mycluster
max instances = 2
master device = /opt/sybase/rawdevices/mycluster.master
config file = /opt/sybase/ASE-15_0/mycluster.config
interfaces path = /opt/sybase
primary protocol = udp
secondary protocol = udp

[management nodes]
hostname = blade1.sybase.com
hostname = blade2.sybase.com

[instance]
id = 1
name = ase1
```

```
node = blade1.sybase.com
primary address = 192.169.0.1
primary port start = 15015
secondary address = 192.169.1.1
secondary port start = 15015
errorlog = /opt/sybase/ASE-15_0/install/ase1.log
additional run parameter = -M/opt/sybase/ASE-15_0

[instance]
id = 2
name = ase2
node = blade2.sybase.com
primary address = 192.169.0.2
primary port start = 16015
secondary address = 192.169.1.2
secondary port start = 16015
errorlog = /opt/sybase/ASE-15_0/install/ase2.log
additional run parameter = -M/opt/sybase/ASE-15_0
```

すべてのインスタンスが単一のノード上に存在するクラスタ入力ファイルの例については、『Cluster ユーザーズ・ガイド』を参照してください。

interfaces ファイルの設定

interfaces ファイルを使用する場合は、クラスタ・サーバとすべてのインスタンスのエントリがファイルに含まれている必要があります。

interfaces ファイルの構文は次のようになります。

```
        instance_name
master network_protocol
        machine_name
        port_number
query  network_protocol
        machine_name
        port_number
. . .
cluster_server_name
query  network_protocol ether machine_name
        port_number
```

各パラメータの意味は、次のとおりです。

- **instance_name** – この interfaces ファイルのエントリを行うインスタンス
- **network_protocol** – インスタンスに使用されるネットワーク・プロトコル
- **machine_name** – インスタンスが実行されるマシンの名前
- **port_number** – このインスタンスへの接続に使用されるポート番号
- **cluster_server_name** – クラスタ・サーバの名前

この例では、クラスタ "mycluster" で稼働中のマシン "blade1"、"blade2"、"blade3" のインスタンス "ase1"、"ase2"、"ase3" が示されています。

```

ase1
  master tcp ether blade1 19786
  query tcp ether blade1 19786
ase2
  master tcp ether blade2 19786
  query tcp ether blade2 19786
ase3
  master tcp ether blade3      19786
  query tcp ether blade3 19786
mycluster
  query tcp ether blade1 19786
  query tcp ether blade2 19786
  query tcp ether blade3 19786

```

マスタ・デバイスとクォーラム・デバイスの構築

共有ディスク領域のロー・デバイスに、クォーラム・デバイスとマスタ・デバイスを作成します。

クォーラム・デバイスを作成する場合、すべてのマシンで同じデバイス名および同じ最大数と最小数を使用します。各ロー・デバイスには、各ノードから同じパスを使用してアクセスできなければなりません。「クラスタ入力ファイルの作成」のトピックで説明する例では、ロー・デバイス `raw11` がクォーラム・ディスクとして指定されます。次の例でも、`raw11` がクォーラム・ディスクとして使用されます。

次に例を示します。

```

dataserver
...

--quorum-dev /dev/raw/raw11

```

Cluster Edition マスタ・デバイスとクォーラム・デバイスを作成するための構文は、次のようになります。

```

dataserver
--cluster-input= cluster_input_filename_and_path
--quorum-dev= quorum_device_and_path
--master-device-size= master_device_size
--logical-page-size= page_size
--instance= instance_name
--buildquorum

```

各パラメータの意味は、次のとおりです。

- **--master-device-size=<size spec>** - マスタ・デバイスのサイズを指定します。
- **--cluster-input=<cluster input file>** - 入力ファイルで指定されたクラスタ設定をクォーラム・デバイスにロードします。
- **--quorum-dev= path_to_quorum_device** - クォーラム・デバイスへのフル・パスを指定します。
- **--instance=instance_name** - インスタンスの名前を指定します。

- `--logical-page-size= page_size` – ページ・サイズを指定します。
- `--buildquorum` – 新しいクォーラム・デバイスの構築を指定します。

注意： `dataserver` パラメータの詳細については、『ユーザーズ・ガイド』の「システムの変更点」の章にある「コマンドの変更」の項の「`dataserver`」を参照してください。

`dataserver` を使用して、マスタ・デバイスとクォーラム・デバイスを作成します。この例では、4K のページが設定された "ase1" という名前のインスタンス、500MB のマスタ・デバイス、およびクォーラム・デバイスが作成されます。

```
/opt/sybase/ASE-15_0/bin/ dataserver¥
--quorum-dev=/dev/raw/raw11¥
--instance=ase1
--cluster-input=/opt/sybase/mycluster.inp
--master-device-size=500M
--logical-page-size=4k
--buildquorum
```

sybsystemprocs の初期化と作成

ここでは、`sybsystemprocs` の初期化と作成の方法について説明します。

`disk init` を使用して `sybsystemprocs` 用にデバイスを初期化してから、`sybsystemprocs` データベースを作成します。

1. `sybsystemprocs` に対して、150MB 以上のロー・デバイスを初期化します。

```
disk init name = "sysprocsdev",
physname = "/dev/raw/raw13",

size = "150MB"
```

2. `sybsystemprocs` データベースを作成します。

```
create database sybsystemprocs on sysprocsdev = 150
```

システム・ストアド・プロシージャのインストール

ここでは、システム・ストアド・プロシージャをインストールする方法について説明します。

`$$SYBASE/$$SYBASE_ASE/scripts` ディレクトリから `InstallAnywhere` を実行して、システム・ストアド・プロシージャをインストールします。 `installmaster` は、任意のインスタンスから実行できます。

```
isql -U sa -P sa_password -S server_name -n
-i $$SYBASE/$$SYBASE_ASE/scripts/installmaster
-o output_file
```


runserver ファイルの作成

マスタ・デバイスとクォーラム・デバイスを作成した後、クラスタ内の各インスタンスに runserver ファイルを作成します。後でこれらのファイルを使用して、インスタンスを起動します。

各インスタンスに runserver ファイルを作成します。

1. runserver ファイルを作成します。

この例では、インスタンス ase1 に RUN_ase1 が作成されます。runserver ファイルを 1 行に入力します。「¥」は、スペースを空けずに行が続くことを表します。

```

$SYBASE/ASE-15_0/bin/dataserver
--quorum-dev=/dev/raw/raw11
--instance=ase1

```

注意：すべてのデバイス、データベース・デバイス、およびクォーラム・デバイスを、共有ディスク上にロー・デバイスとして作成します。

2. クラスタ内の各 Adaptive Server に、runserver ファイルのコピーを作成します。たとえば、クラスタ "mycluster" の 3 つのインスタンスに、RUN_ase1、RUN_ase2、RUN_ase3 という名前の runserver ファイルが作成されているとします。すべてのインスタンスに、必ず同じクォーラム・デバイスを含めてください。各ファイル内の **--instance** は適切なインスタンス名を示すように変更します。

ローカル・システム・テンポラリ・データベースの設定

共有ディスク・クラスタで、各インスタンスにローカル・システム・テンポラリ・データベースを設定する必要があります。

Cluster Edition では、インスタンスにローカル・システム・テンポラリ・データベースが設定されていない場合、それがクラスタ内で最初にブートするインスタンスである場合にのみブートします。

1. Adaptive Server を起動します。次に例を示します。

```
startserver -f $SYBASE/$SYBASE_ASE/install/RUN_ase1_coord
```

2. Adaptive Server にログインします。
3. テンポラリ・データベースにマスタ・デバイスを使用しない場合、ローカル・システム・テンポラリ・データベース用のデバイスを作成します。ローカル・システム・テンポラリ・データベースは共有ディスクにのみ作成できます。テンポラリ・データに対してローカル・ディスクを使用する方法の詳細については、『Cluster Edition ユーザ・ガイド』の「第 6 章テンポラリ・データベースの使用」を参照してください。

Sybase では、これらのデータベースのログとデータに別々のデバイスを使用することをおすすめします。次に例を示します。

