

IBM® DB2® Universal Database™



Video Central for e-business チュートリアル

バージョン 8.1

IBM® DB2® Universal Database™



Video Central for e-business チュートリアル

バージョン 8.1

ご注意!

本書および本書で紹介する製品をご使用になる前に、87 ページの『付録 C. 特記事項』に記載されている情報をお読みください。

本書には、IBM の専有情報が含まれています。その情報は、使用許諾条件に基づき提供され、著作権により保護されています。本書に記載される情報には、いかなる製品の保証も含まれていません。また、本書で提供されるいかなる記述も、製品保証として解釈すべきではありません。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

原 典 :	IBM® DB2® Universal Database™ IBM Video Central for e-business tutorial Version 8.1
発 行 :	日本アイ・ビー・エム株式会社
担 当 :	ナショナル・ランゲージ・サポート

第1刷 2002.10

この文書では、平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、
平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 2001, 2002. All rights reserved.

© Copyright IBM Japan 2002

目次

図.	v	第 4 章 WebSphere Studio Application Developer の使用開始	17
第 1 章 このチュートリアルについて	1	所要時間	17
チュートリアルの概要	1	概説.	17
前提条件となるスキル	1	前提条件	17
チュートリアルの期間	1	Video Central の作業	17
始める前に.	2	作業 1: WAS リモート・サーバー・インス タンスの定義.	17
このチュートリアルで使用される規則.	2	作業 2: Video Central の Web プロジェク トの定義とソース・コードのインポート.	23
第 2 章 Video Central のサンプル・アプリケ ーション	3	学習のまとめ.	29
所要時間	3	次のステップ.	29
始める前に.	3	第 5 章 Video Central の設計の検討	31
概説	3	所要時間	31
Web サービスの発見	3	概説.	31
Web サービスとは	3	Video Central の設計	31
Web サービスへのアクセス方法.	5	Web インターフェース層 (WIL)	31
Video Central のシナリオ	6	ビジネス論理層 (BLL) の開発	31
ビジネスの背景	6	データのモデル化	41
Video Central のアーキテクチャー	7	学習のまとめ.	43
学習のまとめ.	10	次のステップ.	44
次のステップ.	10	第 6 章 Video Central のデプロイメント	45
第 3 章 Video Central 用 DB2 環境の使用 可能化	11	所要時間	45
所要時間	11	前提条件	45
開始前の準備.	11	概要.	45
システムの前提条件	11	Video Central の作業	45
概説.	11	作業 1: WebSphere Studio Application Developer から Video Central をデプロイす る環境の準備.	45
Video Central の作業	11	作業 2: WebSphere Application Server に対 する Web サービスのデプロイ.	47
作業 1: 'ivc' ユーザー ID の作成.	11	学習のまとめ.	54
作業 2: サンプル・データベースの作成.	12	次のステップ.	55
作業 3: サンプル・データベースでの DB2 XML Extender の使用可能化	12	第 7 章 Video Central 用の Web ベースの サンプル・クライアントの作成と実行	57
作業 4: サンプル・データベースでの DB2 Net Search Extender の使用可能化.	14	所要時間	57
作業 5: Video Central 表の作成と移植	14	前提条件	57
学習のまとめ.	15	概説.	57
次のステップ.	15		

Video Central サンプル・クライアントの設計	57	付録 A. 参照情報	79
Video Central サンプル・クライアントのク		DB2 XML Extender とは	79
ラスとシーケンスのダイアグラム	58	DB2 Net Search Extender とは	80
Video Central の作業	60	IBM WebSphere Studio Application Developer	
作業 1: クライアント・プロジェクトの準備	60	とは	81
作業 2: WebSphere Studio Application Developer を使用したプロキシの作成	68	Web サービス環境について	81
作業 3: WebSphere Application Server へのサンプル・クライアントのデプロイ	71	XML 開発環境について	82
作業 4: サンプル・クライアントの実行	72	付録 B. 用語集	85
学習のまとめ	73	付録 C. 特記事項	87
第 8 章 共通問題の解決	75	IBM と連絡をとる	89
第 9 章 追加情報	77	製品情報	89



1. Web サービスのコンポーネント	4	22. デプロイする Web サービスの選択	48
2. Video Central のアーキテクチャー	9	23. デプロイする Java Bean	49
3. 新しいサーバー・インスタンスの作成と 構成	19	24. Web サービスとしての Java Bean の構 成	50
4. サーバー・インスタンスの作成	20	25. デプロイするパブリック・メソッドの選 択	51
5. データ・ソースの構成	22	26. Web サービス・バインディング・プロキ シーの生成	52
6. 外部 JAR ファイルの追加	24	27. WSDL ファイルの内容	53
7. サーバーへの Web プロジェクトの追加	25	28. デプロイされるすべての Web サービス	54
8. サーバーへの Web プロジェクトの追加 (つづき)	26	29. Web サービスのコンポーネント	58
9. JAR ファイルのインポート前	27	30. Video Central サンプル・クライアント Video Client のクラス・ダイアグラム	59
10. インポートするリソースの選択	28	31. Video Central サンプル・クライアント Video Client のシーケンス・ダイアグラ ム	60
11. Video Central プロジェクトの準備後	29	32. 外部 JAR ファイルの追加	62
12. Video Central のクラス・ダイアグラム全 体	32	33. サーバーへのクライアント Web プロジ ェクトの追加	63
13. カスタマー違反行為 Web サービスのク ラス・ダイアグラム	34	34. ソースの強調表示	64
14. 希望商品リスト Web サービスのクラ ス・ダイアグラム	36	35. JAR ファイルから VideoClient プロジェ クトへのクライアント・コードのインポ ート	65
15. カスタマー違反行為の追加のシーケン ス・ダイアグラム	38	36. webApplication の強調表示	66
16. カスタマー違反行為の照会のシーケン ス・ダイアグラム	39	37. 「ファイル・システム (File System)」画 面	67
17. 希望商品リストの追加のシーケンス・ダ イアグラム	40	38. Video Central クライアント (VideoClient) のインポートの成功	68
18. データ・モデルのダイアグラム - 第 1 部	42	39. BusinessRegistration プロキシシーの生成	70
19. データ・モデルのダイアグラム - 第 2 部	43	40. すべての WSDL ファイルに対して作成 されたプロキシシー	71
20. HTTP サーバーの検査	46	41. クライアント・サーブレットの追加	72
21. WAS Administration Server が稼働中かど うかの検査	47		

第 1 章 このチュートリアルについて

チュートリアルの概要

IBM® Video Central チュートリアルは、ステップバイステップのガイドと完全な実働アプリケーションを含み、DB2® を Web サービス・プロバイダーとして示しています。Web サービスは、以下を含む新しいテクノロジーに基づいています。すなわち、Simple Object Access Protocol (SOAP)、Web サービス記述言語 (WSDL)、および Universal Description, Discover and Integration (UDDI) 仕様です。SOAP は、Web サービスの起動に使用される標準メッセージ・フォーマットです。Video Central チュートリアルのアプリケーションは、インターネット標準プロトコルである Hyper Text Transfer Protocol (HTTP) を使用して SOAP メッセージを移送します。Web サービス・リクエスターは、Web サービス・プロバイダーの WSDL 文書を使用して開発できます。Web サービスの詳細については、<http://www.ibm.com/software/webservices> を参照してください。

Web サービスは、アプリケーション統合用の新しい開発モデルです。これは分散コンピューティング環境でビジネス論理を記述、公開、発見、および起動するための標準手法を提供します。このチュートリアルには、6 つの Web サービスを WebSphere® Application Server (WAS) に首尾よくデプロイするために必要な完全な資料が含まれています。作動中の Web サービスを例示するために、クライアント・アプリケーションも備わっています。DB2 XML Extender の動的 XML マッピング機能、および DB2 Net Search Extender の高速フルテキスト検索エンジンが、このチュートリアルで例示されています。

前提条件となるスキル

このチュートリアルは、以下に関する実際的な知識を持つプロの Java™ プログラマーを対象としています。

- WebSphere Application Server
- Web サービスの概念 (詳細については、『第 3 章、Video Central 用 DB2 環境の使用可能化』を参照してください)
- DB2 Universal Database™
- オブジェクト指向プログラミングの概念
- Java プログラム言語

チュートリアルの期間

このチュートリアルは、約 2 時間で完了します。

始める前に

以下の製品はこのチュートリアルで使用されるので、開始する前にインストールしておいてください。

- WebSphere Application Server (WAS) バージョン 4.0 Advanced Edition Single Server
- DB2 Universal Developer's Edition (DB2 UDE) バージョン 8.1。これには、DB2 XML Extender および DB2 Net Search Extender が含まれます (XML および Net Search Extender は同梱されていますが別の CD 上にあり、別個にインストールする必要があります)
- WebSphere Studio Application Developer (WSAD) バージョン 4.0

このチュートリアルで使用される規則

このチュートリアル全体で、以下の規則が適用されます。

- <DB2_install_dir> は DB2 のインストール・ディレクトリーを指すものとして使用されています。
- <WAS_install_dir> は、 WebSphere Application Server のインストール・ディレクトリーを指すものとして使用されています。
- <WSAD_install_dir> は、 WebSphere Studio Application Developer のインストール・ディレクトリーを指すものとして使用されています。

第 2 章 Video Central のサンプル・アプリケーション

所要時間

この章は、約 20 分で完了します。

始める前に

オブジェクト指向プログラミングの概念を理解している必要があります。

概説

この章では、Web サービスおよび Simple Object Access Protocol (SOAP) について学びます。さらに、Video Central にインプリメントされた Web サービス、および Video Central のアーキテクチャーについても学びます。

Web サービスの発見

以下のセクションでは、Web サービスについて概説しています。

Web サービスとは

Web サービスは、他のアプリケーションから起動できる関連した複数のアプリケーション機能のセットです。HTML を使用して情報を表示する従来の Web アプリケーションとは異なり、Web サービスは XML を使用して情報を表示します。Web サービスは多くの異なるプログラム言語およびプラットフォームを使用して、インプリメントおよびアクセスすることができます。このチュートリアルでは Java 言語を活用して、Web サービス・リクエスターおよびプロバイダーを Windows® プラットフォーム上にインプリメントします。

Web サービスは、特定のタスクを実行し、HTTP などのオープン・プロトコルを介してアクセス可能なモジュラー・アプリケーションです。企業は Web サービスを動的に混合および突き合わせして、フレキシブルなプログラミング・モデルを使用した複雑な操作を実行することができます。

Web サービスの概念は、3 つのコンポーネントの役割に依存しています (図 1 を参照)。

サービス・プロバイダー

- e-business サービスを提供します。
- これらのサービスの可用性をレジストリーを介して公開します (オプション)。

サービス・ブローカー

- サービスの公開および位置決めをサポートします。

サービス・リクエスター

- サービス・ブローカーを使用して必要なサービスを見付けます。
- サービス・プロバイダーから使用可能なサービスにバインドします。
- 定義済みインターフェースを使用してサービスを呼び出します。

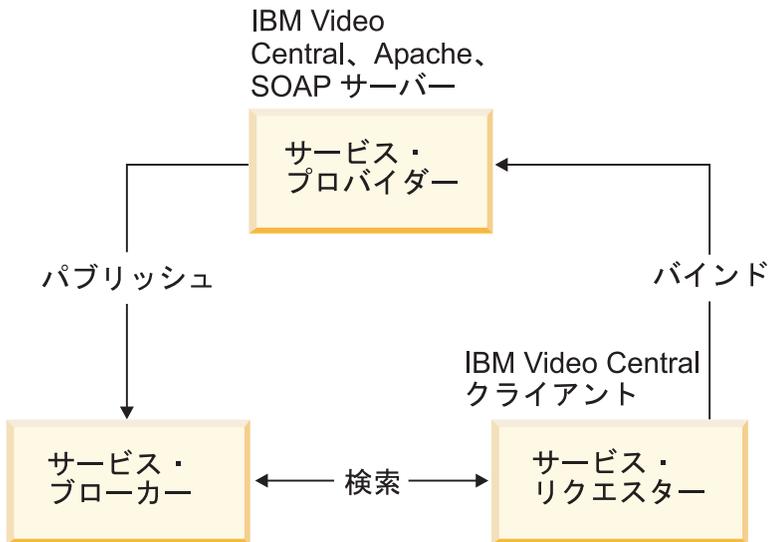


図1. Web サービスのコンポーネント

Web サービスでビジネスがパートナーシップを形成する合流点は、 Universal Description, Discovery and Integration (UDDI) レジストリーです。 UDDI の仕様は、プラットフォームに依存しないオープンな標準を定義しています。ビジネスでこれを使って、グローバル Business Registry にある情報の共有、相互のサービスの発見、およびそれらのインターネット上での相互作用の方法を定義することができます。このレジストリーは機能の膨大なライブラリーになることができ、開発者はそれを使って動的なアプリケーションを迅速に作成できます。 IBM は、すでに業界の数十のリーダーが支持している UDDI 仕様の定義作業での主要な参加メンバーの中の 1 社です。

Web サービスは、Web サービス記述言語 (WSDL) を使用して、UDDI Business Registry に発行されます。それらは Simple Open Access Protocol (SOAP) を使用して相互に対話を行います。 Web サービスは、次の 3 つのカテゴリーに分けることができます。

- **ビジネス情報** - ここでは、ビジネスは消費者や他のビジネスと情報を共有します。この場合、ビジネスは Web サービスを使ってビジネスの範囲の拡張を行います。ビジネス情報の Web サービスの例としては、ニュース・ストリーム、地方の天気予報、株式市況などがあります。
- **ビジネス・インテグレーション** - ここでは、ビジネスはトランザクションのサービスをその消費者に対して無料で提供します。この場合、ビジネスはサプライヤーのグローバル値ネットを使用でき、取引を行うときにそれを活用することができます。ビジネス・インテグレーション Web サービスの例としては、入札およびオークションの e-マーケットプレイス、予約システム、および信用調査などがあります。
- **ビジネス・プロセス外面化** - ここでは、ビジネスはグローバル値チェーンを作成することによって、競争相手との違いを明確にします。この場合、ビジネスは Web サービスを使って、そのプロセスを統合します。

UDDI についての詳細は、<http://www.uddi.org> を参照してください。

Web サービスへのアクセス方法

企業は、そのビジネス機能を発行して Web 上に提供することができます。潜在的なカスタマーやクライアントは、UDDI Business Registry などのレジストリーを介して、Web 上でそれらのサービスの可用性を発見します。

Web サービスは、分散モデルおよびクライアント/サーバー・モデルが自然に拡張されたものです。Web サービスでは、メソッドは SOAP と呼ばれる XML 標準形式を使って、リモート・オブジェクト上で呼び出されます。SOAP メッセージは、MQSeries[®] や HTTP などの各種トランスポート・プロトコルを使って交換されます。UDDI レジストリーは、使用可能なサービスおよびサービス・プロバイダーのディレクトリーを提供します。このインターフェースは、XML 標準からは独立している言語である WSDL を使用して定義されます。

IBM では、主要製品で Web サービス機能を使用できるようにしています。Web サービスの自動的な生成、発行、検索、テスト、および呼び出しを行うツールが使用可能です。このチュートリアルでは、それらのツールの幾つかについて使用方法を解説します。IBM では SQL ステートメントをサブミットし、(選択した場合は) 戻されたデータの形式を制御するための別の方法も提供しています。このサポートは、Web services Object Runtime Framework (WORF) および Document Access Definition eXtension (DADX) と呼ばれます。詳しくは、次を参照してください。

<http://www.ibm.com/software/data/pubs/papers/db2webservices/db2webservices.pdf>

Video Central のシナリオ

Video Central は、ビデオ・レンタル・アプリケーション (企業間取引 (B2B) アプリケーション) の包括的な Web サービス・プロバイダーです。Video Central の目的は、Web ベースの登録済みのアプリケーションがアクセスできる中央データ・リポジトリを提供することです。Video Central アプリケーションは、クライアントの 2 つの領域のビジネス・ニーズに対応できる一連の Web サービスを提供します。1 つはクライアントの管理上のニーズに対応する **ビジネス・サービス**、もう 1 つはクライアントが自身のカスタマーへのサービスの提供と管理の両方を行える **カスタマー・サービス** です。

ビジネスの背景

ビデオ・レンタル・ビジネスの所有者は、レンタル・ビジネスの日常の業務で、次の制約的な要因に直面しています。

- 新規レンタルのタイトル情報の手動によるデータ入力の繰り返し。
- タイトルの説明が限定される (ストーリー、キャラクター、製作スタッフなどの情報の手動入力)。
- カスタマーのレンタル履歴や好みに基づいてカスタマーにタイトルを推奨できない。
- 付加価値サービス (希望商品リストやこれまでのレンタル履歴) の中に、ローカル・データに限定されるものがある。
- 高価なレンタル品目の場合に必要なら、新規または既存のカスタマーの信用の評価を限定的にしか行えない。
- 新規カスタマー (借りて姿を消す) によるタイトルの盗難が原因の収益の損失。

これらをまとめると、レンタル・ビジネスの所有者のデータ・アクセスは、所有者自身の独自のデータベース上の情報に限られているということです。現段階の Video Central は、以下のサービスを提供します。

Video Central ビジネス・サービス

ビジネスに関連した Web サービスが 1 つ設計されており、このチュートリアルで紹介されています (ビジネス登録)。

ビジネス登録: この Web サービスを使用すると、クライアント (たとえばビデオ店) は、そのビジネスを Video Central に登録することができます。ビジネスを Video Central に登録すると、クライアントは他の使用可能な Web サービスを利用することができます。

Video Central カスタマー・サービス

カスタマーに関連した特定の Web サービスが 5 つ設計されています (カスタマー登録、カスタマー違反行為、カスタマー・レンタル・リスト、カスタマー希望商品リスト、ムービー検索)。

カスタマー登録: この Web サービスを使用すると、登録済みビジネス (つまりビデオ店) は、自身のカスタマーを Video Central に登録することができます。カスタマーを登録すると、使用可能な他のサービスを利用して、カスタマーに関する有用な情報 (たとえばレンタル履歴、希望商品リスト、違反行為) の保管、アクセス、および分析が可能です。

カスタマー違反行為: この Web サービスを使用すると、登録済みビジネスは、カスタマーが犯した違反行為を Video Central リポジトリに追加することができます。これによって、ビジネスは自身のカスタマーの違反行為を、そのデータをローカル・データベースに実際に保管せずに把握することができます。また、この Web サービスでは、登録済みビジネスは潜在的または既存のカスタマーの違反行為をいつでも照会することができます。Video Central は多数のビデオ・ストアの中央リポジトリであるため、カスタマーに新規アイテムを貸し出す前に、そのカスタマーに別の店で違反行為があるかどうかを見付けることができます。

カスタマー・レンタル・リスト: この Web サービスを使用すると、登録済みビジネスは、カスタマーごとにこれまでに貸し出したビデオのリストを追跡記録することができます。また、この Web サービスでは、登録済みビジネスはカスタマーのレンタルをいつでも照会することができます。

カスタマー希望商品リスト: この Web サービスを使用すると、登録済みビジネスは、カスタマーが画面で関心を示したビデオのタイトルを追加することができます。これによってカスタマーのそれらのタイトルを記録し、カスタマーの希望商品リストをいつでも照会することができます。さらに、ビデオ店はこのリストを使用して、自身のカスタマーが要望しているビデオを判断し、在庫を変更することができます。

ムービー検索: この Web サービスを使用すると、登録済みビジネスは、ムービーのタイトルとムービーのプロットの要約を検索することができます。すべてのデータを手動でローカル・データベースに入力する代わりに、常に更新される最新のムービー情報をオンラインで検索できます。この Web サービスはストアのスタッフやカスタマーが利用して、ビデオ・レンタルの意思決定時の参考にすることができます。

Video Central のアーキテクチャー

Video Central アプリケーションは、Web サービスのアーキテクチャーに基づいています。Video Central の設計についての詳細は、『第 5 章 Video Central の設計の検討』で説明します。このアーキテクチャーの一環になっていることは、Video Central は本質的には、(インターネット上の) e-business サービスの動的で柔軟なプロバイダーであるということです。

Video Central が提供するサービスを使用するサービス・リクエスターを開発することができます。サービス・リクエスターが Video Central のサービスを要求するときは、その要求をクライアント・マシン上のサービス・プロキシ・オブジェクトに送信します。プロキシはその要求と関連データを XML 形式にパックして、その XML データ

を SOAP でエンコードされたメッセージに入れて送信します。 SOAP ランタイム・サーブレットは WAS バージョン 4.x で提供されています。 SOAP ランタイム・サーブレットが要求を受信すると、その要求を処理する関連サービスがインプリメントされている該当 Java Bean にその要求を委任します。

Video Central アプリケーションは、次の 3 層のアプリケーションとしてインプリメントされています。

1. Web インターフェース層 (WIL)
2. ビジネス・ロジック層 (BLL)
3. データ・アクセス層 (DAL)

この 3 層設計により、 Video Central のインプリメンテーションはコンポーネント・ベースの柔軟なものになっています。この拡張可能なフレームワークにより、将来の拡張が容易に行えます。

Web インターフェース層は、Web サービス、HTTP サーブレット、 JavaServer Pages (JSP) などの各種 Web インターフェースをサポートするようにインプリメントすることができます。このことは、エンド・ユーザーのインターフェース設計に最大限の柔軟性を与えます。 Video Central は、Web サービスの呼び出しに SOAP ランタイムを使用することにより、 Web サービスをインプリメントしています。

