

SYBASE®

Web Services ユーザーズ・ガイド

Adaptive Server® Enterprise

15.5

ドキュメント ID : DC20144-01-1550-01

改訂 : 2009 年 10 月

Copyright © 2010 by Sybase, Inc. All rights reserved.

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

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

マニュアルの注文

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

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

Java および Java 関連の商標は、米国およびその他の国における Sun Microsystems, Inc. の商標または登録商標です。

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

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

はじめに	vii	
第 1 部	概要	
第 1 章	Adaptive Server Enterprise Web Services について 3	
	概要	3
	Adaptive Server Enterprise Web Services.....	4
	ASE Web Services の利点.....	4
	ストアド・プロシージャと関数	4
	SQL.....	4
	セキュリティ	5
	LDAP	5
	ユーザ定義 Web サービス	5
	Web サービス標準.....	5
	XML	6
	WSDL.....	8
	SOAP	10
第 2 章	ASE Web Services Engine について 11	
	Web サービスのプロデューサ	12
	プロデューサ・コンポーネント	12
	プロデューサ Web メソッド	13
	ユーザ定義 Web サービス	14
	Web サービスのコンシューマ	15
	コンシューマ・コンポーネント	15
	プロキシ・テーブル	17

第 2 部	設定	
第 3 章	ASE Web Services の設定	21
	設定	21
	インストール中の設定	22
	インストール後の設定	22
	ライセンス	23
	設定ファイル	23
	ws.properties ファイル	23
	logging.properties ファイル	23
	wsmsg.properties ファイル	24
	セキュリティ	24
	SSL の設定	24
	Microsoft .NET の証明書のインストール	25
第 3 部	管理	
第 4 章	ASE Web Services の使用	29
	ASE Web Services Engine の使用	29
	ASE Web Services Engine の起動と停止	29
	ASE Web Services のメソッド	31
	sp_webservices の使用方法	35
	Web サービスの呼び出し	39
	ユーザ定義 Web サービスの使用	42
	ユーザ定義 Web サービスのコマンド	42
	ユーザ定義 Web サービスでの sp_webservices の使用	46
	ユーザ定義 Web サービスのセキュリティ	49
	ユーザ定義 Web サービスの監査	49
	ASE Web Services のロギング	51
	ASE Web Services のログ・ファイル	51
	ログ・ファイルのロールオーバー	52
	Sybase Central を使用する場合	52
第 5 章	サンプル・アプリケーション	53
	Apache サンプル・クライアント	53
	サンプル・クライアントの作成	53
	runexecute の使用	54
	Microsoft .NET のサンプル・クライアント	57
	サンプル・クライアントの作成	57
	Execute.exe の使用	58

第 4 部	トラブルシューティング	
第 6 章	トラブルシューティング	63
	トラブルシューティングの問題	63
	リモート・サーバ・クラス定義の設定	63
	マップを解除された RPC/encoded Web メソッド	64
	トランケートされた document/literal の結果	64
	ASE Web Services Engine の起動	64
	WSDL の検索	65
	ws.properties でのエントリの指定	65
	Windows NT コマンド・ライン引数	66
	実行スクリプトまたは停止スクリプトの失敗	66
	null パスワード	66
	SSL での SOAP 終了ポイントの指定	67
	sp_webservices 'add' の異常終了	67
	Web Services のプロキシ・テーブルの制約	67
	sysattributes テーブル・エントリ	68
	診断ツール	68
	詳細なロギング	69
	JDBC レベルのトレースの有効化	69
	メッセージ	70
第 5 部	付録	
付録 A	インストール内容	75
	ASE Web Services のディレクトリ・ツリー	75
	bin ディレクトリの内容	76
	lib ディレクトリの内容	76
	logs ディレクトリの内容	77
	producer ディレクトリの内容	77
	props ディレクトリの内容	77
	samples ディレクトリの内容	78
付録 B	設定プロパティ	79
	ws.properties	79
	myres.properties	81
	properties ファイルでのエントリの指定	82

付録 C	SOAP と Adaptive Server Enterprise のデータ型のマッピング	83
	データ型のマッピング	83
	SOAP から ASE へのデータ型のマッピング	83
	create service コマンド用の ASE から SOAP への データ型マッピング	85
用語解説		87
索引		89

はじめに

対象読者

このマニュアルは、ユーザ定義の Web services または Adaptive Server® Enterprise が提供する Web services のアクセスに関心を持つユーザ、または Adaptive Server Enterprise を使用して他のアプリケーションの Web services にアクセスするユーザを対象としています。

このマニュアルの内容

このマニュアルは次の部分から構成されています。

- 「第 1 部 概要」
 - 「第 1 章 Adaptive Server Enterprise Web Services について」
 - 「第 2 章 ASE Web Services Engine について」
- 「第 2 部 設定」
 - 「第 3 章 ASE Web Services の設定」
- 「第 3 部 管理」
 - 「第 4 章 ASE Web Services の使用」
 - 「第 5 章 サンプル・アプリケーション」
- 「第 4 部 トラブルシューティング」
 - 「第 6 章 トラブルシューティング」
- 「第 5 部 付録」
 - 「付録 A インストール内容」
 - 「付録 B 設定プロパティ」
 - 「付録 C SOAP と Adaptive Server Enterprise のデータ型のマッピング」

関連マニュアル

Adaptive Server® Enterprise には次のマニュアルが用意されています。必要に応じて参照してください。

- 使用しているプラットフォームの『リリース・ノート』— マニュアルには記載できなかった最新の情報が記載されています。

このリリース・ノートの最新バージョン（英語版）を入手できます。製品の CD がリリースされた後で、製品またはマニュアルに関する重要な情報が追加されているかを確認するには、Sybase Product Manuals Web サイトを使用してください。

-
- 使用しているプラットフォームの『インストール・ガイド』 – すべての Adaptive Server および関連する Sybase 製品のインストール、アップグレード、設定の手順について説明しています。
 - 『新機能ガイド』 – Adaptive Server の新しい機能について説明しています。また、新しい機能をサポートするためのシステム変更や、既存のアプリケーションに影響を与える可能性がある変更についても説明しています。
 - 『Active Messaging ユーザーズ・ガイド』 – Active Messaging を使用して、Adaptive Server Enterprise データベースでトランザクション (データ変更) を取得し、外部アプリケーションにイベントとしてリアルタイムで渡す方法について説明しています。
 - 『コンポーネント統合サービス・ユーザーズ・ガイド』 – コンポーネント統合サービスを使用して、リモートの Sybase データベースおよび Sybase 以外のデータベースに接続する方法について説明しています。
 - 使用しているプラットフォームの『設定ガイド』 – 特定の設定作業の手順について説明しています。
 - 『用語解説』 – Adaptive Server マニュアルで使用されている技術用語について説明しています。
 - 『Historical Server ユーザーズ・ガイド』 – Historical Server を使用して、Adaptive Server のパフォーマンス情報を入手する方法について説明しています。
 - 『Adaptive Server Enterprise における Java』 – Adaptive Server データベースで Java クラスをデータ型、関数、ストアド・プロシージャとしてインストールして使用する方法について説明しています。
 - 『Job Scheduler ユーザーズ・ガイド』 – コマンド・ラインまたはグラフィカル・ユーザ・インタフェース (GUI) を使用して、ローカルまたはリモートの Adaptive Server でジョブのインストール、設定、作成、スケジュールを行う方法について説明しています。
 - 『マイグレーション技術ガイド』 – 別のバージョンの Adaptive Server にマイグレートするための方法とツールについて説明しています。
 - 『Monitor Client Library プログラマーズ・ガイド』 – Adaptive Server のパフォーマンス・データにアクセスする Monitor Client Library アプリケーションの記述方法について説明しています。
 - 『Monitor Server ユーザーズ・ガイド』 – Monitor Server を使用して、Adaptive Server のパフォーマンス統計を取得する方法について説明しています。
 - 『モニタリング・テーブル・ダイアグラム』 – モニタリング・テーブルと、そのエンティティの関係をポスター形式で図解しています。フル・サイズのダイアグラムは印刷版だけで参照できます。コンパクト版は PDF 形式で参照できます。

- 『パフォーマンス&チューニング・シリーズ』－ Adaptive Server で最高のパフォーマンスを実現するためのチューニング方法について説明しています。
 - 『基本』－ Adaptive Server のパフォーマンスに関する問題の理解と調査の基本について説明しています。
 - 『統計的分析によるパフォーマンスの向上』－ Adaptive Server で統計情報がどのように保存され、表示されるかについて説明しています。また、`set statistics` コマンドを使用して、サーバの統計情報を分析する方法について説明しています。
 - 『ロックと同時実行制御』－ ロック・スキームを使用してパフォーマンスを向上させる方法と、同時実行性を最小限に抑えるようにインデックスを選択する方法について説明しています。
 - 『sp_sysmon による Adaptive Server の監視』－ `sp_sysmon` を使用してパフォーマンスをモニタリングする方法について説明しています。
 - 『モニタリング・テーブル』－ Adaptive Server のモニタリング・テーブルに統計情報や診断情報を問い合わせる方法について説明しています。
 - 『物理データベースのチューニング』－ データの物理的配置、データに割り付けられた領域、テンポラリ・データベースの管理方法について説明しています。
 - 『クエリ処理と抽象プラン』－ オプティマイザがクエリを処理する方法と、抽象プランを使用してオプティマイザのプランの一部を変更する方法について説明しています。
- 『クイック・リファレンス・ガイド』－ コマンド、関数、システム・プロシージャ、拡張システム・プロシージャ、データ型、ユーティリティの名前と構文の包括的な一覧表を記載したポケット版 (PDF 版は通常サイズ) のマニュアルです。
- 『ASE リファレンス・マニュアル』－ 詳細な Transact-SQL® 情報を記載しています。このマニュアルは以下の 4 冊に分かれています。
 - 『ビルディング・ブロック』－ データ型、関数、グローバル変数、式、識別子とワイルドカード、予約語について説明しています。
 - 『コマンド』－ コマンドについて説明しています。
 - 『プロシージャ』－ システム・プロシージャ、カタログ・ストアド・プロシージャ、システム拡張ストアド・プロシージャ、`dbcc` ストアド・プロシージャについて説明しています。
 - 『テーブル』－ システム・テーブル、モニタリング・テーブル、`dbcc` テーブルについて説明しています。

-
- 『システム管理ガイド』でさらに詳しく説明しています。
 - 『第1巻』－ 設定パラメータ、リソースの問題、文字セット、ソート順、システムの問題の診断方法に関する説明を含め、システム管理の基本の概要について説明しています。『第1巻』の後半は、セキュリティ管理に関する詳細な説明です。
 - 『第2巻』－ 物理的なリソースの管理、デバイスのミラーリング、メモリとデータ・キャッシュの設定、マルチプロセッサ・サーバとユーザ・データベースの管理、データベースのマウントとマウント解除、セグメントの作成と使用、**reorg** コマンドの使用、データベース一貫性の検査方法についての手順とガイドラインを説明しています。『第2巻』の後半では、システムとユーザ・データベースをバックアップおよびリストアする方法について説明しています。
 - 『システム・テーブル・ダイアグラム』－ システム・テーブルと、そのエンティティとの関係をポスター形式で図解しています。フル・サイズのダイアグラムは印刷版だけで参照できます。コンパクト版は PDF 形式で参照できます。
 - 『Transact-SQL ユーザーズ・ガイド』－ リレーショナル・データベース言語の拡張版である Sybase の Transact-SQL について説明しています。まだ経験の浅いデータベース管理システムのユーザは、このマニュアルをガイドブックとして使用してください。pubs2 および pubs3 サンプル・データベースの詳細も説明しています。
 - 『トラブルシューティング・シリーズ』－
 - 『トラブルシューティング：エラー・メッセージと詳細な解決方法』－ 発生する可能性のある問題について、トラブルシューティング手順を説明しています。このマニュアルで取り上げられている問題は、Sybase 製品の保守契約を結んでいるサポート・センタに最も頻繁に寄せられるものです。
 - 『トラブルシューティング&エラー・メッセージ・ガイド』－ 発生頻度が高い Adaptive Server のエラー・メッセージの解決方法について詳しい手順を説明しています。
 - 『暗号化カラム・ユーザーズ・ガイド』－ Adaptive Server を使用して暗号化カラムを設定し、使用方法について説明しています。
 - 『インメモリ・データベース・ユーザーズ・ガイド』－ インメモリ・データベースの設定および使用方法について説明しています。
 - 『Adaptive Server 分散トランザクション管理機能の使用』－ 分散トランザクション処理環境での Adaptive Server DTM 機能の設定、使用、トラブルシューティングについて説明しています。
 - 『IBM® Tivoli® Storage Manager と Backup Server の使用』－ IBM Tivoli Storage Manager を設定および使用して Adaptive Server のバックアップを作成する方法について説明しています。

- 『高可用性システムにおける Sybase フェールオーバーの使用』 – Sybase のフェールオーバー機能を使用して、Adaptive Server を高可用性システムのコンパニオン・サーバとして設定する方法について説明しています。
- 『Unified Agent および Agent Management Console』 – Unified Agent について説明しています。Unified Agent は、分散 Sybase リソースを管理、モニタ、制御するためのランタイム・サービスを提供します。
- 『ASE ユーティリティ・ガイド』 – オペレーティング・システム・レベルで実行される `isql` や `bcp` などの、Adaptive Server のユーティリティ・プログラムについて説明しています。
- 『Web Services ユーザーズ・ガイド』 – Adaptive Server 用の Web サービスの設定、使用、トラブルシューティング方法について説明しています。
- 『XA インタフェース統合ガイド for CICS, Encina, TUXEDO』 – X/Open XA トランザクション・マネージャを備えた Sybase DTM XA インタフェースを使用する方法について説明しています。
- 『Adaptive Server Enterprise における XML サービス』では、データベースに XML 機能を導入する、Sybase ネイティブの XML プロセッサと Sybase Java ベースの XML のサポートについて、また XML サービスに準拠したクエリとマッピング用の関数について説明しています。

その他の情報

Sybase Getting Started CD、SyBooks™ CD、Sybase® Product Manuals Web サイトを利用すると、製品について詳しく知ることができます。

- Getting Started CD には、PDF 形式のリリース・ノートとインストール・ガイド、SyBooks CD に含まれていないその他のマニュアルや更新情報が収録されています。この CD は製品のソフトウェアに同梱されています。Getting Started CD に収録されているマニュアルを参照または印刷するには、Adobe Acrobat Reader が必要です (CD 内のリンクを使用して Adobe の Web サイトから無料でダウンロードできます)。
- SyBooks CD には製品マニュアルが収録されています。この CD は製品のソフトウェアに同梱されています。Eclipse ベースの SyBooks ブラウザを使用すれば、使いやすい HTML 形式のマニュアルにアクセスできます。

一部のマニュアルは PDF 形式で提供されています。これらのマニュアルは SyBooks CD の PDF ディレクトリに収録されています。PDF ファイルを開いたり印刷したりするには、Adobe Acrobat Reader が必要です。

SyBooks をインストールして起動するまでの手順については、Getting Started CD の『SyBooks インストール・ガイド』、または SyBooks CD の『*README.txt*』ファイルを参照してください。

- Sybase Product Manuals Web サイトは、SyBooks CD 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。また、製品マニュアルのほか、EBFs/Updates、Technical Documents、Case Management、Solved Cases、ニュース・グループ、Sybase Developer Network へのリンクもあります。

Technical Library Product Manuals Web サイトにアクセスするには、Product Manuals (<http://www.sybase.com/support/manuals/>) にアクセスしてください。

Web 上の Sybase 製品の動作確認情報

Sybase Web サイトの技術的な資料は頻繁に更新されます。

❖ 製品認定の最新情報にアクセスする

- 1 Web ブラウザで Technical Documents を指定します。
(<http://www.sybase.com/support/techdocs/>)
- 2 左側のナビゲーション・バーから [Products] を選択します。
- 3 製品リストから製品名を選択し、[Go] をクリックします。
- 4 [Certification Report] フィルタを選択し、時間枠を指定して [Go] をクリックします。
- 5 [Certification Report] のタイトルをクリックして、レポートを表示します。

❖ コンポーネント認定の最新情報にアクセスする

- 1 Web ブラウザで Availability and Certification Reports を指定します。
(<http://certification.sybase.com/>)
- 2 [Search By Base Product] で製品ファミリーとベース製品を選択するか、[Search by Platform] でプラットフォームとベース製品を選択します。
- 3 [Search] をクリックして、入手状況と認定レポートを表示します。

❖ Sybase Web サイト (サポート・ページを含む) の自分専用のビューを作成する

MySybase プロファイルを設定します。MySybase は無料サービスです。このサービスを使用すると、Sybase Web ページの表示方法を自分専用カスタマイズできます。

- 1 Web ブラウザで Technical Documents を指定します。
(<http://www.sybase.com/support/techdocs/>)
- 2 [MySybase] をクリックし、MySybase プロファイルを作成します。

Sybase EBF とソフトウェア・メンテナンス

❖ EBF とソフトウェア・メンテナンスの最新情報にアクセスする

- 1 Web ブラウザで Sybase Support Page (<http://www.sybase.com/support>) を指定します。
- 2 [EBFs/Maintenance] を選択します。MySybase のユーザ名とパスワードを入力します。
- 3 製品を選択します。

- 4 時間枠を指定して [Go] をクリックします。EBF/Maintenance リリースの一覧が表示されます。

鍵のアイコンは、「Technical Support Contact」として登録されていないため、一部の EBF/Maintenance リリースをダウンロードする権限がないことを示しています。未登録でも、Sybase 担当者またはサポート・コンタクトから有効な情報を得ている場合は、[Edit Roles] をクリックして、「Technical Support Contact」の役割を MySybase プロファイルに追加します。

- 5 EBF/Maintenance レポートを表示するには [Info] アイコンをクリックします。ソフトウェアをダウンロードするには製品の説明をクリックします。

表記規則

表 1 に構文の規則を示します。

表 1: このマニュアルでのフォントと構文規則

要素	例
コマンド名、プロシージャ名、ユーティリティ名、その他のキーワードは sans serif フォントで表記する。	<code>select</code> <code>sp_configure</code>
データベース名とデータ型は sans serif フォントで表記する。	<code>master</code> データベース
ファイル名、変数、パス名は斜体で表記する。	システム管理ガイド <i>sql.ini</i> ファイル <i>column_name</i> SSYBASE/ASE ディレクトリ
変数 (ユーザが入力する値を表す語) がクエリまたは文の一部である場合は Courier フォントの斜体で表記する。	<code>select column_name</code> <code>from table_name</code> <code>where search_conditions</code>
カッコはコマンドの一部として入力する。	<code>compute row_aggregate (column_name)</code>
2つのコロンと等号は、構文が BNF 表記で記述されていることを示す。この記号は入力しない。「~と定義されている」ことを意味する。	<code>::=</code>
中カッコで囲まれたオプションの中から必ず 1つ以上を選択する。コマンドには中カッコは入力しない。	<code>{cash, check, credit}</code>
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。	<code>[cash check credit]</code>
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。	<code>cash, check, credit</code>
パイプまたは縦線は複数のオプションのうち 1つだけを選択できることを意味する。	<code>cash check credit</code>

