

IBM DB2 Universal Database



Tutoriel IBM Video Central pour l'e-business

Version 8.1

IBM DB2 Universal Database



Tutoriel IBM Video Central pour l'e-business

Version 8.1

Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à l'Annexe C, «Remarques» à la page 93.

Remarque : Les captures d'écrans et les graphiques de cette brochure ne sont pas disponibles en français à la date d'impression.

Première édition – avril 2002

LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT". IBM DECLINE TOUTE RESPONSABILITE, EXPRESSE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE QUALITE MARCHANDE OU D'ADAPTATION A VOS BESOINS. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.can.ibm.com> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France
Direction Qualité
Tour Descartes
92066 Paris-La Défense Cedex 50*

© Copyright IBM France 2002. Tous droits réservés.

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

Table des matières

Figures.	v	Chapitre 4. Initiation à WebSphere Studio Application Developer	17
Chapitre 1. Préface	1	Durée	17
Présentation du tutoriel	1	Présentation	17
Connaissances requises.	1	Logiciels requis	17
Durée du tutoriel	2	Tâches relatives à Video Central	18
Logiciels requis	2	Tâche 1 : Définition de l'instance de serveur éloigné WAS	18
Conventions utilisées dans ce tutoriel	2	Tâche 2 : Définition du projet Web Video Central et importation du code source	22
Chapitre 2. Exemple d'application Video Central	3	Résumé de la leçon	29
Durée	3	Étape suivante	29
Avant de commencer	3	Chapitre 5. Conception de l'application Video Central	31
Présentation	3	Durée	31
Présentation des services Web	3	Présentation	31
Qu'est-ce qu'un service Web ?	3	Conception de Video Central	31
Comment accéder à un service Web	5	Couche WIL	31
Scénario Video Central	6	Développement de la couche BLL.	31
Objectifs commerciaux	6	Modélisation des données	43
Architecture de l'application Video Central	7	Résumé de la leçon	46
Résumé de la leçon	10	Étape suivante	46
Étape suivante	10	Chapitre 6. Déploiement de l'application Video Central	47
Chapitre 3. Préparation de l'environnement DB2 en vue de l'utilisation de Video Central	11	Durée	47
Durée	11	Logiciels requis	47
Avant de commencer	11	Présentation	47
Logiciels requis	11	Tâches relatives à Video Central	47
Présentation	11	Tâche 1 : Préparation de l'environnement au déploiement de Video Central à partir de WebSphere Studio Application Developer.	47
Tâches relatives à Video Central	11	Tâche 2 : Déploiement des services Web sur le serveur WebSphere Application Server	49
Tâche 1 : Création de l'ID utilisateur "ivc"	11	Résumé de la leçon	57
Tâche 2 : Création de l'exemple de base de données	12	Étape suivante	57
Tâche 3 : Préparation de l'exemple de base de données en vue de l'utilisation de DB2 Extension XML	12	Chapitre 7. Création et exécution d'un exemple de client Web pour Video Central	59
Tâche 4 : Préparation de l'exemple de base de données en vue de l'utilisation de DB2 Extension Net Search	14	Durée	59
Tâche 5 : Création et remplissage des tables de Video Central	15	Avant de commencer	59
Résumé de la leçon	15	Présentation	59
Étape suivante	16		

Conception de l'exemple de client Video Central.	59
Schémas de classes et de séquences de l'exemple de client de Video Central	60
Tâches relatives à Video Central	63
Tâche 1 : Préparation du projet client	63
Tâche 2 : Création des proxy à l'aide de WebSphere Studio Application Developer	71
Tâche 3 : Déploiement de l'exemple de client sur le serveur WebSphere Application Server	75
Tâche 4 : Exécution de l'exemple de client	76
Résumé de la leçon	77
Chapitre 8. Résolution des incidents courants	79
Chapitre 9. Sources d'informations complémentaires	81

Annexe A. Informations de référence	83
Qu'est-ce que DB2 Extension XML ?.	83
Qu'est-ce que DB2 Extension Net Search ?.	85
Qu'est-ce qu'IBM WebSphere Studio Application Developer ?.	86
Environnement des services Web	86
Environnement de développement XML	89
Annexe B. Glossaire.	91
Annexe C. Remarques	93
Comment contacter IBM	95
Informations produit	95