```
disk init name="tempdbdev1",  
physname="/dev/raw/raw14",  
size="400M"
```

ログ・デバイスの場合：

```
disk init name="temp_log_dev1",  
physname="/dev/raw/raw15",  
size="200M"
```

4. クラスタ内の各インスタンスに、ローカル・システム・テンポラリ・データベースを作成します。たとえば、インスタンス "ase1"、"ase2"、"ase3" に、3つのローカル・システム・テンポラリ・データベース "ase1_tdb1"、"ase2_tdb1"、"ase3_tdb1" をそれぞれ作成するには、次のように入力します。

```
create system temporary database ase1_tdb1 for instance ase1 on  
tempdbdev1 = 100  
log on temp_log_dev1 = 50  
create system temporary database ase2_tdb1 for instance ase2 on  
tempdbdev1 = 100  
log on temp_log_dev2 = 50  
create system temporary database ase3_tdb1 for instance ase3 on  
tempdbdev1 = 100  
log on temp_log_dev3 = 50
```

5. **shutdown cluster** コマンドを使用してクラスタを停止します。

クラスタへの自動クラスタ継承の設定

クラスタ全体の突然の障害から自動的にリカバリするようにクラスタを設定します。

automatic cluster takeover は、インスタンスがクラスタへの接続を試み、次の条件が満たされる場合にトリガされます。

- **automatic cluster takeover** が有効である。
- クォーラム・デバイスはクラスタが実行中であることを示しているが、インスタンスはハートビートを検出していない。

automatic cluster takeover 設定パラメータによって、クラスタへの接続を試みるインスタンスは、新しいクラスタを形成し、クラスタ・コーディネータを起動し、データベースをリカバリできます。**automatic cluster takeover** の構文は次のとおりです。

```
sp_configure "automatic cluster takeover", [1 | 0]
```

automatic cluster takeover を 1 に設定すると、自動クラスタ継承が有効になります (デフォルト)。0 に設定すると、**automatic cluster takeover** が無効になります。

クラスタの手動による再起動については、「クラスタの起動」のトピックを参照してください。

I/O フェンシングが有効になっている環境では、**automatic cluster takeover** が安全な動作であることが保証されます。I/O フェンシングが有効になっていない環境では、アルゴリズムの不具合によりデータ破壊が生じる場合があります。したがって、この設定パラメータは、アルゴリズムで不具合が発生した場合にアルゴリズムを無効にするために存在します。I/O フェンシング機能のない環境では常にデータ破壊のリスクが存在し、自動クラスタ継承を無効にしてもそのリスクが完全に解消されるわけではありません。

クラスタの起動

クラスタ内の各ノードのクラスタに関連したすべてのインスタンスを起動して、クラスタを起動します。

クラスタは次の場合に起動できます。

- 通常は、適切な停止の後。
- システム障害の後。「システム障害後のクラスタの起動」のトピックを参照してください。

注意： Sybase では、通常のクラスタの起動に使用される `runserver` ファイルを変更しないことをおすすめします。

1. 各クラスタ・インスタンスを、稼働中のノードから起動します。

たとえば、"ase1" インスタンスを起動する場合は、次のように指定します。

```
startserver -f $SYBASE/$SYBASE_ASE/install/RUN_ase1
```

2. 別のインスタンスを稼働させる予定の各ノードにログインし、**startserver** を実行します。

たとえば、"blade2" で "ase2" インスタンスを起動する場合は、次のように指定します。

1. "blade2" で \$SYBASE ディレクトリに移動します。

2. **startserver** コマンドを発行します。

```
startserver -f $SYBASE/$SYBASE_ASE/install/RUN_ase2
```

システム障害後のクラスタの起動

ここでは、システムで障害が発生した後でサーバを起動する方法について説明します。

システム障害によりクラスタが停止し、**automatic cluster takeover** 設定パラメータを有効にしている場合は、手順1を実行します。automatic cluster takeover 設定パラメータを有効にしていない場合は、手順2を実行します。

1. **automatic cluster takeover** パラメータをデフォルトの1に設定します。クラスタへの接続を試みるインスタンスは、クラスタを再起動し、クラスタ・コーディネータとして機能し、データベースをリカバリします。自動的に再起動するようにクラスタを設定する方法については、「クラスタへの自動クラスタ継承の設定」のトピックを参照してください。
2. **automatic cluster takeover** を有効にしていない場合は、**dataserver . . . --cluster-takeover** パラメータを使用してクラスタを再起動する必要があります。たとえば、この例ではクラスタの起動に使用されていた **runserver** ファイルに **-cluster-takeover** パラメータを追加しています。

```
$SYBASE/ASE-15_0/bin/dataserver ¥  
--quorum-dev=/dev/raw/raw11¥  
--instance=ase1¥  
--cluster-takeover
```

3. クラスタが起動した後、通常どおりにすべてのインスタンスを再起動します。

設定後の作業

ここでは、手動でサーバを設定した後に行う必要のある作業について説明します。設定したインスタンスが稼働しているかどうかを判断するために、クラスタにログインし、次のコマンドを入力します。

```
sp_cluster show
```

その後で、次の手順に従ってインスタンスに接続できることを確認します。

1. **SYBASE.sh** を読み込みます。
2. **isql** を使用してサーバに接続します。コマンド・プロンプトで次のように入力します。

```
isql -Usa -P -Sserver_name
```

server_name はインスタンス名です。ログインに成功すると、コマンド・プロンプトが表示されます。

3. Adaptive Server のバージョン番号を表示するには、次のように入力します。

```
1> select @@version
2> go
```

エラーが発生する場合は、『トラブルシューティング&エラー・メッセージ・ガイド』を参照してください。

クラスタまたはインスタンスの停止

クラスタを停止すると、クラスタに関連するすべてのインスタンスが停止します。

クラスタを停止するには、次の手順に従います。

1. インスタンスにログインします。次に例を示します。

```
isql -Usa -P -Sase2 -I$SYBASE/interfaces
```

2. **shutdown** コマンドを発行します。

```
shutdown cluster
```

インスタンスの停止

shutdown コマンドを使用して、クラスタ内の個々のインスタンスを停止します。クラスタ内の別のインスタンスから、インスタンスを停止することもできます。

1. インスタンスにログインします。次に例を示します。

```
isql -Usa -P -Smycluster -I$SYBASE/interfaces
```

2. **shutdown** コマンドを発行します。

```
shutdown ase2
```

クラスタの再設定

クラスタの再設定により、インスタンスの追加、トレース・フラグの変更などを実行できます。クラスタ入力ファイルを編集し、次に、最初に起動する予定のインスタンスに、**dataserver** と **cluster-input** オプションを含む新しい実行ファイルを作成します。このオプションにより、新しい設定情報のクォーラム・デバイスへの書き込みが Adaptive Server に指示されます。

インスタンスにログインし、クラスタを停止します。

注意： **sybcluster** または Adaptive Server プラグインを使用する場合、クラスタを再設定する前に停止する必要はありません。

1. **qrmutil** ユーティリティに **-extract-config** コマンドを使用して、現在の設定をファイルに抽出します。詳細については、『Cluster ユーザーズ・ガイド』を参照してください。
2. クラスタ入力ファイルを編集します。
3. ブート予定のインスタンスの実行ファイルをコピーし、**--cluster-input** オプションを **dataserver** 文に追加します。たとえば、次の方法で **RUN_ase1** を変更します。

```
$SYBASE/$SYBASE_ASE/bin/dataserver¥  
--cluster-input=/--quorum-dev=/dev/raw/raw1¥  
--instance=ase1¥
```

4. クラスタを起動します。

手動設定後の **sybcluster** とプラグインの有効化

手動設定後に **sybcluster** または Adaptive Server プラグインを使用してクラスタを管理するには、クラスタで使用される各ノード上で Unified Agent を起動し、その後で各ノードに接続エージェントを展開します。

1. クラスタで使用される各ノードで、Unified Agent を起動します。\$SYBASE ディレクトリから、次のように入力します。