要素

省略記号 (...) は、直前の要素を必要な回数だけ繰り返し指定できることを意味する。

例

```
buy thing = price [cash | check | credit]
[, thing = price [cash | check | credit]]...
```

この例では、製品 (thing) を少なくとも 1 つ購入 (buy) し、価格 (price) を指定する必要があります。支払方法を選択できる。角カッコで囲まれた項目の 1 つを選択する。追加品目を、必要な数だけ購入することもできる。各 buy に対して、購入した製品 (thing)、価格 (price)、オプションで支払方法 (cash、check、credit のいずれか) を指定します。

- 次は、オプション句のあるコマンドの構文の例です。

```
sp_dropdevice [device_name]
```

複数のオプションを持つコマンドの例を示します。

```
select column_name
from table_name
where search_conditions
```

構文では、キーワード (コマンド) は通常のフォントで表記し、識別子は小文字で表記します。ユーザが提供するワードは斜体で表記します。

- Transact-SQL コマンドの使用例は次のように表記します。

```
select * from publishers
```

- 次は、コンピュータからの出力例です。

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

(3 rows affected)

このマニュアルでは、例に使用する文字はほとんどが小文字ですが、Transact-SQL のキーワードを入力するときは、大文字と小文字は区別されません。たとえば、**SELECT**、**Select**、**select** はすべて同じです。

テーブル名などのデータベース・オブジェクトの大文字と小文字を Adaptive Server が区別するかどうかは、Adaptive Server にインストールされたソート順によって決まります。シングルバイト文字セットを使用している場合は、Adaptive Server のソート順を再設定することによって、大文字と小文字の区別の取り扱い方を変更できます。詳細については、『ASE システム管理ガイド』を参照してください。

アクセシビリティ機能

このマニュアルには、アクセシビリティを重視した HTML 版もあります。この HTML 版マニュアルは、スクリーン・リーダーで読み上げる、または画面を拡大表示するなどの方法により、その内容を理解できるよう配慮されています。

Adaptive Server HTML マニュアルは、連邦リハビリテーション法第 508 条のアクセシビリティ規定に準拠していることがテストにより確認されています。第 508 条に準拠しているマニュアルは通常、World Wide Web Consortium (W3C) の Web サイト用ガイドラインなど、米国以外のアクセシビリティ・ガイドラインにも準拠しています。

注意 アクセシビリティ・ツールを効率的に使用するには、設定が必要な場合もあります。一部のスクリーン・リーダーは、テキストの大文字と小文字を区別して発音します。たとえば、すべて大文字のテキスト (ALL UPPERCASE TEXT など) はイニシャルで発音し、大文字と小文字の混在したテキスト (Mixed Case Text など) は単語として発音します。構文規則を発音するようにツールを設定すると便利かもしれません。詳細については、ツールのマニュアルを参照してください。

不明な点があるときは

Sybase のアクセシビリティに対する取り組みについては、[Sybase Accessibility \(http://www.sybase.com/accessibility\)](http://www.sybase.com/accessibility) を参照してください。Sybase Accessibility サイトには、第 508 条と W3C 標準に関する情報へのリンクもあります。

Sybase ソフトウェアがインストールされているサイトには、Sybase 製品の保守契約を結んでいるサポート・センタとの連絡担当の方 (コンタクト・パーソン) を決めてあります。マニュアルだけでは解決できない問題があった場合には、担当の方を通して Sybase のサポート・センタまでご連絡ください。



第 1 部

概要

ここでは、Adaptive Server Enterprise Web Services について説明します。

Adaptive Server Enterprise Web Services について

この章の内容は、次のとおりです。

トピック名	ページ
概要	3
Adaptive Server Enterprise Web Services	4
ASE Web Services の利点	4
Web サービス標準	5

概要

Web サービスは、ネットワーク接続を介して利用できる独立したモジュール形式のアプリケーションです。エンド・ユーザは Web サービスを使用して、Simple Object Access Protocol (SOAP)、Web Services Description Language (WSDL)、Extensible Markup Language (XML) オープン標準に準拠することで、パフォーマンスの代わりに相互運用性を向上させることができます。

実装に使用されているプログラミング言語にかかわらず、さまざまなプラットフォームやオペレーティング・システムから Web サービスを利用できるため、異なるアプリケーションでのデータ共有機能が大幅に強化されます。限定された特定のタスクを扱う各種 Web サービスを使用して、既存のソフトウェアを安全で管理された環境に公開することにより、企業の統合を動的かつ加速的に進めることができます。Web サービスは、リモート・アプリケーションを起動するための標準化された手段を提供することによって、インフラストラクチャに必要なコードの量を減らします。また、Web サービスは、公開されたインタフェース (WSDL) からユーザが実装を取り出せるようにすることによって、サービス指向アーキテクチャ (SOA) の構築に必要なツールを提供します。

Adaptive Server Enterprise Web Services

Adaptive Server Enterprise (ASE) Web Services は、Adaptive Server Enterprise とは独立して実行される ASE Web Services Engine で構成されています。

ASE Web Services Engine には次の機能が備わっています。

- SOAP を使用してクライアント・アプリケーションが Adaptive Server Enterprise の SQL およびストアド・プロシージャにアクセスできる。
- Adaptive Server Enterprise から他のアプリケーションの Web サービスにアクセスできる。これらの外部 Web サービスは、実行時に Adaptive Server Enterprise のプロキシ・テーブルにマップされる。
- ユーザ定義の Web サービスを提供して、Adaptive Server Enterprise で Web ブラウザまたは SOAP クライアントを使用して SQL コマンドを実行できる。

ASE Web Services Engine の詳細については、「[第 2 章 ASE Web Services Engine について](#)」を参照してください。

ASE Web Services の利点

ASE Web Services Engine を使用すると、ユーザはストアド・プロシージャ、ユーザ定義関数、SQL を使用して、データの問い合わせおよび操作ができます。クライアント・アプリケーションは、SQL コマンドを含む SOAP 要求を送信し、SOAP を介して結果を受信できます。データは SQLX 標準に従って返され、クライアント・アプリケーションは XML データ、スキーマ、DTD を受信します。

ストアド・プロシージャと関数

ストアド・プロシージャは、データの内部の論理ビューをビジネス・レベルのロジックから切り離し、SQL の影響とパフォーマンスを拡張します。ストアド・プロシージャは、リモートで実行することもできます。ユーザは、ストアド・プロシージャとユーザ定義関数の両方を使用して、ANSI SQLJ 標準の規定に従って Java メソッドを呼び出したり、標準 XML フォーマットでデータを取得したりできます。

SQL

SQL を使用して XML データを操作できるので、SOAP 対応のクライアント・アプリケーションは ASE Web Services Engine を使用して Adaptive Server Enterprise 内のデータを管理できます。SQL を使用して、ASE Web Services Engine から Web サービスを呼び出すこともできます。

セキュリティ

ASE Web Services のセキュリティ機能には、Secure Sockets Layer (SSL) が組み込まれています。これは、LDAP (Lightweight Directory Access Protocol) によるアクセス制御など、重要なデータベース・セキュリティ機能と権限付与機能を提供します。

LDAP

LDAP は、分散環境でディレクトリにアクセスするためのインターネット・プロトコルです。LDAP サーバは、リソース間の接続の確立に必要なユーザ情報を保存し、ディレクトリへのアクセスを許可するので、クライアント・アプリケーションはその情報を知る必要がありません。ASE Web Services を使用すると、クライアント・アプリケーションが LDAP を使用して Web メソッドにアクセスできます。

ASE Web Services は、LDAP バージョン 3 サーバをサポートしています。LDAP を使用したユーザ認証の有効化や Adaptive Server Enterprise データ・サーバの検索の詳細については、Adaptive Server Enterprise の『システム管理ガイド』を参照してください。

ユーザ定義 Web サービス

ユーザ定義の Web サービスを使用すると、Adaptive Server Enterprise で Web ブラウザまたは SOAP クライアントを使用して SQL コマンドを実行できます。この機能により、Web サービスの名前、実行する SQL、URL のロケーションを定義できます。

ユーザ定義の Web サービスを使用すると、Adaptive Server Enterprise に対する SOA 準拠のインタフェースを作成できます。

Web サービス標準

Web サービスは XML で構築され、WSDL で記述され、SOAP over HTTP で転送されます。ASE Web Services を使用すると、クライアント・アプリケーションが ASE を使用して Web サービスにアクセスできます。また、リモート Web サービスも使用できます。

XML

XML は、データの記述に使用されます。XML は SGML から派生したものであり、HTML などその他のマークアップ言語の特性の一部を備えています。ただし、XML はタグがユーザ定義なので拡張可能であり、通信する複数のアプリケーションが解釈できる構造でデータを交換する場合に適しています。

例

次に示す **pubs2** データベースへの **isql** クエリは、値引きに関する情報を検索します。

```
select * from discounts
```

このクエリは次の結果セットを生成します。

discounttype	stor_id	lowqty	highqty	discount
Initial Customer	NULL	NULL	NULL	10.500000
Volume Discount	NULL	100	1000	6.700000
Huge Volume Discount	NULL	1001	NULL	10.000000
Customer Discount	8042	NULL	NULL	5.000000

この結果セットを XML で表す方法は多数あります。次に、ASE Web Services によって生成され、SQL の ANSI 標準の一部である SQLX でフォーマットされた XML 表現を示します。

```
<?xml version="1.0" encoding="UTF-8">
<ws xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <row>
    <discounttype>Initial Customer</discounttype>
    <discount>10.5</discount>
  </row>
  <row>
    <discounttype>Volume Discount</discounttype>
    <lowqty>100</lowqty>
    <highqty>1000</highqty>
    <discount>6.7</discount>
  </row>
  <row>
    <discounttype>Huge Volume Discount
  </discounttype>
    <lowqty>1001</lowqty>
    <discount>10.0</discount>
  </row>
  <row>
    <discounttype>Customer Discount</discounttype>
    <stor_id>8042</stor_id>
    <discount>5.0</discount>
  </row>
</ws>
```

最初の行は、XML のバージョンと文字コードを示します。残りのタグはユーザ定義であり、ドキュメントの構造とデータの両方を記述します。これらのユーザ定義タグにより、値引き情報を使用して価格を計算するアプリケーションなど、特定のアプリケーション用にドキュメントをカスタマイズすることができます。

XML ドキュメント構造

正しい形式の XML ドキュメントでは、ユーザ定義要素とその配置がドキュメント・タイプ定義 (DTD) または XML スキーマによって定義されます。

次に、前に説明した値引き情報の例の DTD を示します。

```
<!DOCTYPE ws [
<!ELEMENT ws (row*)>
<!ELEMENT row (discounttype, stor_id?, lowqty?, highqty?,
discount)>
<!ELEMENT discounttype (#PCDATA)>
<!ELEMENT stor_id (#PCDATA)>
<!ELEMENT lowqty (#PCDATA)>
<!ELEMENT highqty (#PCDATA)>
<!ELEMENT discount (#PCDATA)>
]>
```

次に、前に説明した値引き情報の例の XML スキーマの一部を示します。

```
<xsd:schema
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:sqlxml="http://www.iso-
standards.org/mra/9075/sqlx">
<xsd:import
namespace="http://www.w3.org/2001/XMLSchema"
schemaLocation="http://www.iso-
standards.org/mra/9075/sqlx.xsd" />
<xsd:complexType name="RowType.ws">
<xsd:sequence>
<xsd:element name="discounttype"
type="VARCHAR_40" />
<xsd:element name="stor_id" type="CHAR_4"
minOccurs="0" maxOccurs="1"/>
<xsd:element name="lowqty" type="SMALLINT"
minOccurs="0" maxOccurs="1"/>
<xsd:element name="highqty" type="SMALLINT"
minOccurs="0" maxOccurs="1"/>
<xsd:element name="discount" type="FLOAT" />
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="TableType.ws">
<xsd:sequence>
<xsd:element name="row" type="RowType.ws"
minOccurs="0" maxOccurs="unbounded"/>
```

```

    </xsd:sequence>
  </xsd:complexType>
  <xsd:simpleType name="VARCHAR_40">
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="40"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="VARCHAR_4">
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="4"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="SMALLINT">
    <xsd:restriction base="xsd:integer">
      <xsd:maxInclusive value="32767"/>
      <xsd:minInclusive value="-32768"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="FLOAT">
    <xsd:restriction base="xsd:float"/>
  </xsd:simpleType>
  <xsd:element name="ws" type="TableType.ws"/>
</xsd:schema>

```

XML スキーマまたは DTD は、記述する XML ドキュメントの一部として組み込むことも、個別のファイルとして参照することもできます。XML スキーマと DTD のファイル・サフィックスは、それぞれ `.xsd` と `.dtd` です。

XML の詳細については、次の資料を参照してください。

- World Wide Web Consortium (W3C): <http://www.w3.org>
- W3C, Extensible Markup Language (XML): <http://www.w3.org/XML/>

WSDL

WSDL ドキュメントは、XML を使用して Web サービスを記述します。WSDL 記述は、Web サービスのロケーションを指定するだけでなく、Web サービスが提供するメソッドや、Web サービスが使用するメッセージ、データ型、通信プロトコルも次のタグで指定します。

- `<service>` – Web サービスの名前を定義します。たとえば、Web サービス名 `ExecuteStoredProcService` は次のようになります。

```

<wsdl:service name="ExecuteStoredProcService">
  <wsdl:port binding="impl:aseSoapBinding" name="ase">
    <wsdlsoap:address location="http://myserver:8181/services/ase"/>
  </wsdl:port>
</wsdl:service>

```


WSDL ドキュメントには、1 つ以上の **<service>** タグを組み込むことができます。ASE Web Services Engine の場合は、“ase” という名前のサービスが 1 つあるだけです。

- **<binding>** – 使用する通信プロトコルを定義します。次の例では、SOAP プロトコルを使用しています。

```
<wsdl:binding name="aseSoapBinding" type="impl:ExecuteStoredProc">
...
</wsdl:binding>
```

WSDL は、HTTP プロトコルと MIME プロトコルの仕様もサポートしています。

- **<port>** – Web サービスのアドレスを指定します。次に例を示します。

```
<wsdl:port binding="impl:aseSoapBinding" name="ase">
  <wsdlsoap:address location="http://myserver:8181/services/ase"/>
</wsdl:port>
```

<port> タグには名前とバインドの属性があります。

- **<message>** – 使用するメッセージを定義します。次に例を示します。

```
<wsdl:message name="executeRequest">
  <wsdl:part name="service" type="xsd:string"/>
  <wsdl:part name="userName" type="xsd:string"/>
  <wsdl:part name="password" type="xsd:string"/>
  <wsdl:part name="sqlOptions" type="xsd:string"/>
  <wsdl:part name="sql" type="xsd:string"/>
</wsdl:message>
```

これは、メソッド **executeRequest** の要求メッセージです。**<part>** タグは、要求メッセージのメソッド呼び出しのパラメータ値と応答の戻り値に対応しています。

- **<operation>** – メッセージを Web メソッドの要求または応答に関連付けます。次に例を示します。

```
<wsdl:operation name="execute" parameterOrder="service userName
password sqlOptions sql">
  <wsdl:input message="impl:executeRequest" name="executeRequest"/>
  <wsdl:output message="impl:executeResponse" name="executeResponse"/>
</wsdl:operation>
```

- **<portType>** – 提供されるメソッドを定義します。**<operation>** タグは、**<portType>** の子要素です。次に例を示します。

```
<wsdl:portType name="ExecuteStoredProc">
  <wsdl:operation name="execute" parameterOrder="aseServerName asePortNumber
...
  </wsdl:operation>
</wsdl:portType>
```

- `<types>` – 使用されるデータ型を定義します。WSDL は、XML スキーマ構文を使用してデータ型を定義します。

WSDL は、通常、ASE Web Services Engine によって自動的に生成され、次のロケーションで Web ブラウザに表示できます。

```
http://myserver:producer_port/services/ase?wsdl
```

ここで、*myserver* は ASE Web Services Engine が動作しているホストの名前、*producer_port* はポート番号です。

SOAP

SOAP は、プラットフォームや言語に依存しない、XML ベースのプロトコルであり、アプリケーション間でのメッセージやデータの送信に使用されます。SOAP はメッセージの構造を定義し、メッセージの処理方法を記述し、アプリケーション定義のデータ型をコード化するためのルールを提供します。SOAP を使用すると、アプリケーションは、HTTP などの標準トランスポート・レイヤ・プロトコルでリモート・プロシージャ・コール (RPC) を送受信できます。

SOAP メッセージ構造

SOAP メッセージはヘッダと本文で構成されます。ヘッダと本文は SOAP エンベロープに含まれています。通常、SOAP 要求メッセージにはヘッダ情報は含まれませんが、前の要求メッセージに対応する応答メッセージにはヘッダが含まれ、必ずしもメッセージの本文は表示されません。

SOAP の詳細については、次の資料を参照してください。

- Simple Object Access Protocol (1.1)
<http://www.w3.org/TR/SOAP/>
- Simple Object Access Protocol (1.2) Part 1
<http://www.w3.org/TR/soap12-part1/>
- Simple Object Access Protocol (1.2) Part 2
<http://www.w3.org/TR/soap12-part2/>

ASE Web Services Engine について

この章の内容は、次のとおりです。

トピック名	ページ
Web サービスのプロデューサ	12
Web サービスのコンシューマ	15

ASE Web Services Engine には次の機能が用意されています。

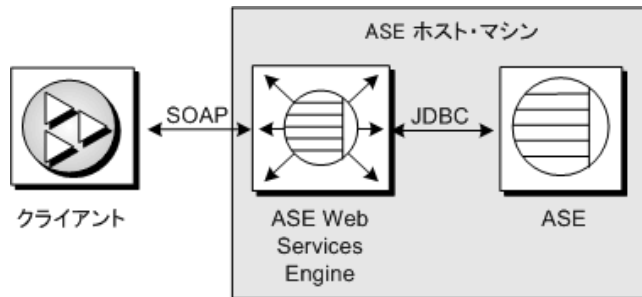
- [Web サービスのプロデューサ](#) – クライアント・アプリケーションが SOAP を使用して Adaptive Server Enterprise のストアド・プロシージャおよび SQL にアクセスできる。
- [Web サービスのコンシューマ](#) – Adaptive Server Enterprise から Web メソッドにアクセスして実行することができる。

Web サービスのプロデューサ

ASE Web Services Engine をプロデューサとして使用すると、クライアント・アプリケーションは、SOAP を介して Adaptive Server Enterprise のストアド・プロシージャと SQL にアクセスできます。ASE Web Services Engine の出力は、SQL の ANSI 仕様の一部として定義された SQLX に準拠します。

注意 ASE Web Services Engine は、Adaptive Server Enterprise と同じマシンで実行することをおすすめします。

図 2-1: クライアントが ASE にアクセスする場合の ASE Web Services Engine



クライアントは SQL またはストアド・プロシージャを SOAP 要求として送信し、その結果はすべて SOAP 応答として返されます。SOAP 応答内のデータは SQLX 標準に準拠します。