Figures

1. Composants des services Web.	4	22. Choix du service Web à déployer.	50
2. Architecture de l'application Video Central.	9	23. Java Bean à déployer.	51
3. Création et configuration d'une instance de serveur.	19	24. Configuration du Java Bean en tant que service Web.	52
4. L'instance de serveur a été créée.	20	25. Sélection des méthodes publiques à déployer.	53
5. Configuration de la source de données.	21	26. Génération du proxy de lien au service Web..	54
6. Ajout de fichiers JAR externes.	23	27. Contenu du fichier WSDL.	55
7. Ajout du projet Web au serveur.	24	28. Tous les services Web déployés.	56
8. Ajout du projet Web au serveur (suite).	25	29. Composants des services Web.	60
9. Avant d'importer les fichiers JAR.	26	30. Schéma des classes de l'exemple de client de Video Central, appelé Video Client.	61
10. Sélection des ressources à importer.	27	31. Schéma de séquence de l'exemple de client de Video Central, appelé Video Client.	62
11. Après préparation du projet Web Video Central..	28	32. Ajout de fichiers JAR externes.	64
12. Schéma général des classes de l'application Video Central.	32	33. Ajout du projet Web client au serveur.	65
13. Schéma des classes du service Web Infractions d'un client..	34	34. Sélectionnez Source.	66
14. Schéma des classes du service Web Liste des souhaits d'un client.	36	35. Importation du code du client dans le projet VideoClient à partir d'un fichier JAR.	67
15. Schéma de séquence du service Web Ajout d'une infraction d'un client.	38	36. Sélection de webApplication.	68
16. Schéma de séquence du service Web Interrogation des infractions d'un client..	40	37. Écran File system.	69
17. Schéma de séquence du service Web Ajout d'un titre à la liste des souhaits d'un client.	42	38. Après l'importation du client de Video Central, appelé VideoClient..	70
18. Schéma du modèle de données — première partie.	44	39. Génération du proxy BusinessRegistration.	72
19. Schéma du modèle de données — seconde partie.	45	40. Proxy créés pour tous les fichiers WSDL.	74
20. Vérification de HTTP server..	48	41. Ajout des servlets client.	75
21. Vérification du fonctionnement su serveur d'administration de WAS.	49		

Chapitre 1. Préface

Présentation du tutoriel

Le tutoriel IBM Video Central, qui comprend un guide d'apprentissage pas à pas ainsi qu'une application opérationnelle complète, illustre l'utilisation de DB2 en tant que fournisseur de services Web. Les services Web s'appuient sur des technologies nouvelles : le protocole SOAP (Simple Object Access Protocol), le langage de description WSDL (Web Services Description Language) et la spécification UDDI (Universal Description, Discovery and Integration). SOAP est un format de message standard utilisé pour appeler des services Web. L'application fournie dans le tutoriel Video Central transporte des messages SOAP en utilisant le protocole Internet standard HTTP (Hyper Text Transfer Protocol). Des demandeurs de services Web peuvent être développés à l'aide du document WSDL du fournisseur de services Web. Pour plus d'informations concernant les services Web, consultez le site <http://www.ibm.com/software/webservices>.

Les services Web constituent un nouveau modèle de développement pour l'intégration d'applications. Ils fournissent une technique standard pour décrire, publier, découvrir et appeler une logique applicative dans un environnement informatique réparti. Le présent tutoriel comprend toute la documentation requise pour déployer six services Web sur un serveur WAS (WebSphere Application Server). Une application client est également fournie, qui permet d'illustrer le fonctionnement de ces services Web. Ce tutoriel montre également les possibilités des fonctions de mappage XML dynamiques de DB2 Extension XML, et du moteur d'extraction en texte intégral hautes performances de DB2 Extension Net Search.

Connaissances requises

Ce tutoriel s'adresse aux programmeurs Java professionnels qui possèdent de bonnes connaissances pratiques dans les domaines suivants :

- WebSphere Application Server,
- concepts de services Web (pour plus d'informations, reportez-vous au "Chapitre 2. Préparation de l'environnement DB2 en vue de l'utilisation de Video Central"),
- DB2 Universal Database,
- concepts de la programmation orientée objet,
- langage de programmation Java.

Durée du tutoriel

L'étude de ce tutoriel nécessite environ deux heures.