ビジネス・ロジック層は抽象概念レベルの Web サービスを使用して設計されています。各 Web サービスで一連のビジネス・ロジックが定義されており、インターフェースは WSDL (Web サービス記述言語) を使って記述されています。そして WSDL インターフェースは、公開または専用 UDDI (Universal Description, Discovery, and Integration) レジストリーに発行することができます。 Video Central アプリケーションの主要コンポーネントは、 DB2 XML Extender を使って、カスタマー違反行為リポジトリーの保管と検索を管理します。そのほか Video Central は DB2 Net Search Extender を使用しており、その高速テキスト検索エンジンによってムービー検索 Web サービスが容易になっています。

データ・アクセス層は、 IBM Data Access beans (Java Database Connectivity (JDBC) 用ラッパー) を使用して、 Data Access コンポーネント内に統合されています。ビジネス・ロジックがインプリメントされている JavaBeans™ は、 Data Access コンポーネントを使用してデータの保管と検索を行います。 Video Central アプリケーションのデータベース・サーバーとして、DB2 UDB を使用します。

図 2 は Video Central アプリケーションのアーキテクチャーを示したものです。

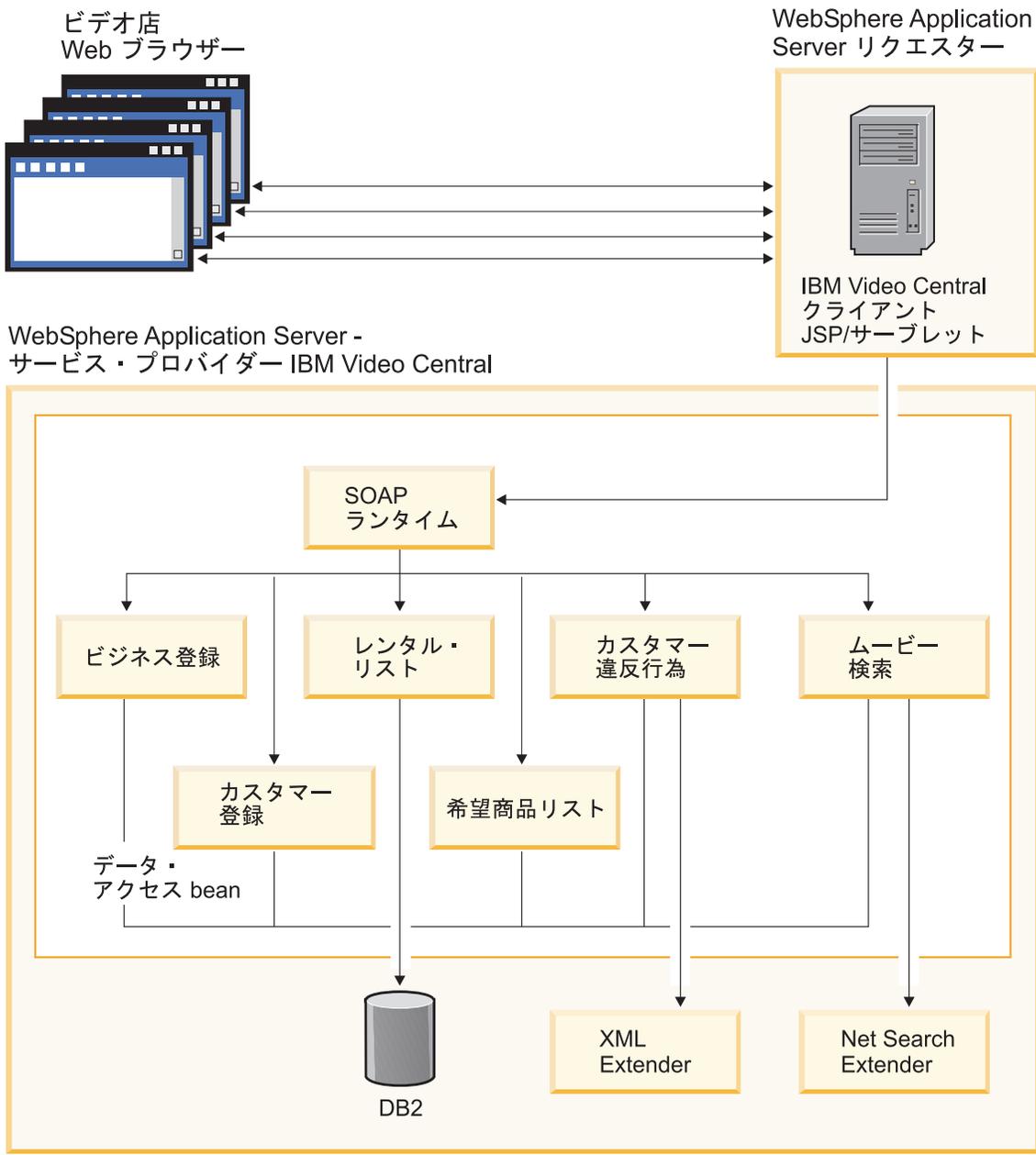


図2. Video Central のアーキテクチャー

学習のまとめ

この章では、以下の点を習得しました。

- Web サービスについて学習しました。
- Video Central アプリケーションの高レベルのアーキテクチャーと設計を理解しました。

次のステップ

次は、必要な DB2 開発環境について学習し、Video Central アプリケーションを配置するための DB2 開発環境を準備します。

第 3 章 Video Central 用 DB2 環境の使用可能化

所要時間

この章を完了するのに要する時間は、約 15 分です。

開始前の準備

システムの前提条件

- Microsoft® Windows NT®, Version 4.0 (Service Pack 6a 以上)、または Windows 2000
 - Java 2 SDK Standard Edition、バージョン 1.3
 - DB2 Universal Developer's Edition (DB2 UDE) バージョン 8.1
 - DB2 XML Extender (DB2 UDE バージョン 8.1 に別個の CD として同梱されており、別個のインストールが必要)
 - DB2 Net Search Extender (DB2 UDE v8.1 に別個の CD として同梱されており、別個のインストールが必要)
-

概説

このセクションでは Video Central 用の DB2 環境のセットアップを解説します。ここには次の内容が含まれています。

- Video Central データベースで DB2 XML Extender および DB2 Net Search Extender が使用できるようにする。

さらに次のことも行います。

- 名前が `ivc` の新規ユーザーを作成する。
 - Video Central が必要とする表を作成する。
-

Video Central の作業

作業 1: 'ivc' ユーザー ID の作成

これをまだ行っていない場合は、ご使用のオペレーティング・システム上で `ivc` ユーザー ID を作成します。

Windows 2000 の場合:

1. 「スタート」 -> 「設定」 -> 「コントロール パネル」とクリックします。

2. 「管理ツール」 -> 「コンピューターの管理」 -> 「ローカル ユーザーとグループ」と選択します。
3. メニュー・バーから右クリックで、「ユーザー」 -> 「新しいユーザー」と選択します。
4. 名前が ivc、パスワードが VideoCentral の新規ユーザーを作成します。「ユーザーは次回ログオン時にパスワードの変更が必要」の選択を解除します。

Windows NT の場合:

1. 「スタート」 -> 「管理ツール」 (共通) -> 「ユーザー マネージャ」とクリックします。
2. メニュー・バーから右クリックで、「ユーザー」 -> 「新しいユーザー」と選択します。
3. 名前が ivc、パスワードが VideoCentral の新規ユーザーを作成します。「ユーザーは次回ログオン時にパスワードの変更が必要」の選択を解除します。

作業 2: サンプル・データベースの作成

このチュートリアルでは、Video Central データベースとして sample データベースを使用します。sample データベースを作成するには、次のいずれかを実行します。

- コマンド行から db2samp1 を実行します。
- 「スタート」 -> 「プログラム」 -> 「IBM DB2」 -> 「ファースト・ステップ (First Steps)」とクリックして、「ファースト・ステップ (First Steps)」ウィンドウの「サンプル・データベースの作成 (Create Sample Databases)」をクリックします。

作業 3: サンプル・データベースでの DB2 XML Extender の使用可能化

XML Extender で DB2 の XML 文書の保管や検索を行うには、まずデータベースで XML を使用できるようにする必要があります。

データベースで XML (DB2 XML Extender) を使用できるようにするには、以下のことを行います。

- すべてのユーザー定義タイプ (UDT) およびユーザー定義関数 (UDF) を作成します。
- コントロール表を作成して、XML Extender が必要とするメタデータを取り込みます。
- db2xml スキーマを作成し、必要な特権を割り当てます。

sample データベースで DB2 XML Extender を使用できるようにするステップを以下に示します。

1. DB2 コマンド・ウィンドウで、
<DB2_install_dir>%tutorial%VideoCentral%db%dbEnableXML.bat を実行します。
このスクリプトによって次のことが行われます。

- a. sample データベースへの接続。
- b. sample データベースへの XML Extender バインド・ファイルのバインド。
- c. XML Extender の使用可能化。

注: データベースで XML を使用不可にする場合は、スクリプト

```
<DB2_install_dir>%tutorial%VideoCentral%db%dbDisableXML.bat を指定します。このチュートリアル目的ではこれを行う必要はありません。
```

2. DB2 コマンド・ウィンドウで次のコマンドを入力して、DB2 XML Extender が使用可能であることを検査します。
 - a. db2 connect to sample
 - b. db2 select procname from syscat.procedures where procschema='DB2XML'

以下のストアード・プロシージャ名のリストが表示されます。

- DXXDISABLECOLL
- DXXDISABLECOLUMN
- DXXENABLECOLL
- DXXENABLECOLUMN
- DXXGENXML
- DXXGENXMLCLOB
- DXXINSERTXML
- DXXRETRIEVEXML
- DXXRETRIEVEXMLCLOB
- DXXSHREDXML

Video Central アプリケーションは、これらのストアード・プロシージャの中の DXXGENXML と DXXSHREDXML の 2 つを使用します。これらは両方ともカスタマー違反行為 Web サービスで使用されます。DXXGENXML はカスタマー違反行為の照会時に、カスタマー違反行為を DB2 から XML 文書として検索するときに使用されます。DXXSHREDXML は、XML 文書として受信したカスタマー違反行為を DB2 に追加するときに使用されます。

XML Extender に関する追加情報は、*XML Extender Administration and Programming* を参照してください。このマニュアルは、<http://www.ibm.com/software/data/db2/extenders/xmlext/library.html> で見ることができます。

作業 4: サンプル・データベースでの DB2 Net Search Extender の使用可能化

1. DB2 コマンド・ウィンドウで、
<DB2_install_dir>%tutorial%VideoCentral%db%dbEnableNX.bat を実行します。
このスクリプトによって次のことが行われます。
 - a. DB2 の開始。
 - b. sample データベースへの接続。
 - c. DB2 Net Search Extender デーモンの開始。
 - d. データベースでの Net Search Extender の使用可能化。
 - e. テキスト列の使用可能化 (Video Central では、Net Search は ivc.titleinfo.name と ivc.titleplot.plotssummary の列に索引付けされます)。

注: データベースで DB2 Net Search Extender を使用不可にする場合は、スクリプト <DB2_install_dir>%tutorial%VideoCentral%db%dbDisableNX.bat を指定します。このチュートリアルの目的ではこれを行う必要はありません。

2. <DB2_install_dir>%tutorial%VideoCentral%db%movieSearch.log を開き、エラーを検査して、DB2 Net Search Extender が使用可能であることを検査します。

注: エラーがある場合は、『第 8 章 共通問題の解決』を参照してください。

Net Search Extender に関する追加情報は、
<http://www.ibm.com/software/data/db2/extenders/netsearch> 上のマニュアル *Net Search Extender Administration and Programming*、および『付録 A 参照情報』を参照してください。

作業 5: Video Central 表の作成と移植

1. DB2 コマンド・ウィンドウを開いてバッチ・ファイルを実行します。
2. DB2 コマンド・ウィンドウで、
<DB2_install_path>%tutorial%VideoCentral%db%tblCreate.bat を実行します。
このスクリプトによって次のことが行われます。
 - a. sample データベースへの Video Central に必要な表の作成。
 - b. 必要な索引と制約の作成。
 - c. 必要な表 (コード表、DAD 表、Movie Search 表) の移植。

注: Video Central の実行を止めて Video Central 表を除去する場合は、スクリプト <DB2_install_dir>%tutorial%VideoCentral%db%tblClean.bat を指定します。このチュートリアルの目的ではこれを行う必要はありません。

3. DB2 コマンド・ウィンドウで、次のコマンドを入力します。
 - a. DB2 connect to sample

b. DB2 list tables for schema ivc

次の 10 の表のリストが表示されます。

- BUSINESSINFO
- DAD
- QUERY_RESULT_TAB
- TITLEINFO
- TITLEPLOT
- USERINFO
- USERINFRACT
- USERINFRACT_TMP
- USERRENTALS
- USERWISHES

学習のまとめ

この章では、以下の点を習得しました。

- 名前が ivc の新規ユーザーを作成しました。
- sample データベースを作成しました。
- Video Central データベースで DB2 XML Extender および DB2 Net Search Extender を使用できるようにしました。
- Video Central が必要とする表を作成して移植しました。

次のステップ

次は、必要な開発環境について学習し、 Video Central を実行するための開発環境を準備します。

第 4 章 WebSphere Studio Application Developer の使用開始

所要時間

この章を完了するのに要する時間は、約 20 分です。

概説

この章では、IBM WebSphere Studio Application Developer を簡潔に紹介します。Java 開発環境を Web サービス用にセットアップする方法を説明します。

前提条件

- WebSphere Application Server (WAS) バージョン 4.0、Advanced Edition Single Server
- WebSphere Studio Application Developer (WSAD) バージョン 4.0

Video Central の作業

作業 1: WAS リモート・サーバー・インスタンスの定義

この作業で WebSphere Studio Application Developer 内に Video Central サーバー・プロジェクトを作成し、WAS リモート・サーバーの定義を作成します。

WebSphere Studio Application Developer の開始

「スタート」 → 「プログラム」 → 「IBM WebSphere Studio Application Developer」 → 「IBM WebSphere Studio Application Developer」とクリックして、WebSphere Studio Application Developer を開始します。

Video Central サーバーの作成と構成

1. Video Central サーバーを作成するには、次のようにします。
 - a. 「サーバー (Server)」パースペクティブで、「ファイル (File)」 → 「新規 (New)」 → 「サーバー・プロジェクト (Server Project)」をクリックしてサーバー・プロジェクトを作成します。
 - b. このプロジェクトを VideoCentralServer と命名します。「完了 (Finish)」をクリックします。
2. 次のようにして、新しいサーバー・インスタンスを作成して構成します。

- a. 「サーバー (Server)」 パースペクティブで、「サーバー構成ナビゲーション (Server Configuration navigation)」 パネル内の「**サーバー構成 (Server Configurations)**」をマウスの右ボタンでクリックしてから、「**新規 (New)**」 → 「**サーバー・インスタンスと構成 (Server Instance and Configuration)**」を選択します。
- b. サーバーに VideoCentralServer という名前を付けてから、「サーバー・インスタンス・タイプ (Server instance type)」内の「**WebSphere サーバー (WebSphere Servers)**」 → 「**WebSphere v4.0 リモート・サーバー (WebSphere v4.0 Remote Server)**」を選択します。次のような表示になります (図 3 を参照)。

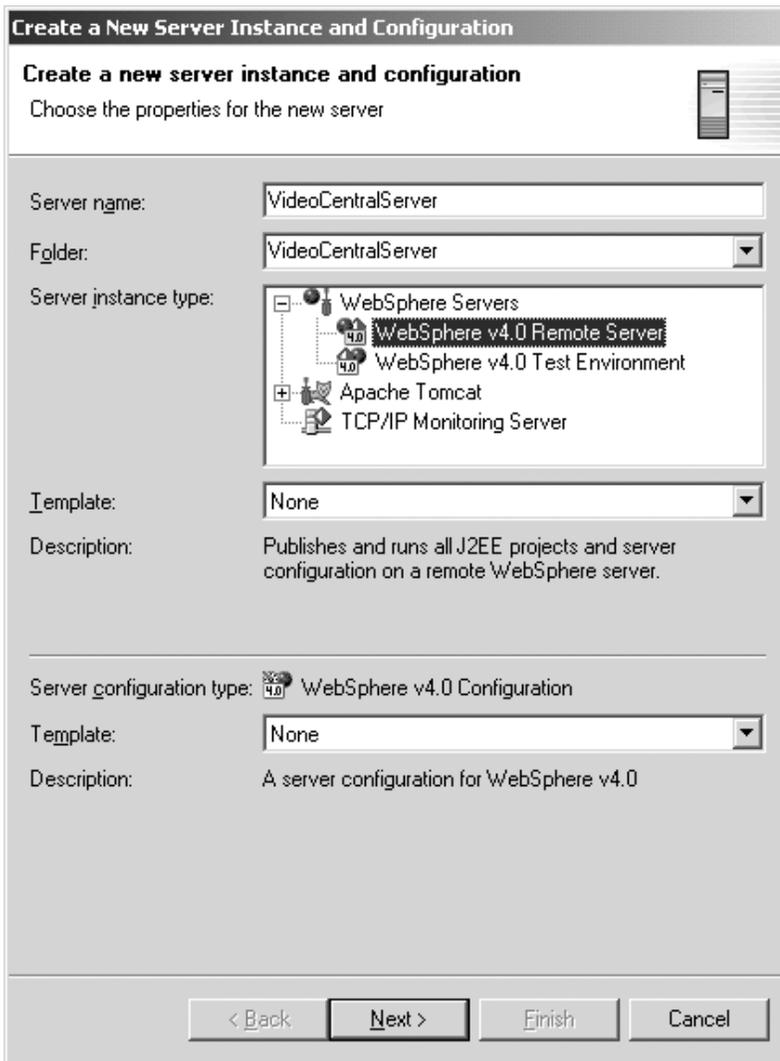


図3. 新しいサーバー・インスタンスの作成と構成

- c. 「次へ (Next)」をクリックします。
- d. 「ホスト・アドレス (Host Address)」フィールドで、127.0.0.1 を localhost に置き換えます。
- e. 「WebSphere インストール・ディレクトリー (WebSphere installation directory)」フィールドで、WebSphere Application Server 4.0 Advanced Edition Single Server をインストールした場所 (つまり、<WAS_install_dir>) を入力します。「次へ (Next)」をクリックします。

- f. 「新規リモート・ファイル転送インスタンスの作成 (Create a new remote file transfer instance)」 → 「ファイル転送機構のコピー (Copy file transfer mechanism)」を選択します。「次へ (Next)」をクリックします。
- g. 「リモート・ターゲット・ディレクトリー (Remote target directory)」フィールドで、WebSphere Application Server 4.0 Advanced Edition Single Server をインストールした場所 (つまり、<WAS_install_dir>) を入力します。「次へ (Next)」をクリックします。
- h. HTTP ポート番号を 8080 ではなく 9080 に設定します。「完了 (Finish)」をクリックします。これで、ナビゲーター・パネルは次のような表示になります (図 4 を参照)。

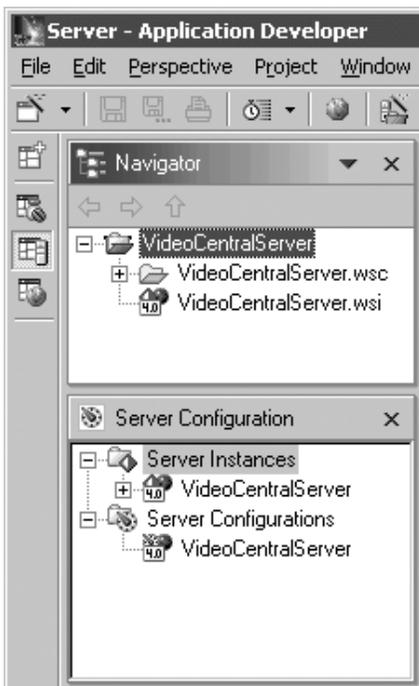


図 4. サーバー・インスタンスの作成

3. 「サーバー構成 (Server Configurations)」パネルの「サーバー構成 (Server Configurations)」フォルダーの下にある「VideoCentralServer」→「オープン (Open)」を右マウス・ボタンでクリックします。
 - a. 「その他 (General)」タブで「Administration Client の使用可能化 (Enable Administration Client)」を選択します。
 - b. EJB タブで「EJB テスト・クライアントの使用可能化 (Enable EJB test client)」の選択を解除します。

- c. 「ファイル (File)」 → 「VideoCentralServer の保管 (Save VideoCentralServer)」をクリックします。
 - d. 「VideoCentralServer」ウィンドウをクローズします。
4. 次のようにして、データ・ソースを構成します。
- a. 同じウィンドウで「データ・ソース (Data source)」タブを選択してから、「JDBC ドライバー (JDBC driver)」リストで「Db2JdbcDriver」を強調表示します。
 - b. 次のようにして、サンプル・データベースを追加します。
 - 1) 「上記で選択した JDBC ドライバーに定義されているデータ・ソース (Data source defined in the JDBC driver selected above)」ボックスの横の「追加 (Add)」ボタンをクリックします。
 - 2) 「名前 (Name)」フィールドに sample と入力します。
 - 3) 「JNDI 名 (JNDI name)」フィールドに jdbc/Sample と入力します (注: 「JNDI 名 (JNDI name)」フィールドでは大文字小文字が区別されるので、ここでは Sample の先頭文字を大文字化する必要があります)。
 - 4) 「データベース名 (Database name)」フィールドに sample と入力します。
 - 5) 「デフォルト・ユーザー ID (Default User ID)」フィールドに ivc と入力します。
 - 6) In the 「デフォルト・ユーザー・パスワード (Default User password)」フィールドに VideoCentral と入力します。
 - 7) 「OK」をクリックします。

注: Video Central アプリケーションは、VideoCentral.properties に保管されているユーザー ID とパスワードを使って DB2 への接続を確立します。