プロデューサ・コンポーネント

Web サービスのプロデューサとして動作する場合、ASE Web Services Engine は、HTTP ハンドラ、SOAP ハンドラ、XML マップの 3 つのコンポーネントを使用します。

図 2-2: プロデューサとしての ASE Web Services Engine



HTTP ハンドラ

HTTP ハンドラは HTTP 1.1 をサポートします。これは、HTTP POST メソッドと GET メソッドを使用して送信される要求を受信します。HTTP ハンドラは、SSL 接続もサポートします。

注意 GET HTTP 要求は使用しないでください。これらのコマンドはすべての引数を URL 内に埋め込むため、コマンドは暗号化できません。POST HTTP を使用すると、すべての引数は HTTP 要求の本文に移動し、内容全体を暗号化できます。

SOAP ハンドラ

SOAP ハンドラは SOAP 1.2 をサポートします。これは、SOAP 要求を処理します。SOAP ハンドラは、Web サービスを記述する WSDL ファイルも生成します。

XML マップ

XML マップは、JDBC を介して Adaptive Server Enterprise から返されるリレーショナル・データを、SQLX 標準に準拠した XML にコード化します。XML マップは、データを記述するために DTD と XML スキーマも生成します。

プロデューサ Web メソッド

ASE Web Services Engine には次のメソッドが用意されています。

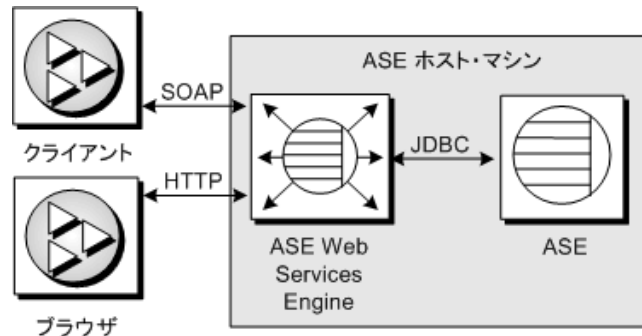
- `execute` – SQL 文またはストアド・プロシージャを実行します。
- `login` – Adaptive Server Enterprise への永続的な接続を確立します。
- `logout` – Adaptive Server Enterprise の接続を明示的に終了します。

これらの Web メソッドの使用方法については、「[第 4 章 ASE Web Services の使用](#)」を参照してください。

ユーザ定義 Web サービス

ASE Web Services Engine で提供されている Web メソッドに加えて、ASE Web Services を利用すると、Web サービスを作成でき、Web ブラウザまたは SOAP クライアントを使用して、Adaptive Server Enterprise で SQL コマンドを実行することができます。

図 2-3: ユーザ定義 Web サービス



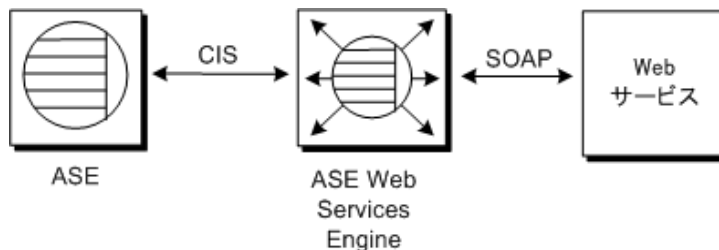
ユーザ定義 Web サービスは、`create service` コマンドを使用して作成できます。実行する SQL を指定したり、`grant` コマンドでパーミッションを制御できる優れたオブジェクトを作成したり、Web ブラウザまたは SOAP クライアントでサービスを呼び出せるかどうかを制御したりできます。ASE Web Services Engine によって、ユーザ定義 Web サービス用の WSDL が自動的に生成されます。ユーザ定義 Web サービスの作成と使用の詳細については、「[第 4 章 ASE Web Services の使用](#)」を参照してください。

注意 GET HTTP 要求は使用しないでください。これらのコマンドはすべての引数を URL 内に埋め込むため、コマンドは暗号化できません。POST HTTP を使用すると、すべての引数は HTTP 要求の本文に移動し、内容全体を暗号化できます。

Web サービスのコンシューマ

Adaptive Server Enterprise による Web メソッドへのアクセスと実行を有効にすると、ASE Web Services Engine は Web サービスのコンシューマとして動作します。Web メソッドは、Web メソッド用の WSDL ファイル内の情報を使用して Adaptive Server Enterprise プロキシ・テーブルに Web メソッドをマップするとアクセス可能になります。その後、Web メソッドは、プロキシ・テーブルで select を使用して呼び出すことができます。

図 2-4: リモート Web サービスへのアクセス

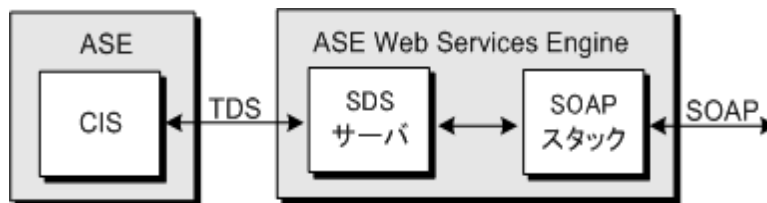


注意 Web Services には、ファイアウォールの内部にあるか外部にあるかを問わずアクセスできます。

コンシューマ・コンポーネント

Web サービスのコンシューマとして動作する場合、ASE Web Services Engine は Specialty Data Store (SDS) サーバと WSDL パーサを使用します。SDS は、Web メソッドをプロキシ・テーブルにマップするための、コンポーネント統合サービス (CIS : Component Integration Service) として使用されます。プロキシ・テーブルは、Apache Axis で解析された WSDL ファイルを使用して構築されます。

図 2-5: コンシューマとしての ASE Web Services Engine



SDS サーバ

Adaptive Server Enterprise は、Web メソッド・プロキシ・テーブル用の `select` 文を受信すると、Tabular Data Stream™ (TDS) を使用して ASE Web Services Engine に要求を転送します。CIS のサーバとして動作する SDS サーバによって、ASE Web Services Engine は Adaptive Server Enterprise からの TDS を傍受して処理できるようになります。

SOAP スタック

SOAP スタックは、XML にコード化されたデータの直列化と転送をまとめて処理する、レイヤ化された機能セットです。SOAP スタックは Web メソッド用の WSDL ファイルを使用して、対応するプロキシ・テーブルの構造を決定します。また、SOAP スタックは、Adaptive Server Enterprise に送信された `select` 文に対応する SOAP 要求を生成し、この SOAP 要求を SOAP サーバに送信します。SOAP スタックは、RPC/encoded と document/literal の両方の Web メソッドをサポートしています。

RPC/encoded メソッド

RPC/encoded Web メソッドの SOAP メッセージには、各メソッド・パラメータの XML 要素が含まれています。RPC/encoded Web メソッドのメッセージは、SOAP 仕様に従ってコード化されます。RPC/encoded Web メソッドを表すプロキシ・テーブルには、各入力パラメータと出力パラメータのカラムが含まれています。

注意 RPC/encoded Web メソッドに入力パラメータまたは出力パラメータがない場合、プロキシ・テーブルにマッピングできません。パラメータを持たない Web メソッドのプロキシ・テーブルには、カラムはありません。カラムのないテーブルを Adaptive Server Enterprise で作成することはできません。

注意 現在、カラムにマッピングできるのは、単純な型に限られています。RPC/encoded Web メソッドで複雑な型または配列を使用すると、Web メソッドをプロキシ・テーブルにマッピングできなくなります。

document/literal メソッド

document/literal Web メソッドの SOAP メッセージでは、通信者が、WSDL ファイルに組み込まれている XML スキーマに従って転送およびフォーマットされるデータを指定します。document/literal Web メソッドのメッセージは、Web メソッド用の WSDL ファイルに従って直列化および非直列化されます。document/literal Web メソッドを表すプロキシ・テーブルには、`_inxml` と `outxml` の 2 つのカラムが含まれています。

相互運用性の実用化に向けたガイドラインを定義している Web Services Interoperability (WSI) Organization は、移植性を向上するため document/literal Web メソッドを使用することを推奨しています。Sybase では、この推奨を支持しています。

プロキシ・テーブル

Web Services のプロキシ・テーブルは Web メソッドを指すので、型は `procedure` であり、この型に対するすべての制約を受けます。

- コマンド – 型が `procedure` のテーブルに対して `delete`、`update`、または `insert` コマンドを発行できません。また、`create index`、`truncate table`、または `alter table` コマンドも発行できません。
- ジョイン – Web Services のプロキシ・テーブルがジョインできるのは、Web Services のプロキシ・テーブル以外の、別の1つのテーブルです。
- クエリ – アンダースコア (`_`) で始まるカラム名は、入力パラメータを指定するために使用されます。このカラムは、パラメータ・カラムとして参照されます。これは、`select` 文の `where` 句内にある必要があります。

`procedure` テーブルの制約の詳細については、『コンポーネント統合サービス・ユーザーズ・ガイド』を参照してください。

設定

ここでは、Adaptive Server での Web Services の設定方法について説明します。

この章の内容は、次のとおりです。

トピック名	ページ
設定	21
ライセンス	23
設定ファイル	23
セキュリティ	24

ASE Web Services は、Adaptive Server Enterprise のインストールの一環としてインストールされます。ASE Web Services は、Adaptive Server Enterprise のインストール中に設定ウィザードを使用して設定することも、インストール後に Sybase Central から設定することもできます。Adaptive Server Enterprise のインストール方法については、Adaptive Server Enterprise の『インストール・ガイド』を参照してください。

注意 特に指定しない限り、この章および後続の章で示すディレクトリは、UNIX の場合は `$SYBASE/WS-15_0` ディレクトリ、Windows の場合は `%SYBASE%\WS-15_0` ディレクトリの下にあると想定します。

設定

ASE Web Services の設定を行う設定ユーティリティは、ASE Web Services のインストール中またはインストール後に使用できます。

- **インストール中の設定** – ASE Web Services 機能のカスタム・インストール時に、ASE Web Services を設定できます。
- **インストール後の設定** – Web Services 機能をインストール中に設定しない場合は、インストール完了後に Sybase Central から機能を設定してください。Sybase Central には、インストール手順で使用する設定ユーティリティに類似した GUI 設定ユーティリティがあります。Web Services 機能は、サイレント・モードまたはコマンド/コンソール・モードでも設定できます。

インストール中の設定

Web Services を設定するには、Web Services 機能のインストール中に設定ユーティリティをアクティブにし、ウィザードの手順に従います。

インストール後の設定

ASE Web Services をすでにインストールしてある場合は、Sybase Central の Adaptive Server Enterprise plug-in から設定ユーティリティを起動します。Web Services の設定には、インストール GUI を使用したり、サイレント・モードまたはコマンド/コンソール・モードを使用したりできます。

❖ Sybase Central からの設定ユーティリティ GUI の起動

- 1 Sybase Central の Adaptive Server Enterprise プラグインの [フォルダ] ビューにある [ユーティリティ] フォルダをクリックします。
- 2 [フォルダ] ビューの右側の詳細ビューに [Web Services の設定] アイコンがあります。[Web Services の設定] のアイコンを選択し、設定ユーティリティを起動します。
- 3 ウィザードの手順に従います。

❖ コマンド/コンソール・モードでの設定ユーティリティの起動

- 1 コンソール・ウィンドウを開きます。
- 2 次のコマンドを入力します。

```
aseplugin -I
```
- 3 コンソールに表示される指示に従います。

❖ サイレント・モードでの設定ユーティリティの起動

- 1 *myres.properties* ファイルのエントリを編集して、希望の値を指定します。*myres.properties* エントリに “=” とプロパティ値を追加して、プロパティを設定します。*myres.properties* ファイルの内容の詳細については、「[付録 B 設定プロパティ](#)」の *myres.properties* を参照してください。
- 2 コンソール・ウィンドウを開きます。
- 3 次のコマンドを入力します。

```
aseplugin -s path
```

ここで、*path* は *myres.properties* ファイルへのパスです。

サイレント・モードでは、Web Services の設定にその他のユーザ・アクションは必要ありません。

ライセンス

ASE Web Services のライセンス入力、Adaptive Server Enterprise の InstallShield インストールによって処理されます。

SySAM から ASE Web Services ライセンスを借り出すには、ASE Web Services Engine を実行する前に、次の手順に従います。

- 1 Adaptive Server Enterprise で isql セッションを確立します。
- 2 isql で次のコマンドを入力して、ASE Web Services をアクティブにします。

```
sp_configure 'enable webservices', 1
```

設定ファイル

props ディレクトリには次の設定ファイルが含まれています。

- [ws.properties](#) ファイル
- [logging.properties](#) ファイル
- [wsmsg.properties](#) ファイル

ws.properties ファイル

ws.properties ファイルには、ASE Web Services の設定が含まれています。これらの設定プロパティの詳細については、「[付録 B 設定プロパティ](#)」を参照してください。

logging.properties ファイル

logging.properties ファイルは、ロギング出力の送信先を定義します。出力は、ログ・ファイルまたはコンソールに送信できます。出力をログ・ファイルに送信する場合は、新しいログ・ファイルにいつロールオーバーするかを指定できます。また、ロギング・メッセージのフォーマットも制御できます。ログのロールオーバー・ポリシーについては、「[ログ・ファイルのロールオーバー](#)」(52 ページ)を参照してください。

logging.properties ファイルのエントリとロギング動作は、Apache log4j パッケージの場合と同じです。詳細については、<http://jakarta.apache.org/log4j/docs/documentation.html> で log4j のマニュアルを参照してください。

wsmmsg.properties ファイル

wsmmsg.properties ファイルは、ASE Web Services での内部使用が目的です。

セキュリティ

ASE Web Services の安全なオペレーションを保証するために、次のことを行うことをおすすめします。

- ASE Web Services を Adaptive Server Enterprise と同じマシンにインストールします。
- SSL を使用して ASE Web Services Engine に接続します。SSL の設定方法については、「[SSL の設定](#) (24 ページ) を参照してください。

ASE Web Services は、Adaptive Server Enterprise でサポートされるすべての認証方式をサポートしています。

注意 ASE Web Services のセキュリティに関する最新の情報は、Adaptive Server Enterprise の『リリース・ノート』を参照してください。

SSL の設定

注意 インストール中に InstallShield からデフォルトで“sybase”として2つの証明書パスワードが作成されます。これらは後で Sybase Central から変更できます。

SSL は、InstallShield または Sybase Central から設定ユーティリティを使用して自動的に設定されます。ただし、SSL は手動で設定することもできます。ASE Web Services の SSL を手動で設定するには、*bin* ディレクトリにある *configssl* スクリプトを次のように実行します。

```
configssl -d <domain_hostName> -k <keystore>  
-h <httpsPort> -f <property_file>  
-c <certificate_password> -s <keystore_password>
```

各パラメータの意味は次のとおりです。

- *domain_hostName* は、SSL を使用して接続する URL のホスト名です。たとえば、次の URL の *domain_hostName* は、mydomainhostname です。

```
http://mydomainhostname:8183/services/ase
```

このパラメータ値のデフォルトはありません。

- *keystore* は、証明書を保存するロケーションとファイルです。デフォルトのロケーションは、UNIX の場合は `$$SYBASE/WS-15_0/props/keystore`、Windows の場合は `%SYBASE%\WS-15_0\props\keystore` です。
- *httpsPort* は、SSL 接続を受信するポートです。デフォルトは 8182 です。
- *property_file* は、更新する properties ファイルのロケーションと名前です。デフォルトのロケーションは、UNIX の場合は `$$SYBASE/WS-15_0/props/ws.properties`、Windows の場合は `%SYBASE%\WS-15_0\props\ws.properties` です。
- *certificate_password* は、証明書のパスワードです。このパラメータ値のデフォルトはありません。スクリプトが呼び出されたときにパスワードが入力されていない場合は、値の入力を求めるプロンプトが表示されます。
- *keystore_password* は、キーストアのパスワードです。このパラメータ値のデフォルトはありません。スクリプトが呼び出されたときにパスワードが入力されていない場合は、値の入力を求めるプロンプトが表示されます。

注意 SSL 用に独自の証明書を追加することもできます。独自の証明書の追加方法については、*keystore* ファイルを操作する JRE で **keytool** ユーティリティのマニュアルを参照してください。ASE Web Services に付属している JRE はバージョン 1.4 です。

Microsoft .NET の証明書のインストール

Microsoft .NET クライアントで SSL を使用して ASE Web Services Engine にアクセスするには、証明書が必要です。Microsoft .NET の証明書をインストールするには、次の手順を実行します。

❖ Microsoft .NET の証明書のインストール

- 1 ASE Web Services Engine を SSL を使用して起動します。ASE Web Services Engine の起動方法については、「[ASE Web Services Engine の起動と停止](#)」(29 ページ)を参照してください。

- 2 Microsoft Internet Explorer の [アドレス] バーに、次のように入力します。

```
https://<producer_host>:<SSL_port>
```

各パラメータの意味は次のとおりです。

- *producer_host* は、ASE Web Services Engine を実行するホストです。
- *SSL_port* は、ASE Web Services Engine 用のポートです。

[セキュリティの警告] ダイアログ・ボックスが表示されます。

- 3 [証明書の表示] をクリックします。[証明書] ダイアログ・ボックスが表示されます。

- 4 [証明書のインストール] をクリックします。[証明書マネージャのインポート] ウィザードが起動します。
- 5 [証明書マネージャのインポート] ウィザードで証明書が正常にインストールされたことが示され、[証明書] ダイアログ・ボックスに戻るまで、[次へ] をクリックします。
- 6 [OK] をクリックします。ブラウザが [セキュリティの警告] ダイアログ・ボックスに戻ります。
- 7 [はい] をクリックします。ブラウザ・ウィンドウに、「Welcome to the ASE Web Services」 ページが表示されます。

❖ **証明書のインストールの確認**

- 1 ブラウザ・ウィンドウをすべて閉じます。
- 2 Microsoft Internet Explorer を再起動します。
- 3 Microsoft Internet Explorer の [アドレス] バーに、次のように入力します。

```
https://<producer_host>:<SSL_port>
```

各パラメータの意味は次のとおりです。

- *producer_host* は、ASE Web Services Engine を実行するホストです。
- *SSL_port* は、ASE Web Services Engine 用のポートです。

[セキュリティの警告] ダイアログ・ボックスは表示されません。

第 3 部

管理

ここでは、製品の機能と管理作業について説明します。

この章の内容は、次のとおりです。

トピック名	ページ
ASE Web Services Engine の使用	29
ユーザ定義 Web サービスの使用	42
ASE Web Services のログイン	51
Sybase Central を使用する場合	52

「[第 3 章 ASE Web Services の設定](#)」の設定作業が完了していることを確認してから、ASE Web Services を使用してください。

ASE Web Services Engine の使用

この項の内容は、次のとおりです。

- [ASE Web Services Engine の起動と停止](#)
- [ASE Web Services のメソッド](#)
- [sp_webservices の使用方法](#)
- [Web サービスの呼び出し](#)