Logiciels requis

Avant de poursuivre, assurez-vous que les logiciels suivants, qui sont utilisés dans le présent tutoriel, sont installés sur votre poste de travail :

- WebSphere Application Server (WAS) version 4.0 Advanced Edition Single Server
- DB2 Universal Developer's Edition (DB2 UDE), version 8.1, y compris DB2 Extension XML et DB2 Extension Net Search (ces deux extensions sont fournies avec DB2, mais elles figurent sur un CD distinct et doivent être installées séparément)
- WebSphere Studio Application Developer (WSAD), Version 4.0

Conventions utilisées dans ce tutoriel

Dans la totalité du présent tutoriel :

- <rép_install_DB2> représente le répertoire d'installation de DB2,
- <rép_install_WAS> représente le répertoire d'installation de WebSphere Application Server,
- <rép_install_WSAD> représente le répertoire d'installation de WebSphere Studio Application Developer.

Chapitre 2. Exemple d'application Video Central

Durée

L'étude de ce chapitre nécessite environ 20 minutes.

Avant de commencer

Vous devez connaître les concepts de la programmation orientée objet.

Présentation

Ce chapitre présente les services Web et le protocole SOAP (Simple Object Access Protocol). Il décrit également les services Web mis en œuvre dans Video Central, et l'architecture de cette application.

Présentation des services Web

La section qui suit définit le concept de service Web.

Qu'est-ce qu'un service Web ?

Un service Web est un ensemble de fonctions d'application connexes pouvant être appelées à partir d'une autre application. Contrairement aux applications Web traditionnelles, qui utilisent HTML pour présenter les informations, les services Web utilisent XML pour échanger des informations. De nombreux langages de programmation et plateformes peuvent être utilisés pour les mettre en œuvre et y accéder. Dans le présent tutoriel, le langage Java est utilisé pour mettre en œuvre le demandeur et le fournisseur de service Web sur une plateforme Windows.

Les services Web sont des applications modulaires qui effectuent des tâches spécifiques et sont accessibles via des protocoles ouverts tels que HTTP. Les entreprises peuvent combiner dynamiquement plusieurs services Web pour effectuer des opérations complexes en utilisant un modèle de programmation souple.

Le concept de service Web repose sur la définition des trois rôles suivants (voir Figure 1) :

Fournisseur de service

- Fournit les services d'e-business
- Publie la disponibilité de ces services dans un registre (facultatif)

Courtier de services

- Fournit une assistance à la publication et à la recherche de services

Demandeur de services

- Localise les services requis en utilisant le courtier de services
- Se connecte aux services disponibles auprès du fournisseur de services
- Appelle le service souhaité en utilisant l'interface définie

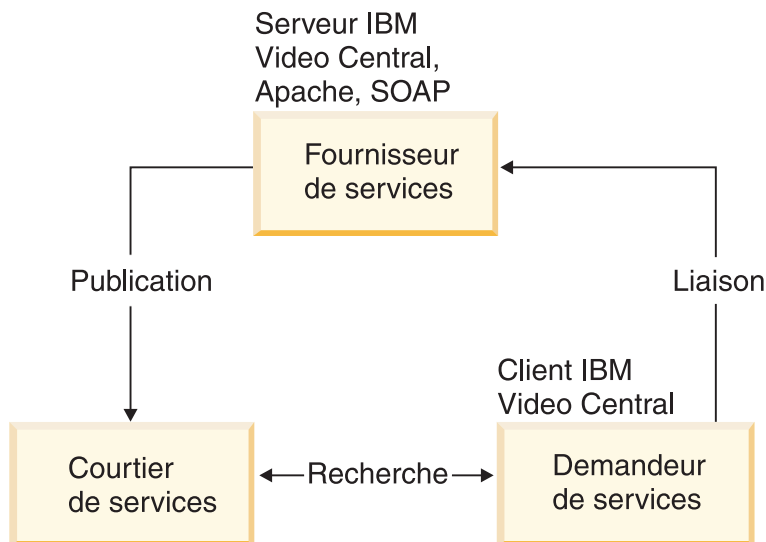


Figure 1. Composants des services Web.