次のような表示になります (図 5 を参照)。

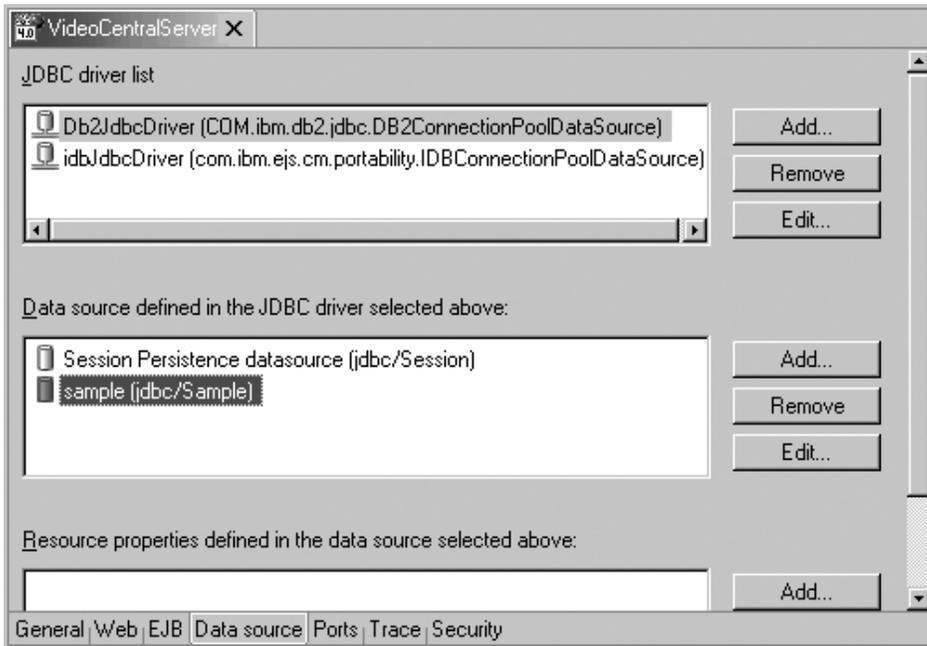


図 5. データ・ソースの構成

- c. **X** をクリックし、変更を保管するかどうかを尋ねるウィンドウがポップアップしたら、「はい (Yes)」を選択して「VideoCentralServer」をクローズします。
5. 「サーバー構成 (Server Configuration)」パネルの「サーバー・インスタンス (Server Instances)」フォルダーの下にある「VideoCentralServer」→「オープン (Open)」を右マウス・ボタンでクリックします。
 - a. 「パス (Paths)」タブを選択し、「外部 JAR の追加 (Add External JARs)」をクリックして以下を追加します。
 - <DB2_install_dir>%java%db2java.zip
 - <DB2_install_dir>%tools%databean.jar
 - <WAS_install_dir>%lib%j2ee.jar

注: db2java.zip ファイルは、DB2 データベースへのアクセスのための JDBC 2.1 サポートとして機能する一連のクラスです。それらのクラスは、JDBC タイプ 2 ドライバー・アーキテクチャーを使って DB2 サーバーと通信することになります。
 - b. **X** をクリックし、変更を保管するかどうかを尋ねるウィンドウがポップアップしたら、「はい (Yes)」を選択して「VideoCentralServer」をクローズします。

作業 2: Video Central の Web プロジェクトの定義とソース・コードのインポート

Video Central Web プロジェクトの作成

1. 次のようにして、VideoCentral Web プロジェクトを作成します。
 - a. Web パースペクティブで「ファイル (File)」 -> 「新規 (New)」 -> 「Web プロジェクト (Web project)」をクリックし、新しい Web プロジェクトを作成します。そのプロジェクトを VideoCentral と命名し、Enterprise Application Project を VideoCentralEAR と命名します。「次へ (Next)」をクリックします。
 - b. 「モジュールの依存関係 (Module Dependencies)」画面が表示されます。すべて、デフォルトのままにします。「次へ (Next)」をクリックします。
 - c. これで、「Java ビルド設定の定義 (Define Java Build Settings)」画面が表示されるはずです。「ライブラリー (Libraries)」タブを選択してから「外部 JAR の追加 (Add External JARs)」をクリックし、「JAR 選択 (JAR Selection)」ポップアップ・ウィンドウ内で次のような外部 JAR を追加します。

•

```
<WSAD_install_dir>%plugins%com.ibm.etools.webservices%runtime%soap.jar
```

- <DB2_install_dir>%tools%databean.jar

注: soap.jar が Video Central にとって必要なのは、Web サービスから要求された SOAP メッセージの構文解析を実行するためです。databean.jar は、Java から簡単に DB2 にアクセスできるようにするための IBM 提供の一連のクラスです。

次のような表示になります (図 6 を参照)。

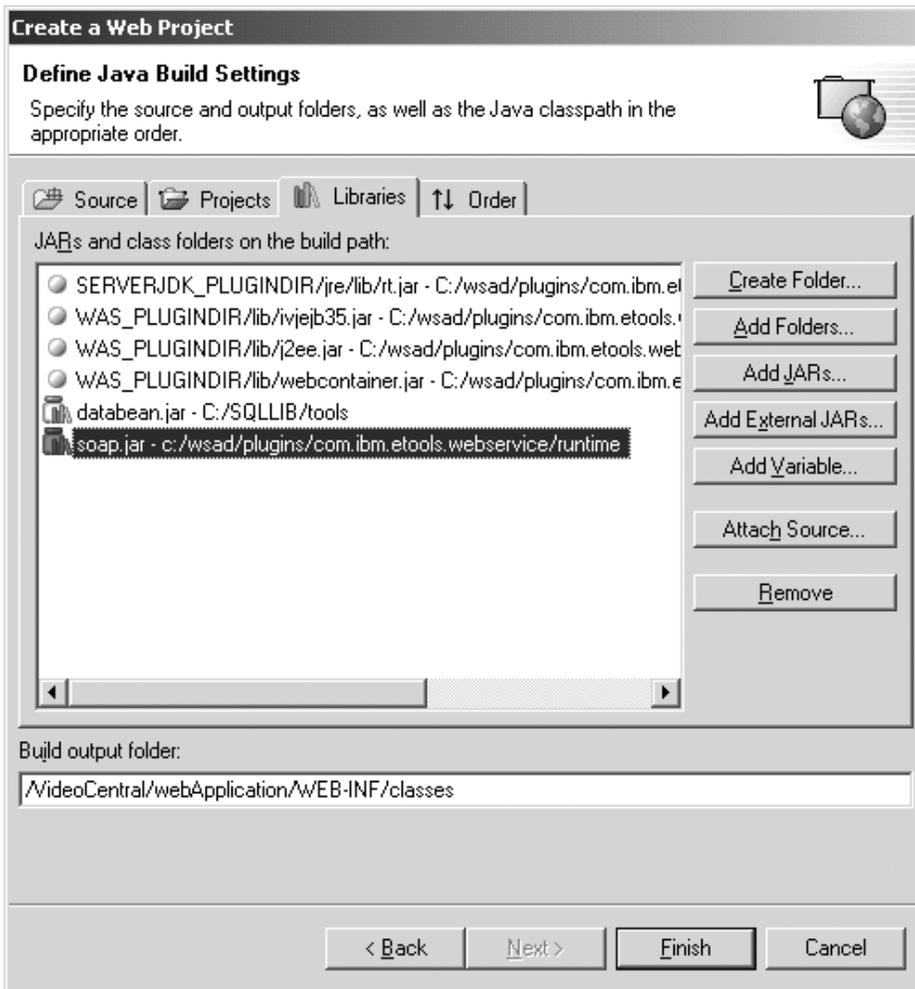


図6. 外部 JAR ファイルの追加

- d. 「完了 (Finish)」をクリックします。
2. 次のようにして、「VideoCentral」 Web プロジェクトをサーバーに追加します。
 - a. 「サーバー構成 (Server Configurations)」パネルの「サーバー構成 (Server Configurations)」を拡張表示してから「VideoCentralServer」を右マウス・ボタンでクリックします。「プロジェクトの追加 (Add Project)」 -> 「VideoCentralEAR」を選択します。

これで、「サーバー構成 (Server Configuration)」パネル・ウィンドウに次のように表示されるはずです (図 7 を参照)。

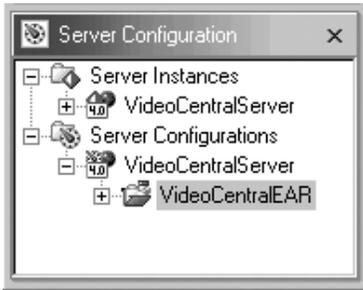


図7. サーバーへの Web プロジェクトの追加

- b. ナビゲーター・パネルの「VideoCentral」 Web プロジェクトを右マウス・ボタンでクリックし、「プロパティ (Properties)」 → 「サーバー設定 (Server Preferences)」を選択します。ボックス内の「VideoCentralServer」を強調表示し、「適用 (Apply)」をクリックしてから「OK」をクリックします (図 8 を参照)。

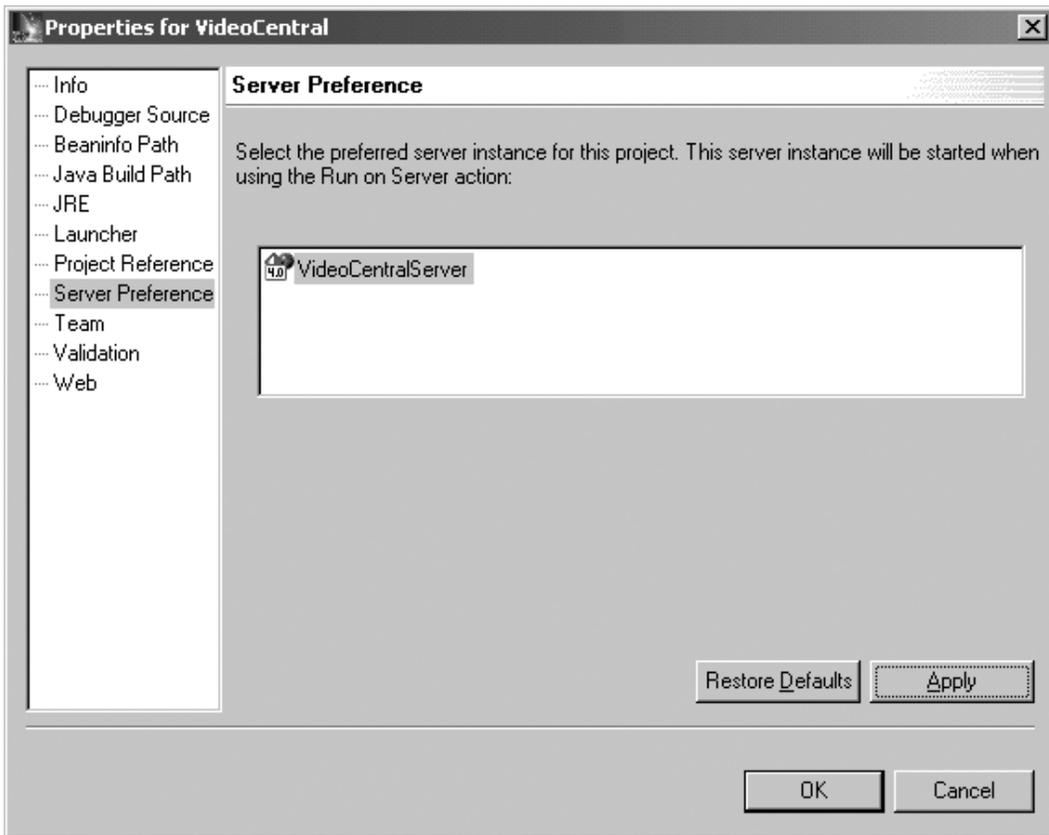


図8. サーバーへの Web プロジェクトの追加 (つづき)

WebSphere Studio Application Developer への VideoCentral.jar のインポート

1. 新しく作成した **VideoCentral** Web プロジェクトを拡張表示し、「ソース (source)」を強調表示します (図 9 を参照)。

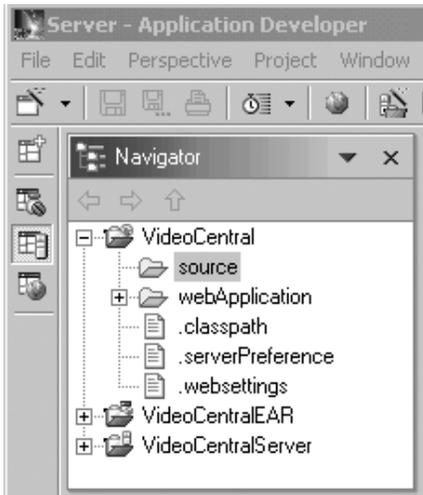


図9. JAR ファイルのインポート前

2. 「ファイル (File)」 -> 「インポート (Import)」 を選択します。
3. インポート・ソースには、「ZIP ファイル (ZIP file)」 を選択します。「次へ (Next)」 をクリックします。
4. <DB2_install_dir>%tutorial%VideoCentral%lib%VideoCentral.jar をブラウズして 見つけ出します。
5. 「全選択 (Select All)」 ボタンをクリックします(図 10 を参照)。

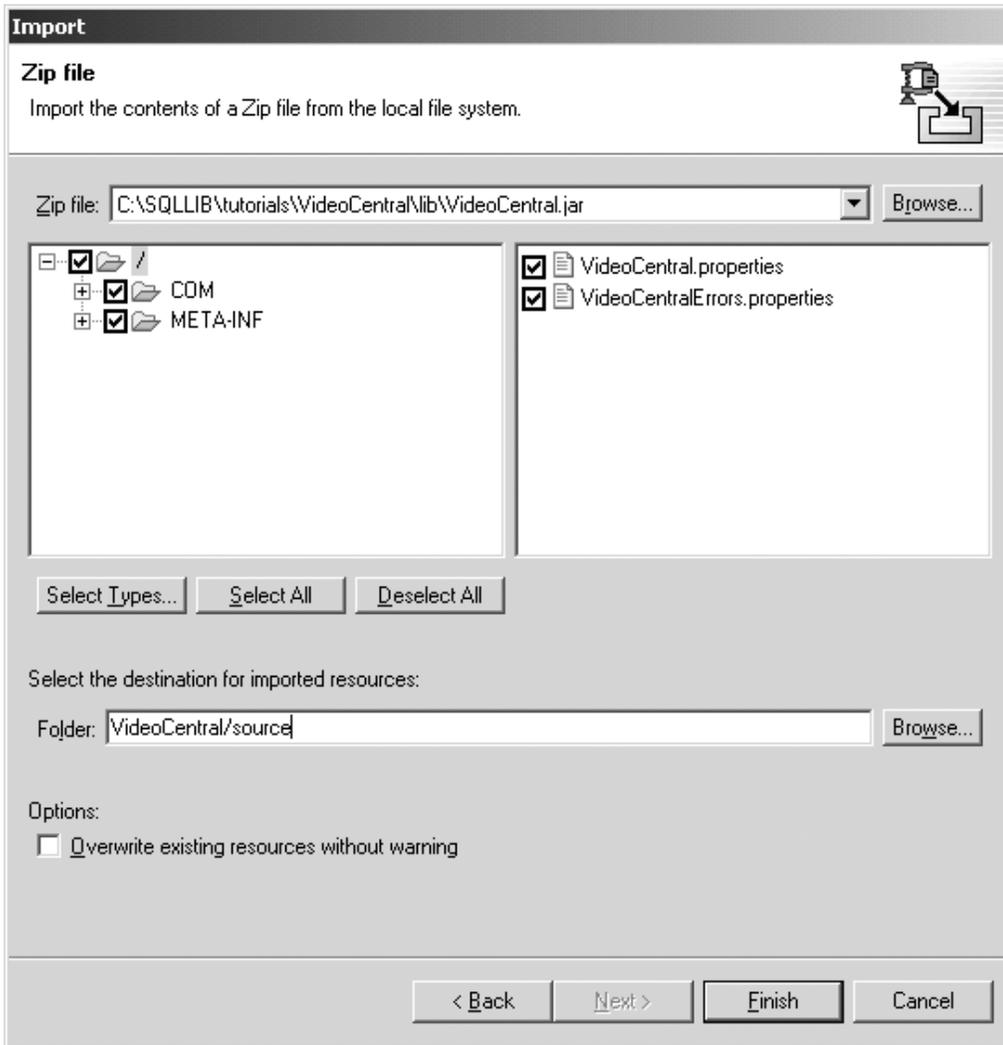


図 10. インポートするリソースの選択

6. 「完了 (**Finish**)」をクリックします。

これでファイルは WebSphere Studio Application Developer にインポートされたので、ファイルはコンパイル中であることがアクション・バーに示されます。

7. VideoCentral Web プロジェクトへのファイルのインポートが完了してから、「ソース (source)」フォルダーを拡張表示すると、以下のような構造になっているはずです (図 11 を参照)。

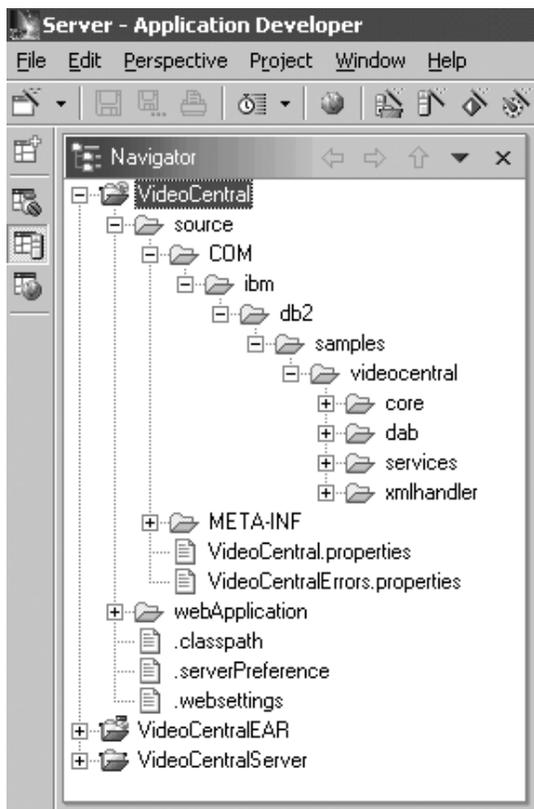


図 11. Video Central プロジェクトの準備後

学習のまとめ

この章では、以下の点を習得しました。

- WebSphere Studio Application Developer の習得
- アプリケーションのデプロイとテストのための WebSphere Studio Application Developer 内での WAS サーバー参照の作成
- WebSphere Studio Application Developer への Video Central コードのインポート

次のステップ

次に、Video Central アプリケーションの設計について詳しく説明します。

第 5 章 Video Central の設計の検討

所要時間

この章を完了するのに要する時間は、約 20 分です。

概説

この章では、以下のようなソフトウェア・コンポーネント層形式の Video Central の設計とインプリメンテーションについて説明します。

1. Web インターフェース層 (WIL)
 2. ビジネス論理層 (BLL)
 3. データ・アクセス層 (DAL)
-

Video Central の設計

ここでは、Video Central の設計について説明します。分かりやすくするために、いくつかのクラス・ダイアグラムとシーケンス・ダイアグラムが示されています。

Web インターフェース層 (WIL)

Web インターフェース層は、Video Central から消費者に提供されるさまざまな Web サービスを公開しています。そのような Web サービスを公開するには、上位レベルの抽象層を使用します。Video Central 自体の設計には Web サービス用のこの抽象層は組み込まれていませんが、SOAP ランタイムがこの層の働きをします。

ビジネス論理層 (BLL) の開発

Video Central でインプリメントされている Java クラスのクラス・ダイアグラムとシーケンス・ダイアグラムをここに示してあります。ちなみに、実際の Java ファイルは、本チュートリアルのもので、`<DB2_install_dir>%tutorial%VideoCentral\lib\VideoCentral.jar` および `<DB2_install_dir>%tutorial%VideoCentral\lib\VideoClient.jar` ファイル内にあります。