ASE Web Services Engine の起動と停止

ASE Web Services の ASE Web Services Engine を起動するには、*bin* ディレクトリにある *runws* スクリプトを実行します。

```
runws -U <ase_username> -P <ase_password>
-S <ase_server_name> -f <property_file> -v
```

ASE Web Services の ASE Web Services Engine を停止するには、*bin* ディレクトリにある *stopws* スクリプトを実行します。

```
stopws -U <ase_username> -P <ase_password>
-S <ase_server_name> -f <property_file> -v
```

runws スクリプトと *stopws* スクリプトは同じパラメータを取ります。

- *ase_username* は、Adaptive Server Enterprise のユーザ名です。このパラメータ値のデフォルトはありません。このパラメータに値を指定していない場合、指定を求めるプロンプトが表示されます。
- *ase_password* は、Adaptive Server Enterprise のパスワードです。このパラメータ値のデフォルトはありません。このパラメータに値を指定していない場合、指定を求めるプロンプトが表示されます。
- *ase_server_name* は、Web サービスの名前です。このパラメータ値のデフォルトはありません。このパラメータに値を指定していない場合、指定を求めるプロンプトが表示されます。
- *property_file* は、更新する *properties* ファイルのロケーションと名前です。デフォルトのロケーションは、UNIX の場合は `$$SYBASE/WS-15_0/props/ws.properties`、Windows の場合は `%SYBASE%\WS-15_0\props\ws.properties` です。
- *-v* は、ASE Web Services Engine が起動またはシャットダウン時にバージョン情報を表示するように指定します。

条件

ASE Web Services Engine は、次の条件を満たす場合に起動または停止します。

- 提供された *ase_server_name* が、*libtcl.cfg* ファイルが指す LDAP サーバ上または Adaptive Server Enterprise の *interfaces* ファイル内にある。

ASE Web Services は、*libtcl.cfg* ファイルが指す LDAP サーバ上で、*ase_server_name* の値があるエントリを最初に検索します。ASE Web Services は、*ws.properties* ファイルの `com.sybase.ase.ws.libtcl` エントリを使用して *libtcl.cfg* ファイルを検索します。LDAP サーバにエントリが見つからない場合、ASE Web Services は、Adaptive Server Enterprise の *interfaces* ファイル内のエントリを検索します。

注意 Windows システムでは、*interfaces* ファイルの名前が *sql.ini* になります。

ASE Web Services は、*ws.properties* ファイルの `com.sybase.ase.ws.interfaces` エントリを使用して *interfaces* ファイルを検索します。

- 提供された *ase_username* と *ase_password* を使用したログインが成功する。

注意 Adaptive Server Enterprise ユーザのパスワードには null 文字列を設定できません。sa のログインでは、デフォルトで null 文字列のパスワードが許可されます。しかし、null パスワードの使用はおすすめできません。

- Adaptive Server Enterprise のログインアカウントに `sa` ロールがある。
- Adaptive Server Enterprise の `isql` で、次のストアド・プロシージャ・コマンドを実行した。

```
sp_configure 'enable webservices', 1
```

検証

`runws` スクリプトが正常に実行されてから、ASE Web Services が有効で ASE Web Services Engine が動作していることを確認します。

❖ ASE Web Services が有効であることの検証

ASE Web Services が有効であることを検証するには、次の手順に従います。

- Adaptive Server Enterprise で次のコマンドを実行します。

```
sp_configure 'enable webservices'
```

`sp_configure` の戻り値が 1 である場合は、Web Services 機能が有効になっています。戻り値が 0 である場合は、機能が有効になっていません。

❖ ASE Web Services Engine が動作していることの確認

- `logs` ディレクトリの `producer.log` ファイルまたは `consumer.log` ファイルで、ASE Web Services Engine が動作していることを示すメッセージを確認します。次に例を示します。

```
2004-03-29 16:29:29.522 INFO [main] - Starting HTTP Server on Port:8181
```

SSL の場合、ログには HTTPS ポートおよび関連する SSL 情報が示されます。次に例を示します。

```
2004-03-29 16:29:29.532 INFO [main] - Https Port [8182], KeyPassword:...
```

注意 `runproducer`、`stopproducer`、`runconsumer`、`stopconsumer` の各スクリプトは、以前のリリースの ASE Web Services との互換性を保つために 15.0 のリリースでも残してあります。しかし、15.0 のリリースでは、これらのスクリプトは `runws` スクリプトおよび `stopws` スクリプトを呼び出します。

ASE Web Services のメソッド

ASE Web Services にアクセスするには、クライアントは、ASE Web Services Engine によって公開されたメソッドを使用する必要があります。これらのメソッドは、SOAP で `rpc` としてマップされます。

```
<soap:binding style="rpc" ...>
```

メッセージのデータはコード化されます。

```
<soap:body use="encoded" ....>
```

ASE Web Services Engine には次のメソッドが用意されています。

- **execute** – SQL 文またはストアド・プロシージャを実行します。
- **login** – Adaptive Server Enterprise への永続的な接続を確立します。
- **logout** – Adaptive Server Enterprise の接続を明示的に終了します。

これらのメソッドは Adaptive Server Enterprise においてデフォルトでサポートされており、1つの Web サービス (1つの WSDL ファイル) として提供されています。これらのメソッドの構文は、HTTP と SSL のどちらを使用して呼び出された場合でも同じです。

execute

execute メソッドは、Adaptive Server Enterprise で Transact-SQL 文またはストアド・プロシージャを実行します。

構文

```
execute aseServerName userName password sqlxOptions sql
```

パラメータ

- *aseServerName*

interfaces ファイル内の Adaptive Server Enterprise サーバ、または LDAP サーバの名前を示す SOAP string です。

execute メソッドを呼び出すたびに、ASE Web Services は、ASE Web Services Engine を起動または停止するときに *ase_server_name* を使用するのと同様に、*aseServerName* の値を使用します。詳細については、「[ASE Web Services Engine の起動と停止](#)」(29 ページ)を参照してください。

- *userName*

Adaptive Server Enterprise へのログインに必要なユーザ ID を示す SOAP string です。

- *password*

Adaptive Server Enterprise へのログインに必要なパスワードを示す SOAP string です。

- *sqlxOptions*

1つ以上の **option** パラメータを示す SOAP string です。このパラメータは、SQLX 結果セットの特性を指定します。有効な option パラメータは、次のとおりです。

- **general**
- **binary**={hex | base64}
- **columnstyle**={element | attribute}

- entitize={yes | no | cond}
- format={yes | no}
- header={yes | no}
- multipleentitize={yes | no}
- multiplieresults={all | data}
- ncr={non_ascii | no}
- nullstyle={attribute | omit}
- prefix="value"
- root={yes | no}
- rowname="value"
- schemaloc="value"
- statement={yes | no}
- tablename="value"
- targetns="value"
- xsidecl={yes | no}

value には値を指定してください。SQLX の関数とオプションの詳細については、『Adaptive Server Enterprise における XML Services』を参照してください。

- *sql*

Adaptive Server Enterprise で実行する SQL 文またはストアド・プロシージャを示す SOAP string です。*sql* パラメータで指定されている SOAP string のサイズは、*ws.properties* ファイルの *com.sybase.ase.ws.maxpostsize* プロパティの設定値によって制限されます。このプロパティおよびその他のプロパティについては、「付録 B 設定プロパティ」を参照してください。

例 1

この例では、Adaptive Server Enterprise のバージョン番号を調べます。

```
execute johndoe-sun sa password "tablename=ws" "select
@@version"
```

この例では、Web メソッドを直接呼び出します。ASE Web Services は、XML スキーマ、DTD、文の実行結果が含まれている結果セットを返します。

例 2 この例では、pubs2 データベース内のテーブルで左ジョインを計算します。

```
execute johndoe-sun sa password "tablename=ws"  
"select title, price, au_fname, au_lname from (titles  
left join titleauthor on titles.title_id =  
titleauthor.title_id ) left join authors on  
titleauthor.au_id = authors.au_id and titles.price >  
$15.00"
```

login

login メソッドは、Adaptive Server Enterprise への永続的な接続を確立します。

構文 `login aseServerName userName password`

パラメータ

- *aseServerName*

SQL 文またはストアド・プロシージャを実行する Adaptive Server Enterprise の名前を示す SOAP string です。

login メソッドでは、ASE Web Services は execute メソッドの場合と同様の方法で、*aseServerName* の値を使用します。

- *username*

Adaptive Server Enterprise へのログインに必要なユーザ ID を示す SOAP string です。

- パスワード

Adaptive Server Enterprise へのログインに必要なパスワードを示す SOAP string です。

使用法

まず接続を確立してから、Adaptive Server Enterprise で SQL 文またはストアド・プロシージャを実行してください。ただし、login メソッドはオプションです。login メソッドを呼び出さずに execute メソッドを呼び出すと、ASE Web Services は Adaptive Server Enterprise への非永続的な接続を自動的に確立します。login メソッドは、Adaptive Server Enterprise への永続的な接続を開始します。接続は、logout メソッドで終了します。永続的な接続は、アクティブでない状態が 60 秒間続くと自動的に終了します。

logout

logout メソッドは、Adaptive Server Enterprise への永続的な接続を終了します。

構文 `logout`

使用法

logout メソッドは、login メソッドで確立した Adaptive Server Enterprise への永続的な接続を終了します。

sp_webservices の使用方法

sp_webservices ストアド・プロシージャは、ASE Web Services Engine で使用するプロキシ・テーブルを作成および管理します。この項では、sp_webservices のオプションとパラメータについて説明します。

sp_webservices ストアド・プロシージャには次のオプションがあります。

- **add** – プロキシ・テーブルを作成します。
- **help** – sp_webservices の使用方法を表示します。
- **list** – WSDL ファイルにマップされるプロキシ・テーブルをリストします。
- **modify** – タイムアウト設定を変更します。
- **remove** – WSDL ファイルにマップされるプロキシ・テーブルを削除します。

この他にも、ユーザ定義 Web サービスで使用する sp_webservices オプションがあります。これらのオプションの詳細については、「[ユーザ定義 Web サービスでの sp_webservices の使用](#)」(46 ページ) を参照してください。

注意 Web Services のプロキシ・テーブルの制約については、「[第 6 章 トラブルシューティング](#)」の [Web Services のプロキシ・テーブルの制約](#) を参照してください。

add

add オプションは、WSDL ファイルで指定された Web メソッドのプロキシ・テーブルを作成するために使用します。add オプションが正常に実行されると、list オプションが自動的に呼び出され、新しいプロキシ・テーブルのスキーマが示されます。

構文

```
sp_webservices 'add', 'wsdl_uri' [, sds_name]
[, 'method_name=proxy_table
[,method_name=proxy_table ]* ' ]
```

パラメータ

- *wsdl_uri*
新しいプロキシ・テーブルにマップされる WSDL ファイルのロケーションです。このパラメータを指定すると、Web Services は、syswsdl テーブルに URI が存在することを確認します。
- *sds_name*
interfaces または *sql.ini* ファイルで ASE Web Services Engine 用に指定された名前です。デフォルト値は **ws** です。sysattributes テーブルにエントリがない場合は、エラーが発生します。

- *method_name*

プロキシ・テーブルにマップされる Web メソッドの名前です。指定する *method_name* は、関連する WSDL ファイルで指定された Web メソッドの名前にします。

- *proxy_table*

method_name で指定された Web メソッドがマップされるプロキシ・テーブルの名前です。

使用法

Web メソッドの *method_name* 値と *proxy_table* 値を指定しない場合、その Web メソッド用に生成されるプロキシ・テーブルは、デフォルトで、WSDL ファイルに指定した Web メソッドの名前になります。この Web メソッドの名前を持つプロキシ・テーブルがすでにある場合は、次のような名前の新しいプロキシ・テーブルが生成されます。

method_nameN

ここで、*method_name* はデフォルトのプロキシ・テーブル名、*N* は Web メソッドの一連の各マッピングを示す 1～9 の数字です。プロキシ・テーブルの複製は 99 個まで可能です。

Web メソッドの *method_name* 値と *proxy_table* 値を指定する場合、プロキシ・テーブル名は新しい名前にします。*proxy_table* で指定した名前を持つプロキシ・テーブルがすでにある場合はエラーが発生し、**add** オプションで指定した Web メソッドがプロキシ・テーブルにマップされません。

add オプションからの出力には、プロキシ・テーブルに正常にマップされたメソッドとマップされなかったメソッドがリストされます。マップされなかった Web メソッドのプロキシ・テーブルの名前は、**add** オプションの出力で NULL として示されます。

注意 入力と出力に使用されるカラムは、RPC/encoded Web メソッド用に生成されたプロキシ・テーブルと document/literal Web メソッド用に生成されたプロキシ・テーブルでは異なります。RPC/encoded Web メソッドを表すプロキシ・テーブルには、各入力パラメータと出力パラメータのカラムが含まれています。document/literal Web メソッドを表すプロキシ・テーブルには、*_inxml* と *outxml* の 2 つのカラムが含まれています。

注意 データ型のマッピングの詳細については、「[付録 C SOAP と Adaptive Server Enterprise のデータ型のマッピング](#)」を参照してください。

help

help オプションは、`sp_webservices` ストアド・プロシージャの使用法と使用例を示します。

構文

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

パラメータ

- *option*
詳細な方法の表示対象となるオプションです。有効な値は、`add`、`list`、`remove`、`modify` です。

使用法

option の値を指定せずに **help** オプションを使用すると、`add`、`addalias`、`deploy`、`dropalias`、`list`、`listalias`、`listudws`、`modify`、`remove`、`undeploy` オプションに関する構文の簡単な説明が出力されます。

list

list オプションは、WSDL ファイルに記述された Web メソッドをリストするために使用します。

構文

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

パラメータ

- *wsdl_uri*
マップされる WSDL ファイルの URI です。*wsdl_uri* の値が指定されない場合、**list** オプションは、プロキシ・テーブルにマップされたすべての Web メソッドに関する情報を表示します。
- *sds_name*
interfaces または *sql.ini* ファイルで ASE Web Services Engine 用に指定された SDS サーバの名前です。デフォルト値は `ws` です。`sysattributes` テーブルにエントリがない場合は、エラーが発生します。

wsdl_uri パラメータと *sds_name* パラメータの両方を指定しない場合は、`sysattributes` テーブル内のすべてのエントリが `wsdlid` の順にリストされます。

使用法

WSDL ファイルに記述された Web メソッドがすでにプロキシ・テーブルにマップされている場合、**list** オプションは各プロキシ・テーブルに関する情報を出力します。WSDL ファイルに記述された Web メソッドがプロキシ・テーブルにマップされていない場合、**list** オプションはプロキシ・テーブルの作成に使用できる SQL を出力します。

modify

modify オプションは、WSDL ファイルの属性情報を変更するために使用します。

構文

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

パラメータ

- *wsdl_uri*
属性情報を変更する WSDL ファイルの URI です。
- *time*
オペレーションがアポートされる前に Web メソッドが応答する秒単位の時間間隔です。

remove

remove オプションは、Web メソッドのプロキシ・テーブル・マッピングを削除するために使用します。

構文

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

パラメータ

- *wsdl_uri*
プロキシ・テーブルを削除する WSDL ファイルの URI です。
- *sds_name*
interfaces または *sql.ini* ファイルで ASE Web Services Engine 用に指定された SDS サーバの名前です。デフォルト値は **ws** です。**sysattributes** テーブルにエントリがない場合は、エラーが発生します。

Web サービスの呼び出し

ASE Web Services Engine を使用して Web サービスを呼び出すには、次の手順を実行します。

❖ Web サービスの呼び出し

- 1 ASE Web Services Engine を起動します。
- 2 `sp_webservices` の `add` オプションを使用して、Adaptive Server Enterprise のプロキシ・テーブルに Web サービスをマップします。
- 3 `sp_help` を使用して、Web メソッドの呼び出しに必要な入力パラメータと出力パラメータを確認します。
- 4 プロキシ・テーブルで `select` 文を使用して Web メソッドを呼び出します。

例 1

この例では、RPC/encoded Web メソッドを呼び出して 2 つの通貨間の為替レートを表示します。

`sp_webservices` の `add` オプションを使用して、プロキシ・テーブルに Web メソッドをマップします。

```
1> sp_webservices 'add', 'http://www.xmethods.net/sd/2001/CurrencyExchangeService.wsdl'  
2> go
```

Web メソッド `getRate` は、同じ名前のプロキシ・テーブルにマップされます。

プロキシ・テーブルからこの Web メソッドを選択して呼び出します。

```
1> select * from getRate where _country1 = 'usa' and _country2 = 'india'  
2> go
```

この `select` から返された結果には、パラメータで指定した為替レートが次のように示されます。

```
Result          _country1      _country2  
43.000000      usa            india  
(1 row affected)
```

例 2

この例では、Web メソッドを呼び出して XML ドキュメント内で株式情報を表示します。

`sp_webservices` の `add` オプションを使用して、プロキシ・テーブルに Web メソッドをマップします。

```
1> sp_webservices "add" , "http://www.webservicex.net/stockquote.asmx?WSDL"
2> go
```

Web メソッド `GetQuote` は、同じ名前のプロキシ・テーブルにマップされます。

`GetQuote` プロキシ・テーブルの `outxml` カラムを選択して、この Web メソッドを呼び出します。

```
1> select outxml from GetQuote where _inxml = '<?xml version="1.0" encoding="utf-8"?>
2>     <GetQuote xmlns="http://www.webserviceX.NET/">
3>         <symbol>SY</symbol>
4>     </GetQuote>'
5> go
```

この `select` の結果、株価情報が XML ドキュメントに次のように示されます。

outxml

```
<?xml version="1.0" encoding="UTF-8" ?><GetQuoteResponse
xmlns="http://www.webserviceX.NET/"><GetQuoteResult><StockQuotes><Stock><Symbol>SY</Sym
bol><Last>21.48</Last><Date>7/21/2005</Date><Time>4:01pm</Time><Change>+1.72</Change><O
pen>20.00</Open><High>21.60</High><Low>19.91</Low><Volume>2420100</Volume><MktCap>1.927
B</MktCap><PreviousClose>19.76</PreviousClose><PercentageChange>+8.70%</PercentageChang
e><AnnRange>12.75 - 20.44</AnnRange><Earnings>0.706</Earnings><P-E>27.99</P-E><Name>SYBASE
INC</Name></Stock></StockQuotes></GetQuoteResult></GetQuoteResponse>
```

(1 row affected)

例 3

この例では、前の例でプロキシ・テーブルにマップされた **GetQuote** Web メソッドをビューを介して呼び出し、株式情報を表示します。

この Web サービスを使用するには、株式を示す記号を保持するテーブルを作成してください。

```
1> create table stocksymbol(symbol varchar(100))
2> go
```

stocksymbol テーブルにデータを挿入します。

```
1> insert stocksymbol values("SY")
2> insert stocksymbol values("ORCL")
3> go
```