L'espace de rencontre des services Web dans lequel les entreprises formeront des partenariats est le registre UDDI (Universal Description, Discovery and Integration). La spécification UDDI définit des normes ouvertes et indépendantes de la plateforme qui permettent aux entreprises de partager des informations dans un registre commercial global, de découvrir les services proposés par les autres entreprises et de définir comment elles interagissent via Internet. Ce registre a pour vocation de devenir une vaste bibliothèque de fonctions que les développeurs pourront utiliser pour créer rapidement des applications dynamiques. IBM joue un rôle clé dans la définition de la spécification UDDI, que des dizaines d'entreprises de premier plan ont déjà adoptée.

Les services Web sont publiés dans le registre commercial UDDI en utilisant le langage WSDL (Web Services Description Language). Ils interagissent entre eux en utilisant le protocole SOAP (Simple Open Access Protocol). Les services Web peuvent être regroupés en trois catégories :

- *Informations commerciales.* Il s'agit des informations qu'une entreprise partage avec ses clients ou d'autres entreprises. Dans ce cas, l'entreprise utilise les services Web pour développer le nombre de ses clients potentiels. Parmi les services Web d'information commerciale, on peut citer les lettres d'information, les bulletins météo locaux et le suivi des cours de la bourse.
- *Intégration commerciale.* Il s'agit de la fourniture par une entreprise de services transactionnels payants à ses clients. Dans ce cas, l'entreprise crée un réseau global de fournisseurs utilisables pour le commerce. Parmi les services Web d'intégration commerciale, on peut citer les places de marché d'offres et d'enchères, les systèmes de réservation et la vérification de crédit.
- *Externalisation des processus métier.* Il s'agit de la création par une entreprise d'une chaîne globale afin de se démarquer de ses concurrents. Dans ce cas, l'entreprise utilise les services Web pour intégrer dynamiquement ses processus.

Pour plus d'informations concernant la spécification UDDI, consultez le site Web <http://www.uddi.org>.

Comment accéder à un service Web

Une entreprise peut publier (rendre publiques) les fonctions commerciales qu'elle propose sur le Web. Les consommateurs ou clients potentiels découvrent alors la disponibilité de ces services sur le Web en effectuant des recherches dans des registres tels que le registre commercial UDDI.

Les services Web sont une extension naturelle des modèles informatiques réparti et client-serveur. Dans le contexte des services Web, pour appeler des méthodes sur des objets éloignés, on utilise un format XML standard appelé SOAP. Un message SOAP peut être échangé en utilisant différents protocoles de transmission, dont MQSeries et HTTP. Le registre UDDI fournit un annuaire des services disponibles et des fournisseurs de services. L'interface est définie à l'aide d'une norme XML indépendante du langage, WSDL.

IBM est en train d'intégrer à ses produits clés des fonctions de services Web. Des outils permettant de générer, publier, rechercher, tester et appeler automatiquement des services Web sont déjà disponibles. Le présent tutoriel va vous apprendre à utiliser certains de ces outils. IBM offre également une autre méthode pour soumettre des instructions SQL et, si vous le souhaitez, pour contrôler le format des données renvoyées. Cette prise en charge est appelée WORF (Web services Object Runtime Framework) et DADX (Document Access Definition eXtension). Pour plus d'informations, consultez le site Web suivant :

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

Scénario Video Central

Video Central est un fournisseur de services Web générique destiné aux applications de location de cassettes vidéo (applications Business-to-Business). L'objectif de Video Central est de fournir un référentiel de données central auquel les applications Web inscrites peuvent accéder. L'application Video Central fournit un ensemble de services Web qui répondent aux besoins des clients dans deux domaines : les **services d'entreprise** satisfont les besoins administratifs des clients, et les **services clients** permettent aux clients de servir et de gérer leur clientèle.

Objectifs commerciaux

Dans leur travail quotidien, les loueurs de cassettes vidéo sont confrontés aux limitations suivantes :

- saisie manuelle et répétitive des nouveaux titres proposés,
- description limitée des titres (saisie manuelle des informations telles que le résumé, les personnages et les acteurs),
- impossibilité de suggérer des titres à leurs clients en fonction de leurs choix précédents et de leurs goûts,
- fourniture de certains services à valeur ajoutée (liste de souhaits ou historique de location) uniquement sur la base des données enregistrées localement,
- capacité limitée à évaluer le degré de confiance pouvant être accordé aux clients nouveaux ou existants pour la location d'articles coûteux,
- perte de revenus liée aux titres volés par de nouveaux clients (location sans retour).