Video Central のインプリメンテーションの検討

この後の項では、Video Central のビジネス論理層についてさらに掘り下げて考察します。

Video Central のクラス・ダイアグラム: 図 12 は、Video Central のクラス・ダイアグラム全体を示しています。

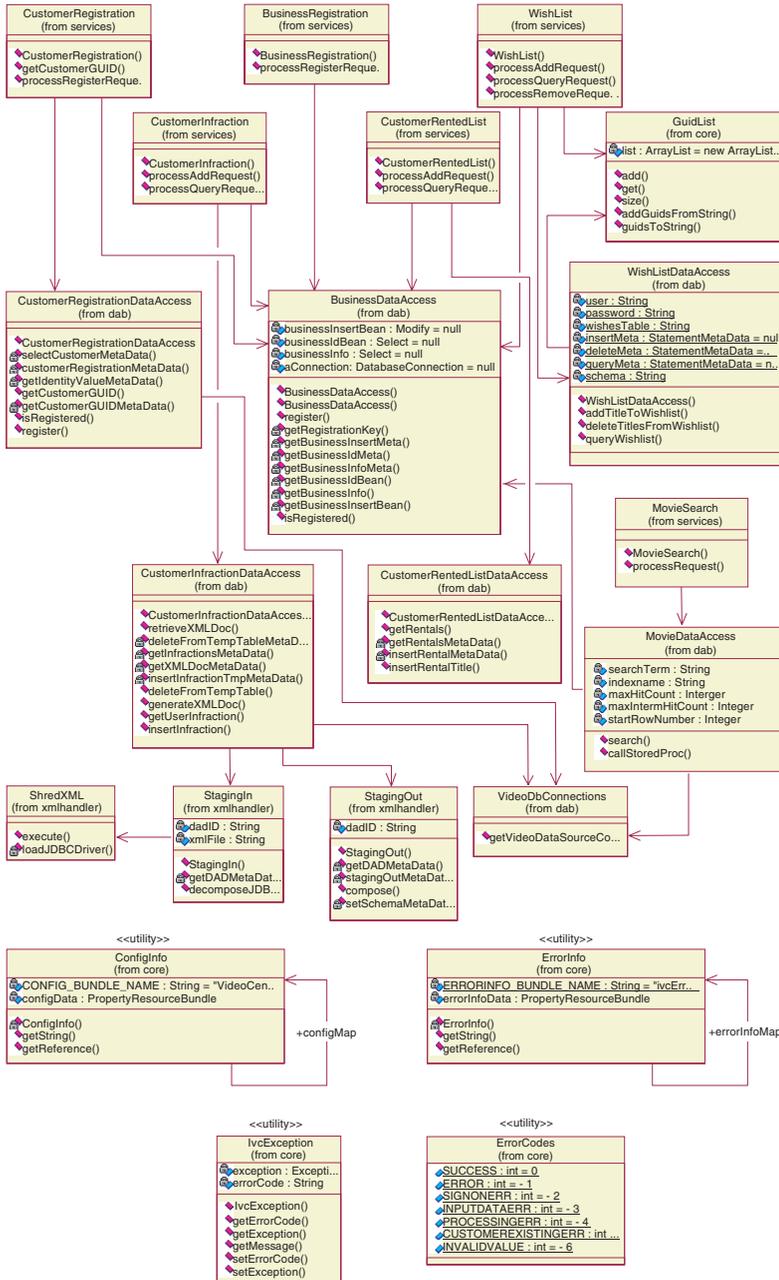


図 12. Video Central のクラス・ダイアグラム全体

カスタマー違反行為 Web サービスのクラス・ダイアグラム: 基本的に、どの Web サービスも同じ方法でインプリメントされます。ただしカスタマー違反行為 Web サービスは、XML Extender の使用法を示すという点で特異なサービスです。というわけでこのサービスに関しては、XML ストアード・プロシージャーへのアクセスとその使用法を示すために、特に重点を置く価値があります。以下のクラス・ダイアグラムは、カスタマー違反行為 Web サービスのインプリメンテーションに焦点を当てていません (図 13 を参照)。

う XML Extender が関与しています。VideoDbConnections は、データ・ソースからのデータベース接続を確立するための機能を有します。BusinessDataAccess は、取引の妥当性検査用として各 Web サービスに携わります。

希望商品リスト Web サービスのクラス・ダイアグラム: 希望商品リスト Web サービスのインプリメンテーションでは、Data Access JavaBeans (パッケージ com.ibm.db.* 内にあります) が使用されます。他のサービスはすべて、IBM Data Access bean (パッケージ com.ibm.db.uibeans.* 内にあります) を使ってインプリメントされます。DB2 UDE に付属している databean.jar ファイルには、Video Central のデータ・アクセス層に必要なクラスが入っています。以下のクラス・ダイアグラムは、希望商品リスト Web サービスのインプリメンテーションに焦点を当てています (図 14 を参照)。

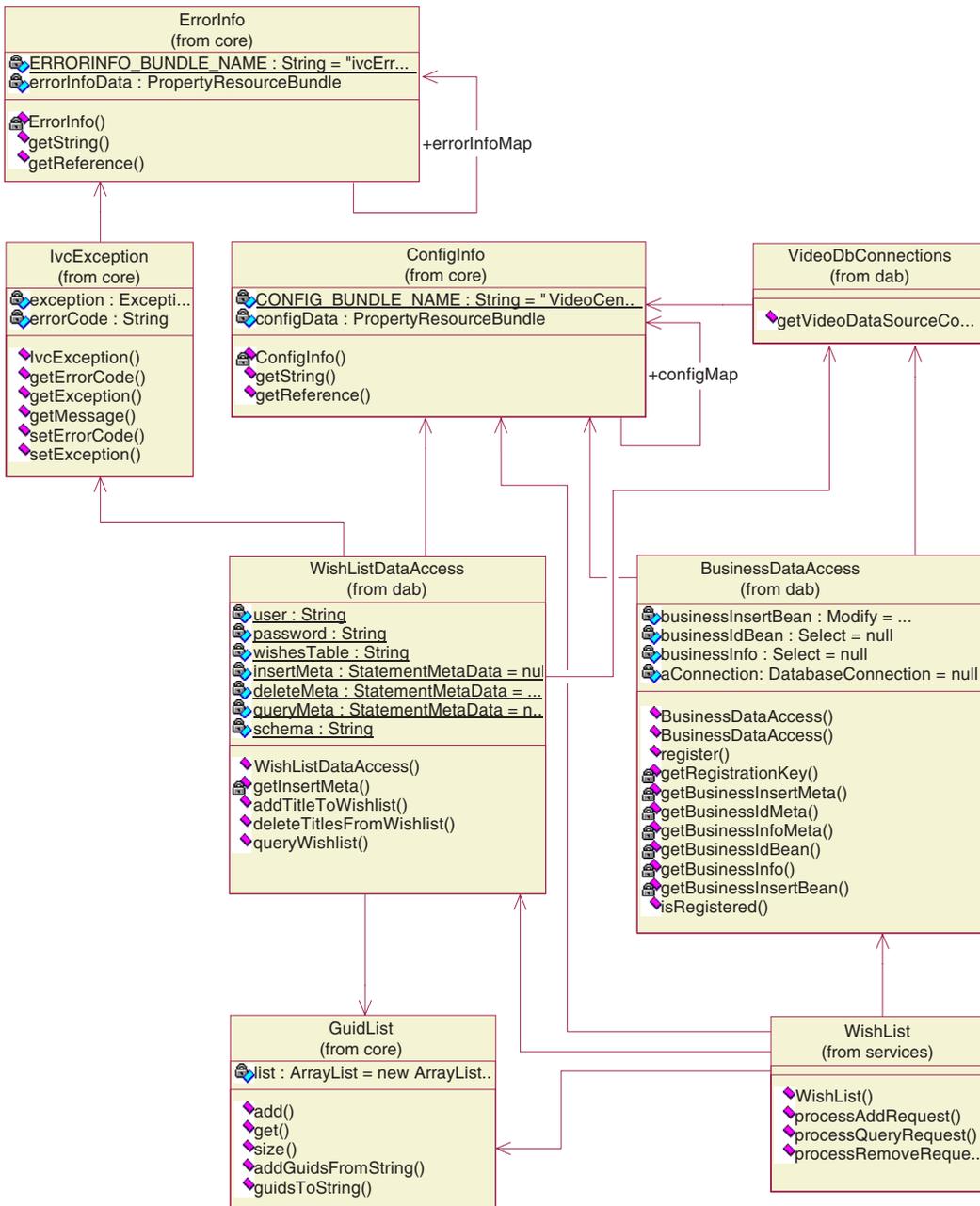


図 14. 希望商品リスト Web サービスのクラス・ダイアグラム

カスタマー違反行為の追加のシーケンス・ダイアグラム: 概してどのような Web サービスも同じ方法でインプリメントされるので、6 種類のサービスのシーケンス・ダイアグラムをすべて示す必要はないと考えられます。ここでは、興味を引きそう

なものをいくつか示してあります。中でも、AddCustomerInfraction のシーケンス・ダイアグラムは特に注目に値します。これは、XML Extender の分解ストアード・プロシージャ DXXSHREXML を使用するからです (図 15 を参照)。ShredXML クラス内の execute() メソッドを調べれば、XML Extender のこの分解ストアード・プロシージャの用法が明らかになります。

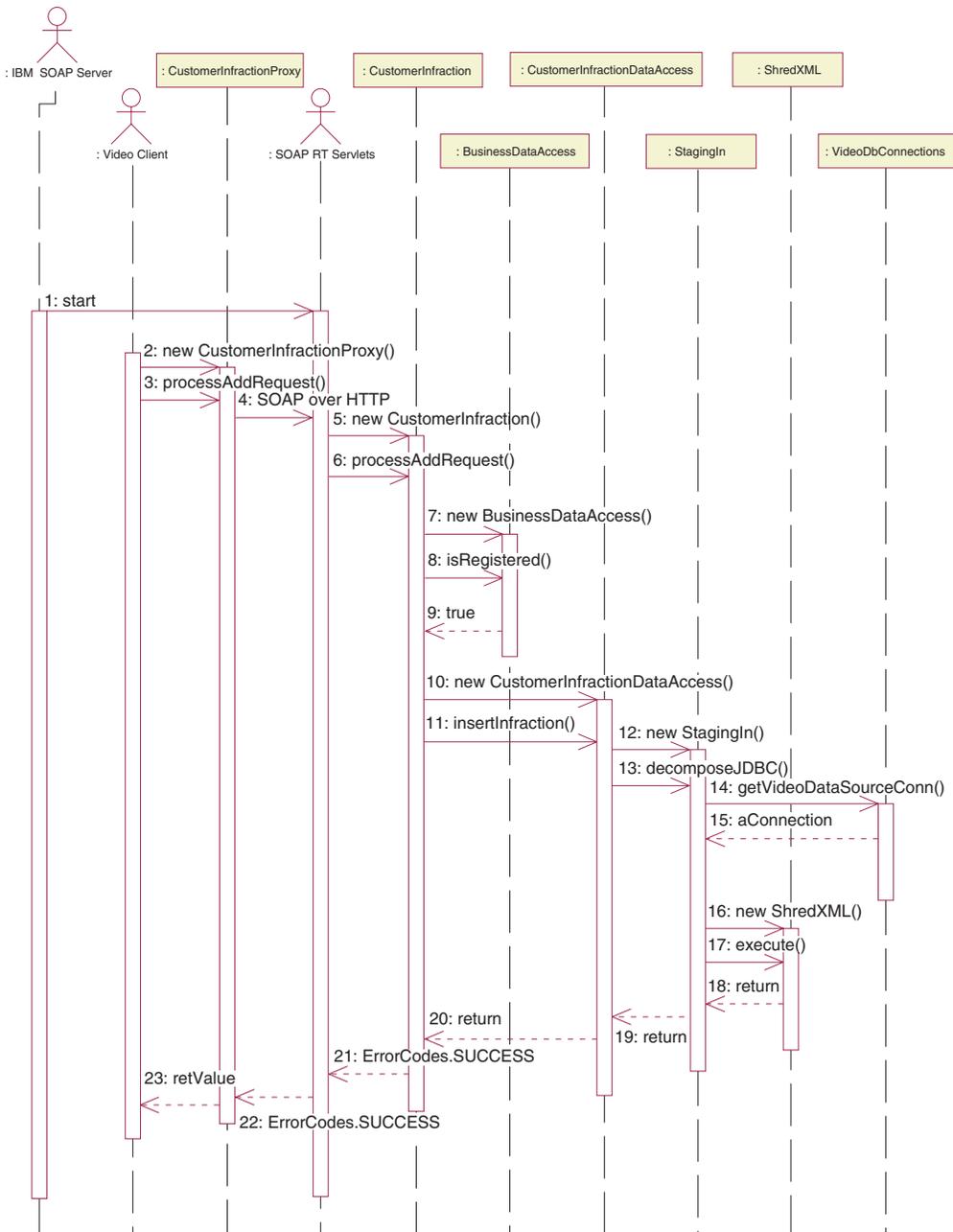


図 15. カスタマー違反行為の追加のシーケンス・ダイアグラム

カスタマー違反行為の照会のシーケンス・ダイアグラム: また、
 QueryCustomerInfraction のシーケンス・ダイアグラムも特別な注目に価します。これは、
 XML Extender の合成ストアード・プロシージャー DXXGENXML を使用するか

らです (図 16 を参照)。StagingOut クラス内の compose() メソッドを調べれば、XML Extender のこの合成ストアード・プロシーチャーの用法が明らかになります。

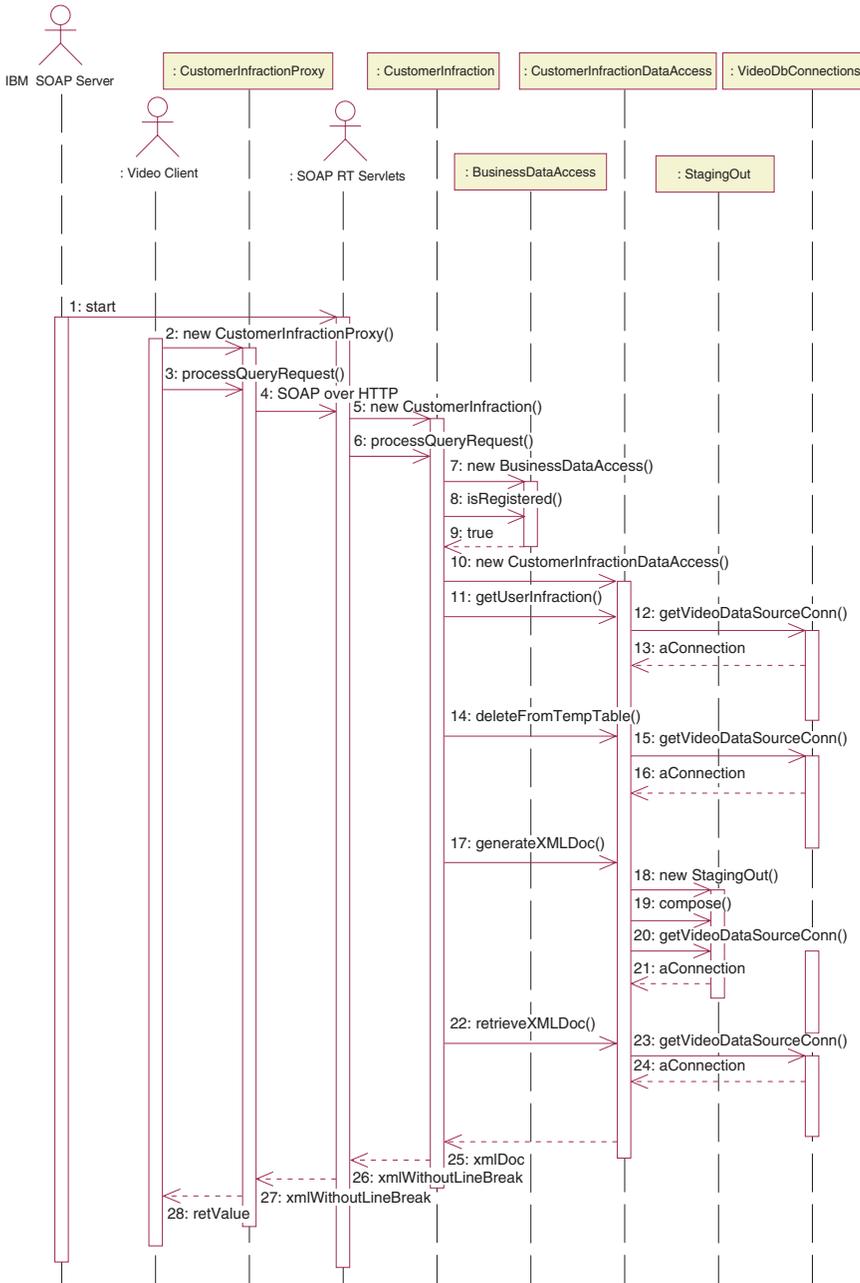


図 16. カスタマー違反行為の照会のシーケンス・ダイアグラム

希望商品リストの追加のシーケンス・ダイアグラム: ここでは、Web サービスの典型的な流れを見せるために、希望商品リストの追加のシーケンス・ダイアグラムを示してあります。希望商品リスト・サービスは、パッケージ com.ibm.db.* 内にある Data Access bean よりも下位レベルの Data Access クラスを使ってインプリメントされていますが、このサービスのシーケンスの流れは似通っています。図 17 を参照してください。ステップ 10 ~ 14 は、データ・アクセス層を通して顧客の希望商品リストにタイトルを挿入するためのステップです。

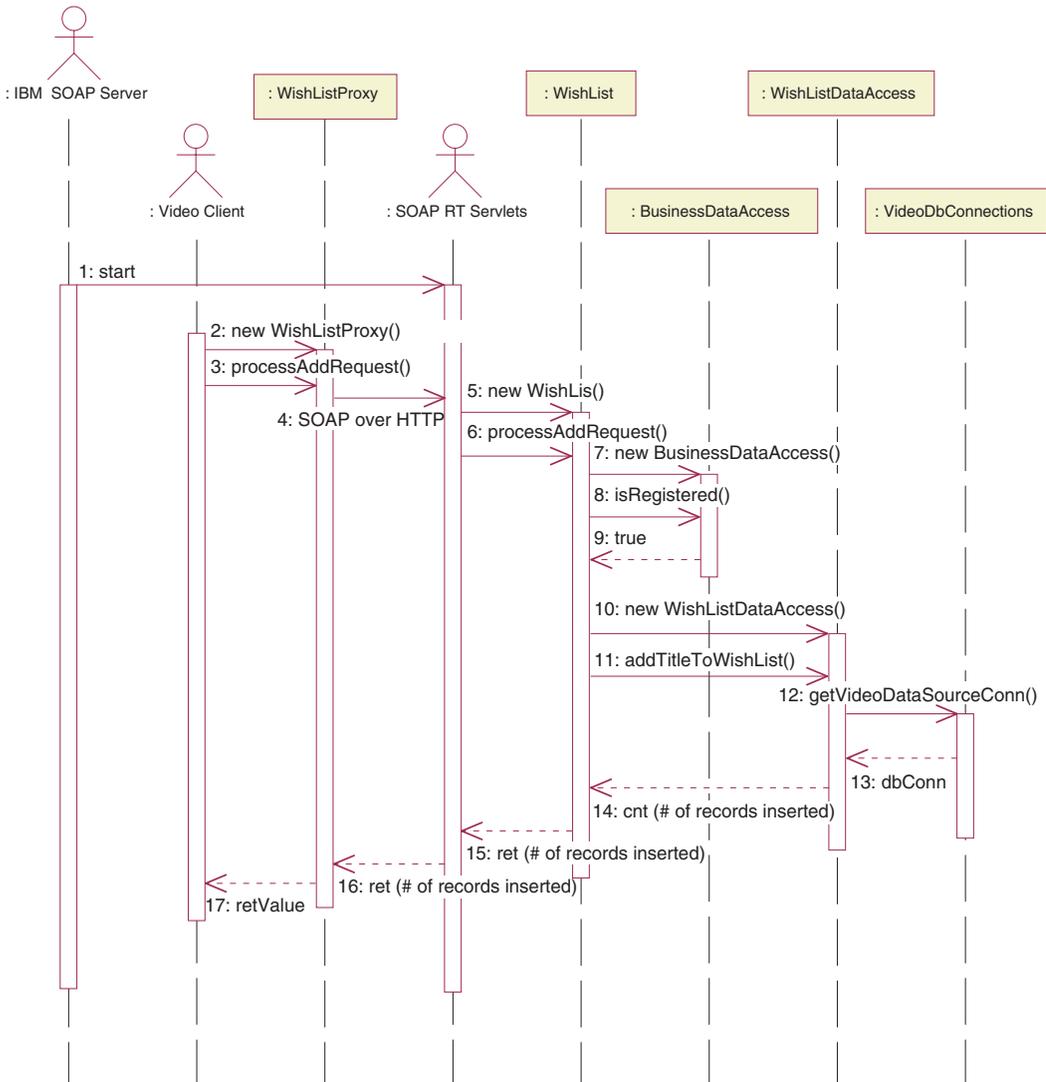


図 17. 希望商品リストの追加のシーケンス・ダイアグラム

データのモデル化

ここでは、Video Central に必要な表のデータ・モデルについて説明します。詳しくは、`<DB2_install_dir>%tutorial%VideoCentral%db` に置かれているスクリプト・ファイルを参照してください。