次に、**GetQuote** Web メソッドを呼び出すビューを作成します。

```
1> CREATE VIEW getstockvw as
2> select Symbol = xmlextract('//Stock/Symbol/text()',outxml returns varchar(5)),
3>    Name = xmlextract('//Stock/Name/text()',outxml returns varchar(20)),
4>    Time = xmlextract('//Stock/Time/text()',outxml returns varchar(10)),
5>    Date = xmlextract('//Stock/Date/text()',outxml returns date),
6>    High = xmlextract('//Stock/High/text()',outxml returns decimal(15,2)),
7>    Low = xmlextract('//Stock/Low/text()',outxml returns decimal(15,2))
8> FROM GetQuote ,stocksymbol
9> WHERE _inxml = '<GetQuote
xmlns="http://www.webserviceX.NET/"><symbol>'+symbol+'</symbol></GetQuote>'
10> go
```

getstockvw ビューから選択して、**GetQuotes** メソッドからの出力を表示します。

```
1> select * from getstockvw
2> go
```

この **select** の結果、株価情報がビュー定義で指定されたパラメータにより次のように示されます。

Symbol	Name	Time	Date	High	Low
SY	SYBASE INC	4:01pm	Jul 21 2005	21.60	19.91
ORCL	ORACLE CORP	4:00pm	Jul 21 2005	14.05	13.54
MSFT	MICROSOFT CP	4:00pm	Jul 21 2005	26.48	26.19

(3 rows affected)

ユーザ定義 Web サービスの使用

この項では、ユーザ定義 Web サービスに特有の機能に関する次の項目について説明します。

- [ユーザ定義 Web サービスのコマンド](#)
- [ユーザ定義 Web サービスでの `sp_webservices` の使用](#)
- [ユーザ定義 Web サービスのセキュリティ](#)
- [ユーザ定義 Web サービスの監査](#)

ユーザ定義 Web サービスのコマンド

次のコマンドを使用するとユーザ定義 Web サービスの作成、削除、変更を行います。

- `create service`
- `drop service`

`create service`

`create service` コマンドは、提供された SQL 文を指定された名前とパラメータを持つストアド・プロシージャでラップします。次の点以外は、結果として得られるストアド・プロシージャは `create procedure` コマンドで作成されたストアド・プロシージャと同じように動作し、既存のストアド・プロシージャに関する実行、複写、`sp_helptext`、再コンパイルのルールに従います。また `isql` から実行できます。

- 結果として得られるストアド・プロシージャは、`drop procedure` コマンドではなく、`drop service` コマンドでのみ削除できる。
- `syscomments` テーブルには、`create service` コマンドを再作成するために必要な DDL が格納される。
- 指定されたサービス名では、ストアド・プロシージャ・グループが作成されない場合がある。

注意 ASE Web Services Engine を介してユーザ定義 Web サービスを使用できるようにするには、`sp_webservices` の `deploy` オプションを使用します。ただし、ユーザ定義 Web サービスのストアド・プロシージャは、展開されていなくても `isql` からアクセスできます。`sp_webservices` の `deploy` オプションの詳細については、[sp_webservices の使用方法の「deploy」\(47 ページ\)](#) を参照してください。

構文	<pre> create service <i>service-name</i> [secure <i>security_options</i>] [, userpath <i>path</i>] [, alias <i>alias-name</i>] type { xml raw soap } [[(@<i>parameter_name</i> datatype [(length) (precision [, scale])] [= default][output] [, @<i>parameter_name</i> datatype [(length) (precision [, scale])] [= default][output]...[...])] as <i>SQL_statements</i> <i>security_options</i> ::= (<i>security_option_item</i> [<i>security_option_item</i>]) </pre>
パラメータ	<ul style="list-style-type: none"> • <i>service-name</i> – ユーザ定義 Web サービスの名前です。サービス名には、ストアド・プロシージャとして有効な任意の名前を指定できます。drop procedure コマンドがこのサービス名を使用して呼び出されると、対応するストアド・プロシージャが削除されます。既存のサービスの名前を指定すると、例外が発生します。 • <i>security_option_item</i> – clear または ssl のどちらかを指定します。 <ul style="list-style-type: none"> • clear は Web サービスへのアクセスに HTTP が使用されることを示します。 • ssl は Web サービスへのアクセスに HTTPS が使用されることを示します。 • <i>path</i> – Web サービスにアクセスする URL に追加されるユーザ定義パスをリテラル文字列で指定します。このパスはデフォルトでは null です。 • <i>alias-name</i> – ユーザ定義 Web サービスのエイリアスをリテラル文字列で指定します。 • <i>parameter_name</i> – ユーザ定義 Web サービスの引数の名前です。このパラメータの値は Web サービスが実行されるときに渡されます。パラメータ名の前には @ 記号を付けてください。また、識別子の規則に従っていません。これらの条件は、create procedure コマンドの <i>parameter_name</i> パラメータの場合と同じです。 • <i>SQL_statements</i> – ユーザ定義 Web サービスが実行するアクションです。create view、create default、create rule、create procedure、create trigger、use を除き、あらゆる SQL 文をいくつでも指定できます。これらの条件は、create procedure コマンドの <i>SQL_statements</i> パラメータの場合と同じです。 • <i>type</i> – soap、raw、xml のいずれかになります。 <ul style="list-style-type: none"> • soap は、HTTP の POST 要求を意味し、SOAP のすべてのルールに準拠する必要があります。返されるデータは、SQL/XML フォーマットです。 • raw は、出力が何らかの変更や再フォーマットをまったく受けずに送信されることを示します。これは HTTP の GET 要求を意味します。呼び出されたストアド・プロシージャは、出力を厳密に指定できます。

- xml は、結果セットの出力が SQL/XML フォーマットで返されることを示します。これは HTTP の GET 要求を意味します。

注意 ASE スタアド・プロシージャと SOAP ユーザ定義 Web サービス間でのデータ型のマッピングについては、「[付録 C SOAP と Adaptive Server Enterprise のデータ型のマッピング](#)」の `create service` コマンド用の ASE から SOAP へのデータ型マッピングを参照してください。

例 1 この例では、`raw` 型のユーザ定義 Web サービス `rawservice` を作成し、現在のデータベースのバージョンを返すようにします。`pubs2` データベースに対し `isql` コマンド・ラインから `create service` コマンドを次のように入力します。

```
1> use pubs2
2> go
1> create service rawservice type raw as select '<html><h1>' +
@@version + '</h1></html>'
2> go
```

次に、新しく作成したユーザ定義 Web サービスを展開します。

```
1> sp_webservices 'deploy', 'all'
2> go
```

注意 `sp_webservices` の `deploy` オプションの詳細については、「[ユーザ定義 Web サービスでの `sp_webservices` の使用](#)」(46 ページ)を参照してください。

新しく作成したユーザ定義 Web サービスの WSDL は次の URL にあります。

```
http://myhost:8181/services/pubs2?wsdl
```

新しく作成したユーザ定義 Web サービスは次の URL から利用可能です。

```
http://myhost:8181/services/pubs2?method=rawservice&
username=bob&password=bob123
```

ここで、`bob` と `bob123` は、ユーザ定義 Web サービスの作成者のユーザ ID とパスワードです。

出力である Adaptive Server Enterprise のバージョン文字列は、ブラウザ・ウィンドウの HTML `<h1>` タグ内に表示されます。

例 2

この例では、xml 型のユーザ定義 Web サービス `xmlservice` を作成し、現在のデータベースのバージョンを返すようにします。`pubs2` データベースに対し `isql` コマンド・ラインから `create service` コマンドを次のように入力します。

```
1> use pubs2
2> go
1> create service xmlservice userpath "testing" type xml as
select @@version
2> go
```

次に、新しく作成したユーザ定義 Web サービスを展開します。

```
1> sp_webservices 'deploy', 'xmlservice'
2> go
```

注意 `sp_webservices` の `deploy` オプションの詳細については、「[ユーザ定義 Web サービスでの `sp_webservices` の使用](#)」(46 ページ)を参照してください。

ユーザ定義 Web サービスの WSDL は次の URL にあります。

```
http://myhost:8181/services/pubs2/testing?wsdl
```

ユーザ定義 Web サービスはブラウザで次の URL にアクセスして呼び出します。

```
http://myhost:8181/services/pubs2/testing?method=xmlservice&
username=bob&password=bob123
```

ここで、`bob` と `bob123` は、ユーザ定義 Web サービスの作成者のユーザ ID とパスワードです。出力はブラウザ・ウィンドウに XML として表示されます。

例 3

この例では、ユーザ定義 Web サービスを使用して、SOAP クライアントから、ストアド・プロシージャ `sp_who` を実行できるようにします。このストアド・プロシージャは引数を 1 つ取り、オプションで `userpath` トークンを指定します。

```
create service sp_who_service userpath 'myservices/args' type
soap @loginname varchar(30) as exec sp_who @loginname
```

Web サービスは、`pubs2` データベース内に `sp_who_service` という名前で作成され、展開された後に、次の URL からアクセス可能になります。

```
http://localhost:8181/pubs2/myservices/args/sp_who_service
```

サービスの WSDL は次の URL から利用可能です。

```
http://localhost:8181/pubs2/myservices/args?wsdl
```

WSDL ファイルに記述されているこの Web メソッドのシグニチャは、次のようになります。

```
DataReturn[] sp_who_service (xsd:string username, xsd:string
password, xsd:string loginname)
```

新しいサービスは SOAP クライアントから、`varchar(30)` 型の `loginname` というパラメータを 1 つ使用して呼び出されます。

drop service

drop service コマンドは、現在のデータベースからユーザ定義 Web サービスを削除します。メタデータと対応するストアド・プロシージャの両方が削除されます。

注意 ユーザ定義 Web サービスを削除する前に、展開を解除する必要があります。**sp_webservices** の **undeploy** オプションの詳細については、「[ユーザ定義 Web サービスでの sp_webservices の使用](#)」(46 ページ) を参照してください。

構文

```
drop service service-name
```

パラメータ

- **service-name** – ユーザ定義 Web サービスの名前です。サービス名には、ストアド・プロシージャとして有効な任意の名前を指定できます。存在しないサービスの名前を指定すると、例外が発生します。現在別のセッションで使用中のサービスは削除できません。

例

この例では **sp_who_service** という名前のユーザ定義 Web サービスを削除します。

```
drop service sp_who_service
```

ユーザ定義 Web サービスでの sp_webservices の使用

ユーザ定義 Web サービスでは、次の **sp_webservices** オプションを使用できます。

- **addalias** – データベース・エイリアスを作成します。
- **deploy** – ユーザ定義 Web サービスを展開します。
- **dropalias** – データベース・エイリアスを削除します。
- **listudws** – WSDL ファイルにマップされるプロキシ・テーブルをリストします。
- **listalias** – データベース・エイリアスをリストします。
- **undeploy** – ユーザ定義 Web サービスの展開を解除します。

addalias

add alias オプションは、データベース名を表すエイリアスの作成に使用します。

構文

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

- パラメータ**
- *alias_name*
指定したデータベースのエイリアスです。このパラメータは必須です。
 - *database_name*
エイリアスを指定するデータベースの名前です。このパラメータは必須です。
- 使用法**
- エイリアスは、データベース名を表す URL の部分を指定する場合に役に立ち、**create service** コマンドの **userpath** オプションと共に使用することで、エイリアスはユーザ定義 Web サービスへのアクセスに使用される URL を完全に制御することができます。

deploy

deploy オプションは、ユーザ定義 Web サービスを展開し、HTTP または HTTPS を介して ASE Web Services Engine からアクセス可能にするために使用されます。

注意 GET HTTP 要求は使用しないでください。これらのコマンドはすべての引数を URL 内に埋め込むため、コマンドは暗号化できません。POST HTTP を使用すると、すべての引数は HTTP 要求の本文に移動し、内容全体を暗号化できます。

構文

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

パラメータ

- **all**
すべてのユーザ定義 Web サービスを現在のデータベース用に展開することを指定します。
- *service_name*
展開するユーザ定義 Web サービスの名前です。

使用法

deploy および **undeploy** オプションは、ユーザ定義 Web サービスを利用可能にするかどうかを制御するために使用します。このオプションを使用するには、**webservices_role** 権限が必要です。

all パラメータが指定されると、ASE Web Services Engine はユーザ定義 Web サービスの内部キャッシュを削除し、ユーザ定義 Web サービスに関するすべてのメタデータを Adaptive Server Enterprise から再度読み込みます。

注意 現在展開中のユーザ定義 Web サービスの削除または名前変更はできません。

dropalias

dropalias オプションは、データベース名を表すエイリアスの削除に使用します。

構文 `sp_webservices 'dropalias' alias_name`

パラメータ

- *alias_name*
削除するエイリアスです。

使用法 展開したユーザ定義 Web サービスによって参照されている場合、エイリアスは削除できません。エイリアスを削除するには、そのエイリアスを参照しているユーザ定義 Web サービスの展開を解除する必要があります。

listudws

listudws オプションは、現在のデータベースのユーザ定義 Web サービスをリストするために使用します。

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

パラメータ

- *service_name*
リストするユーザ定義 Web サービスの名前です。

使用法 *service_name* パラメータが指定されていないと、すべてのユーザ定義 Web サービスがリストされます。

listalias

listalias オプションは、すべてのエイリアスをリストするために使用します。

構文 `sp_webservices 'listalias'`

使用法 すべてのエイリアスがリストされます。

undeploy

undeploy オプションは、ユーザ定義 Web サービスが HTTP または HTTPS を介して ASE Web Services Engine からアクセスできないようにするために使用されます。

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

パラメータ	<ul style="list-style-type: none"> • <code>all</code> すべてのユーザ定義 Web サービスを現在のデータベースから展開解除することを指定します。 • <code>service_name</code> 展開解除するユーザ定義 Web サービスの名前です。
使用法	<code>deploy</code> および <code>undeploy</code> オプションは、ユーザ定義 Web サービスを利用可能にするかどうかを制御するために使用します。このオプションを使用するには、 <code>webservices_role</code> 権限が必要です。

ユーザ定義 Web サービスのセキュリティ

Web Services 機能向けの Adaptive Server Enterprise に、システム役割 `webservices_role` が追加されました。この役割は、`sp_webservices` の `deploy` および `undeploy` オプションを使用するために必要になります。ユーザ定義 Web サービスを実行するには、対応するストアド・プロシージャを実行するための有効なログインと権限が必要です。

ユーザ定義 Web サービスの作成、削除、実行を行うには、Adaptive Server Enterprise でストアド・プロシージャを作成、削除、実行するために必要な権限と同じ権限が必要です。`grant` および `revoke` コマンドを使用して適切な権限を設定する方法の詳細については、Adaptive Server Enterprise の『システム管理ガイド』を参照してください。

注意 ASE Web Services のセキュリティに関する最新の情報は、Adaptive Server Enterprise の『リリース・ノート』を参照してください。

ユーザ定義 Web サービスの監査

ユーザ定義 Web サービスは、Adaptive Server Enterprise のストアド・プロシージャをモデルにしています。ユーザ定義 Web サービスの操作において、Adaptive Server Enterprise は既存のストアド・プロシージャ向けの監査範囲を使用して、次のイベントを生成します。

- ユーザ定義 Web サービスの作成 – イベント 11 「プロシージャの作成」を生成
- ユーザ定義 Web サービスの削除 – イベント 28 「プロシージャの削除」を生成
- ユーザ定義 Web サービスの実行 – イベント 38 「ストアド・プロシージャの実行」を生成

既存の監査機能の詳細については、Adaptive Server Enterprise の『システム管理ガイド』を参照してください。

既存の監査機能に加えて、Adaptive Server Enterprise では `sp_webservices` の `deploy` オプションおよび `undeploy` オプション用に2つの監査イベントが用意されています。

監査レコードは `sybsecurity` システム・テーブルに保管されます。Web サービスの監査を有効にするには、次のコマンドを実行します。

```
sp_audit "security", "all", "all", "on"
```

`sp_webservices 'deploy'` の監査

監査イベントの番号 110 は、`sp_webservices` の `deploy` オプションに対応しています。

例 1 この例では、ユーザ `bob` によって `pubs2` データベースに入力された次のコマンドに対する監査テーブル・エントリを示します。

```
sp_webservices 'deploy', 'all'
```

対応する監査テーブル・エントリには、110、`bob`、`pubs2` が、`event`、`loginname`、`dbname` カラムの値としてそれぞれリストされます。`extrainfo` カラムの内容は次のようになります。

```
webservices_role; deploy_all; ; ; ; bob/ase;
```

例 2 この例では、ユーザ `bob` によって `pubs2` データベースに入力された次のコマンドに対する監査テーブル・エントリを示します。

```
sp_webservices 'deploy', 'rawservice'
```

対応する監査テーブル・エントリには、110、`bob`、`pubs2` が、`event`、`loginname`、`dbname` カラムの値としてそれぞれリストされます。`extrainfo` カラムの内容は次のようになります。

```
webservices_role; deploy; ; ; ; bob/ase;
```

`sysaudits` テーブルのカラムの詳細な説明については、Adaptive Server Enterprise の『システム管理ガイド』を参照してください。

`sp_webservices 'undeploy'` の監査

監査イベントの番号 111 は、`sp_webservices` の `undeploy` オプションに対応しています。

例 1 この例では、ユーザ `bob` によって `pubs2` データベースに入力された次のコマンドに対する監査テーブル・エントリを示します。

```
sp_webservices 'undeploy', 'all'
```

対応する監査テーブル・エントリには、111、bob、pubs2 が、event、loginname、dbname カラムの値としてそれぞれリストされます。extrainfo カラムの内容は次のようになります。

```
webservices_role; undeploy_all; ; ; ; bob/ase;
```

例 2

この例では、ユーザ bob によって pubs2 データベースに入力された次のコマンドに対する監査テーブル・エントリを示します。

```
sp_webservices 'undeploy', 'rawservice'
```

対応する監査テーブル・エントリには、111、bob、pubs2 が、event、loginname、dbname カラムの値としてそれぞれリストされます。extrainfo カラムの内容は次のようになります。

```
webservices_role; deploy; ; ; ; bob/ase;
```

sysaudits テーブルのカラムの詳細な説明については、Adaptive Server Enterprise の『システム管理ガイド』を参照してください。

ASE Web Services のロギング

デフォルトでは、ASE Web Services は情報メッセージとエラー・メッセージのみをログに記録します。その他の詳細な情報をログに記録する方法については、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。

この項の内容は、次のとおりです。

- [ASE Web Services のログ・ファイル](#)
- [ログ・ファイルのロールオーバー](#)

ASE Web Services のログ・ファイル

ASE Web Services は、次の3つのファイルにアクティビティを記録します。

- [consumer.log](#)
- [producer.log](#)
- [http.log](#)

consumer.log

consumer.log ファイルには、Adaptive Server Enterprise がプロキシ・テーブルを介してリモート Web サービスを呼び出したことによる情報メッセージとエラー・メッセージが保存されます。

producer.log

producer.log ファイルには、Adaptive Server Enterprise で受信された SOAP 要求からの情報メッセージとエラー・メッセージがすべて保存されます。

http.log

http.log ファイルには、すべての HTTP 要求が NCSA Request Log 形式で保持されます。HTTP 要求は、呼び出された Web メソッドごとに存在します。