```
UAF-2_5/bin/uafstartup.sh
```

2. **sybcluster** を起動します。たとえば、次のように入力します。

```
sybcluster -U uafadmin -P -C mycluster  
-F "blade1, blade2,blade2"
```

3. **deploy plugin** を実行します。次に例を示します。

```
deploy plugin agent "blade1,blade2,blade3"
```

sybcluster と ASE プラグインの完全な構文と使用方法の詳細については、『Cluster ユーザーズ・ガイド』の「**sybcluster** ユーティリティ」の章を参照してください。

sybcluster のサンプル・セッションの設定値

ここでは、**sybcluster** を使用して一般的な共有ディスク・クラスタを設定する方法の例を示します。

この例では、プライマリ・ネットワークとセカンダリ・ネットワークがあります。セカンダリ・ネットワークが指定されない場合、**sybcluster** は開始ポート番号を要求し、インスタンス間で通信するのに必要なポート番号の数を計算します。指定する番号と次の必要なポート番号が、他のアプリケーションで使用されていないことを確認します。**sybcluster** を使用した必要なポートの数の計算については、「**sybcluster** を使用したクラスタ・サーバの作成」のトピックを参照してください。

表 15: サンプル・セッション用に入力された sybcluster プロンプトおよび設定値

パラメータ	値
クラスタ名	mycluster
インスタンスの数	
エージェントの数	
クラスタ・ノード ID	1
設定タイプ	
プライベート・インストールの \$SYBASE ディレクトリを使用してクラスタを設定する	N
クォーラム・デバイス	
クォーラム・デバイスのフル・パスと名前	/dev/rhdisk11
ページ・サイズ	
ページ・サイズ (キロバイト単位)	2kB
マスタ・デバイス	
マスタ・デバイスのフル・パスと名前	/dev/rhdisk12
マスタ・デバイスのサイズ [MB]	30MB
マスタ・データベースのサイズ [MB]	13MB
PCI デバイス (オプション)	

sybcluster のサンプル・セッションの設定値

パラメータ	値
PCI データベース・デバイスへのフル・パス	/dev/raw/raw20
PCI データベース・デバイスのサイズ [MB]	24MB
PCI データベースのサイズ [MB]	24MB
Sybase システム・プロシージャ・デバイス	
sybssystemprocs デバイスのフル・パスと名前	/dev/rhdisk13
sybssystemprocs デバイスのサイズ (MB)	160MB
sybssystemprocs データベースのサイズ (MB)	152MB
システム・データベース・デバイス	
システムのデータベース・デバイスのフル・パスと名前	/dev/rhdisk14
システム・データベース・デバイスのサイズ (MB)	6MB
システム・データベースのサイズ (MB)	6MB

このクラスタはセカンダリ・ネットワークを持っているかどうか [Y]	Yes
ファイルのロケーションを確認	
Sybase ホーム・ディレクトリのロケーション	/remote/var/sybase/
環境変数スクリプトのフル・パス	/remote/var/sybase/ SYBASE.sh
\$SYBASE_ASE へのパス	/remote/var/sybase/ ASE-15_0
interfaces ファイル・ディレクトリへのパス	/remote/var/sybase
dataserver 設定ファイルへのパス	/remote/var/sybase/ mycluster.cfg
インスタンス情報	
ノード名	blade1

パラメータ	値
sybcluster による表示: Cluster: mycluster - Node: blade1 - Agent blade1:9999	
インスタンス名	ase1
ase1 のクエリ・ポート番号	19786
ase1 のプライマリ・プロトコル・アドレス	000.000.001.001
ase1 のセカンダリ・プロトコル・アドレス	000.000.001.002
ローカル・システム・テンポラリ・データベース	
ローカル・システム・テンポラリ・データベースのデバイス名(ローカル・システム・テンポラリ・データベースの Adaptive Server データベース・デバイスの名前を入力します)。ローカル・システム・テンポラリ・データベースのデバイスには、共有ディスクを使用してください。	mycluster1_tempdb
ローカル・システム・テンポラリ・データベースのデバイスのパス	/dev/rhdisk15
ローカル・システム・テンポラリ・データベースのデバイスのサイズ(MB)	40MB
ローカル・システム・テンポラリ・データベース名	mycluster_tdb_1
ローカル・システム・テンポラリ・データベースのサイズ(MB)	40MB

別のインスタンスが必要かどうか(YまたはN)	Yes
ノード名	blade 2
sybcluster による表示: Cluster: mycluster - Node: blade2 - Agent blade2:9999	
インスタンス名	ase2
ase2 のクエリ・ポート番号	19786
ase2 のプライマリ・プロトコル・アドレス	000.000.002.001
ase2 のセカンダリ・プロトコル・アドレス	000.000.002.002

sybcluster のサンプル・セッションの設定値

パラメータ	値
ローカル・システム・テンポラリ・データベース	
ローカル・システム・テンポラリ・データベースのデバイス名(ローカル・システム・テンポラリ・データベースの Adaptive Server データベース・デバイスの名前を入力します)。ローカル・システム・テンポラリ・データベースのデバイスには、共有ディスクを使用してください。	mycluster2_tempdb
ローカル・システム・テンポラリ・データベースのデバイスのパス	/dev/rhdisk16
ローカル・システム・テンポラリ・データベースのデバイスのサイズ(MB)	40MB
ローカル・システム・テンポラリ・データベース名	mycluster_tdb_2
ローカル・システム・テンポラリ・データベースのサイズ(MB)	40MB

別のインスタンスが必要かどうか(Y または N)	Yes
ノード名	blade3
sybcluster による表示: Cluster: mycluster - Node: blade3 - Agent blade3:9999	
インスタンス名	ase3
クエリ・ポート番号	19786
ase3 のプライマリ・プロトコル・アドレス	000.000.003.001
ase3 のセカンダリ・プロトコル・アドレス	000.000.003.002
ローカル・システム・テンポラリ・データベース	
ローカル・システム・テンポラリ・データベースのデバイス名(ローカル・システム・テンポラリ・データベースの Adaptive Server データベース・デバイスの名前を入力します)。ローカル・システム・テンポラリ・データベースのデバイスには、共有ディスクを使用してください。	mycluster3_tempdb

パラメータ	値
ローカル・システム・テンポラリ・データベースのデバイスのパス	/dev/rhdisk17
ローカル・システム・テンポラリ・データベースのデバイスのサイズ (MB)	40MB
ローカル・システム・テンポラリ・データベース名	mycluster_tdb_3
ローカル・システム・テンポラリ・データベースのサイズ (MB)	40MB

別のインスタンスが必要かどうか (Y または N)	N
設定情報をファイルに保存する [Y]	Yes
設定ファイルへのフル・パス	/remote/var/sybase/ mycluster.xml
クラスタを今すぐ作成 [Y]	Yes
Veritas Cluster Server サポート (オプション - VCS がサポートされているシステムでしか表示されない)	
Cluster Edition サーバを VCS と統合する [Y]	N
各クラスタ・デバイスに IO フェンシング機能が備わっているかどうか確認 [Y]	

sybcluster のサンプル・セッション

ここでは、**sybcluster** サンプル・セッションを示します。

このサンプル・セッションでは、以下を前提とします。

- 共有ファイル
- サーバ検索用の `interfaces` ファイル
- データベースでの Java サポート

```

sybcluster -U uafadmin -P -F
hpcblade2:9009,hpcblade1:9009,hpcblade3:9009,hpcblade4:9009
> create cluster
Enter the name of the cluster: mycluster
Cluster mycluster - Enter the maximum number of instances: [ 4 ]
How many agents will participate in this cluster: [ 4 ] 4
Verifying the supplied agent specifications...

```

sybcluster のサンプル・セッションの設定値