データ・モデルの概要

図 18 および 19 は、Video Central で使用される表を示しています。

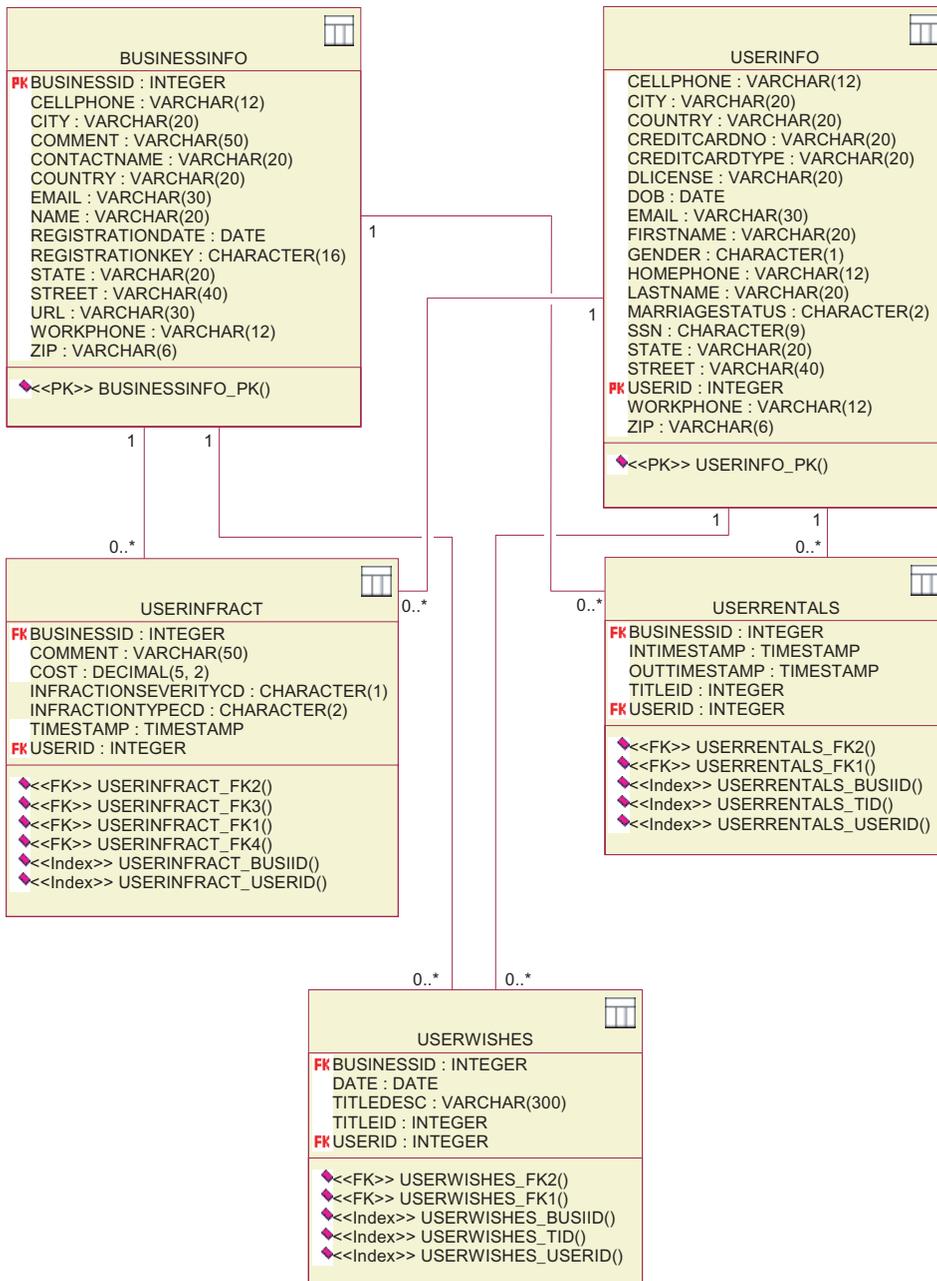


図 18. データ・モデルのダイアグラム - 第 1 部

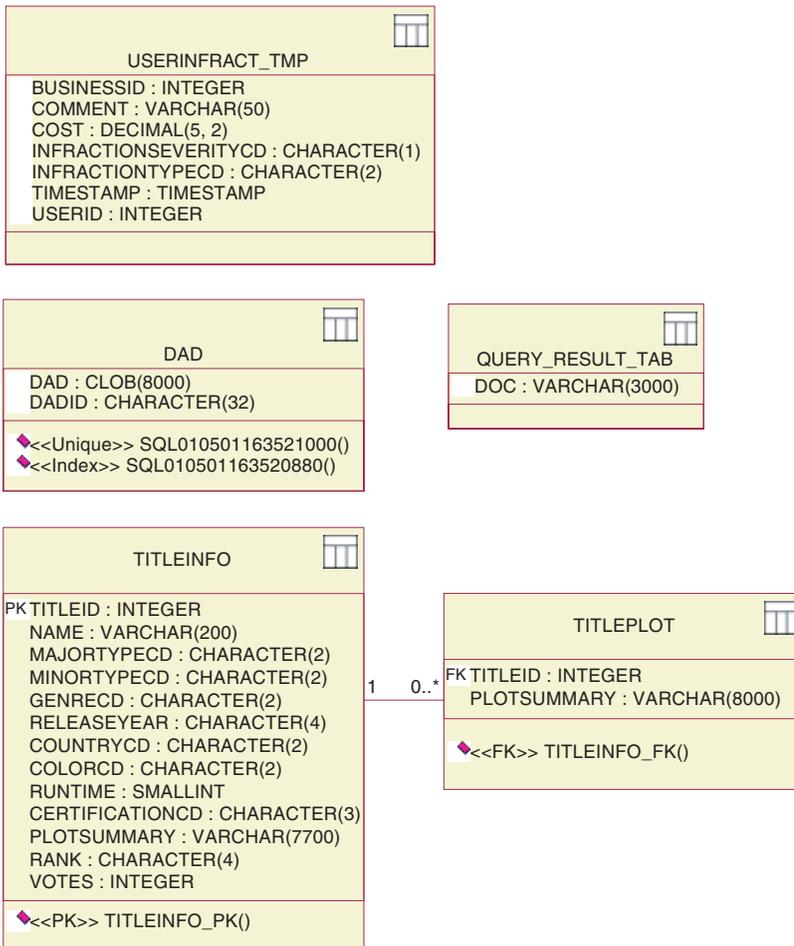


図 19. データ・モデルのダイアグラム - 第 2 部

DAD 表は、XML Extender を介した合成と分解に使用される DAD (Document Access Definition: 文書アクセス定義) ファイルを保管します。USERINFRACT_TMP と QUERY_RESULT_TAB の各表は、カスタマー違反行為照会でのカスタマー違反行為レポートを保管するのに使用されます。この 2 つの表は、XML Extender の DxxGenXMLClob フィーチャーを使用すれば除去することができます。TITLEINFO 表と TITLEPLOT 表は、ビデオのタイトルとプロットに関する情報を保管し、列 TITLEINFO.NAME および TITLEPLOT.PLOTSUMMARY 上の索引は、ムービー検索 Web サービスの提供のために Net Search Extender で使用されます。

学習のまとめ

この章では、Video Central の設計およびデータ・モデルについての理解を深めました。

次のステップ

次に、WebSphere Studio Application Developer の Web サービスを WebSphere Application Server に対してデプロイします。

第 6 章 Video Central のデプロイメント

所要時間

この章は、完了するのに約 20 分かかります。

前提条件

- WebSphere Application Server 4.0、Advanced Edition Single Server
- IBM HTTP Server、バージョン 1.3.12。これは、<http://www.ibm.com/software/webservers/httservers/download.html> から入手することができます。
- WebSphere Studio Application Developer および VideoCentral.jar (WebSphere Studio Application Developer への VideoCentral.jar のインポートについては、第 4 章に説明されています)。

概要

この章では、WebSphere Studio Application Developer 環境内の WebSphere Application Server に対して Video Central コードをデプロイします (前述の第 4 章で定義されています)。

Video Central の作業

ここでは、WebSphere Studio Application Developer から Web サービスを WebSphere Application Server に対してデプロイする方法を習得します。

作業 1: WebSphere Studio Application Developer から Video Central をデプロイする環境の準備

1. DB2 サーバーと HTTP サーバーが開始済みであることを確認します。
 - a. 「Windows サービス (Windows Services)」を表示し、DB2 サーバーと HTTP サーバーが開始済みであることを確認します。まだ開始していなければ、ここで開始します。
 - b. Web ブラウザーをオープンし、URL に localhost と入力します。次のような表示になります (図 20 を参照)。

Welcome to the IBM HTTP Server

powered by Apache

*Everything you need to start
and use this server begins here...*

- **Configure server**
- **View documentation**
- **Visit our Web site**



図 20. HTTP サーバーの検査

2. WebSphere Studio Application Developer から、「サーバー (Servers)」パースペクティブの「サーバー (Servers)」パネルを選択します。このパネルの「VideoCentralServer」を右マウス・ボタンでクリックしてから、「開始 (Start)」を選択します。「コンソール (Console)」パネルを選択し、メッセージ Default server open for e-business が示されているかどうかを調べます (図 21 を参照)。

```
Console
A DYNA0011E: Servlet cache file dynacache.xml not found; ca
A SRVE0169I: Loading Web Module: admin.
I SRVE0091I: [Servlet LOG]: JSP 1.1 Processor: init
I SRVE0091I: [Servlet LOG]: SimpleFileServlet: init
I SRVE0091I: [Servlet LOG]: action: init
A SRVE0169I: Loading Web Module: IBMVideoCentral.
I SRVE0091I: [Servlet LOG]: JSP 1.1 Processor: init
I SRVE0091I: [Servlet LOG]: SimpleFileServlet: init
I SRVE0091I: [Servlet LOG]: InvokerServlet: init
A SRVE0169I: Loading Web Module: IBMVideoCentralClient.
I SRVE0091I: [Servlet LOG]: JSP 1.1 Processor: init
I SRVE0091I: [Servlet LOG]: SimpleFileServlet: init
I SRVE0091I: [Servlet LOG]: InvokerServlet: init
A SRVE0171I: Transport http is listening on port 9,080.
A SRVE0171I: Transport http is listening on port 9,090.
U *** Server has started ***
A WSWR0023I: Server Default Server open for e-business
```

図 21. WAS Administration Server が稼働中かどうかの検査

3. WebSphere Studio Application Developer から、「サーバー (Servers)」パースペクティブの「サーバー (Servers)」パネルを選択します。このパネルの「VideoCentralServer」を右マウス・ボタンでクリックしてから、「停止 (Stop)」を選択します。

注: VideoCentralServer を停止するのは、Web サービスのデプロイ中にこのサーバーは自動的に開始しているの、サーバーが開始済みであると、「ポートは使用中 (port in use)」のエラーが起きるからです。

作業 2: WebSphere Application Server に対する Web サービスのデプロイ

WebSphere Studio Application Developer を使用した Web サービス (Java bean ベース) の作成とデプロイ

1. 次のようにして、WebSphere Studio Application Developer から WebSphere Application Server へ、Web サービスをデプロイします。

- a. Web サービス・タイプの選択

WebSphere Studio Application Developer で、Web パースペクティブに移動し、VideoCentral Web プロジェクトを拡張表示して次のとおり選択します。

```
VideoCentral/source/COM/ibm/db2/samples/videocentral/services/  
BusinessRegistration.java
```

これらは、デプロイする最初の Web サービスとして選択します (図 22 を参照)。

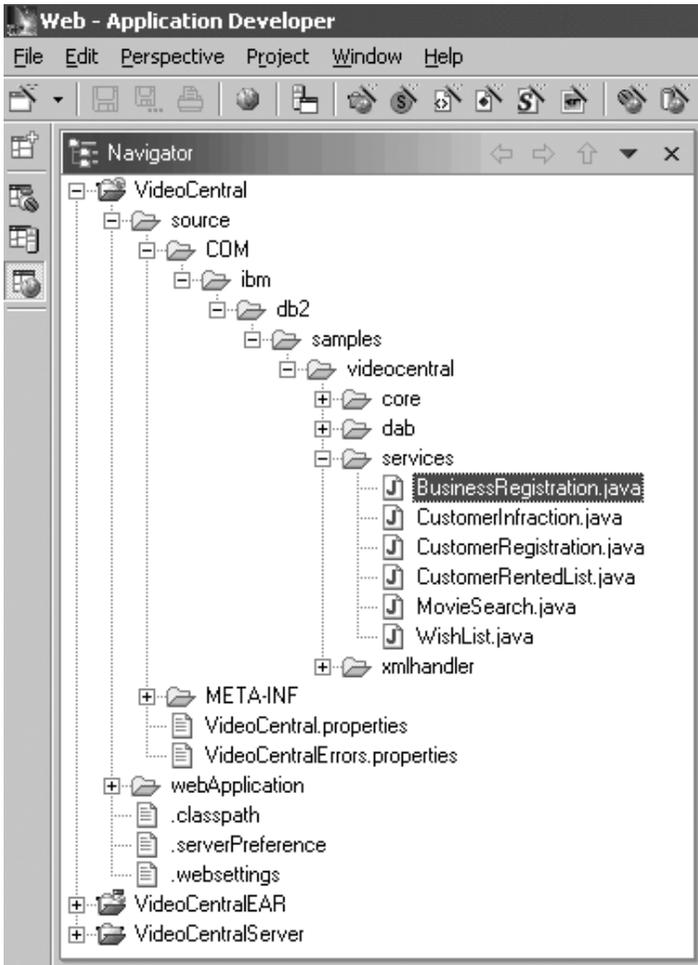


図 22. デプロイする Web サービスの選択

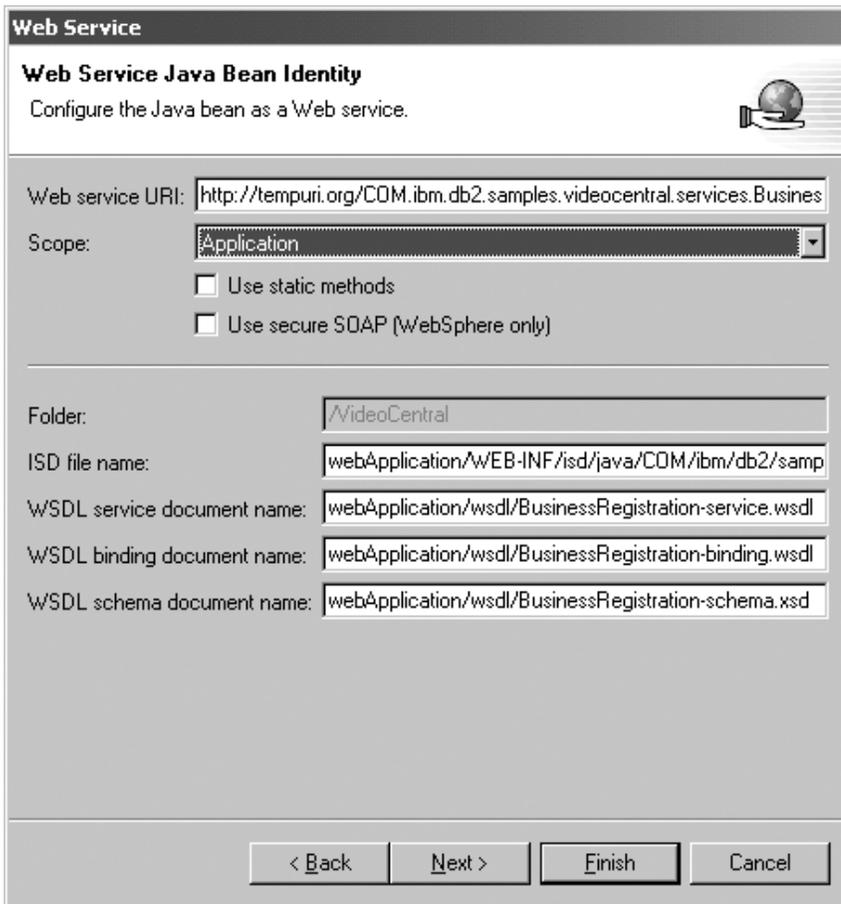
- b. **BusinessRegistration.java** を右クリックし、「新規 (New)」→「その他 (Other)」を選択し、表示される「新規 (New)」ウィンドウで「Web サービス (Web service)」をクリックします。すべてデフォルトのままにして、「次へ (Next)」をクリックします。
2. 「Web サービス Java bean の選択 (Web Services Java Bean Selection)」ウィンドウで、bean がステップ 1 で選択したものであることを確認します (図 23 を参照)。



図 23. デプロイする Java Bean

「次へ (Next)」をクリックします。

3. Java Bean を Web サービスとして構成します。デフォルト設定のままにして「次へ (Next)」をクリックします (図 24 を参照)。



The image shows a dialog box titled "Web Service Java Bean Identity". The subtitle is "Configure the Java bean as a Web service." The dialog contains several input fields and checkboxes. The "Web service URI" field is filled with "http://tempuri.org/COM.ibm.db2.samples.videocentral.services.Busines". The "Scope" dropdown menu is set to "Application". There are two unchecked checkboxes: "Use static methods" and "Use secure SOAP (WebSphere only)". Below these are five text input fields for "Folder", "ISD file name", "WSDL service document name", "WSDL binding document name", and "WSDL schema document name", each containing a file path. At the bottom, there are four buttons: "< Back", "Next >", "Finish", and "Cancel".

Web Service

Web Service Java Bean Identity
Configure the Java bean as a Web service.

Web service URI:

Scope:

Use static methods

Use secure SOAP (WebSphere only)

Folder:

ISD file name:

WSDL service document name:

WSDL binding document name:

WSDL schema document name:

< Back Next > Finish Cancel

図 24. Web サービスとしての Java Bean の構成

4. すべてのメソッドを選択状態のままにします (図 25 を参照)。

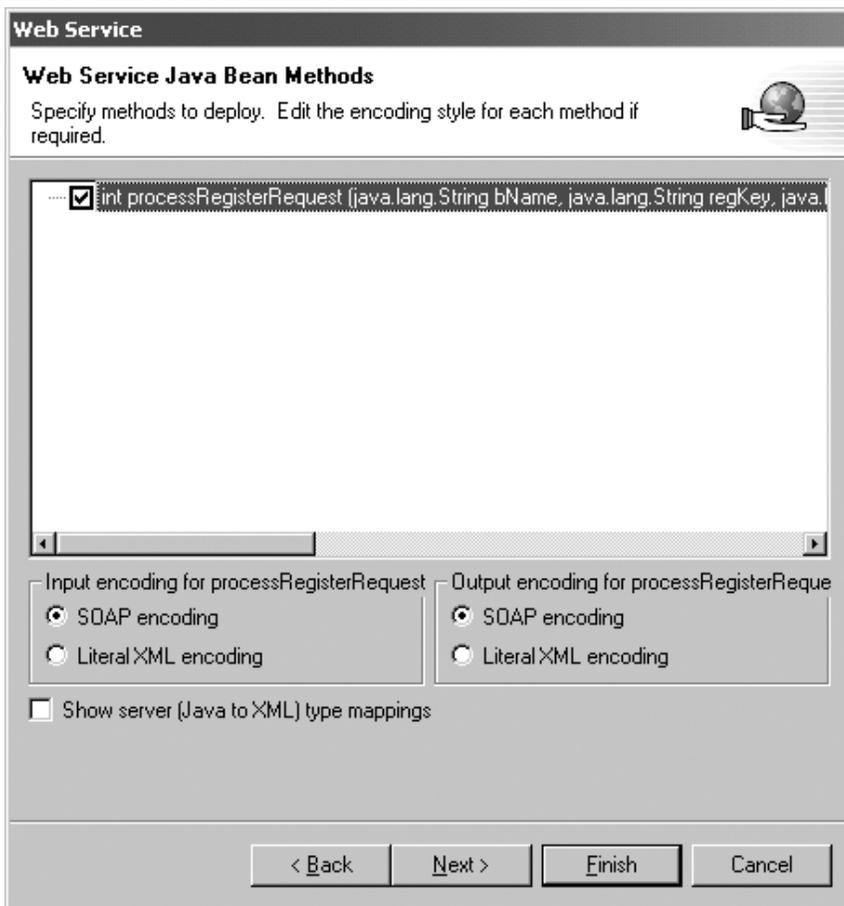


図 25. デプロイするパブリック・メソッドの選択

「次へ (Next)」をクリックします。

5. 「Web サービス・バインディング・プロキシの生成 (Web Service Binding Proxy Generation)」ページで、「クラス (Class)」フィールドの以下のクラスを proxy.soap.COM.ibm.db2.sample.videocentral.services.