ログ・ファイルのロールオーバー

ロギングは、Apache log4j フレームワークを使用して ASE Web Services に実装されます。特定の log4j パラメータとロールオーバー・ポリシーの実装については、Web マニュアル (<http://jakarta.apache.org/log4j/docs/>) を参照してください。

Sybase Central を使用する場合

ASE Web Services 用の Sybase Central プラグインを使用すると、次の作業を実行できます。

- Sybase Central からの Web Services の設定
- Web サービスの利用
- Web サービスの実行
- Web メソッドの削除
- ユーザ定義 Web サービスの追加
- エイリアスの追加
- Web メソッドのプロパティの表示

詳細については、Sybase Central で ASE Web Services のオンライン・ヘルプを参照してください。

この章の内容は、次のとおりです。

トピック名	ページ
Apache サンプル・クライアント	53
Microsoft .NET のサンプル・クライアント	57

サンプル・クライアントを作成および実行するためのツールは、*samples* ディレクトリの下にあります。この章では、ASE Web Services Engine 用に提供されているサンプル・アプリケーションについて説明します。

Apache サンプル・クライアント

この項では、UNIX の `$$SYBASE/WS-15_0/samples/apacheclient` ディレクトリまたは Windows の `%SYBASE%\WS-15_0\samples\apacheclient` ディレクトリにあるサンプル・クライアントとスクリプトについて説明します。

注意 ASE Web Services がインストールされていないマシンで Apache サンプル・クライアントを実行する場合は、`/apacheclient/lib` ディレクトリの内容をそのマシンにコピーしてください。

サンプル・クライアントの作成

付属のサンプル・スクリプトを使用するには、まずサンプル・クライアントを作成します。

❖ サンプル・クライアントの作成

- 必要に応じて *apacheclient* ディレクトリ内のすべてのスクリプトの変数定義を変更して、必ず JRE 変数が使用する JRE を指すようにしてください。JRE バージョン 1.4.2 以降を使用してください。デフォルトでは、UNIX の `$$SYBASE_JRE` ディレクトリまたは Windows の `%SYBASE_JRE%` ディレクトリに格納されている JRE が使用されます。

ASE Web サービス・クライアントを作成すると、サンプル・スクリプトを実行してストアド・プロシージャと SQL 文を実行できます。このスクリプトは、*apacheclient* ディレクトリにあります。

runexecute の使用

runexecute スクリプトは、ASE Web Services を使用して Adaptive Server Enterprise でストアド・プロシージャまたは Transact-SQL 文を実行します。このサンプル・アプリケーションは **execute** Web メソッドを呼び出します。

構文

```
runexecute "web_service_URL" aseServerName user_ID password  
"SQLX_option" output_class count "sql_statement"
```

パラメータ

- *web_service_URL*
使用している Web サービスのロケーションです。
- *aseServerName*
interfaces ファイル内の Adaptive Server Enterprise サーバ、または LDAP サーバの名前を示す SOAP string です。
- *user_ID*
Adaptive Server Enterprise へのログインに必要なユーザ ID です。
- パスワード
Adaptive Server Enterprise へのログインに必要なパスワードです。
- *SQLX_option*
1 つ以上の *option* パラメータを示す文字列です。このパラメータは、SQLX 結果セットの特性を指定します。有効な *option* パラメータは、次のとおりです。
 - `binary={hex | base64}`
 - `columnstyle={element | attribute}`
 - `format={yes | no}`
 - `header={yes | no}`
 - `nullstyle={attribute | omit}`
 - `prefix="value"`
 - `root={yes | no}`
 - `rowname="value"`
 - `schemaloc="value"`
 - `statement={yes | no}`
 - `tablename="value"`
 - `targetns="value"`

value には値を指定してください。SQLX の関数とオプションの詳細については、『Adaptive Server Enterprise における XML Services』を参照してください。

- *output_class*

希望の出力の種類です。このパラメータに有効な値は、次のとおりです。

- *schema* – XML スキーマを返します。
- *dtd* – XML DTD を返します。
- *data* – 結果セットを返します。
- *all* – スキーマ、DTD、データを返します。

- *count*

文を実行する回数です。*count* の値が 1 より大きい場合は、セッションが作成され、永続的な接続が使用されます。

- *sql_statement*

Adaptive Server Enterprise で実行する文です。文を区切る場合は、二重引用符を使用してください。

例 1

この例では、**select** 文を使用して Adaptive Server Enterprise のバージョン番号を調べます。

```
runexecute "http://johndoe-sun:8183/services/ase" johndoe-sun
sa nopasswordspecified "tablename=ws" all 1 "select @@version"
```

ASE Web Services は、XML スキーマ、DTD、文の実行結果が含まれている結果セットを返します。

例 2

この例では、**pubs2** データベースでストアド・プロシージャ **booksales** を実行します。ストアド・プロシージャは、指定された本のタイトル ID の販売部数を返します。

```
runexecute "http://johndoe-sun:8183/services/ase" johndoe-sun
sa nopasswordspecified
"columnstyle=attribute,format=no,rowname=wsrow,prefix=Unnamed
col,nullstyle=attribute,header=yes" all 1 "execute booksales
MC2222"
```

ASE Web Services は、XML スキーマ、DTD、文の実行結果が含まれている結果セットを返します。

次の結果セットが返されます。

```
<?xml version="1.0" ?>
<resultset
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<wsrow title="Silicon Valley Gastronomic Treats"
total_sales="2032" Unnamedcol1="Books sold"/>
</resultset>
```

次の DTD が返されます。

```
<!DOCTYPE ws [  
<!ELEMENT resultset (row*)>  
<!ELEMENT row (title, total_sales, Unnamedcoll)>  
<!ELEMENT title (#PCDATA)>  
<!ELEMENT total_sales (#PCDATA)>  
<!ELEMENT Unnamedcoll (#PCDATA)>  
>]
```

次のスキーマが返されます。

```
<?xml version="1.0" ?>  
<xsd:schema  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
  xmlns:sqlxml="http://www.iso-standards.org/mra/9075/  
sqlx">  
  <xsd:import  
    namespace="http://www.w3.org/2001/XMLSchema"  
    schemaLocation="http://www.iso-standards.org/mra/  
9075/sqlx.xsd" />  
  <xsd:complexType  
    name="RowType.resultset">  
    <xsd:attribute name="title"  
      type="VARCHAR_80" use="required"/>  
    <xsd:attribute name="total_sales" type="INTEGER"  
      use="optional"/>  
    <xsd:attribute name="Unnamedcoll"  
      type="VARCHAR_24" use="optional"/>  
  </xsd:complexType>  
  <xsd:complexType  
    name="TableType.resultset">  
    <xsd:sequence>  
      <xsd:element name="wsrow"  
        type="RowType.resultset" minOccurs="0"  
        maxOccurs="unbounded"/>  
    </xsd:sequence>  
  </xsd:complexType>  
  <xsd:simpleType name="VARCHAR_80">  
    <xsd:restriction base="xsd:string">  
      <xsd:maxLength value="80"/>  
    </xsd:restriction>  
  </xsd:simpleType>  
  <xsd:simpleType name="INTEGER">  
    <xsd:restriction base="xsd:integer">  
      <xsd:maxInclusive value="2147483647"/>  
      <xsd:minInclusive value="-2147483648"/>  
    </xsd:restriction>  
  </xsd:simpleType>  
  <xsd:simpleType name="VARCHAR_24">  
    <xsd:restriction base="xsd:string">  
      <xsd:maxLength value="24"/>  
    </xsd:restriction>  
  </xsd:simpleType>  
</xsd:schema>
```



```

</xsd:restriction>
</xsd:simpleType>
<xsd:element name="resultset" type="TableType.resultset"/>
</xsd:schema>

```

例 3

この例では、**pubs2** データベースで SQL クエリを実行します。クエリは、出版社と同じ都市に住んでいない作家の姓と居住都市を返します。

```

runexecute "http://johndoe-sun:8183/services/ase" johndoe-sun
sa nopasswordspecified
"tablename=ws,header=yes,schemaloc='http://www-
edm/remote/svr/xmltestdir/resultset.xsd',targetns='http://www-
edm/remote/svr/xmltestdir/'" data 1 "select distinct
au_lname, authors.city from publishers, authors where
authors.city not in (select city from publishers where
authors.city = publishers.city)"

```

ASE Web Services は、文の実行結果が含まれている結果セットを返します。

Microsoft .NET のサンプル・クライアント

この項では、UNIX の `$$SYBASE/WS-15_0/samples/ms.net/Execute/bin/Release` ディレクトリまたは Windows の `%SYBASE%\WS-15_0\samples\ms.net\Execute\bin\Release` ディレクトリにあるサンプル・クライアントとスクリプトについて説明します。

Microsoft .NET は、Microsoft の Web サイト (<http://msdn.microsoft.com/library/default.asp?url=/downloads/list/netdevframework.asp>) からダウンロードできます。

注意 この URL は、このマニュアルの発行日時点では最新ですが、変更される可能性があります。

サンプル・クライアントの作成

付属のサンプル・スクリプトを使用するには、まずサンプル・クライアントを作成します。サンプル・クライアントを作成すると、この項に示すサンプル・スクリプトを実行できます。このスクリプトは、*Release* ディレクトリにあります。

Execute.exe の使用

Execute.exe は、ASE Web Services を使用して Adaptive Server Enterprise でストアド・プロシージャまたは Transact-SQL 文を実行します。このサンプル・アプリケーションは **execute** Web メソッドを呼び出します。

構文 `Execute.exe "web_service_URL" aseServerName user_ID password "SQLX_option" output_class count "sql_statement"`

パラメータ

- *web_service_URL*
使用している Web サービスのロケーションです。
- *aseServerName*
sql.ini ファイル内の Adaptive Server Enterprise サーバ、または LDAP サーバの名前を示す SOAP string です。
- *user_ID*
Adaptive Server Enterprise へのログインに必要なユーザ ID です。
- パスワード
Adaptive Server Enterprise へのログインに必要なパスワードです。
- *SQLX_option*
1 つ以上の **option** パラメータを示す文字列です。このパラメータは、SQLX 結果セットの特性を指定します。有効な option パラメータは、次のとおりです。
 - `binary={hex | base64}`
 - `columnstyle={element | attribute}`
 - `format={yes | no}`
 - `header={yes | no}`
 - `nullstyle={attribute | omit}`
 - `prefix="value"`
 - `root={yes | no}`
 - `rowname="value"`
 - `schemaloc="value"`
 - `statement={yes | no}`
 - `tablename="value"`
 - `targetns="value"`

value には値を指定してください。SQLX の関数とオプションの詳細については、『Adaptive Server Enterprise における XML Services』を参照してください。

- *output_class*

希望の出力の種類です。このパラメータに有効な値は、次のとおりです。

- *schema* – XML スキーマを返します。
- *dtd* – XML DTD を返します。
- *data* – 結果セットを返します。
- *all* – スキーマ、DTD、データを返します。

- *count* (カウント)

実行する回数です。

- *sql_statement*

Adaptive Server Enterprise で実行する文です。文を区切る場合は、二重引用符を使用してください。

例 1

この例では、Adaptive Server Enterprise のバージョン番号を調べます。

```
Execute.exe "http://johndoe-sun:8183/services/ase" johndoe-
sun sa nopasswordspecified "tablename=ws" all 1 "select
@@version"
```

ASE Web Services は、XML スキーマ、DTD、文の実行結果が含まれている結果セットを返します。

例 2

この例では、pubs2 データベースでストアド・プロシージャ **booksales** を実行します。ストアド・プロシージャは、指定された本のタイトル ID の販売部数を返します。

```
Execute.exe "http://johndoe-sun:8183/services/ase" johndoe-
sun sa nopasswordspecified
"columnstyle=attribute,format=no,rowname=wsrow,prefix=Unnamed
col,nullstyle=attribute,header=yes" all 1 "execute booksales
MC2222"
```

ASE Web Services は、XML スキーマ、DTD、文の実行結果が含まれている結果セットを返します。

例 3

この例では、**pubs2** データベースで SQL クエリを実行します。クエリは、出版社と同じ都市に住んでいない作家の姓と居住都市を返します。

```
Execute.exe "http://johndoe-sun:8183/services/ase" johndoe-  
sun sa nopasswordspecified  
"tablename=ws,header=yes,schemaloc='http://www-  
edm/remote/svr/xmltestdir/resultset.xsd',targetns='http://www-  
edm/remote/svr/xmltestdir/'" data 1 "select distinct  
au_lname, authors.city from publishers, authors where  
authors.city not in (select city from publishers where  
authors.city = publishers.city)"
```

ASE Web Services は、文の実行結果が含まれている結果セットを返します。

第 4 部

トラブルシューティング

ここでは、トラブルシューティング手順とエラー・メッセージについて説明します。

この章の内容は、次のとおりです。

トピック名	ページ
トラブルシューティングの問題	63
診断ツール	68
メッセージ	70

トラブルシューティングの問題

ASE Web Services のトラブルシューティングに役立つ問題は、次のとおりです。

リモート・サーバ・クラス定義の設定

問題

プロキシ・テーブルの生成時に、`sp_webservices add` コマンドを実行すると、次のエラーが発生することがあります。

```
Warning:Row size (3347 bytes) could exceed row size limit, which is 1962 bytes.
Msg 208, Level 16, State 1:
Server 'JMALVARADO', Line 1:
tempdb..ws_4338e6e122cd4ef0a not found.Specify owner.objectname or uses to check
whether the object exists (sp_help may produce lots of output).
No proxy tables were created for the WSDL URL:
[http://www.xignite.com/xquotes.asmx?WSDL]
(return status = 0)
```

このエラーは、ASE Web Services Engine を表すリモート・サーバが、“sds”以外のクラスの `sp_addserver` を使用して追加されているために発生します。これを確認するには、次のように、`isql` で `sp_helpserver` を使用します。

```
sp_helpserver ws
```

ここで、`ws` は、ASE Web Services Engine の名前です。これはデフォルト値です。リモート・サーバ・クラスが、結果の指定されたカラムに返されます。

```
name network_name class ...
---- -
ws ws null ...
```

ユーザのアクション 次のように、`isql` で `sp_dropserver` と `sp_addserver` を使用して、リモート・サーバのクラスを “sds” に変更します。

```
sp_dropserver ws_name
...
sp_addserver ws_name, sds, ws_name
```

ここで、`ws_name` は、ASE Web Services Engine 用に選択された名前です。

マップを解除された RPC/encoded Web メソッド

問題 RPC/encoded Web メソッドに入力パラメータまたは出力パラメータがない場合、プロキシ・テーブルにマッピングできません。パラメータを持たない Web メソッドのプロキシ・テーブルには、カラムはありません。カラムのないテーブルを Adaptive Server Enterprise で作成することはできません。

ユーザのアクション Web メソッドを変更して、入力パラメータまたは出力パラメータを組み込みます。

問題 カラムにマッピングできるのは、単純な型に限られています。RPC/encoded Web メソッドで複雑な型または配列を使用すると、Web メソッドをプロキシ・テーブルにマッピングできなくなります。

ユーザのアクション 単純な型と配列のみを使用するように Web メソッドを変更します。

トランケートされた document/literal の結果

問題 Web サービスが `@@textsize` グローバル変数の値より多くのデータを返した場合、データは `@@textsize` で指定されたサイズにトランケートされます。その結果、返されたデータが有効な XML ドキュメントの形式にならない場合があります。

ユーザのアクション テキストまたはイメージ・データが `select` コマンドでトランケートされた場合、これは Adaptive Server Enterprise で予期されている動作なのでエラーは発生しません。ただし、警告が ASE Web Services Engine のログに記録されます。`consumer.log` ファイルを確認してください。

ASE Web Services Engine の起動

問題 `runws` スクリプトが ASE Web Services Engine を正常に起動しません。

ユーザのアクション 1 使用するポートがすでに別のプロセスで使用されていないことを確認してください。

- 2 正しい JRE がインストールされていることを確認してください。ASE Web Services には JRE 1.4.2 以降が必要です。

JRE のバージョンを確認するには、コマンド・プロンプトで次のように入力します。

```
java -version
```

- 3 `ws.properties` 以外のプロパティ・ファイルで Web Services を実行する場合は、ファイルの絶対パスを指定します。たとえば、別のプロパティ・ファイルで ASE Web Services Engine を実行する場合は、次のようになります。

```
C:\sybase\WS-15_0\bin\runws -f
C:\sybase\WS-15_0\props\myfile.properties
```

問題

ASE Web Services Engine は、指定された `ase_service_name` を `interfaces` ファイル内で検索しますが、`producer.log` に次のエラー・メッセージが表示されています。

```
INFO [main] - Error locating libtcl.cfg file.
INFO [main] - java.io.FileNotFoundException:LDAP config File
does not exist
```

WSDL の検索

問題

Web ブラウザを使用して ASE Web Services Engine に接続しているクライアントが、WSDL ファイルを検索できません。または、ASE Web Services Engine が WSDL ファイルを検索してプロキシ・テーブルのマッピングを実行できません。

ユーザのアクション

ASE Web Services Engine が動作していることを確認します。ブラウザから ASE Web Services Engine を使用している場合は、ブラウザの URL に、SSL 接続の場合は `https://`、標準接続の場合は `http://` と表示されていることを確認してください。

ws.properties でのエントリの指定

問題

円記号 (¥) はエスケープ文字として使用されるので、単一の円記号を使用するエントリは正しく解釈されません。次に例を示します。

```
com.sybase.ase.ws.interfaces = d:\sybase\ini\sql.ini
```

ユーザのアクション

円記号を別の円記号でエスケープします。

```
com.sybase.ase.ws.interfaces = d:\¥sybase¥ini¥sql.ini
```

スラッシュを使用することもできます。

```
com.sybase.ase.ws.interfaces = d:/sybase/ini/sql.ini
```

Windows NT コマンド・ライン引数

問題 引数と引数値の間にスペースがない場合、Windows NT ではスクリプトは実行されません。たとえば、*configssl* スクリプトを次のように呼び出した場合、スクリプトは実行されません。

```
configssl -dhostname
```

ユーザのアクション 引数と引数値の間にスペースを入れます。

```
configssl -d hostname
```

実行スクリプトまたは停止スクリプトの失敗

問題 *runws* または *stopws* スクリプトの実行が失敗します。

ユーザのアクション これらのいずれかのスクリプトを実行できない場合は、次の手順を実行します。

- Adaptive Server Enterprise が実行中であることを確認します。
- Adaptive Server Enterprise にログインするために指定したユーザ名とパスワードが有効であることを確認します。
- *producer.log* または *consumer.log* ファイルでエラー・メッセージを確認します。
- 提供された *ase_service_name* が、*libtcl.cfg* ファイルが指す LDAP サーバ上または Adaptive Server Enterprise の *interfaces* ファイル内にあることを確認します。

注意 Windows システムでは、*interfaces* ファイルの名前が *sql.ini* になります。

- ユーザ名に *sa_role* 権限があることを確認します。

null パスワード

問題 Adaptive Server Enterprise ユーザのパスワードに null 文字列が設定されている場合があります。

ユーザのアクション *runws* スクリプトや *stopws* スクリプトなど、パスワードが必要な場合には、トークン *nopasswordspecified* を使用します。