```
1) hpcblade1.sybase.com 9009 2.5.0 Linux
2) hpcblade2.sybase.com 9009 2.5.0 Linux
3) hpcblade3.sybase.com 9009 2.5.0 Linux
4) hpcblade4.sybase.com 9009 2.5.0 Linux
Enter the number representing the cluster node 1: [ 4 ] 1
2) hpcblade2.sybase.com 9009 2.5.0 Linux
3) hpcblade3.sybase.com 9009 2.5.0 Linux
4) hpcblade4.sybase.com 9009 2.5.0 Linux
Enter the number representing the cluster node 2: [ 4 ] 2
3) hpcblade3.sybase.com 9009 2.5.0 Linux
4) hpcblade4.sybase.com 9009 2.5.0 Linux
Enter the number representing the cluster node 3: [ 4 ] 3
4) hpcblade4.sybase.com 9009 2.5.0 Linux
Enter the number representing the cluster node 4: [ 4 ] 4
Will this cluster be configured using private SYBASE
installations? (Y/N) : [ N ]
----- Quorum Device -----
The quorum device is used to manage a cluster. It contains
information shared between instances and nodes.
Enter the full path to the quorum disk: /hpcblade_cfs/q/
pd16218942/d3.dbs
Enter any traceflags:
----- Page Size -----
Enter the page size in kilobytes: [ 2 ] 8
----- Master Database Device -----
The master database device controls the operation of the Adaptive
Server and stores information about all user databases and their
associated database devices.
Enter the full path to the master device: /hpcblade_cfs/q/
pd16218942/d4.dbs
Enter the size the Master Device (MB): [ 120 ] 500
Enter the size the Master Database (MB): [ 52 ] 100
----- Sybase System Procedure Device -----
Sybase system procedures (sybssystemprocs) are stored on a device.
Enter the System Procedure Device path: /hpcblade_cfs/q/
pd16218942/d5.dbs
Enter System Procedure Device size (MB): [ 152 ] 200
Enter the System Procedure Database size (MB): [ 152 ] 200
----- System Database Device -----
The system database (sybssystemdb) stores information about
distributed transactions.
Enter the System Database Device path: /hpcblade_cfs/q/pd16218942/
d6.dbs
Enter the System Database Device size (MB): [ 24 ] 100
Enter the System Database size (MB): [ 24 ] 100
----- PCI Device -----
Pluggable Component Interface (PCI) provides support for Java in
database by loading off-the-shelf JVMs from any vendor. If you
want to use JVM, create a device for it.
Enable PCI in Adaptive Server (Y/N): [ N ] y
Enter the full path to the PCI device: /hpcblade_cfs/q/pd16218942/
pci.dbs
Enter the size the PCI Device (MB): [ 96 ]
Enter the size the PCI Database (MB): [ 96 ]
-----
Does this cluster have a secondary network: [ Y ] n
```

```

Enter the port number from which this range will be applied:
[ 15100 ] 17005
-----
Enter the SYBASE home directory: [ /remote/quasr5/adong/aries/
release/lamce_s1 ]
Enter the environment shell script path: [ /remote/quasr5/adong/
aries/release/lamce_s1/SYBASE.sh ]
Enter the ASE home directory: [ /remote/quasr5/adong/aries/
release/lamce_s1/ASE-15_0 ]
Enter path to the dataserver configuration file: [ /remote/
quasr5/adong/aries/release/lamce_s1/mycluster.cfg ]
-----
You will now be asked for the instance information on a node by
node basis.
-- Cluster: mycluster - Node: hpcbladel1.sybase.com - Agent:
hpcbladel1.sybase.com:9009 --
Enter the name of the cluster instance: instance1
Enter the interface file query port number for instance instance1:
10665
Enter the primary protocol address for instance1:
[ hpcbladel1.sybase.com ]
----- Local System Temporary Database -----
The Local System Temporary Database Device contains a database for
each instance in the cluster.
Enter the LST device name: LST
Enter the LST device path: /hpcblade_cfs/q/pd16218942/d7.dbs
Enter LST device size (MB): 200
Enter the LST database name: [ mycluster_tdb_1 ]
Enter the LST database size (MB): [ 200 ] 50
Do you want to add another instance to this node? (Y or N): [ N ]
-- Cluster: mycluster - Node: hpcblade2.sybase.com - Agent:
hpcblade2.sybase.com:9009 --
Enter the name of the cluster instance: instance2
Enter the interface file query port number for instance instance2:
15465
Enter the primary protocol address for instance2:
[ hpcblade2.sybase.com ]
----- Local System Temporary Database -----
The Local System Temporary Database Device contains a database for
each instance in the cluster.
Enter the LST device name: [ LST ]
Enter the LST database name: [ mycluster_tdb_2 ]
Enter the LST database size (MB): [ 150 ] 50
Do you want to add another instance to this node? (Y or N): [ N ]
-- Cluster: mycluster - Node: hpcblade3.sybase.com - Agent:
hpcblade3.sybase.com:9009 --
Enter the name of the cluster instance: instance3
Enter the interface file query port number for instance instance3:
16730
Enter the primary protocol address for instance3:
[ hpcblade3.sybase.com ]
----- Local System Temporary Database -----
The Local System Temporary Database Device contains a database for
each instance in the cluster.
Enter the LST device name: [ LST ]
Enter the LST database name: [ mycluster_tdb_3 ]

```

sybcluster のサンプル・セッションの設定値

```
Enter the LST database size (MB): [ 100 ] 50
Do you want to add another instance to this node? (Y or N): [ N ]
-- Cluster: mycluster - Node: hpcblade4.sybase.com - Agent:
hpcblade4.sybase.com:9009 --
Enter the name of the cluster instance: instance4
Enter the interface file query port number for instance instance4:
15220
Enter the primary protocol address for instance4:
[ hpcblade4.sybase.com ]
----- Local System Temporary Database -----
The Local System Temporary Database Device contains a database for
each instance in the cluster.
Enter the LST device name: [ LST ]
Enter the LST database name: [ mycluster_tdb_4 ]
Enter the LST database size (MB): [ 50 ]
Would you like to save this configuration information in a file?
[Y]
Enter the name of the file to save the cluster creation
information: [ /hpcblade_cfs/q/s16218942/mycluster.xml ]
-----
Create the cluster now? [Y]
-----
```

入力ファイルを使用したクラスタの設定

セッションの最後で、**sybcluster** によって現在のセッションの値を外部ファイルに保存するかどうか尋ねられます。このファイルを使用して同じクラスタを再作成したり、ファイル内の値を編集して別のクラスタを作成したりできます。構文は次のとおりです。

```
create cluster cluster_name file xml_input_file
```

この例では、`mycluster.xml` という名前の入力ファイルを使用して "mycluster2" を作成します。

```
create cluster mycluster2 file ./mycluster.xml
```

Linux でのロー・パーティション

データベース・デバイスをロー・ディスク IO 用に作成しロー・バインド・デバイス上にマウントできます。ロー・ディスク IO を使用すると、アドレス空間からディスク上の物理セクタへのダイレクト・メモリ・アクセスが可能になります。その一方で、ユーザ・アドレス空間からカーネル・バッファへの不必要なメモリ・コピー操作を省略できます。

ロー・ディスク IO では、論理 IO と物理 IO が同時であること、およびシステム `write()` 呼び出しが返されたときに書き込みが確実にディスクにフラッシュされることを前提としています。次のガイドラインに従って、ロー・パーティション・デバイスを準備します。

1. Sybase インストール・ソフトウェアが格納されているパーティションで、データベース・デバイスを初期化しない。初期化すると、そのパーティションの既存のファイルがすべて破壊される。
2. Sybase が使用するロー・パーティションは、ファイル・システムやスワップ領域など、オペレーティング・システムのその他の目的で使用するようマウントすることはできない。
3. Sybase 設定ユーティリティまたは `disk init` コマンドを使用してパーティションの一部をデータベース・デバイスとして初期化すると、そのパーティション全体が別の目的で使用できなくなる。このデバイスに指定されたサイズを超えたパーティション上の領域の残り部分は、`disk resize` コマンドを使用して再使用できる。
4. パーティション・マップが格納されているパーティションの使用を防ぐために、シリンダ 0 を使用しない。
5. サーバのリカバリ・システムはバッファされていないシステム IO を必要とするため、データベース・デバイスを文字型デバイスに配置する。
6. デバイスがブロック・デバイスかそれとも文字型デバイスかを判別するには、次のコマンドを実行します。