BusinessRegistrationProxy

次のものに置き換えます。

COM.ibm.db2.samples.videoclient.proxies.BusinessRegistrationProxy

図 26 を参照してください。このようにして、VideoCentral プロジェクト内で、COM.ibm.db2.samples.videoclient.proxies.* パッケージの中にプロキシ・クラスを作成することができます。

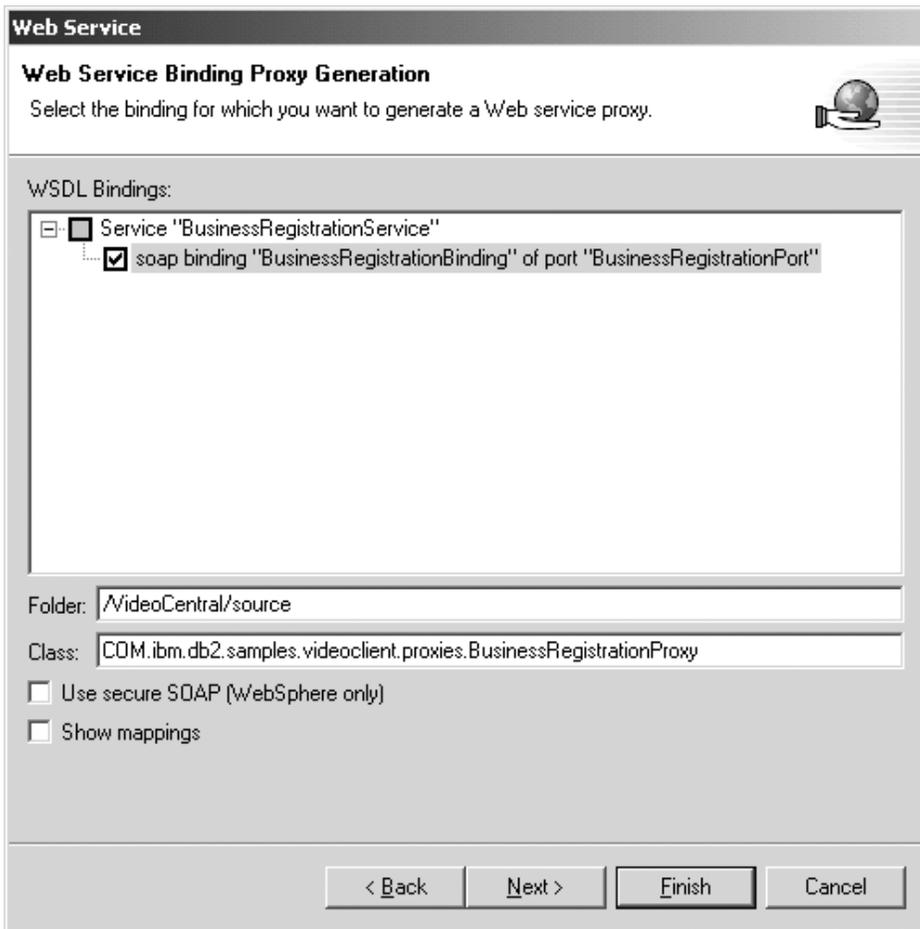


図 26. Web サービス・バインディング・プロキシの生成

6. 「完了 (**Finish**)」をクリックしてサービスをデプロイします。
7. 「サーバー (Servers)」パースペクティブの「サーバー (Servers)」パネルを選択します。このパネル内の VideoCentralServer を右クリックし、「停止 (**Stop**)」を選択します。

注: デプロイメントが成功しなかった場合は、『第 8 章、共通問題の解決』で、いくつかの考えられる原因を参照してください。

この時点で、WSDL が作成されます。VideoCentral Web プロジェクトの中にそれが表示されます。

- VideoCentral > webApplication > wsdl > BusinessRegistration-binding.wsdl
- VideoCentral > webApplication > wsdl > BusinessRegistration-service.wsdl

デフォルト・エディターを使用して WSDL ファイルを見たい場合には、WSDL ファイルをダブルクリックして、「ソース (source)」ボタンを選択します (図 27 の BusinessRegistration-service.wsdl を参照)。バインディング文書は、特定の portType によって定義される、オペレーションおよびメッセージ用のプロトコルとフォーマットを定義します。サービス文書は、バインディング文書内にバインディングするための関連エンドポイントの集合です。



```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="BusinessRegistrationService" targetNamespac
  <import namespace="http://www.businessregistration.com/defin
  <service name="BusinessRegistrationService">
    <port name="BusinessRegistrationPort" binding="binding:Bus
      <soap:address location="http://localhost:9080/VideoCentr
    </port>
  </service>
</definitions>
```

図 27. WSDL ファイルの内容

残りの 5 つの Web サービスについて、作業 2 のステップ 1~7 を繰り返します。Video Central のこの段階では、次の 6 つのサービスが使用可能です。

- COM/ibm/db2/samples/videocentral/services/BusinessRegistration.java
- COM/ibm/db2/samples/videocentral/services/CustomerInfraction.java
- COM/ibm/db2/samples/videocentral/services/CustomerRegistration.java
- COM/ibm/db2/samples/videocentral/services/CustomerRentedList.java
- COM/ibm/db2/samples/videocentral/services/WishList.java
- COM/ibm/db2/samples/videocentral/services/MovieSearch.java

すべての Web サービスをデプロイしたら、Web プロジェクト内のそれぞれのサービスごとに 2 つの WSDL ファイルが表示されます (図 28 を参照)。

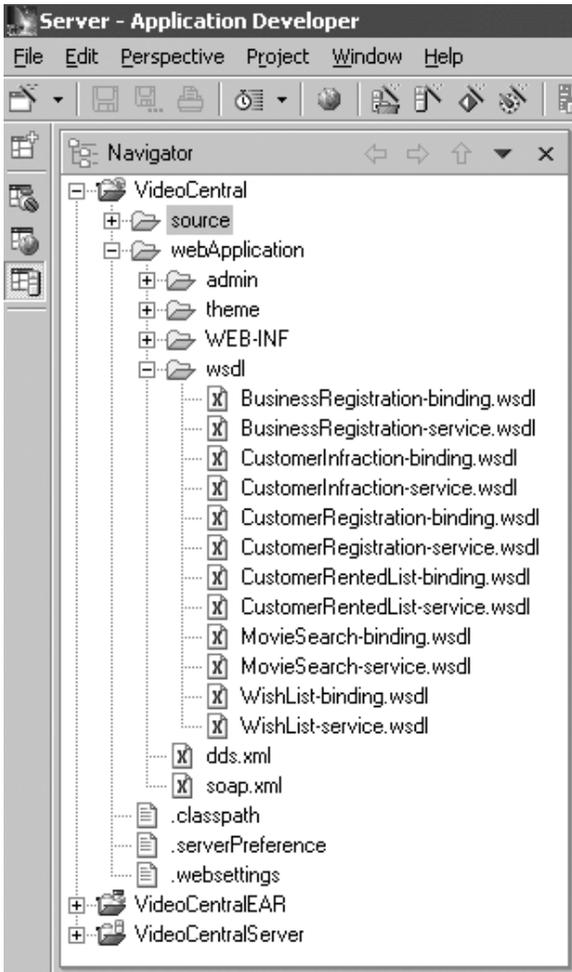


図 28. デプロイされるすべての Web サービス

学習のまとめ

この章では、以下の点を習得しました。

- Video Central をデプロイするための環境を検査しました。
- WebSphere Studio Application Developer から WebSphere Application Server へ、6 つの Video Central Web サービスをデプロイしました。
- Video Central の Web サービスへのアクセスに使用するプロキシ・クラスを生成しました。

- それぞれの用の WSDL ファイルを生成しました。WSDL ファイルを UDDI レジストリーにパブリッシュしたり (第 2 章)、サービス・リクエスターに提供することができました。

次のステップ

次に、使用中の Web サービスを表示するための Video Central サービスを起動する、Web ベースのサンプル・クライアントを作成します。

第 7 章 Video Central 用の Web ベースのサンプル・クライアントの作成と実行

所要時間

この章を完了するのに要する時間は、約 25 分です。

前提条件

『第 4 章、WebSphere Studio Application Developer の使用開始』と『第 6 章、Video Central のデプロイメント』のステップを完了していることが必要です。

概説

この章では、Video Central 用の Web ベースのサンプル・クライアント (サービス・リクエスター) の設計と、提供された Web サービスを使用するために必要なステップを説明します。これらのステップには、プロキシー・クラスの再生成と、WebSphere Application Server へのクライアントのデプロイメントが含まれています。第 6 章では、プロキシー・クラスと WSDL ファイルを Java Beans から生成しました。この章では、WSDL ファイルからプロキシー・クラスを再生成します。

Video Central サンプル・クライアントの設計

次のセクションは、サービス・リクエスター (Video Client) のアプリケーション・サーバーで実行されます。クライアントは、Web サービス・プロバイダーが提供する Web サービス用のプロキシー・クラスを作成します (図 29 を参照)。

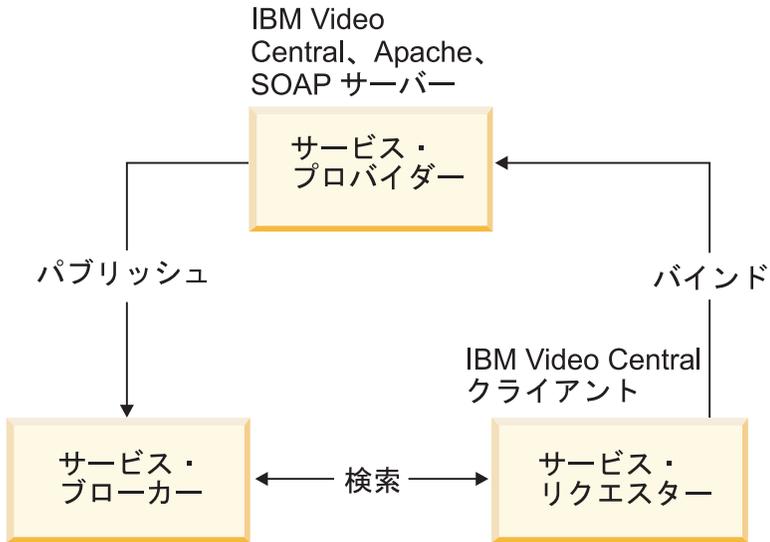


図 29. Web サービスのコンポーネント

この時点まで、すべてのファイルは Video Central サーバー (サービス・プロバイダー) 上にあります。このチュートリアル次のセクションでは、Video Client サーバー (サービス・リクエスター) を扱います。このチュートリアルでは、同じ WebSphere Application Server を使用していますが、実際には Video Client は、それぞれのビデオ店にある別々のアプリケーション・サーバーにあります。

Video Central サンプル・クライアントのクラスとシーケンスのダイアグラム

図 30 は、Video Central サンプル・クライアント Video Client の完全なクラス・ダイアグラムを示しています。

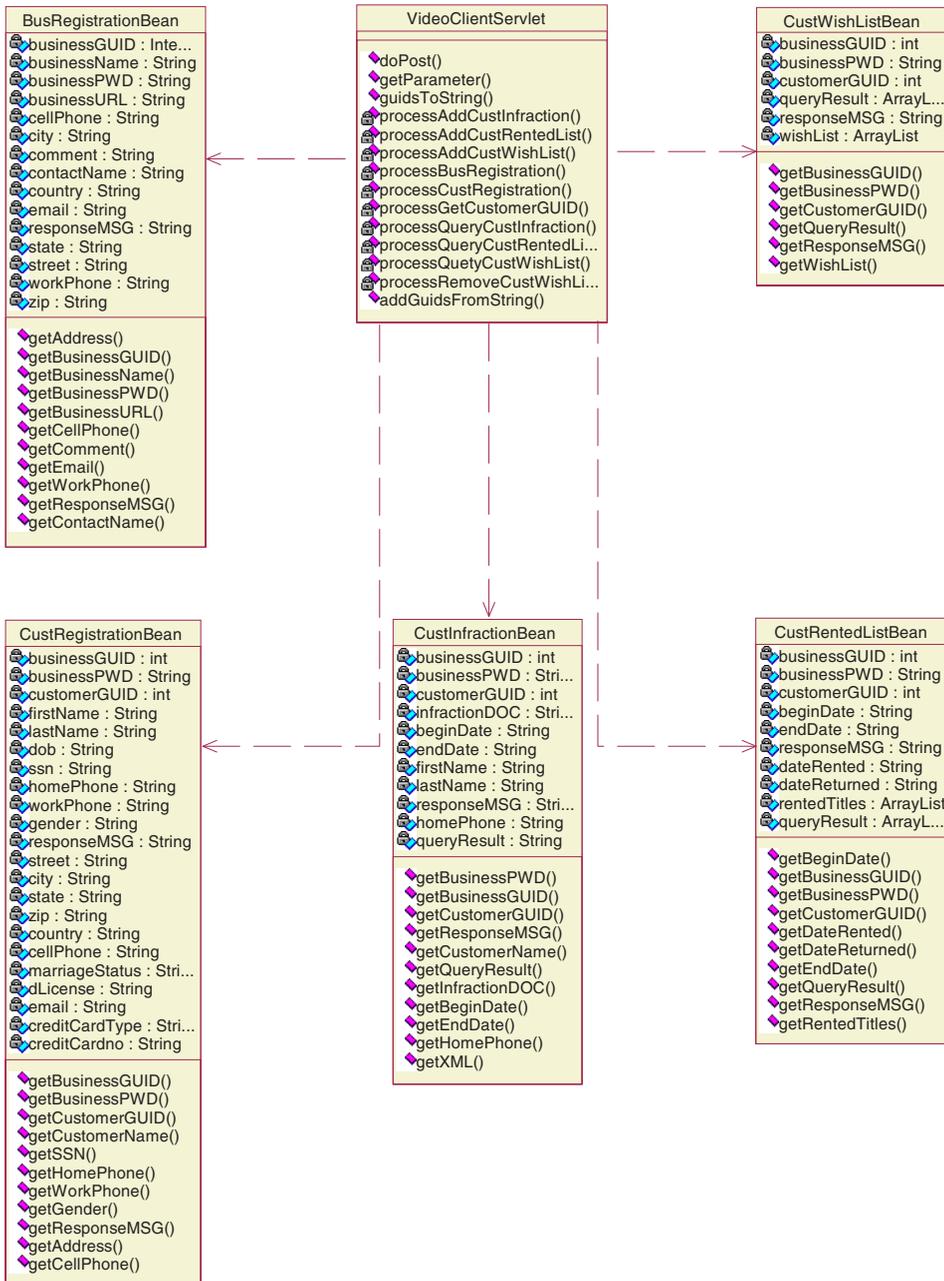


図 30. Video Central サンプル・クライアント Video Client のクラス・ダイアグラム

図 31 は、Video Central サンプル・クライアント Video Client の完全なシーケンス・ダイアグラムを示しています。

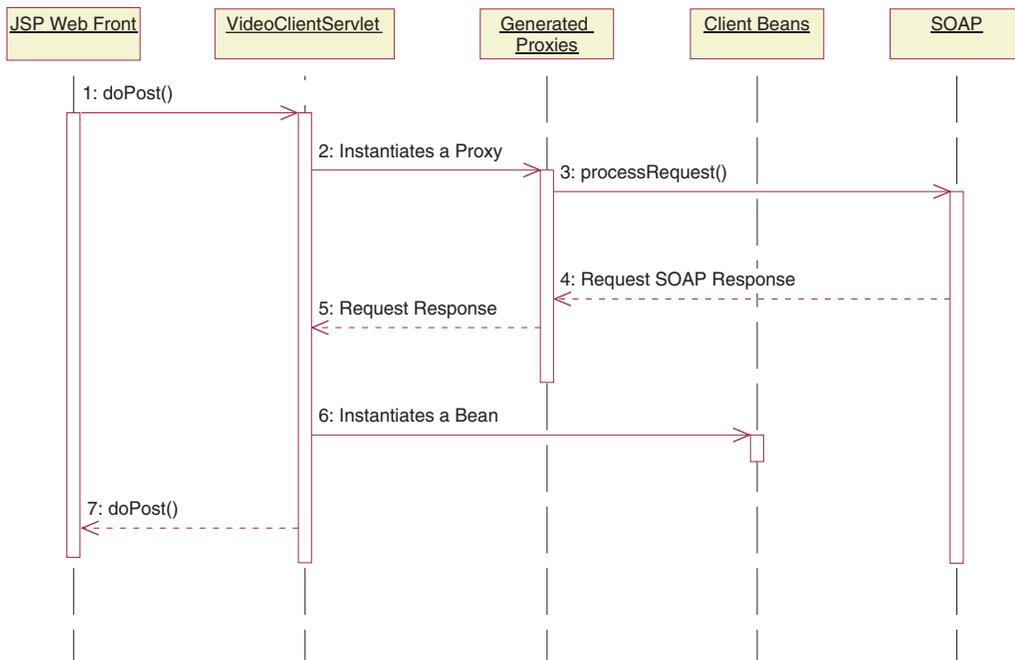


図 31. Video Central サンプル・クライアント Video Client のシーケンス・ダイアグラム

Video Client において、すべての要求は VideoClientServlet に送られます。それぞれの要求を処理する際、VideoClientServlet は対応するプロキシー要求を起動し、そしてプロキシー要求はリモート Web サービスを起動します。応答に入れて戻される情報を使用することにより、VideoClientServlet はクライアント Bean をインスタンス生成します。最後に、JavaServer Page を使用して、結果がエンド・ユーザーに表示されます。

Video Central の作業

ここで、以下の実行方法を学びます。

- 必要なプロキシー・クラスを作成する。
- クライアント・コードを WebSphere Application Server にデプロイする

作業 1: クライアント・プロジェクトの準備

WebSphere Studio Application Developer の中で、以下のステップを実行します。

1. 次のようにして、VideoClient Web プロジェクトを作成します。
 - a. Web パースペクティブで、**VideoCentral** Web プロジェクトを強調表示します (これは、WebSphere Studio Application Developer のツリーにおける一番上のレベルです)。

- b. 「ファイル (File)」 → 「新規 (New)」 → 「Web プロジェクト (Web Project)」 の順にクリックして、Web プロジェクトを作成します。
- c. Web プロジェクトに VideoClient という名前を付け、Enterprise Application Project に VideoClientEAR という名前を付けます。
- d. 「次へ (Next)」 をクリックします。
- e. 「Web 設定 (Web Settings)」 画面が表示されます。すべてのデフォルトのままにして、「次へ (Next)」 をクリックします。
- f. 「Java 設定 (Java Settings)」 画面が表示されます。「ライブラリー (Libraries)」 パネルを選択します。「外部 JAR の追加 (Add External JARs)」 をクリックし、ポップアップされる 「JAR 選択 (JAR Selection)」 ウィンドウで、以下の外部 JAR を、示されているとおりの順番で追加します。
 - <WAS_install_dir>%lib%xerces.jar
 - <WSAD_install_dir>%plugins%com.ibm.etools.webservice%runtime%soap.jar

xerces.jar は、カスタマー違反行為 Web サービス・リクエスターにより必要とされます (違反行為はメモリー内の XML オブジェクトとして扱う必要があるため)。また、xerces は IBM 提供の XML Document Object Model (DOM) パーサーです。

以下のような表示になります (図 32 を参照)。

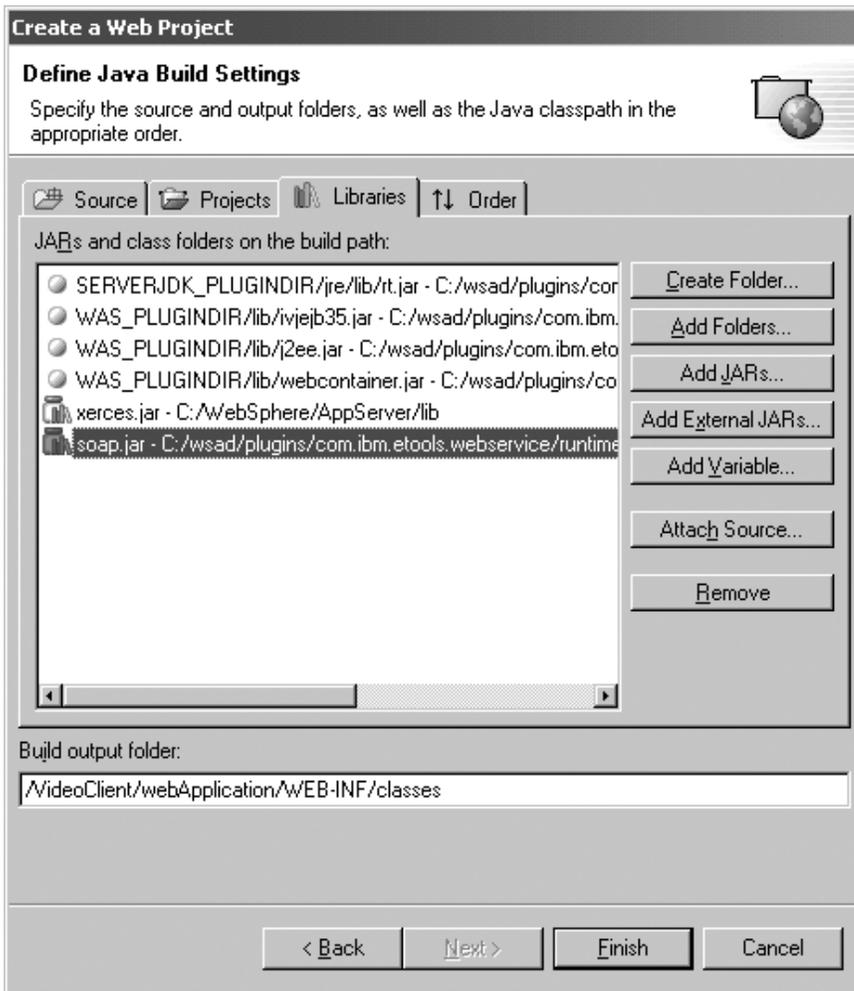


図 32. 外部 JAR ファイルの追加

- g. 「完了 (**Finish**)」をクリックします。
2. 次のようにして、VideoClient をサーバーに追加します。
 - a. 「サーバー構成 (Server Configuration)」パネルで、「サーバー構成 (**Server Configurations**)」を拡張表示し、**VideoCentralServer** を右クリックします。「プロジェクトの追加 (**Add Project**)」→ **VideoClientEAR** の順に選択します (図 33 を参照)。

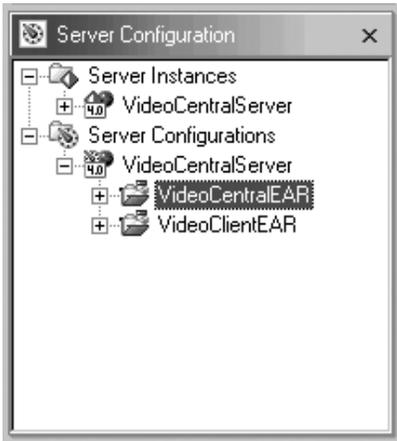


図 33. サーバーへのクライアント Web プロジェクトの追加

- b. ナビゲーター・パネルで、**VideoClient** プロジェクトを右クリックし、「プロパティ (Properties)」→「サーバー・プリファレンス (Server Preference)」の順に選択します。ボックス内の **VideoCentralServer** を強調表示します。
 - c. 「適用 (Apply)」をクリックし、次いで「OK」をクリックします。
 3. 次のようにして、クライアント・コードを VideoClient にインポートします。
 - a. 新たに作成した VideoClient Web プロジェクトを拡張表示し、Web プロジェクトの「ソース (source)」を強調表示します (図 34 を参照)。

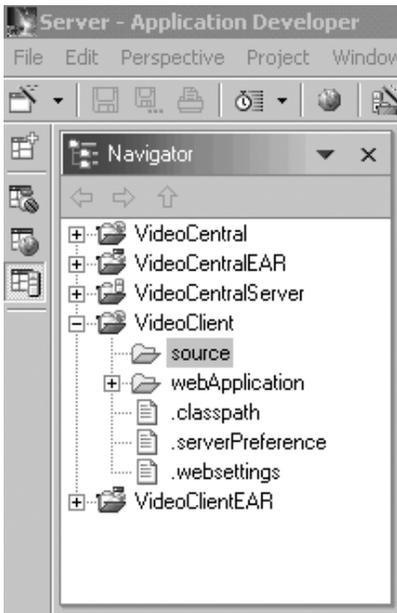


図 34. ソースの強調表示

- b. 「ファイル (File)」 -> 「インポート (Import)」 をクリックします。
- c. 「インポート・ソース (Import Source)」には、「ZIP ファイル (ZIP file)」を選択します。「次へ (Next)」をクリックします。
- d. `<DB2_install_dir>%tutorial%VideoCentral\lib\VideoClient.jar` をブラウズします。「OK」をクリックします。「ZIP ファイル (ZIP file)」画面が表示されず (図 35 を参照)。