SSL での SOAP 終了ポイントの指定

問題 呼び出し時に *aseServerName* を指定したときに、ASE Web Services のメソッドまたはサンプル・アプリケーションが結果を返しません。

ユーザのアクション *aseServerName* 名が有効な SOAP 終了ポイントであることを確認します。DNS エイリアスを使用している場合は、エイリアスが有効な SOAP 終了ポイントに解決されることを確認します。SSL を使用する場合は、*aseServerName* で指定された終了ポイントが、*configssl* スクリプトで SSL 証明書を作成するときに指定した名前と同じであることを確認します。次に例を示します。

```
configssl -d mydomainhostname -h 8182
```

ここで、ASE Web Services のメソッドまたはサンプル・アプリケーションの呼び出し時に指定する *aseServerName* の値は “https://mydomainhostname:8182” にします。SSL を使用している場合に *aseServerName* の “localhost” または IP アドレスを置換すると、メソッドまたはサンプル・アプリケーションは結果を返しません。

sp_webservices ‘add’ の異常終了

問題 *sp_webservices* ‘add’ オプションの実行中に作成されたプロキシ・テーブルは、[Ctrl+C] による中断または Adaptive Server Enterprise のクラッシュの場合と同様に、*sp_webservices* の異常終了後にも残ります。

ユーザのアクション *sp_webservices* が異常終了した場合に *add* オプションによって作成されたプロキシ・テーブルを削除するには、*remove* オプションを使用します。

Web Services のプロキシ・テーブルの制約

問題 *sp_webservices* の *add* オプションを実行して生成した Web Services のプロキシ・テーブルと同じ名前を持つプロキシ・テーブルが、Web Services のプロキシ・テーブルとして誤って指定される場合があります。Web Services のプロキシ・テーブルのジョインの処理方法は、通常のプロキシ・テーブルのジョインとは異なります。

ユーザのアクション Web Services のプロキシ・テーブルをユニークなデータベースに格納します。

問題 予約語 *or* は、Web Services のプロキシ・テーブルに対するクエリの *where* 句ではサポートされません。そのため、クエリが予約語 *or* を使用するクエリに内部的に変換されると、エラーが発生する場合があります。たとえば、次のクエリは Web Services のプロキシ・テーブル *testint* から選択します。

```
select * from testint where _in0 in (1, 2)
```

このクエリは、内部的に次のように変換されます。

```
select * from testint where _in0 = 1 or _in0 = 2
```

内部変換では予約語 *or* が使用されるので、ユーザが送信したクエリによってエラーが発生します。

ユーザのアクション

内部変換で予約語 `or` が使用されないようにクエリを変更します。たとえば、前に説明したようなユーザが送信したクエリは、テンポラリ・テーブルの使用に変更できます。

```
create table a (col int)
insert into a values (1)
insert into a values (2)
select * from testint where _in0 in (select col from a)
```

sysattributes テーブル・エントリ

問題

`sp_webservices 'add'` を実行すると、次のエラーが発生します。

```
Msg 5629, Level 16, State 1:
Line 1:
Cannot start a remote distributed transaction participant as
the local server is not named.Please contact a user with System
Administrator role.
```

ユーザのアクション

`sysattributes` テーブルに Adaptive Server Enterprise 用のエントリが必要です。エントリを提供するには、`sp_addserver` ストアド・プロシージャを使用します。

```
sp_addserver ase_entry, local, ase_entry
```

ここで、`ase_entry` は、`interfaces` または `sql.ini` ファイルにある Adaptive Server Enterprise のローカル名です。次に Adaptive Server Enterprise を再起動します。

診断ツール

ASE Web Services には、診断用の詳細なログギングと接続の問題の特定に役立つ JDBC レベルのトレースがあります。

詳細なロギング

診断用の詳細なロギングを有効にするには、*props* ディレクトリの *logging.properties* ファイルを適切に変更してから、ASE Web Services Engine を再起動します。

次に、ASE Web Services に付属する *logging.properties* ファイルの内容の例を示します。

```
# Set logging level for ASE Web Services Consumer
log4j.logger.com.sybase.ase.ws.sds=INFO, C
#log4j.logger.com.sybase.ase.ws.sds=DEBUG, C
log4j.additivity.com.sybase.ase.ws.sds=false

# Set logging level for ASE Web Services Producer
log4j.logger.com.sybase.ase.ws.producer=INFO, P
#log4j.logger.com.sybase.ase.ws.producer=TRACE, P
log4j.additivity.com.sybase.ase.ws.producer=false
```

診断用の詳細なロギングを行うには、*logging.properties* を次のように変更します。

```
# Set logging level for ASE Web Services Consumer
#log4j.logger.com.sybase.ase.ws.sds=INFO, C
log4j.logger.com.sybase.ase.ws.sds=DEBUG, C
log4j.additivity.com.sybase.ase.ws.sds=false

# Set logging level for ASE Web Services Producer
#log4j.logger.com.sybase.ase.ws.producer=INFO, P
log4j.logger.com.sybase.ase.ws.producer=TRACE, P
log4j.additivity.com.sybase.ase.ws.producer=false
```

JDBC レベルのトレースの有効化

JDBC レベルのトレースを有効にする方法については、該当する jConnect のマニュアルを参照してください。

メッセージ

表 6-1 に ASE Web Services のメッセージを示します。

表 6-1: ASE Web Services のメッセージ

メッセージ番号	メッセージ
15200	WSDL URI [%s] に対してプロキシ・テーブルにマップされている Web メソッドがありません。
15201	ほとんどの場合、WSDL URI には、サフィックス ?WSDL が付きます。WSDL URI を検証してください。
15202	Web Method [%s] は、データ型がサポートされないため、プロキシ・テーブルにマップされません。
15203	不明なプロシージャ [%s] を実行する要求を受け取りました。
15204	IOException が検出されました。通常これは、ASE と ASE Web Services Engine との間の通信エラーを表します。詳細 :[%s]
15205	SQLException が検出されました。通常これは、ASE からメタ・データを取得するときのエラーを表します。詳細 :[%s]
15206	認識できない例外が検出されました : 詳細 :[%s]
15207	Remote Web Method Exception (AxisFault) が検出されました。これは、リモート Web メソッドのエラーを表します。詳細 :[%s]
15208	マッピング例外が検出されました。これは、Web メソッド引数を ASE 型にマップするときのエラーを表します。詳細 :[%s]
15209	サービス例外が検出されました。通常これは、不正な WSDL ファイルを表します。詳細 :[%s]
15210	良好に形成されていない Web メソッドに対する XML 入力を受け取りました。
15211	Web メソッド (MalformedURL) の起動エラーです。詳細 :[%s]
15212	RemoteException が検出されました。通常これは、ネットワーク転送エラーを表します。詳細 :[%s]
15213	Web メソッドの起動エラーです (不明な例外) : 詳細 [%s]
15214	プロキシ・テーブル [%s] からの Web メソッド [%s] の呼び出しを中止します。これは、Web メソッドが [%s] 引数を期待していたのに、[%s] を受け取ったためです。
15215	パラメータの数/型が一致しません。組み込み関数 ws_admin に渡されたパラメータの数と型をチェックしてください。
15220	その名前によるユーザ定義の Web サービスはすでに存在します。
15221	サービスから参照されているエイリアスは削除できません。
15216	組み込み関数 ws_admin に不明なオペレーション '%.*s' が指定されました。パラメータの綴りと配置をチェックしてください。
15217	sysattributes に対する更新/挿入/削除で障害が発生しました。
15218	現在のデータベース名を取得できません。
15219	'%.*s' のオブジェクト ID を取得できません。
19307	WSDL URI の sds [%1!] を使用してプロキシ・テーブルを生成します : [%2!]
19308	[%1!] の WSDL Match が見つかりました。

メッセージ番号	メッセージ
19309	WSDL URI [%!1] がシステム内に見つかりません。
19310	[%2!] を使用して WSDL URI [%!1] のタイムアウト・エントリを更新します。
19311	すべての Web サービス・メタ・データを削除しています
19312	WSDL URI [%!1] のエントリを削除します。
19313	webservice を削除するには、有効な SDS サーバが提供される必要があります。SDS サーバ [%!1] が見つかりませんでした。sp_addserver を使用して SDS サーバを追加してください。
19319	webservice を追加するには、有効な SDS サーバが提供される必要があります。SDS サーバ [%!1] が見つかりませんでした。sp_addserver を使用して SDS サーバを追加してください。
19320	ASE Web Services Engine が動作していることを確認します。
19321	webservice を追加するには、有効な WSDL URI が指定される必要があります。
19322	指定された WSDL URI [%!1] はすでにシステムに存在します。sp_webservices を使用して、最初に削除します。
19323	システムにロードされていない webservice の情報をリストするには、SDS サーバが必要です。SDS サーバ [%!1] が見つかりませんでした。sp_addserver を使用して SDS サーバを追加してください。
19324	特定の wsdl uri を指定して、修正してください。
19325	修正のために変更する部分を指定してください。
19326	[%!1][%!1] は、sp_webservices による修正のための有効なオプションではありません。
19327	修正のために item=value 構文を指定してください。
19408	エイリアスは存在しないので、削除できません。
19409	追加するエイリアスの名前を指定してください。
19410	このエイリアスに関連付けられるデータベース名を指定してください。
19412	展開済みのサービスの名前を変更することはできません。

sp_webservices のヘルプ情報を参照するには、isql コマンド・ラインで sp_webservices help を入力してください。

第 5 部

付録

ここでは、このマニュアルの付録について説明します。

インストール内容

この付録では、ASE Web Services のインストール内容について説明します。

ASE Web Services のディレクトリ・ツリー

ASE Web Services は、Adaptive Server Enterprise ルート・ディレクトリと同じレベルにインストールされます。ASE Web Services のルート・ディレクトリには *WS-15_0* という名前が付けられます。これは、次のサブディレクトリで構成されます。

表 A-1: ASE Web Services のディレクトリ

ディレクトリ名	内容
<i>bin</i>	ASE Web Services を設定および実行するスクリプトです。
<i>lib</i>	ASE Web Services が使用する Java ライブラリとパッケージです。
<i>logs</i>	ログ・ファイルのデフォルトのロケーションです。
<i>producer</i>	実行時における ASE Web Services Engine のファイルとサブディレクトリです。
<i>props</i>	ASE Web Services のプロパティ用のファイルです。
<i>samples</i>	ASE Web Services Engine でサンプル・クライアントを構築および実行するためのサンプル・スクリプトです。

bin ディレクトリの内容

次の表に、*bin* ディレクトリの内容を示します。

表 A-2: bin ディレクトリの内容

ファイル名またはディレクトリ名	機能
<i>configssl</i>	SSL を設定します。
<i>getpass.exe</i>	Adaptive Server Enterprise ログインに使用します。このファイルは Windows でのみ使用できます。
<i>installws</i>	<i>sp_webservices</i> ストアド・プロシージャをインストールします。
<i>runtcpmon</i>	SOAP メッセージをトレースするためのモニタを作成します。
<i>runws</i>	ASE Web Services Engine を起動します。
<i>stopws</i>	ASE Web Services Engine を停止します。

注意 Windows では、これらのファイルにサフィックス *.bat* が付けられます。

lib ディレクトリの内容

次の表に、*lib* ディレクトリの内容を示します。

表 A-3: lib ディレクトリの内容

ファイル名またはディレクトリ名	機能
<i>axis.jar</i>	Apache Axis ファイル
<i>commons-discovery.jar</i>	Apache Axis ファイル
<i>commons-logging.jar</i>	Apache Axis ファイル
<i>javax.servlet.jar</i>	サーブレット・ライブラリ
<i>jaxrpc.jar</i>	Apache Axis ファイル
<i>log4j-1.2.4.jar</i>	log4j ロガー
<i>mail.jar</i>	Apache Axis ファイル
<i>org.mortbay.jetty.jar</i>	HTTP サーバ
<i>saaj.jar</i>	Apache Axis ファイル
<i>sqlx.jar</i>	SQLX ファイル
<i>tools.jar</i>	Java ツール (たとえば、 <i>javac</i> コンパイラ)
<i>ws.jar</i>	ASE Web Services Engine のコード
<i>ws_debug.jar</i>	ASE Web Services Engine 用のデバッガ・コード
<i>wsdl4j.jar</i>	Apache Axis ファイル
<i>xercesImpl.jar</i>	Xerces パーサ
<i>xmlParserAPIs.jar</i>	Xerces パーサ

logs ディレクトリの内容

次の表に、logs ディレクトリの内容を示します。

表 A-4: logs ディレクトリの内容

ファイル名またはディレクトリ名	機能
<i>consumer.log</i>	Adaptive Server Enterprise がプロキシ・テーブルを介してリモート Web サービスを呼び出したことによる情報メッセージとエラー・メッセージが記録されます。
<i>http.log</i>	Web サーバのアクティビティのログGINGを行います。
<i>producer.log</i>	SOAP 要求が Adaptive Server Enterprise で受信されたことによる、すべての情報メッセージとエラー・メッセージが記録されます。

producer ディレクトリの内容

次の表に、producer ディレクトリの内容を示します。

表 A-5: producer ディレクトリの内容

ファイル名またはディレクトリ名	機能
<i>index.html</i>	Web サービス・ブラウザ・インタフェースのホーム・ページです。
<i>keystore</i>	暗号化キーを保持します。
<i>ui</i>	Web サービス・ブラウザ・インタフェースのディレクトリ構造です。
<i>WEB-INF</i>	Jetty に必要なディレクトリ構造です。
<i>wscertificate.cer</i>	SSL 用の自動生成証明書です。このファイルは、 <i>configssl</i> スクリプトを実行した場合にのみ存在します。

props ディレクトリの内容

次の表に、props ディレクトリの内容を示します。

表 A-6: props ディレクトリの内容

ディレクトリ名	機能
<i>logging.properties</i>	log4j 用の設定ファイル
<i>ws.properties</i>	ASE Web Services Engine のすべての設定パラメータ
<i>wsmmsg.properties</i>	メッセージの設定ファイル

samples ディレクトリの内容

samples ディレクトリには、Apache と Microsoft .NET 用のプリコンパイルされたソース・コードが含まれています。ソース・コードは独自のアプリケーションでも使用できます。次の表に、*samples* ディレクトリの内容を示します。

表 A-7: samples ディレクトリの内容

ディレクトリ名	機能
<i>apacheclient</i>	サンプル・クライアントをコンパイルおよび実行するサンプル・スクリプトを含むディレクトリ
<i>ms.net</i>	.NET のサンプルを含むディレクトリ

設定プロパティ

この付録の内容は、次のとおりです。

トピック名	ページ
ws.properties	79
myres.properties	81
properties ファイルでのエントリの指定	82

ws.properties

ws.properties ファイルには、ASE Web Services の次の設定が含まれています。

表 B-1: *ws.properties* のエントリ

<code>com.sybase.ase.ws.producer.httpport</code>	ASE Web Services Engine が HTTP 接続を受信するポートを示します。デフォルトのエントリは 8181 です。
<code>com.sybase.ase.ws.consumer.cisport</code>	ASE Web Services Engine が TDS を受信するポートを示します。デフォルトのエントリは 8183 です。
<code>com.sybase.ase.ws.producer.logfilename</code>	ASE Web Services Engine のログ・ファイルを格納する場所を示します。デフォルトのロケーションは、UNIX の場合は <code>\$\$SYBASE/WS-15_0/logs/producer.log</code> 、Windows の場合は <code>%SYBASE%\WS-15_0\logs\producer.log</code> です。
<code>com.sybase.ase.ws.consumer.logfilename</code>	ASE Web Services Engine のログ・ファイルを格納する場所を示します。デフォルトのロケーションは、UNIX の場合は <code>\$\$SYBASE/WS-15_0/logs/consumer.log</code> 、Windows の場合は <code>%SYBASE%\WS-15_0\logs\consumer.log</code> です。
<code>com.sybase.ase.ws.producer.jettylogfile</code>	HTTP 要求のログ・ファイルを格納する場所を示します。デフォルトのロケーションは、UNIX の場合は <code>\$\$SYBASE/WS-15_0/logs/http.log</code> 、Windows の場合は <code>%SYBASE%\WS-15_0\logs\http.log</code> です。
<code>com.sybase.ase.ws.interfaces</code>	Adaptive Server Enterprise の <i>interfaces</i> または <i>sql.ini</i> ファイルのロケーションを示します。デフォルトのロケーションは、UNIX の場合は <code>\$\$SYBASE/interfaces</code> 、Windows の場合は <code>%SYBASE%\ini\sql.ini</code> です。

com.sybase.ase.ws.libtcl

LDAP サーバの識別に使用する *libtcl.cfg* ファイルのロケーションを示します。デフォルトのロケーションは、32 ビットの UNIX プラットフォームの場合は `$$SYBASE/$SYBASE_OCS/config/libtcl.cfg`、32 ビットの Windows プラットフォームの場合は `%SYBASE%\$SYBASE_OCS%\ini\libtcl.cfg` です。64 ビットの UNIX プラットフォームの場合は `$$SYBASE/$SYBASE_OCS/config/libtcl64.cfg`、64 ビットの Windows プラットフォームの場合は `%SYBASE%\$SYBASE_OCS%\ini\libtcl64.cfg` です。

com.sybase.ase.ws.maxpostsize

受信する SOAP 要求の最大サイズを決定します。デフォルトのエントリは **20000** です。

com.sybase.ase.ws.ui.activate

Web ベースのユーザ・インタフェースがアクティブになっているかどうかを決定します。ユーザ・インタフェースは、`https://hostname:https_port` で入手できます。デフォルトのエントリは **true** です。

com.sybase.ase.ws.producer.tuning.maxthreads

HTTP ポートにサービスを提供するスレッド・プール内の最大スレッド数を示します。デフォルトのエントリは **250** です。

com.sybase.ase.ws.producer.tuning.minthreads

HTTP ポートにサービスを提供するスレッド・プール内の最小スレッド数を示します。デフォルトのエントリは **45** です。

com.sybase.ase.ws.producer.tuning.maxidletime

スレッドがアイドルになる最大時間をミリ秒で示します。デフォルトのエントリは **60000** です。

com.sybase.ase.ws.producer.tuning.ssl.maxthreads

HTTPS ポートにサービスを提供するスレッド・プール内の最大スレッド数を示します。デフォルトのエントリは **250** です。

com.sybase.ase.ws.producer.tuning.ssl.minthreads

HTTPS ポートにサービスを提供するスレッド・プール内の最小スレッド数を示します。デフォルトのエントリは **45** です。

com.sybase.ase.ws.producer.tuning.ssl.maxidletime

スレッドがアイドルになる最大時間をミリ秒で示します。デフォルトのエントリは **60000** です。

com.sybase.ase.ws.producer.ssl.keypassword

SSL 証明書のパスワードを示します。デフォルトはありません。

com.sybase.ase.ws.producer.ssl.keystore

SSL のキーストアのロケーションを示します。デフォルトのロケーションは、UNIX の場合は `$$SYBASE/WS-15_0/producer/keystore`、Windows の場合は `%SYBASE%\$WS-15_0\producer\keystore` です。

com.sybase.ase.ws.producer.ssl.httpsport

ASE Web Services Engine が HTTPS 接続を受信するポートを示します。デフォルトのエントリは **8182** です。

com.sybase.ase.ws.producer.ssl.password

SSL のキーストア・パスワードを示します。デフォルトはありません。

myres.properties

myres.properties ファイルは、Sybase Central プラグインが ASE Web Services 用の設定作業を実行したときに作成されます。*myres.properties* ファイルには、ASE Web Services の次の設定が含まれています。