```
ls -l <device path>
```

ロー・パーティションの選択

ここでは、ロー・パーティションを選択する方法について説明します。

1. 使用可能なロー・パーティションを調べます。
2. ロー・パーティションのサイズを決定します。

3. 使用可能なロー・パーティションのリストから、各デバイスのロー・パーティションを選択します。
4. オペレーティング・システム管理者に、選択したパーティションが使用可能であることを確認します。
5. そのロー・パーティションに対する読み込み権限および書き込み権限が、"sybase" ユーザにあるか確認します。

注意： ロー・パーティションの選択方法の詳細については、オペレーティング・システムのマニュアルを参照してください。

ロー・パーティションの作成

ロー・デバイスを有効にして使用するには、まず特定のシステム管理が必要となります。デバイスの設定に使用できるツールは、配布設定によって異なります。

ロー・デバイスを設定するディスク上のパーティション内に、物理ディスク領域を割り当ててください。物理 IO サブシステムは、SCSI デバイスと EIDE デバイスのどちらでも使用できます。

注意： パーティションの作成には、Linux デフォルトの **fdisk(8)** ユーティリティを使用できます。 **fdisk** コマンドを使用するには、"root" 権限を持っている必要があります。コマンドの詳細については、**fdisk(8)** の man ページを参照してください。

1. **例 1** - 次の例では、システム内の 4 つの SCSI ディスク (*sda*, *sdb*, *sdc*, および *sdd*) 上で、パーティションをロー・デバイスとして設定する方法について説明します。

- a) /dev/sdd: 上で **fdisk** を起動します。

```
# fdisk /dev/sdd
```

システムから次のように返されます。

```
The number of cylinders for this disk is set to 8683
....
Command (m for help):
```

- b) "p" と入力し、現在のパーティション・レイアウトを出力します。出力は次のようになります。

```
Disk /dev/sdd: 64 heads, 32 sectors, 8683 cylinders
Units = cylinders of 2048 * 512 bytes
Device Boot Start End Blocks Id System
/dev/sdd1 1 7499 7678960 83 Linux
/dev/sdd2 7500 8012 525312 82 Linux swap
/dev/sdd4 8013 8683 687104 5 Extended
```

2. **例 2** - この例では、拡張パーティション (*sdd4*) が 8013 ~ 8683 に 687104 個の空きブロックを持っていることを示しています。残りのパーティションは後で割

り当てることができます。この例では、ロー・バインド・ディスク IO 用の追加パーティションを割り当てます。

- a) **n** コマンドを使用して新しいパーティションを作成し、このプロンプトで "logical" に対して "l" と入力します。

```
Command (m for help):n
Command action
l logical (5 or over)
p primary partition (1-4)
```

- b) 次のように表示されたら、[Enter] キーを押してデフォルトを確定します。

```
First cylinder (8013-8683, default 8013):
```

- c) 次のように表示されたら、[Enter] キーを再度押してデフォルトを確定します。

```
Last cylinder or +size or +sizeM or +sizeK
(8013-8683, default 8683): 8269
```

- d) **t** コマンドを使用し、次のプロンプトで "5" と入力します。

```
Partition number (1-8):5
```

- e) 次のプロンプトで、"60" と入力します。

```
Hex code (type L to list codes): 60
```

出力は次のようになります。

```
Changed system type of partition 5 to 60 (Unknown)
```

- f) 上記の手順を繰り返して、ロー・デバイス IO 用のパーティションを 4 つ作成します。
- g) **p** を使用して設定を確認し、それを書き出す前に完全なパーティション・テーブルを出力します。重複するパーティションがないこと、割り当てられていないパーティションのタイプが Unknown type 60 であることを確認してください。

これで、パーティション・テーブルをディスクに書き込むことができようになり、`fdisk(8)`ユーティリティを終了できます。

Red Hat のロー・デバイス管理

Red Hat Enterprise Linux は管理ツールを完備しており、ロー・デバイスを設定しシステム起動時に管理できます。Red Hat Enterprise Linux システムを設定する場合、パーティションは `/etc/sysconfig/rawdevices` ファイルで管理されます。

これはプレーン・テキスト・ファイルで、次のようなコメントと設定例が含まれます。

```
# raw device bindings
# format: rawdev
           major
           minor
#         rawdev
```

```

                blockdev
# example: /dev/raw/raw1 /dev/sda1
#          /dev/raw/raw2 8 5
/dev/raw/raw1 /dev/sdd1
/dev/raw/raw2 /dev/sdd2
/dev/raw/raw3 /dev/sdd3
/dev/raw/raw4 /dev/sdd4

```

1. ロー・デバイスが作成されたら、それをバインドする必要があります。バインドは、`etc/rc.d/init.d/rawdevices` からロー・デバイスを起動して行うことができます。

```

[root@legolas init.d]# cd /etc/rc.d/init.d
[root@legolas init.d]# sh rawdevices start
Assigning devices:
/dev/raw/raw1 --> /dev/sdd5
/dev/raw/raw1: bound to major 3, minor 5
/dev/raw/raw2 --> /dev/sdd6
/dev/raw/raw2: bound to major 3, minor 6
/dev/raw/raw3 --> /dev/sdd7
/dev/raw/raw3: bound to major 3, minor 7
/dev/raw/raw4 --> /dev/sdd8
/dev/raw/raw4: bound to major 3, minor 8
done

```

2. ロー・デバイス・バインドが再起動時に発生することを保証するには、`chkconfig(8)`ユーティリティを使用します。

```
# /sbin/chkconfig rawdevices on
```

SuSE のロー・デバイス管理

`/etc/raw` ファイルでロー・ディスク・パーティションを管理します。これはプレーン・テキスト・ファイルで、次のようなコメントと設定例が含まれます。

```

# /etc/raw
#
# sample configuration to bind raw devices
# to block devices
#
# The format of this file is:
# raw<N>:<blockdev>
#
# example:
# -----
# raw1:hdb1
#
# this means: bind /dev/raw/raw1 to /dev/hdb1
#
# ...
raw1:sda7
raw2:sda8
raw3:sda9

```

作成したら、ロー・デバイスをバインドします。バインドは、スクリプト `/etc/init.d/raw` でロー・デバイスを起動して行うことができます。

```
# cd /etc/init.d
# sh raw start
bind /dev/raw/raw1 to /dev/sdb1... done
bind /dev/raw/raw2 to /dev/sdb2... done
bind /dev/raw/raw3 to /dev/sdb3... done
...
```

chkconfig(8) ユーティリティを使用して、ロー・デバイス・バインドが再起動時に発生することを保証します。

```
# /sbin/chkconfig raw on
```

サーバからのロー・デバイスへのアクセス

パーティションが作成されてデバイスがロー・ディスク IO にバインドされると、Adaptive Server でそれらを使用できます。Adaptive Server がユーザ "sybase" として動作している場合、`/dev/raw/raw#` デバイス・エントリとロー・バインド制御デバイス `/dev/rawctl` への読み込み、書き込み、および所有者の各パーミッションをユーザ "sybase" に適用します。

適切なパーミッションを適用するには、`chown(1)`、`chgrp(1)`、および `chmod(1)` の各コマンドを参照してください。

1. 設定を確認するには、**raw** コマンドを使用して次のように入力し、デバイス・バインドに対するクエリを実行します。