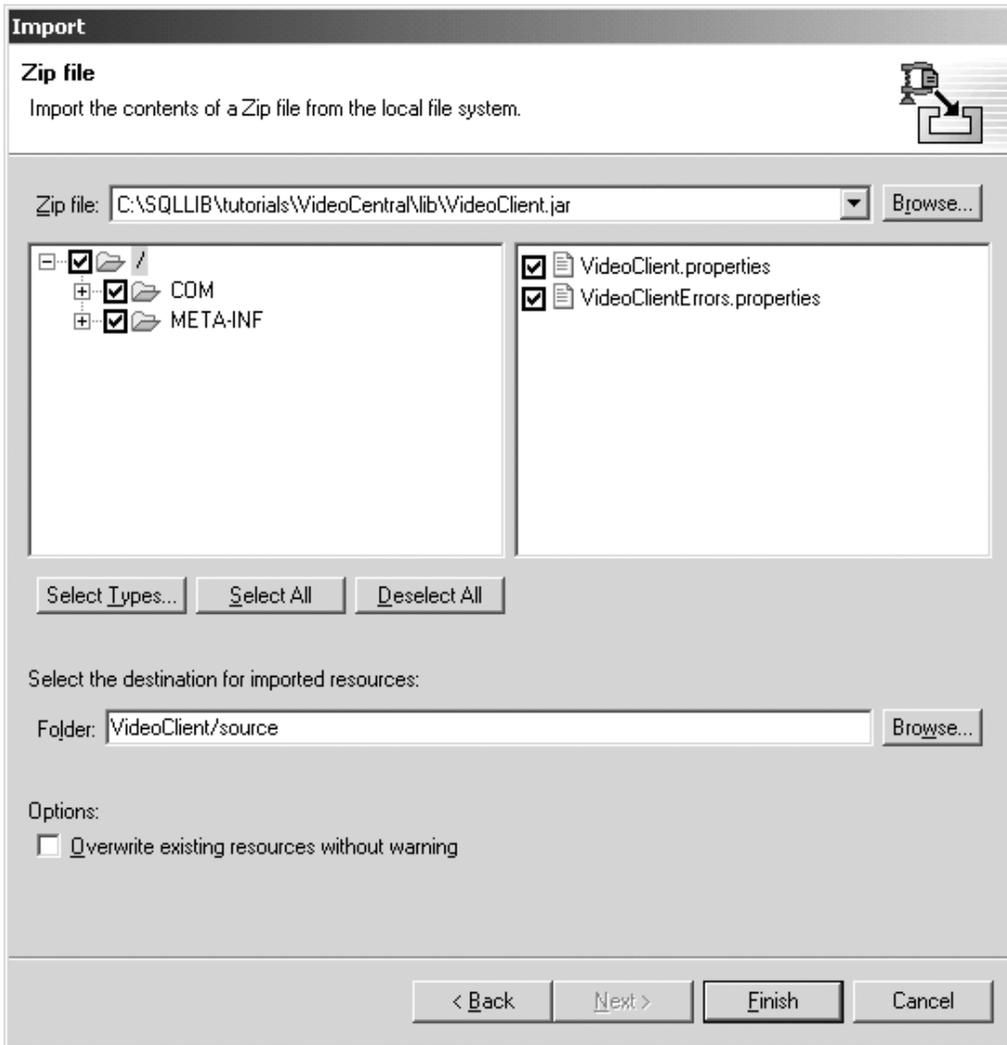


図 35. JAR ファイルから VideoClient プロジェクトへのクライアント・コードのインポート

- e. 「完了 (**Finish**)」をクリックします。これで、ファイルが WebSphere Studio Application Developer にインポートされ、ファイルがコンパイルされていることがアクション・バーに示されます。
4. 次のようにして、クライアント JSP をインポートします。
 - a. VideoClient Web プロジェクトを拡張表示し、Web プロジェクトの **webApplication** を強調表示します (図 36 を参照)。

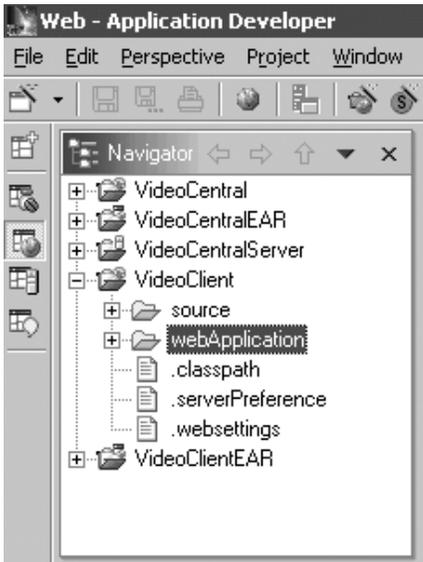


図 36. *webApplication* の強調表示

- b. 「ファイル (File)」 -> 「インポート (Import)」 をクリックします。
- c. 「インポート・ソース (Import Source)」には、「ファイル・システム (File System)」を選択します。「次へ (Next)」をクリックします。
- d. `<DB2_install_dir>%tutorials%VideoCentral%jsp` をブラウザして、「OK」をクリックします。「ファイル・システム (File System)」画面が表示されます。「全選択 (Select All)」をクリックします (図 37 を参照)。

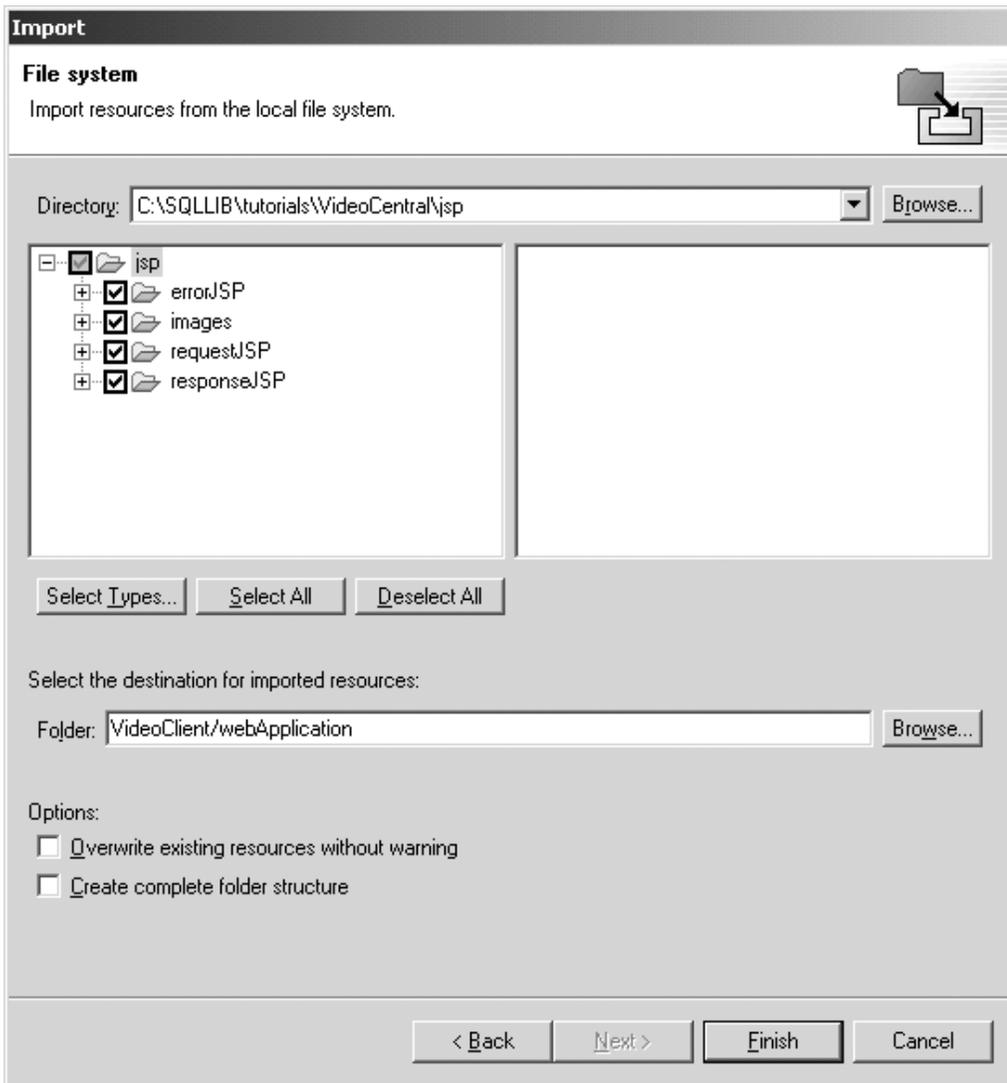


図 37. 「ファイル・システム (File System)」画面

- e. 「完了 (Finish)」をクリックします。

これで、ファイルが WebSphere Studio Application Developer にインポートされ、ファイルがコンパイルされていることがアクション・バーに示されます。

ファイルが VideoClient プロジェクトにインポートされたら、次のような構造になっているはずです (図 38 を参照)。

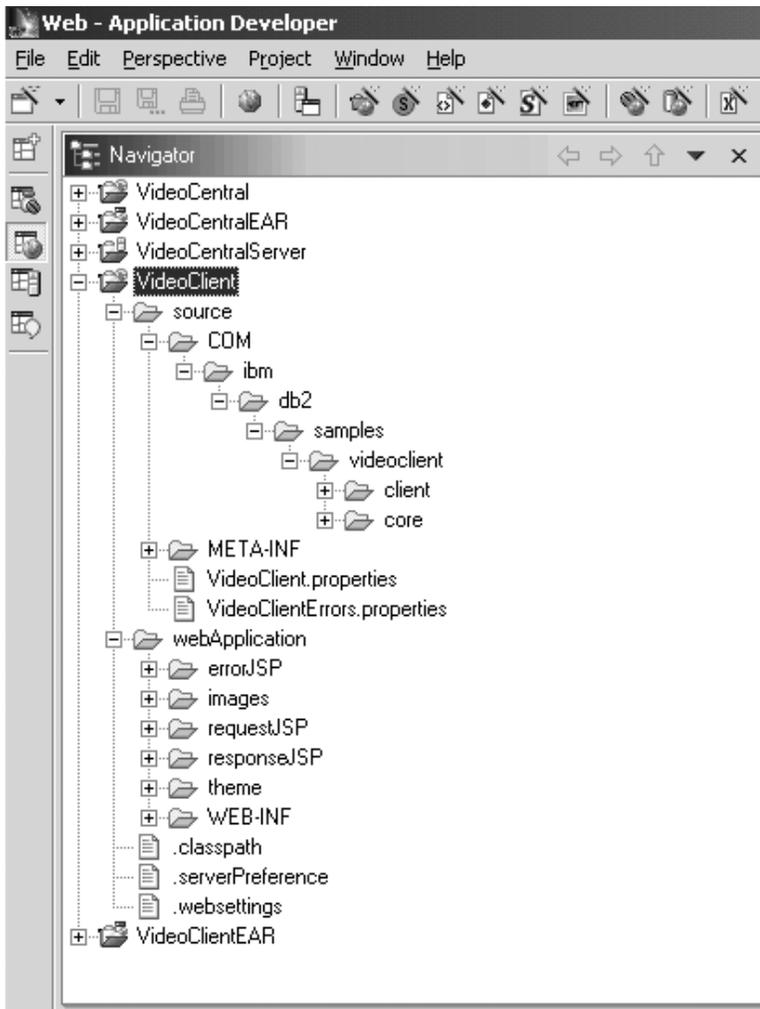


図 38. Video Central クライアント (VideoClient) のインポートの成功

作業 2: WebSphere Studio Application Developer を使用したプロキシの作成

このセクションを実行するためには、第 4 章を完了している必要があります。この作業を実行するには、以前のセクションで作成した WSDL ファイルが必要です。

1. Web パースペクティブから、Web プロジェクト VideoClient の下の、**source/COM/ibm/db2/samples/videoclient** を右クリックします。「新規 (New)」→「フォルダー (Folder)」を選択して、proxies という名前の新規フォルダーを作成します。コンテナは

VideoClient/source/COM/ibm/db2/samples/videoclient/proxies と表示するはずで
す。「完了 (Finish)」をクリックします。

2. 次のようにして、BusinessRegistration サービスのプロキシを生成します。
 - a. Web パースペクティブから、VideoCentral Web プロジェクトの下の **webApplication¥wsdl** を拡張表示し、「**BusinessRegistration-service.wsdl**」を右クリックし、「**新規 (New)**」→「**その他 (Other)**」を選択します。ポップアップ・ウィンドウで、「**Web サービス (Web Services)**」→「**Web Service Client (Web サービス・クライアント)**」の順に選択します。「**次へ (Next)**」をクリックします。
 - b. 「**Web サービス・バインディング・プロキシの生成 (Web Service Binding Proxy Generation)**」ポップアップ・ウィンドウが表示されるまで、「**次へ (Next)**」を再度クリックします。
 - c. 「**フォルダー (Folder)**」フィールドで、/VideoCentral/source を /VideoClient/source に置き換えます。
 - d. 「**クラス (Class)**」フィールドで、 proxy.soap.BusinessRegistrationProxy を以下のとおりに置き換えます。

`COM.ibm.db2.samples.videoclient.proxies.BusinessRegistrationProxy`

注: 便宜のために、残りのプロキシの作成の間、このパスをクリップボードにコピーしておいてください。後続のプロキシのパスには、それを貼り付けることができます。

以下のような表示になります (図 39 を参照)。

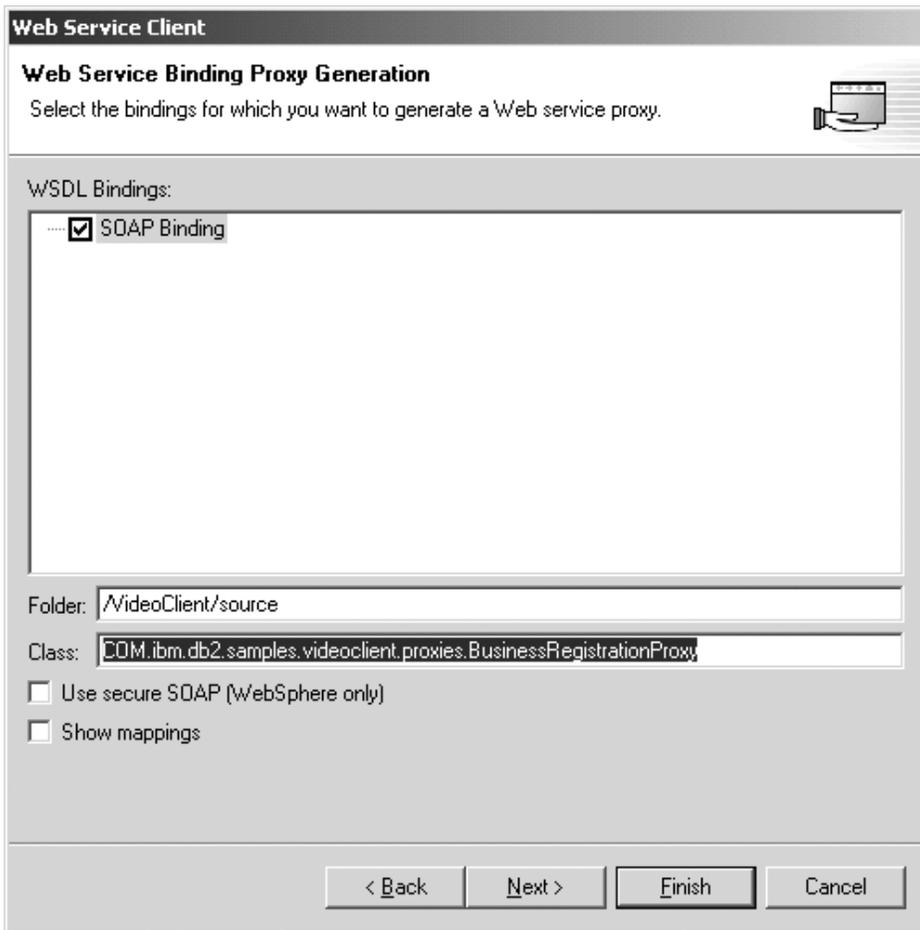


図 39. BusinessRegistration プロキシの生成

e. 「完了 (**Finish**)」をクリックします。

残りの 5 つの WSDL ファイルについて、『作業 2、ステップ 2』を繰り返します。

- CustomerRegistration-service.wsdl
- CustomerInfraction-service.wsdl
- CustomerRentedList-service.wsdl
- WishList-service.wsdl
- MovieSearch-service.wsdl

VideoClient Web プロジェクトに、プロキシ Java ファイルが表示されるはずですが、WebSphere Studio Application Developer には、コンパイラー・エラーが表示されていて

はなりません。ただし、受け入れられるいくつかの警告があるかもしれません (図 40 を参照)。

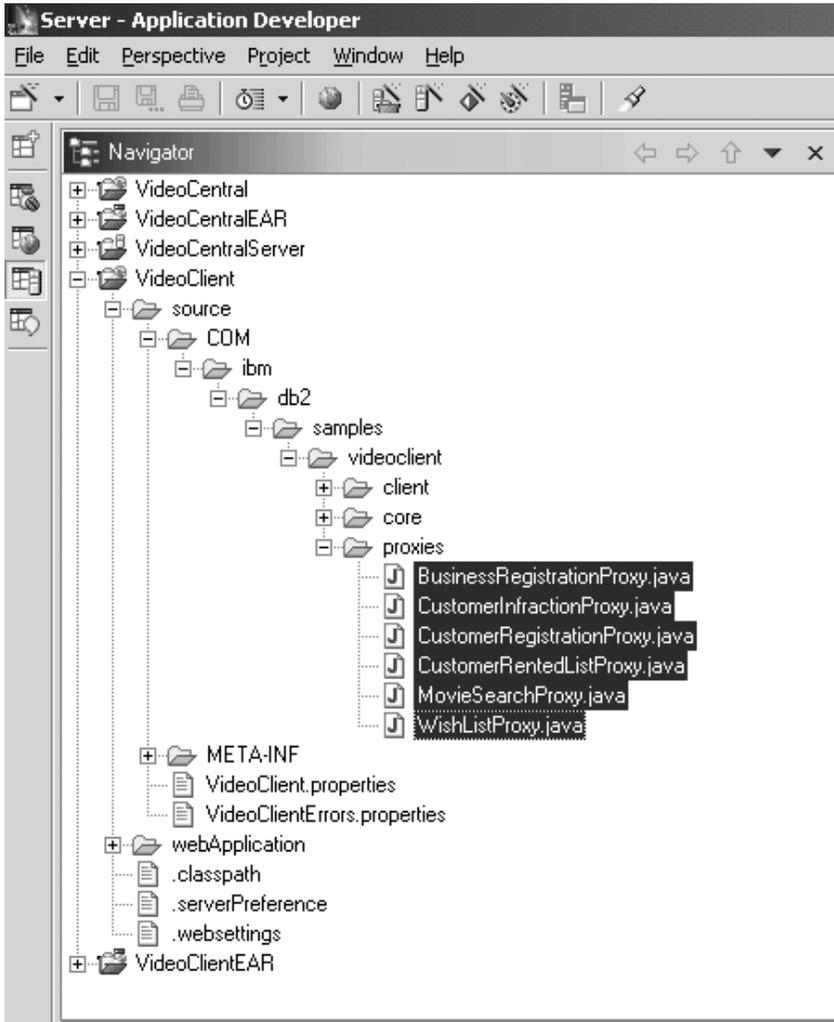


図 40. すべての WSDL ファイルに対して作成されたプロキシ

作業 3: WebSphere Application Server へのサンプル・クライアントのデプロイ

1. Web パースペクティブで、**VideoClient\webApplication\WEB-INF** フォルダを拡張表示します。

2. **Web-INF** フォルダで、web.xml ファイルを開きます。
3. 「サーブレット (Servlets)」タブを選択します。
4. 次のようにして VideoClientServlet を追加します。
 - a. 「追加 (Add)」ボタンをクリックし、**VideoClientServlet** を選択して「OK」をクリックします。
 - b. **VideoClientServlet** を強調表示し、「URL マッピング (URL mappings)」フィールドで、「追加 (Add)」をクリックし、(New URL) のところで servlet/VideoClient と記入します。以下のような表示になります (図 41 を参照)。