表 B-2: *myres.properties* のエントリ

Web Services	一次のエントリを設定して Web Services を設定します。
<code>ws.ini</code>	Adaptive Server Enterprise の <i>interfaces</i> または <i>sql.ini</i> ファイルのロケーションを示します。デフォルトのロケーションは、UNIX の場合は <code>\$\$SYBASE/interfaces</code> 、Windows の場合は <code>%SYBASE%\%ini%sql.ini</code> です。
<code>ws.libtcl</code>	LDAP サーバの識別に使用する <i>libtcl.cfg</i> ファイルのロケーションを示します。デフォルトのロケーションは、32 ビットの UNIX プラットフォームの場合は <code>\$\$SYBASE/\$\$SYBASE_OCS/config/libtcl.cfg</code> 、32 ビットの Windows プラットフォームの場合は <code>%SYBASE%\%SYBASE_OCS%\%ini%libtcl.cfg</code> です。64 ビットの UNIX プラットフォームの場合は <code>\$\$SYBASE/\$\$SYBASE_OCS/config/libtcl64.cfg</code> 、64 ビットの Windows プラットフォームの場合は <code>%SYBASE%\%SYBASE_OCS%\%ini%libtcl64.cfg</code> です。
<code>ws.producer.port</code>	ASE Web Services Engine 用のポートを示します。
<code>ws.producer.log</code>	<i>producer.log</i> ファイルのロケーションを示します。
<code>ws.producer.jettylogfile</code>	HTTP 要求のログ・ファイルを格納する場所を示します。デフォルトのロケーションは、UNIX の場合は <code>\$\$SYBASE/WS-15_0/logs/http.log</code> 、Windows の場合は <code>%SYBASE%\%WS-15_0%\logs%http.log</code> です。
SSL	一次のエントリを設定して SSL を設定します。
<code>ws.ssl.host</code>	アクセスする SSL ホストの名前を示します。
<code>ws.ssl.keystorelocation</code>	SSL のキーストアのロケーションを示します。デフォルトのロケーションは、UNIX の場合は <code>\$\$SYBASE/WS-15_0/producer/keystore</code> 、Windows の場合は <code>%SYBASE%\%WS-15_0%\producer%keystore</code> です。
<code>ws.ssl.certificatepassword</code>	SSL 証明書のパスワードを示します。デフォルトはありません。
<code>ws.ssl.keystorepassword</code>	SSL のキーストア・パスワードを示します。デフォルトはありません。
<code>ws.consumer.name</code>	<i>interfaces</i> ファイルまたは <i>sql.ini</i> ファイルで指定されている ASE Web Services Engine の名前を示します。
<code>ws.consumer.host</code>	ASE Web Services Engine のホスト・マシンを示します。

`ws.consumer.port`

ASE Web Services Engine プロセスのポート番号を示します。

`ws.consumer.log`

`consumer.log` ファイルのロケーションを示します。

`installws` - 次のエントリを設定して `installws` スクリプトを実行します。

`ws.sqlsrv.server_name`

`installws` を実行する Adaptive Server Enterprise の名前を示します。

`ws.sqlsrv.sa_login`

Adaptive Server Enterprise のユーザ・ログインを示します。

`ws.sqlsrv.sa_password`

Adaptive Server Enterprise のパスワードを示します。

properties ファイルでのエントリの指定

円記号“¥”はエスケープ文字として使用されるので、単一の円記号を使用するエントリは正しく解釈されません。次に例を示します。

```
com.sybase.ase.ws.interfaces = d:¥sybase¥ini¥sql.ini
```

この問題に対処するには、円記号を別の円記号でエスケープします。次に例を示します。

```
com.sybase.ase.ws.interfaces = d:¥¥sybase¥¥ini¥¥sql.ini
```

スラッシュを使用することもできます。次に例を示します。

```
com.sybase.ase.ws.interfaces = d:/sybase/ini/sql.ini
```

SOAP と Adaptive Server Enterprise のデータ型のマッピング

この付録では、SOAP と Adaptive Server Enterprise のデータ型について説明します。

データ型のマッピング

次の項では、Web Services 機能に関して SOAP と Adaptive Server Enterprise 間のデータ型のマッピングについて説明します。

SOAP から ASE へのデータ型のマッピング

次の表に、SOAP のデータ型とそれに対応する Adaptive Server Enterprise のデータ型を示します。これらは、RPC/encoded Web サービスを Adaptive Server Enterprise のプロキシ・テーブルにマッピングするときに使用されます。

表 C-1: SOAP と Adaptive Server Enterprise のデータ型

SOAP のデータ型	Adaptive Server Enterprise のデータ型
string	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
boolean	smallint
float	real
double	double precision
decimal	float
duration	datetime
dateTime	datetime
time	datetime
date	datetime
gYearMonth	datetime
gYear	datetime
gMonthDay	datetime
gDay	datetime
gMonth	datetime
hexBinary	サポートされていません。

SOAP のデータ型	Adaptive Server Enterprise のデータ型
base64Binary	サポートされていません。
anyURI	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
QName	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
NOTATION	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
normalizedString	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
token	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
language	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
NMTOKEN	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
Name	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
NCName	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
ID	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
IDREF	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
ENTITY	varchar 長さは Adaptive Server Enterprise のページサイズによって異なります。
integer	integer
nonPositiveInteger	integer
negativeInteger	integer
long	integer
int	integer
short	smallint
byte	tinyint

SOAP のデータ型	Adaptive Server Enterprise のデータ型
nonNegativeInteger	integer
unsignedLong	integer
unsignedInt	integer
unsignedShort	smallint
unsignedByte	tinyint
positiveInteger	integer
soap 配列	サポートされていません。
soap の複雑なデータ型	サポートされていません。

create service コマンド用の ASE から SOAP へのデータ型マッピング

次の表は、ストアド・プロシージャのデータ型として利用可能な型と、SOAP のユーザ定義 Web サービスで使用されるデータ型との関係を示します。

注意 HTTP の GET メソッドを使用してアクセスされる Web サービスは、http 型または raw 型の Web サービスに対応しており、すべてのパラメータが xsd:string 型にマッピングされます。

表 C-2: SOAP と Adaptive Server Enterprise のデータ型

ASE のデータ型	SOAP のパラメータ型
tinyint	xsd:int
smallint	xsd:int
int	xsd:int
bigint	xsd:decimal
numeric (p,s)	xsd:decimal
decimal (p,s)	xsd:decimal
float (precision)	xsd:float
double (precision)	xsd:double
real	xsd:double
smallmoney	xsd:int
money	xsd:int
smalldatetime	xsd:datetime
datetime	xsd:datetime
date	xsd:datetime
time	xsd:datetime
char(n)	xsd:string
varchar(n)	xsd:string
unichar	xsd:string
univarchar	xsd:string

ASE のデータ型	SOAP のパラメータ型
nchar	xsd:string
nvarchar	xsd:string
binary	xsd:byte[]
varbinary	xsd:byte[]
xml	xsd:string
text	サポートされていません。
image	サポートされていません。
Java Data Type	サポートされていません。

用語解説

ASE Web Services Engine	Web サービスのプロデューサおよびコンシューマとして動作する ASE Web Services のコンポーネント。ASE Web Services Engine は、クライアント・アプリケーションが Web メソッドとして Adaptive Server Enterprise スタッド・プロシージャと SQL にアクセスできるようにします。また、ASE Web Services Engine は、クライアント・アプリケーションが SQL select 文で Web メソッドを呼び出せるように、Web メソッドをプロキシ・テーブルにマップします。
document/literal	通信者が WSDL ファイルに組み込まれている XML スキーマに従って転送およびフォーマットされるデータを指定する、Web メソッドのタイプ。
DTD	ドキュメント・タイプ定義 (Document Type Definition)。XML ドキュメントの有効なビルディング・ブロックを定義するために使用されます。DTD は、XML ドキュメント内で宣言することも、外部から参照することもできます。
HTTP	ハイパーテキスト転送プロトコル (Hypertext Transfer Protocol)。インターネット・プロトコル・スイートのアプリケーション・レイヤの一部であり、World Wide Web における情報交換の主要な手段です。
HTTPS	SSL 暗号化または TLS 暗号化を使用した HTTP。
LDAP	Lightweight Directory Access Protocol。インターネット・ベースのディレクトリ・サービスを実現するアプリケーション・レベルのプロトコルです。
RPC/encoded	各メソッド・パラメータの XML 要素を含む SOAP メッセージで呼び出される Web メソッドのタイプ。
SDS	Specialty Data Store。Web メソッドをプロキシ・テーブルにマップするための、コンポーネント統合サービス (CIS : Component Integration Service) として使用されます。
SOAP	Simple Object Access Protocol。XML ベースのネットワーク経由メッセージングの標準です。
SQLX	SQL/XML。SQL データベース・システムのコンテキストにおいて XML を使用できるようにする ANSI および ISO の標準です。
UDDI	Universal Description Discovery and Integration。
URI	Uniform Resource Identifier。インターネット・リソースを識別する文字列です。最も一般的な URI は、インターネット・アドレスを示す Uniform Resource Locator (URL) です。あまり一般的ではない URI として、Universal Resource Name (URN) があります。

URL	Uniform Resource Locator。URI の一種です。
W3C	World Wide Web Consortium。World Wide Web のソフトウェア標準を制定する団体です。
Web サービス	WSDL ファイルによって記述される 1 つ以上の Web メソッド。
Web メソッド	WSDL によって記述され、SOAP メッセージを介して呼び出される機能。
WSDL	Web Services Description Language。Web サービスへのパブリック・インタフェースを記述します。
Xerces	Apache オープンソース XML パーサ。
XML	Extensible Markup Language。W3C によって標準化されたマークアップ言語です。
XML スキーマ	構造と内容の制約から構成される、XML ドキュメントに関する記述。
XPath	XML Path Language。XML ドキュメントの各部分をアドレス指定するための言語です。
XQL	XML 問い合わせ言語。XQuery の原型です。
スキーマ (schema)	XML ドキュメントの構造、内容、セマンティックを定義するアウトライン。
ユーザ定義 Web サービス (User-defined Web service)	エンド・ユーザにより名前を付けられ、SOAP クライアントまたは Web ブラウザで実行される任意の SQL 文です。

索引

記号

() (カッコ) xiii
[] (角カッコ) xiii
{ } (中カッコ) xiii

A

add オプション
 構文 35
 使用法 36
 パラメータ 35
addalias オプション
 構文 46
 使用法 47
 パラメータ 47
Apache
 サンプル・クライアント 53
ASE
 SOAP へのデータ型のマッピング 83
ASE Web Services
 使用 29
 セキュリティ 24
 設定 21
 ディレクトリ・ツリー 75
 メソッド 31
 ライセンス 23
 利点 4
 ログイン 51
 ログ・ファイル 51
ASE Web Services Engine
 HTTP ハンドラ 13
 SOAP ハンドラ 13
 XML マップ 13
 起動 29
 コンシューマ 15
 コンシューマ・コンポーネント 15
 使用 29
 プロキシ・テーブル 17
 プロデューサ 12

プロデューサ Web メソッド 13
プロデューサ・コンポーネント 12
ユーザ定義 Web サービス 14
ASE Web Services の使用 29

B

Backus Naur Form (BNF) 表記 xiii
bin ディレクトリ 76

C

configssl 24
consumer.log 51
create service コマンド 42
 構文 43
 パラメータ 43
 例 44, 45

D

deploy オプション
 監査 50
 構文 47
 使用法 47
 パラメータ 47
document/literal Web メソッド 16
drop service コマンド 46
 構文 46
 パラメータ 46
 例 46
dropalias オプション
 構文 48
 使用法 48
 パラメータ 48

索引

E

- execute メソッド 32
 - 構文 32
 - パラメータ 32
 - 例 33, 34
- Extensible Markup Language 6

H

- help オプション
 - 構文 37
 - 使用法 37
 - パラメータ 37
- HTTP ハンドラ 13
- http.log 52

J

- JDBC レベルのトレース 69

L

- LDAP 5
- lib ディレクトリ 76
- list オプション
 - 構文 37
 - 使用法 37
 - パラメータ 37
- listalias オプション
 - 構文 48
 - 使用法 48
- listudws オプション
 - 構文 48
 - 使用法 48
 - パラメータ 48
- log4j 52
- logging.properties ファイル 23
- login メソッド 34
 - 構文 34
 - 使用法 34
 - パラメータ 34
- logout メソッド 34
 - 構文 34
 - 使用法 34
- logs ディレクトリ 77

M

- Microsoft .NET
 - SSL 証明書のインストール 25
 - サンプル・クライアント 57
- Microsoft .NET の証明書 25
- modify オプション
 - 構文 38
 - パラメータ 38
- myres.properties ファイル
 - 内容 81

N

- null パスワード 66

P

- producer ディレクトリ 77
- producer.log 52
- properties ファイル
 - エントリの指定 82
- props ディレクトリ 77

R

- remove オプション
 - 構文 38
 - パラメータ 38
- RPC/encoded Web メソッド 16
- runws の失敗 66

S

- samples ディレクトリ 78
- Simple Object Access Protocol, 「SOAP」参照 10
- SOAP 10
 - ASE へのデータ型のマッピング 83
 - Web サービス標準 10
 - メッセージ構造 10
- SOAP スタック 16
- SOAP ハンドラ 13

sp_webservices
 add オプション 35
 addalias オプション 46
 deploy オプション 47
 dropalias オプション 48
 help オプション 37
 list オプション 37
 listalias オプション 48
 listudws オプション 48
 modify オプション 38
 remove オプション 38
 undeploy オプション 48
 使用 35
 ユーザ定義 Web サービスでの使用 46
 SQL 4
 SQL での大文字と小文字の区別 xiv
 SSL
 設定 24
 stopws の失敗 66
 Sybase Central 52

U

undeploy オプション
 監査 50
 構文 48
 使用法 49
 パラメータ 49

W

Web Services Description Language、「WSDL」参照 8
 Web サービス
 概要 3
 標準 5
 呼び出し 39
 Web サービスの呼び出し 39
 例 39, 40, 41
 Web メソッド
 document/literal 16
 RPC/encoded 16
 ws.properties ファイル 23
 エントリ 65
 内容 79

WSDL 8
 Web サービス標準 8
 検索 65
 wsmsg.properties ファイル 24

X

XML 6
 Web サービス標準 6
 ドキュメント構造 7
 XML マップ 13

い

インストール
 内容 75

お

大文字と小文字の区別
 SQL xiv

か

角カッコ [] xiii
 角カッコ。「角カッコ []」参照
 監査
 deploy オプション 50
 undeploy オプション 50
 ユーザ定義 Web サービス 49
 関数 4

き

記号 xiii
 規約 xiii
 「構文」参照

こ

構文の表記規則 xiii

索引

さ

- サンプル・アプリケーション 53
 - Apache クライアント 53
 - Execute.exe 58
 - Microsoft .NET クライアント 57
 - runexecute 54

し

- 詳細なロギング 69
- 診断ツール 68
 - JDBC レベルのトレース 69
 - 詳細なロギング 69

す

- ストアド・プロシージャ 4

せ

- セキュリティ 5, 24
 - ユーザ定義 Web サービス 49
- 設定 21
 - logging.properties 23
 - logging.properties ファイル 23
 - ws.properties 23
 - ws.properties ファイル 23
 - wsmsg.properties ファイル 24
 - インストール後 22
 - インストール中 22
 - ファイル 23
 - プロパティ 79

ち

- 中カッコ ({}) xiii

て

- ディレクトリ
 - bin 76
 - lib 76
 - producer 77
 - props 77
 - サンプル 78
 - ログ 77
- データ型
 - create service 用の ASE から SOAP へのマッピング 85
 - SOAP と Adaptive Server Enterprise 83
 - SOAP と ASE 間のマッピング 83
 - マッピング 83

と

- トラブルシューティング 63
- トラブルシューティングの問題 63
 - ASE Web Services Engine の起動 64
 - null パスワード 66
 - runws または stopws の失敗 66
 - sp_webservices の異常終了 67
 - SSL での SOAP 終了ポイントの指定 67
 - sysattributes テーブル・エントリ 68
 - Web Services のプロキシ・テーブルの制約 67
 - ws.properties のエントリ 65
 - WSDL の検索 65
 - コマンド・ライン引数 66
 - トランケートされた document/literal の結果 64
 - マップを解除された RPC/encoded Web メソッド 64
 - リモート・サーバ・クラス定義の設定 63

は

- ハンドラ
 - HTTP 13
 - SOAP 13

ふ

プロキシ・テーブル 17
 プロパティ
 com.sybase.ase.ws.consumer.cisport 79
 com.sybase.ase.ws.consumer.logfilename 79
 com.sybase.ase.ws.interfaces 79
 com.sybase.ase.ws.libtcl 80
 com.sybase.ase.ws.maxpostsize 80
 com.sybase.ase.ws.producer.httpport 79
 com.sybase.ase.ws.producer.jettylogfile 79
 com.sybase.ase.ws.producer.logfilename 79
 com.sybase.ase.ws.producer.ssl.httpport 80
 com.sybase.ase.ws.producer.ssl.keypassword 80
 com.sybase.ase.ws.producer.ssl.keystore 80
 com.sybase.ase.ws.producer.ssl.password 80
 com.sybase.ase.ws.producer.tuning.maxidletime 80
 com.sybase.ase.ws.producer.tuning.maxthreads 80
 com.sybase.ase.ws.producer.tuning.minthreads 80
 com.sybase.ase.ws.producer.tuning.ssl.maxidletime 80
 com.sybase.ase.ws.producer.tuning.ssl.maxthreads 80
 com.sybase.ase.ws.producer.tuning.ssl.minthreads 80
 com.sybase.ase.ws.ui.activate 80
 ws.consumer.host 81
 ws.consumer.log 82
 ws.consumer.name 81
 ws.consumer.port 82
 ws.ini 81
 ws.libtcl 81
 ws.producer.jettylogfile 81
 ws.producer.log 81
 ws.producer.port 81
 ws.sqlsrv.sa_login 82
 ws.sqlsrv.sa_password 82
 ws.sqlsrv.server_name 82
 ws.ssl.certificatepassword 81
 ws.ssl.host 81
 ws.ssl.keystorelocation 81
 ws.ssl.keystorepassword 81

め

メソッド 31
 execute 32
 logout 34
 ログイン 34
 メッセージ 70

ゆ

ユーザー定義 Web サービス 5
 sp_webservices の使用 46
 監査 49
 コマンド 42
 使用 42
 セキュリティ 49

ら

ライセンス 23

ろ

ログイン 51
 consumer.log 51
 http.log 52
 producer.log 52
 ポリシーの設定 52
 ログ・ファイル 51