```
# raw -qa
```

出力は次のようになります。

```
/dev/raw/raw1: bound to major 3, minor 5
/dev/raw/raw2: bound to major 3, minor 6
/dev/raw/raw3: bound to major 3, minor 7
/dev/raw/raw4: bound to major 3, minor 8
```

2. このコマンドを実行するには、root 権限が必要です。root 権限を持っていない場合は、`raw -qa` コマンドの出力が次のように表示されます。

```
Cannot open master raw device '/dev/rawctl'
(Permission denied)
```

3. ロー・デバイスを使用すると、Adaptive Server とインストール/設定ユーティリティ **srvbuild** はそれらのロー・デバイスを自動的に検出してサイズを表示します。`master`、`sybssystemprocs`、`sybtempdb`、およびその他のいかなるデバイスを作成する場合も、ロー・デバイスの絶対パスを入力します。

複製データベースを含むサーバのアップグレード

アップグレードが完了すると、アップグレード前に存在したトランザクション・ログのどの部分もスキャンできなくなります。このため、使用しているサーバに (レプリケート RSSD を含む) 複製プライマリ・データベースがある場合は、この項で説明する手順に従ってください。この手順は、複製データベースからレプリケート・データベースにデータが正常に複製されたことを確認するのに役立ちます。

警告！ 複製データベースをアップグレードするための手順を実行する前に、有効な **dump database** と **dump transaction** を保持しておく必要があります。

ここで説明する手順では、Replication Server 自体はアップグレードされません。Replication Server のアップグレードについては、Replication Server のマニュアルを参照してください。

これらの手順は、データベースのアップグレード手順の前に実行します。各手順は、この項を構成する以下のトピックを表しています。

- トランザクション処理と複製アクティビティのサスペンド
- プライマリ・データベースのトランザクション・ログの排出
- RSSD トランザクション・ログの排出
- セカンダリ・トランケーション・ポイントの無効化

既存のサーバに複製データベースが含まれているかどうかを調べる

アップグレードする前に、既存のサーバに複製データベースが含まれているかどうかを調べます。

1. **isql** を使用して、アップグレードするサーバに接続します。
2. データを複製するために Replication Agent™ が使用されている場合は、サーバに接続して、次のコマンドを実行します。

```
1> sp_config_rep_agent
2> go
```

上記のコマンドを実行すると、Replication Agent に対して有効になっているデータベースのリストが表示されます。

- a) データを複製するために他のアプリケーション、つまり LTM または Mirror Activator を使用している場合は、各データベース (マスタ・データベースも含む) で次のコマンドを実行します。

```
1> dbcc gettrunc
2> go
```

“secondary trunc state” に “1” が返されたデータベースでは、複製が有効になっています。

トランザクション処理と複製アクティビティの中断

複製データベースをアップグレードする前に、トランザクション処理と複製を中断してください。

アップグレード処理が終了するまで、アップグレードしているデータベース内のデータにサブスクリプションを作成するユーザがいないようにします。

1. アップグレードを開始する前に、アップグレードするデータベース内のプライマリ・データを使用して **create subscription** コマンドによって作成されるサブスクリプションが、プライマリ Replication Server で “valid” ステータスに達したことを確認します。 **check subscription** コマンドを使用して、“valid” ステータスを確認します。
2. アップグレード中のデータベースに Replication Server がアクセスを試みることをないように、サブスクリプションの作成中はアップグレード・プロセスを延期します。
3. アップグレードする各 Replication Server システム・データベース (RSSD) で **rs_helproute** を実行します。
4. すべての既存のルートステータスが “Active” になっていなければなりません。“Active” でないルートがある場合は、そのルートに関する問題を解決してから処理を継続してください。ルートに関する問題を診断、解決する方法については、Replication Server のマニュアルを参照してください。解決したら、次の手順に進み、アップグレードするデータベースを使用しているアプリケーションをすべて停止します。
5. アップグレードするデータベースを使用しているアプリケーションをすべて停止します。
6. Replication Server で **admin who** コマンドを使用して、アップグレードしているデータ・サーバへの既存のデータ・サーバ・インタフェース (DSI) 接続を検索します。
7. 各データベースの Replication Server で次のコマンドを入力して、アップグレードする RSSD 以外のデータベースへの DSI 接続をすべて中断します。

```
1> suspend connection to dataserver.database
2> go
```

8. RSSD データベースへの DSI 接続を実行したままにします。

プライマリ・データベースのトランザクション・ログの排出

アップグレードする各プライマリ・データベースについて、Replication Server がアップグレード前のログを完全に処理できるようにします。

次の手順に従い、トランザクション・ログが完全に排出されるようにします。

1. 残っているトランザクションがすべて複製されるまで待機します。
2. 複製テーブルに手動で加えた更新が Replication Server に転送されているかどうかをチェックします。
 - a) データベースが Replication Agent に対して有効になっている場合は、次のように入力します。

```
1> use <dbname>
2> go
1> sp_config_rep_agent <dbname>, "traceon", "9201"
2> go
```

Adaptive Server が実行されているコンソールにアクセスできない場合は、トレース・ログ・ファイルも定義して、Replication Agent から送信されてくる情報をチェックできるようにしてください。これを行うには、次のように入力します。

```
1> sp_config_rep_agent <dbname>, "trace_log_file", "path for
the log file"
2> go
```

- b) データベースによるデータの複製に別の方法を使用している場合は、Replication Server のキューを照会して、データが複製されていることを確認します。そのためには、次の Replication Server コマンドを実行します。

```
1> admin who, sqm
2> go
```

3. 次のように **isql** を使用して、プライマリ・データベースにある 1 つの複製テーブル内の 1 つのローを更新します。

```
1> update table set column = column
2> where key = unique_value
3> go
```

update コマンドを使用すると、複製データベースに対するすべての修正が Replication Server に送信されたかどうかを追跡できます。

注意： Sybase では、定義するプライマリ・キーを持つダミー・テーブルを使用することをおすすめします。テーブルのすべての列を備えた複写定義を定義することをおすすめします。

4. プライマリ Replication Server で、**admin who, sqm** コマンドをインバウンド・キューの最後の *segment: block* エントリが変わるまで実行します。
5. 次の Replication Server コマンドを実行して、この処理の前の手順で作成したダンプ・ファイルにインバウンド・キューの最後のブロックをダンプします。

```
1> sysadmin dump_queue, queue_number,  
2> queue_type, last_seg, block, 1  
3> go
```

6. メモ帳などのテキスト・エディタを使用してダンプ・ファイルを調べ、前の手順で実行した更新に対応するトランザクションが記載されていることを確認します。
7. 更新に対応するトランザクションがダンプ・ファイルに記載されるまで、上記の更新手順を繰り返します。
8. プライマリ・データベースからトランザクション・ログを読み込むアプリケーションまたはプロセスを停止します。使用しているものに応じて次のようにします。

- a) Rep Agent - Adaptive Server にログインして Rep Agent を停止します。

```
1> sp_stop_rep_agent database  
2> go
```

- b) LTM または Mirror Activator – 停止します。

トランザクション・ログの排出後は、データベース内でこれ以上アクティビティを発生させないでください。アクティビティが発生した場合、ログを再び排出してください。

9. Replication Agent、LTM、または Mirror Activator からの着信接続を拒否するように Replication Server で指定します。そのためには、Replication Server にログインして、そのデータベースからの Log Transfer 接続を中断します。

```
1> suspend log transfer from server.database  
2> go
```

RSSD トランザクション・ログの排出

Replication Server に、他の Replication Server へのルートがある場合は、Replication Server が RSSD トランザクション・ログ内のすべてのトランザクションを処理することを確認してから、データベースをアップグレードしてください。

トランザクション・ログが完全に処理されたことを確認するには、プライマリ Replication Server に複写定義を作成し、これがレプリケート Replication Server の RSSD に表示されることを監視します。複写定義がレプリケート RSSD にある場

合、ログは完全に処理されたと考えることができます。RSSD ログが処理されたことを確認するには、次の手順に従います。

1. プライマリ Replication Server にログインし、テンポラリ複写定義を作成します。