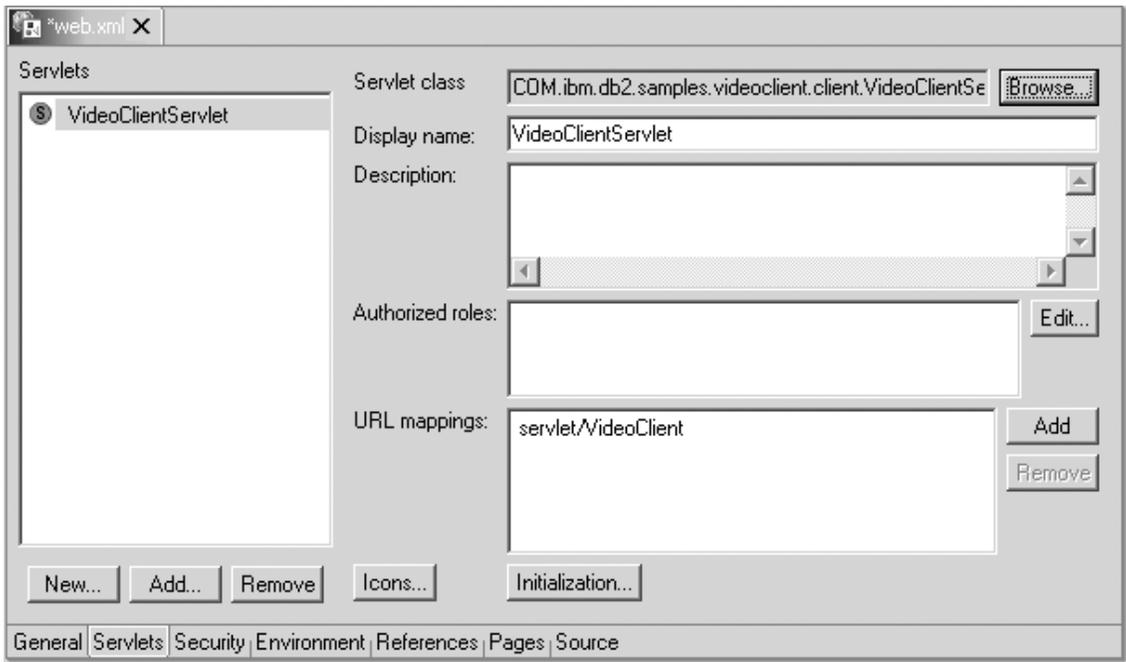


図 41. クライアント・サーブレットの追加

5. 「ファイル (File)」→「web.xml の保管 (Save web.xml)」をクリックします。
6. web.xml ウィンドウを閉じます。

作業 4: サンプル・クライアントの実行

1. クライアントにアクセスするには、Web ブラウザーを開き、
http://localhost:9080/VideoClient/requestJSP/VideoClientRequest.jsp に移動します。
2. 先に進み、クライアントを使用することができます。Video Central サービスを開始するには、ビジネスを登録する必要があります。カスタマーがだれも登録されていない

い場合、次のステップはカスタマーの登録です。カスタマー違反行為を追加するには、XML Extender を使用します。カスタマー違反行為を追加した後は、違反行為の照会を実行することができます。XML 対応の Web ブラウザーを使用している場合は、違反行為を含む XML 文書がブラウザーに表示されます。他にもお望みのサービスがあれば自由に試してみてください。

注: ファイル `info.vc` が `c:\¥` ディレクトリーに作成されます。Video Central の現行の設計では、クライアント・ビジネスを登録するためにこのファイルを読み取りおよび書き込みします。これが最善の方法でないことはわかっていますので、将来のフェーズではこの設計は変更される予定です。Video Central は企業間取引 (B2B) アプリケーションなので、各 Web サービスの起動間で利用できるコンテキストはありません。`info.vc` ファイルを使用すると、それぞれの要求で ID とパスワードを尋ねることなく、ビジネス (クライアント) からの要求を実行することが可能になります。理想的には、それぞれの Web サービス要求に対して信頼のおけるセキュリティ・トークンをエンコードするのが望ましいでしょう。

学習のまとめ

この章では、以下の点を習得しました。

- Video Central サンプル・クライアント設計の概説
- プロキシ・クラスの生成
- クライアント・コードのデプロイメント
- クライアントの実行

第 8 章 共通問題の解決

1. WebSphere Studio Application Developer から WebSphere Application Server へのコードのデプロイメントが成功しない場合は、以下の点を確認してください。
WebSphere インストール・パスが間違っている可能性があります。WebSphere Studio Application Developer における「**Java bean Web サービスのデプロイメント (Java bean Web Service Deployment)**」画面のデフォルトは、`C:\WebSphere\AppServer` です。これを、ご使用の環境での WebSphere Application Server の正しいインストール・ディレクトリに変更してください。
2. 以下のエラー・メッセージを受け取った場合は、下記のようにします。「サーバーの立ち上げ失敗: トレース・サービス・ポート 7000 が使用中です。(Launching the server failed: Trace service port 7000 is in use.) ORB ブートストラップ・ポート 900 が使用中です。(ORB bootstrap port 900 is in use.) 使用中のポート番号を、サーバー構成エディターのポート・ページ上の別の未使用のポートに変更してください。(Change each used port number to another unused port on the ports page of the server configuration editor.) 実行中の別の WebSphere がある場合は、使用されているそれぞれのポート番号を 1 ずつ増やして再試行してください。(In case you have another WebSphere server running you can try to increase each used port number by one and try again)」
WebSphere Studio Application Developer で、「サーバー (Servers)」パースペクティブの「サーバー (Servers)」パネルを選択します。このパネル内の **VideoCentralServer** を右クリックし、「**停止 (Stop)**」を選択します。
3. Video Central が提供する Web サービスにブラウザからアクセスする際に何らかのエラーが生じる場合にまず行うべきことは、サーバー・インスタンスの「サーバー状況 (Server Status)」が「サーバーは同期している (Server is synchronized)」になっているか確認することです。WebSphere Studio Application Developer の中で Web プロジェクトを再公表する必要があるかもしれません。
4. Video Central Web サービスにアクセス中に次のようなエラー・メッセージを受け取ったとします。`"SOAPException: faultCode=SOAP-ENV:Client; msg=Error opening socket: Connection refused: no further information; targetException=java.lang.IllegalArgumentException: Error opening socket: Connection refused: no further information]"` この場合は、サービスと通信するために対応するプロキシ・クラスによって使用されるポート番号が、SOAP ランタイムが聴取するポート番号と同じであること (このチュートリアルでは、ポート番号は 9080) を確認してください。
5. 「DES7300N インデックスが活動化されていません。SQL0438N (DES7300N Index has not been activated, SQL0438N)」のようなエラー・メッセージを受け取った場合は、`VideoCentral.properties` 中の `Net Search` のインデックスの位置のエント

リー mvnets.inxdir を調べ、正しいパスが設定されていることを確認してください。以下のコマンドを実行する必要もあるかもしれません。

- db2nx update index name database sample
 - db2nx update index plot database sample
6. WebSphere Studio Application Developer での何らかのアプリケーション・エラーをデバッグする場合、「サーバー (Servers)」パネルのサーバー・コンソール・ログが、問題解決のための有用な情報を提供します。

第 9 章 追加情報

- IBM DB2: <http://www.ibm.com/software/data/db2/>
- XML Extender ドキュメンテーション:
<http://www.ibm.com/software/data/db2/extenders/xmlxt/library.html>
- Net Search Extender ドキュメンテーション:
http://www.ibm.com/software/data/db2/extenders/net_search/download.html
- SOAP : <http://www.w3.org/TR/SOAP/>
- Java プログラミング: <http://www.javasoft.com>
- WSDL: <http://www.w3.org/TR/wsdl>
- Web サービス情報: <http://www.ibm.com/developerworks/webservices/>
- IBM DeveloperWorks - XML: <http://www.ibm.com/developerworks/xml/>
- WebSphere Developer Domain: <http://www7b.boulder.ibm.com/wsdd/>
- DB2 Developer Domain: <http://www7b.boulder.ibm.com/dmdd/>
- DB2 Web サービス: <http://www.ibm.com/software/data/webservices/>
- WebSphere/DB2 統合: <http://www.ibm.com/software/data/integration/WebSphere>

付録 A. 参照情報

DB2 XML Extender とは

XML Extender は、IBM の DB2 Universal Database の機能と XML の柔軟性とを統合するために役立ちます。DB2 の XML Extender により、XML 文書の保管とアクセス、既存のリレーショナル・データからの XML 文書の生成、および XML 文書からリレーショナル・データへの展開 (分解して、タグを除去した要素または属性内容を保管する) を行うことができます。

XML Extender は、DB2 内の XML データを管理するために、新規のデータ・タイプ、機能、およびストアード・プロシージャを提供します。さらに、複数の表の列として、XML をそのコンポーネント部分に分解し、保管することもできます。検索を高速にするために、XML 文書の要素または属性に索引を定義することができます。さらに、Text Extender を使用して XML 列または分解された部分に対し、テキスト検索およびセクション検索を行うこともできます。

さらに、XML 文書を既存の DB2 表から整形して、企業間取引 (B2B) の環境でのデータ交換に使用することもできます。Net.Data および XML Extender を使用して XML 文書を DB2 から作成し、これらの文書を消費者に送信してブラウザで閲覧できるようにすることができます。

XML Extender には、XML データを DB2 と共に管理および活用するために役立つ以下の機能が備わっています。

- リレーショナル表への XML データの組み込みを管理するために役立つ管理ツール。
- XML データを保管および使用するための方式 : XML 列、XML コレクション。
- XML データの妥当性検査に使用した文書型定義 (DTD) を保管するための、DTD リポジトリ (DTD_REF)。
- XML 文書をリレーショナル・データにマップするための、Document Access Definition (DAD) ファイルと呼ばれるマッピング方式。DAD ファイルは、XML のデータベースを使用可能にするときに作成される XML_USAGE 表を使用して管理されます。
- XML 文書を XML 列に保管したり、そこから検索したり、XML 要素または属性値を抽出するための、強力なユーザー定義機能 (UDF)。UDF はデータベース管理システム内に定義された機能で、その後は SQL 照会で参照できます。XML Extender は、以下のタイプの UDF を提供します。
 - Storage: 未加工の XML 文書を XML 対応の列に XML データ・タイプとして保管します。

- Extract: XML 文書、または指定の要素および属性を基本データ・タイプとして抽出します。
- Update: XML 文書全体または指定の要素および属性値を更新します。

XML Extender のすべての UDF には、接頭部として db2xml1 が付いています。これは DB2 XML Extender UDF のスキーマ名です。

DB2 Net Search Extender とは

DB2 Net Search Extender は、DB2 Universal Database に組み込まれた Net.Data、Java、または DB2 CLI アプリケーションに、高速なフルテキスト検索エンジンの機能を追加します。

Net Search Extender は、非常に大きなボリュームのデータを高速に索引付けし、大量の同時ユーザーによる高速な検索を可能にし、索引付けのときに事前ソートされた表列をメイン・メモリーに保管することにより、検索時に不経済なデータベース・アクセスおよびページングを回避します。

Net Search Extender は、DB2 Text Extender および DB2 Text Information Extender の機能を補完する、以下の機能を提供します。

- 索引付け：
 - 1 つの非常に高速な索引タイプ。(Ngram)
 - 同じテキスト列に複数の索引を設定することが可能。
 - データをロックすることなく索引付けができる。
 - データベース内の変更を反映する動的な索引の更新。
- 検索：
 - UDF の代わりにストアード・プロシージャが備わっている。
 - ワード、句、ステム、またはファジー検索が可能。
 - 特殊タグでマークされた文書内のセクションを識別して、検索をそこだけに制限する。
 - 値の範囲に対して数値検索が可能。
 - ブールおよびワイルドカード操作をサポートしている。
- 検索結果：
 - 索引付けのときに、検索結果のソート方法またはソート用のユーザー提供ランク値を指定することが可能。
 - 大量のボリュームのデータが検索されて大量の結果リストが見込まれているとき、検索結果のサブセットを指定することが可能。
 - 検索用語の数にヒット・カウントの上限による制限を設定することが可能。
 - 検索結果に対する位置決め (カーソル設定) アクセスが可能。

IBM WebSphere Studio Application Developer とは

IBM の WebSphere Studio Application Developer (本書では WSAD と呼ぶこともありま
す) は、WebSphere Studio, Professional and Advanced Editions、および VisualAge® for
Java, Enterprise Edition に対する後継のテクノロジーです。これらの製品は、e-business
アプリケーションのエンドツーエンドの開発、テスト、およびデプロイメントをサポート
しています。

新しい WebSphere Studio 製品は、あらゆるタイプの新しいアプリケーションの要件に
応じるために、基礎から設計されたものです。それらの要件には、オープン標準、
Java、XML、Web サービス、テスト、他のコンポーネントおよび ISV 製品とのさまざま
なレベルでの統合、プラグイン機能、拡張容易性、役割ベースの開発、すべてのユー
ザーのユーザビリティの向上、強化されたチーム・サポート、および市場に出すまで
の速度の向上が含まれます。WebSphere Studio は、Web 開発者、Java 開発者、ビジネ
ス・アナリスト、設計者、企業プログラマーを含むすべての e-business 開発役割のため
に、統合された開発ツールを提供します。

WebSphere Studio は、WebSphere Studio "Classic" および VisualAge for Java の最も評
判の良い機能を統合して、最新のテクノロジーを使用して結合することにより、オー
プン標準、ツールの統合、柔軟性の向上、および既存の複数のアプリケーションを結合す
る機能を提供します。

Web サービス環境について

WebSphere Studio には、Web サービスを素早く開発するためのウィザードその他のツ
ールが備わっています。

WebSphere Studio に備わっている Web サービス開発ツールは、オープンで、複数のプ
ラットフォームにわたる標準に基づいています。

- **Universal Description Discovery and Integration (UDDI):** 企業が自己を表現し、
他の企業とどのような e-business を行いたいかを公開し、必要な商品とサービスを提供
している他の企業を検索することを可能にします。これらはすべて、オンラインの
UDDI レジストリーを介して行われます。
- **Simple Object Access Protocol (SOAP):** ビジネス・アプリケーション相互間でイ
ンターネットを介して、電子ビジネス・メッセージを信頼性のある方法で転送するた
めの標準。

SOAP は、非集中の分散環境で情報を交換するための軽量のプロトコルです。これは
次の 3 つの部分から構成される XML ベースのプロトコルです。それらは、メッセ
ージの内容と処理方法を記述するためのフレームワークを定義するエンベロップ、ア
プリケーション定義のデータ・タイプのインスタンスを表現するためのエンコード・
ルールのセット、リモート・プロシーチャー・コールおよび応答を表す規則です。

SOAP は他のさまざまなプロトコルと組み合わせて使用できます。しかし、本書で説明する唯一のバインディングでは、SOAP を HTTP および HTTP Extension Framework と組み合わせて使用する方法を定義しています。

SOAP バージョン 1.1 の仕様およびその他の SOAP 参照情報については、<http://www.w3.org/TR/SOAP/> を参照してください。

- **Web サービス記述言語 (WSDL):** インターネット (または他のネットワーク) を介してアクセス可能なプログラム、およびそれらと通信するためのメッセージ形式とプロトコルについて記述します。

WebSphere Studio は以下のプロセスを容易にして、Web サービスに対応したアプリケーションの作成およびデプロイメントを支援します。

- **発見。** UDDI ビジネス・レジストリーをブラウズして、統合する既存の Web サービスを見付けます。
- **作成またはトランスフォーム。** bean、データを受け取りそして戻す URL、DB2 XML Extender コール、DB2 ストアード・プロシージャ、SQL 照会などの既存の人工物から Web サービスを作成します。
- **ビルド。** 既存の人工物を SOAP および HTTP GET/POST アクセス可能サービスとしてラップして、それらを WSDL で記述します。Web サービス・ウィザードは SOAP プロキシーを WSDL で記述された Web サービスに生成すること、および bean スケルトンを WSDL から作成することを支援します。
- **デプロイ。** サーバー・ツールを使用して、WebSphere Application Server または Tomcat テスト環境に Web サービスをデプロイします。
- **テスト。** ローカル側またはリモート側で実行している Web サービスをテストして、即座にフィードバックを取得します。
- **開発。** Web サービスのクライアント・アプリケーションの作成に役立つサンプル・アプリケーションを生成します。
- **公開。** Web サービスを UDDI ビジネス・レジストリーに公開して、他の企業がアクセスできるように Web サービスを宣伝します。

Web サービス・ツールの詳細については、WebSphere Studio Application Developer のインストール・システムに付属のオンライン・ヘルプを参照してください。

XML 開発環境について

WebSphere Studio は、DTD、XML スキーマ、および XML ファイルを作成するためのツールを含む広範囲の XML 開発環境を提供します。さらに、リレーショナル・データおよび XML の統合もサポートしています。

以下の XML ツールを使用できます。

XML エディター は、XML ファイルを作成、表示、および検査するためのツールです。これを使用して、新規の XML ファイルを、初めからでも、既存の DTD または既

存の XML スキーマからでも作成することができます。さらにこれを使用して、XML ファイルを編集し、それを DTD またはスキーマに関連付けて、妥当性の検査をすることができます。

DTD エディター は、DTD を作成、表示、および検査するためのツールです。DTD エディターを使用して、DTD 要素、属性、エンティティー、および記法を作成および検査することができます。XML スキーマ・ファイルを作成して、XML スキーマの XML インスタンスを作成するための Java bean を生成できます。さらに DTD エディターを使用して、作成した DTD に基づくデフォルトの HTML フォームを生成することもできます。

XML スキーマ・エディター は、XML スキーマの作成、表示、および検査を容易にします。XML スキーマ・エディターを使用して、XML スキーマ・コンポーネントの作成、XML スキーマのインポートと表示、XML スキーマからの DTD とリレーショナル表定義の生成、XML スキーマの XML インスタンスを作成するための Java bean の生成、および XML スキーマからの DDL の生成などのタスクを実行することができます。

XSL トレース・エディター によって、XSL スタイルシートを XML 文書に適用して結果文書 (HTML または XML) を作成することができます。XML 文書を HTML、テキスト、または他の XML 文書タイプに変換できます。エディターは 3 つの文書 (結果、ソース XML、ソース XSL) を表示して、XSL 変換スクリプト内を見渡しながら、3 つの文書の関係を検討できるようにします。

XML から XML へのマッピング・エディター は、1 つ以上のソース XML 文書を単一のターゲット XML 文書にマップします。ソース・ファイル (DTD または XML) およびターゲット・ファイルを供給して、ソースとターゲットとの間のマッピングを定義することができます。各マッピングはターゲット・フィールド、変換機能、およびソース・フィールドの集合です。マッピングは編集、削除、または後に使用するために保管することができます。マッピングを定義した後、XSLT スクリプトを生成して、それによりソース DTD に合致する任意の XML 文書を結合および変換することができます。

XML および SQL 照会ウィザード を使用して、SQL 照会の結果から XML ファイルを作成することができます。オプションとして、他のアプリケーションで使用するために、XML ファイルが持つ構造を記述する XML スキーマまたは DTD ファイルを作成することもできます。さらに XML および SQL 照会ウィザードを使用して、Web サービス・ツールと共に使用可能な DADX ファイルを作成することもできます。生成された DADX ファイルには、SQL 照会が含まれています。

RDB から XML へのマッピング・エディター は、リレーショナル表と DTD ファイルとの間のマッピングの定義を容易にします。1 つ以上のリレーショナル表内の列を、XML 文書内の要素または属性にマップすることができます。IBM DB2 Extender が既存の DB2 データから XML 文書を形成するため、または XML 文書を分解して DB2

データにするための、 Document Access Definition (DAD) スクリプトを生成できます。さらに、生成された DAD ファイルをテストするためのテスト・プログラムも作成できます。

XML ツールの詳細については、 WebSphere Studio Application Developer のインストールに付属のオンライン・ヘルプを参照してください。

付録 B. 用語集

DAD - 文書アクセス定義 (Document Access Definition). XML 列の索引付け方式または XML コレクションのマッピング方式を定義するために使用します。

DTD - 文書型定義 (Document Type Definition). DTD は XML の宣言構文で記述されたファイル (または一緒に使用される複数のファイル) で、特定のタイプの文書に関する形式が記述されています。ここでは、要素タイプに使用できる名前、それらが出現する位置、および全体が組み合わせられる方法が示されています。

JDBC - Java Database Connectivity. Open Database Connectivity (ODBC) と同じ特性を持ち、Java データベース・アプリケーションで使用するために特に設計されたアプリケーション・プログラミング・インターフェース (API)。

SOAP - Simple Open Access Protocol. XML でエンコードされたデータを渡す一定の方法を定義しているプロトコル仕様。

UDDI - Universal Description, Discovery and Integration Service. クライアントが他の Web サービスを動的に検索するためのレジストリー・メカニズム。

WSDL - Web サービス記述言語 (Web Services Description Language). Web サービスの機能、それが存在する場所、およびその起動方法について記述した XML フォーマット。

XML - eXtensible Markup Language. Web 上の構造化された文書およびデータのための世界共通フォーマット。これは HTML のように固定フォーマットではないので、拡張可能です。これは SGML (Standard Generalized Markup Language) のサブセットです。

XSL - eXtensible Style Language. XML の標準スタイルシート言語を定義します。

付録 C. 特記事項

以下は、IBM Corporation の商標です。

- DB2
- IBM
- VisualAge
- WebSphere

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

他の会社名、製品名およびサービス名などはそれぞれ各社の商標または登録商標です。

IBM と連絡をとる

技術上の問題が生じた場合、お客様サポートに連絡する前に、第 8 章『共通問題の解決』および「DB2 問題判別の手引き」を調べて、そこに提案されている処置を実行してください。「DB2 問題判別の手引き」には、お客様サポートがより良いサービスを提供できるようにユーザーが収集できる情報についての提案が記載されています。

DB2 Universal Database 製品の問い合わせまたは注文については、お近くの IBM 担当者または IBM 特約店 / 販売店に連絡してください。

米国内からは、以下の電話番号を使用できます。

- 1-800-237-5511 (お客様サポート)
- 1-888-426-4343 (使用可能なサービス・オプションについての問い合わせ)

製品情報

DB2 Universal Database 製品の情報は、電話または以下の URL で入手できます。
<http://www.ibm.com/software/data/db2/udb/winos2unix/support>

このサイトには、テクニカル・ライブラリー、書籍の注文、クライアント・ダウンロード、ニュースグループ、フィックスパック、ニュース、および Web リソースへのリンクに関する最新情報が載せられています。

米国内からは、以下の電話番号を使用できます。

- 1-800-IBM-CALL (1-800-426-2255) (製品の注文または一般情報の問い合わせ)
- 1-800-879-2755 (資料の注文)

米国外から IBM に連絡する方法については、IBM の Web ページ
<http://www.ibm.com/planetwide> を調べてください。



Printed in Japan

日本アイ・ビー・エム株式会社

〒106-8711 東京都港区六本木3-2-12