```
1> create replication definition rep_def_name
2> with primary at dataserver.database
3> (column_a int)
4> primary key (column_a)
5> go
```

データ・サーバ名とデータベース名には有効なものを指定しなければなりません。複写定義は実際のテーブルを参照する必要はありません。

2. プライマリ RSSD ではなくレプリケート RSSD にログインし、次のクエリを実行して、複写定義がプライマリ RSSD から届いているかどうかを確認します。

```
1> select * from rs_objects
2> where objname = "rep_def_name"
3> go
```

この **select** 文がローを返した場合、手順 1 で作成された最後の複写定義は、レプリケート RSSD に正常に送信されています。つまり、トランザクション・ログが排出されています。

3. レプリケート Replication Server にログインして、プライマリ RSSD からのログ転送接続を中断します。

```
1> suspend log transfer from server.database
2> go
```

4. Rep Agent を使用している場合は、Adaptive Server にログインして Rep Agent を停止します。

```
1> use database
2> go
```

```
1> sp_stop_rep_agent database
2> go
```

5. LTM を使用している場合は、LTM を停止します。

6. レプリケート RSSD の場合は、RSSD の Replication Server にログインして次のコマンドを発行します。

```
1> sysadmin hibernate_on, 'Replication_Server_name'
2> go
```

セカンダリ・トランザクション・ポイントの無効化

プライマリ・データベースのアップグレード時には、Rep Agent、Log Transfer、および Mirror Activator を停止しなければなりません。また、アップグレード中は、

複写データベースを含むサーバのアップグレード

セカンダリ・トランケーション・ポイントを無効にする必要があります。アップグレード前に Rep Agent と Log Transfer Manager を停止しておく必要があります。

各プライマリ・データベースとレプリケート RSSD に対して、次のようにセカンダリ・トランケーション・ポイントを無効にします。

1. 次のコマンドを実行して、アップグレードする Adaptive Server データベースのセカンダリ・トランケーション・ポイントを無効にします。

```
1> use database
2> go
```

```
1> dbcc settrunc('ltm', 'ignore')
2> go
```

2. **dbcc settrunc** コマンドが失敗する場合は、Rep Agent または LTM が停止していることを確認してください。Rep Agent と LTM が無効になっている場合は、最初の手順の操作をもう一度行ってください。

3. 次のコマンドを実行して、プライマリ・トランザクション・ログをトランケートします。

```
1> dump tran primary_dbname with truncate_only
2> go
```

4. Replication Server を停止し、Replication Server のトランケーション・ポイントをゼロ・クリアします。Replication Server の RSSD で次のコマンドを実行します。

```
1> rs_zeroltm primary_servername, primary_dbname
2> go
```

sybssystemprocs の使用

sysprocsdev は、このデバイスのデフォルトのシステム名です。ただし、Adaptive Server の属性ウィンドウの場合と同様に、*sybssystemprocs* デバイスと呼ばれることもあります。これは、*sybssystemprocs* データベースを格納しているからです。

sybssystemprocs データベースが十分な大きさであることを確認します。アップグレードするには、*sybssystemprocs* のサイズを 140MB より大きくします。または、既存の *sybssystemprocs* データベース、およびアップグレードの対象となる最大サイズのカatalogを格納できる十分な空き領域を確保し、その最大カatalog・サイズの 10% の領域をさらに追加します。追加する 10% の領域は、アップグレード時の変更内容のロギングに使用します。

ユーザ定義のストアド・プロシージャを追加する場合は、さらに多くの領域が必要です。内部データ構造を追加する場合は 140MB が適当ですが、この値では、ユーザ定義システム・プロシージャが大量に作成された場合に対応できません。

sybssystemprocs データベースがこれらの要件を満たしておらず、データベースを必要なサイズにまで大きくするための領域がデバイス上に十分にある場合は、**alter database** コマンドを使用してデータベース・サイズを大きくしてください。

sp_helpdb を使用して、*sybssystemprocs* データベースのサイズを決定します。

```
1> sp_helpdb sybssystemprocs
2> go
```

sp_helpdevice を使用して、*sysprocsdev* デバイスのサイズを決定します。

```
1> sp_helpdevice sysprocdev
2> go
```

db_size の設定が必要な最小値よりも小さい場合は、*sysprocdev* のサイズを大きくします。

Sybssystemprocs データベースのサイズを大きくする

現在の *sybssystemprocs* データベースで使用可能な領域が必要な領域の最小サイズに満たない場合は、次の2つのうちいずれかの方法で、アップグレードを行えるだけの十分な領域を持った新しいデータベースを作成します。ここでは、その方法について説明します。

- 最初の方法は、古いデータベースとデバイスを削除して、新しい *sysprocsdev* デバイスを作成します。

- 2 番目の方法は、古いデータベースとデバイスはそのままにして、追加領域を確保できる十分な大きさのデバイスを新しく追加し、*sybsystemprocs* をそのデバイス上に変更します。現在のデバイスを大きくすることをおすすめします。

sybsystemprocs データベースのサイズを大きくするには、次の手順に従います。

1. 古いデータベースの最新のバックアップがない場合は、ここで作成します。
2. **isql** で **alter database** を使用して *sybsystemprocs* データベースのサイズを大きくします。次に例を示します。

```
1> use master
2> go
1> alter database sybsystemprocs on sysprocsdev=40
2> go
```

この例では、"sysprocsdev" は既存のシステム・プロシージャ・デバイスの論理名、"40" は追加する領域のメガバイト数です。

システム・プロシージャ・デバイスが小さすぎる場合、*sybsystemprocs* データベースのサイズを増やそうとすると、次のようなメッセージが表示されることがあります。

```
Could not find enough space on disks to extend
database sybsystemprocs
```

- a) 別のデバイス上に使用可能な領域がある場合は、そのデバイスまで *sybsystemprocs* を拡張するか、十分な大きさの別のデバイスを初期化します。詳細については、「システム・プロシージャ用のデバイス容量とデータベース容量を増やす」のトピックを参照してください。
3. Adaptive Server が *sybsystemprocs* に十分な領域を割り付けたかどうかを確認するには、次のコマンドを実行します。

```
1> sp_helpdb sybsystemprocs
2> go
```

システム・プロシージャ・データベースが、*sybsystemprocs* データベースのサイズの増加に対応できるだけの十分な大きさを持つ場合は、引き続きその他のアップグレード前の作業を行ってください。

システム・プロシージャ用のデバイス容量とデータベース容量を増やす

サイズを大きくした *sybsystemprocs* データベースがシステム・プロシージャ・デバイスに入りきらない場合は、デバイスのサイズを大きくして、新しいデータベースを作成します。ここでは、その方法について説明します。

この手順には、データベースの削除が含まれます。 **drop database** の詳細については、『リファレンス・マニュアル』を参照してください。

警告！ この手順を実行すると、そのサイトで作成したストアド・プロシージャがすべて削除されます。開始する前に、**defncopy** ユーティリティを使用してローカル・ストアド・プロシージャを保存してください。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。

1. 削除する必要があるデバイスを決定します。

警告！ *sybsystemprocs* 以外のデータベースが使用しているデバイスを削除しないでください。データベースが破損します。

```
select d.name, d.phyname
from sysdevices d, sysusages u
where u.vstart between d.low and d.high
and u.dbid = db_id("sybsystemprocs")
and d.status & 2 = 2
and not exists (select vstart
                from sysusages u2
                where u2.dbid != u.dbid
                and u2.vstart between d.low and d.high)
```

- **d.name** – *sysdevices*. から削除するデバイスの一覧。
- **d.phyname** – コンピュータから削除するファイルの一覧。

このクエリ内の **not exists** 句は、*sybsystemprocs* や他のデータベースで使用されるデバイスを除外します。

以降の手順で使用するデバイスの名前を記録します。

2. *sybsystemprocs* を削除します。

```
1> use master
2> go
1> drop database sybsystemprocs
2> go
```

注意： 15.x より前のバージョンの Adaptive Server Enterprise では、**sysdevices** を使用して、手順 2 の *vstart* を含んでいる仮想ページ範囲 *low* ~ *high* がどのデバイスにあるのかを調べます。

バージョン 15.x では、手順 1 で取得した *dbid* に一致する *sysusages* から *vdevno* を選択します。

3. **sp_configure** を使用して、デバイスを削除します。

```
1> sp_configure "allow updates", 1
2> go
1> delete sysdevices
      where name in ("devname1", "devname2", ...)
2> go
1> sp_configure "allow updates", 0
2> go
```

where 句には、手順 1 のクエリで返されたデバイス名のリストが含まれます。

注意： デバイス名はそれぞれ引用符で囲んでください。たとえば、"devname1" や "devname2" などです。

指定されたデバイスの中にロー・パーティションではなく OS ファイルが含まれている場合は、適切な OS コマンドを使用してそのファイルを削除してください。

4. 返された **d.phyname** のリストのファイルをすべて削除します。

注意： ファイル名が完全なパス名でない可能性があります。相対パスを使用する場合、ファイル名はサーバを起動したディレクトリからの相対値です。

5. 必要な空き領域を持った別の既存のデバイスを探るか、次のような **disk init** コマンドを使用して *sybssystemprocs* 用の追加デバイスを作成します。ここで、`¥sybase¥work¥` は、システム・プロシージャ・デバイスへのフル・パス (絶対パス) です。

```
1> use master
2> go
```

```
1> disk init
2> name = "sysprocsdev",
3> physname = "¥sybase¥work¥sysproc.dat",
4> size = 51200
5> go
```

注意： Server バージョン 12.0.x 以降では、"vdevno=number" を受け付けますが、必須ではありません。12.0.x より前のバージョンでは、**vdevno** の番号が必要です。**vdevno** の値を使用できるかどうかを確認する方法については、『システム管理ガイド』を参照してください。

指定するサイズは、デバイスに必要な領域 (メガバイト単位) の 512 倍です。**disk init** では、サイズを 2K ページ単位で指定する必要があります。この例では、サイズは 112MB (112 x 512 = 57344) です。**disk init** の詳細については、『リファレンス・マニュアル』を参照してください。

6. そのデバイス上に適切なサイズの *sybssystemprocs* データベースを作成します。たとえば、次のように入力します。

```
1> create database sybssystemprocs on sysprocsdev =
    112
```

```
2> go
```

7. 古いサーバ・インストール・ディレクトリで **installmaster** スクリプトを実行します。次のように入力します。

```
isql -Usa -Ppassword -Sserver_name -i$SYBASE/ASE-15_0/scripts/
installmaster
```

追加の説明や情報の入手

Sybase Getting Started CD、Product Manuals サイト (SyBooks™ Online)、オンライン・ヘルプを利用すると、この製品リリースについて詳しく知ることができます。

- Getting Started CD (またはダウンロード) – PDF フォーマットのリリース・ノートとインストール・ガイド、その他のマニュアルや更新情報が収録されています。
- Product Manuals - Sybase マニュアル (<http://sybooks.sybase.com/>) のオンライン版であり、標準の Web ブラウザを使用してアクセスできます。ドキュメントをオンラインで閲覧したり、PDF としてダウンロードしたりできます。また、Web サイトには、製品マニュアルのほか、EBFs/Maintenance、Technical Documents、Case Management、Solved Cases、コミュニティ・フォーラム／ニュースグループなどのリソースへのリンクもあります。
- 製品のオンライン・ヘルプ (利用可能な場合)

PDF 形式のドキュメントを表示または印刷するには、Adobe の Web サイトから無償でダウンロードできる Adobe Acrobat Reader が必要です。

注意：最新のリリース・ノートと、製品のリリース後に追加された製品およびマニュアルに関する重要な情報は、Product Manuals Web サイトで確認できます。

追加の説明や情報の入手

索引

記号

.jar ファイルの更新 113

D

databases 102

devices 102

G

GUI モード 33

I

I/O フェンシング

データ整合性 127

設定 128

有効化 127

infiniband 25

L

LDAP ライブラリ

環境変数 83

ロケーション 83

logical clusters

追加 55

M

minimum size for database devices 183

P

PC クライアント

個別製品 11

製品の説明

ディレクトリ・レイアウト 9

preparing 102

privileges 183

R

raw partitions 183

availability of 183

S

SPDC へのアクセス

Sybase 製品ダウンロード

オンライン Web ポータル 16

Storage Foundation

クラスタの作成 42

sybcluster を使用したクラスタの作成 60

sybsystemprocs データベース

サイズを大きくする 195

U

Unified Agent の停止 42

updating .xml files 113

Y

無人モード 36

あ

新しいライセンス・サーバ 19

アップグレード 102

ASE プラグイン 104, 105

sybcluster 107

データベース内の Java 114

入力ファイル 110

手動 98

概要 97

対話形式 111

アップグレード後

統計の更新

installpcidb 112

アップグレード対応状況の確認 108

アップグレードの互換性

バージョン 87

アップグレード前

データベースでの Java 88

アンインストール

サーバ

レジストリ・エントリ 37, 86
 古いサーバ
 既存のサーバ 37
 アンサーブド・ライセンス
 生成 18

い

インストーラ 33
 インストール 33, 36
 インストールのプランニング 13
 インストール前の作業 27
 引用符付き識別子 92

お

応答ファイル 35

か

環境変数
 Unified Agent 42
 起動 42
 設定 42

き

共有インストーラ 39
 共有メモリ・パラメータ
 調整 29

く

クラスタの再設定 173
 クラスタの作成
 クラスタの起動 39
 クラスタの設定 60

こ

コマンド・ライン 36

さ

サーバ

Cluster Edition より優れた点 1
 インストール・オプション 1
 コンポーネントの説明
 ディレクトリ・レイアウト 6
 概要 1
 単一のシステムとしてアクセス可能なシ
 ステム 1
 負荷管理 1
 サーバ・エディションのオプション
 オプション・パッケージ 2
 サーブド・ライセンス
 生成 16
 サイレント・モード 36
 サポート・センタ
 電話による問い合わせ 139

し

システム・カタログの変更
 アップグレード 90
 システム管理者パスワード
 設定 73
 システムの稼働条件 25
 システム要件
 Linux 22
 PC クライアント 26
 手動設定
 sybcluster の有効化
 プラグインの有効化 174
 取得
 ライセンス 15

せ

設定手順 39

そ

増加
 デバイス容量とデータベース容量
 システム・プロシージャ 196

た

ダウングレード

Cluster Edition 123
Job Scheduler 125
概要 119
機能 123
ダウングレード後 121
ダウングレード手順
 PCI 機能 119
単一ノードでのシミュレートされたクラスタ
 41

と

トラブルシューティング
 SySAM
 猶予期間 131

に

入力ファイル
 作成 162

ね

ネットワーク・プロトコル 25

は

排出
 トランザクション・ログ 191
 RSSD 192
 プライマリ・データベース 191

ひ

表記規則
 フォント 4

ふ

プライベート・インストール 40

プライベート・インストール・モード 33
プライベート相互接続テクノロジー 25
プラグインを使用したクラスタの作成 50

ほ

補助サーバ
 設定 66
ホスト ID
 取得方法
 取得 30

ま

マイグレート 115
 BCP 116
 高可用性 116

む

無効化
 セカンダリ・トランケーション・ポイント
 193

よ

予約語
 競合 92
予約語チェック 92

わ

ワークフロー
 前提条件
 サーバ・インストール 3