En résumé, les loueurs de cassettes vidéo n'ont accès qu'aux informations contenues dans leurs propres bases de données. C'est sur la base de cette constatation qu'ont été développés les services de la version actuelle de Video Central.

Services d'entreprise de Video Central

Nous avons conçu un service Web d'entreprise dans ce tutoriel : l'inscription d'entreprise.

Inscription d'entreprise : Ce service Web permet à un client (par exemple, un vidéoclub) d'inscrire son entreprise auprès de Video Central. Lorsqu'une entreprise est inscrite auprès de Video Central, elle peut utiliser les autres services Web disponibles.

Services clients de Video Central

Nous avons conçus cinq services Web clients spécifiques : inscription d'un client, infractions d'un client, historique des locations d'un client, liste des souhaits d'un client et recherche de films.

Inscription d'un client : Ce service Web permet à une entreprise inscrite (en l'occurrence, le vidéoclub) d'inscrire leurs clients auprès de Video Central. Lorsqu'un client est inscrit, des informations utiles (par exemple, l'historique de ses locations, sa liste de souhaits, ses infractions) le concernant peuvent être enregistrées, consultées et analysées à l'aide des autres services disponibles.

Infractions d'un client : Ce service Web permet à une entreprise inscrite d'ajouter les infractions commises par un client au référentiel de Video Central. Elle peut ainsi suivre les infractions de ses clients sans être obligée de stocker les données correspondantes dans sa base de données locale. Ce service Web lui permet également d'interroger à tout moment l'historique des infractions d'un client existant ou potentiel. Video Central étant utilisé comme référentiel central par de nombreux vidéoclubs, l'entreprise peut ainsi vérifier si un client a commis des infractions dans d'autres vidéoclubs avant d'accepter ou non de lui louer un article nouveau.

Historique des locations d'un client : Ce service Web permet à une entreprise inscrite de tenir la liste des cassettes louées par chacun de ses clients. Il lui permet également d'interroger à tout moment cet historique.

Liste des souhaits d'un client : Ce service Web permet à une entreprise inscrite de tenir la liste des titres qu'un client a souhaité louer. Elle peut ainsi suivre la disponibilité de ces titres pour ses clients et interroger à tout moment la liste des souhaits d'un client. En outre, elle peut utiliser ces listes pour identifier les vidéos les plus demandées et adapter son stock en conséquence.

Recherche de films : Ce service Web permet à une entreprise inscrite de rechercher des titres et des résumés de films. Au lieu d'entrer manuellement toutes ces données dans une base de données locale, elle peut effectuer des recherches en ligne dans un référentiel d'informations cinématographiques constamment mis à jour. Ce service Web peut être utilisé par le personnel ou les clients du vidéoclub et facilite le choix du film à louer.

Architecture de l'application Video Central

L'application Video Central s'appuie sur une architecture de services Web. Sa conception sera décrite plus en détail au "Chapitre 5. Étude de la conception de l'application Video Central". En tant qu'élément de cette architecture, Video Central est essentiellement un fournisseur dynamique et souple de services e-business (via Internet).

Des demandeurs de services peuvent être développés pour tirer parti des services fournis par Video Central. Lorsqu'un demandeur de services a besoin d'un service de Video Central, il envoie sa requête à l'objet proxy de service

de la machine client. Le proxy place la demande et les données associées dans un format XML et envoie ce dernier dans un message codé SOAP. Le servlet d'exécution SOAP est fourni avec WAS version 4.x. Lorsqu'il reçoit la demande, ce servlet la délègue au bean Java approprié, qui met en œuvre le service qui permet de la traiter.

L'application Video Central est mise en œuvre sous forme d'application à trois couches :

1. couche WIL (Web Interface Layer, couche de l'interface Web)
2. couche BLL (Business Logic Layer, couche de la logique applicative)
3. couche DAL (Data Access Layer, couche d'accès aux données)

Cette conception à trois couches permet une mise en œuvre souple, à base de composants, de Video Central, qui facilitera les améliorations ultérieures de l'application grâce à son cadre extensible.

La couche WIL (interface Web) peut être mise en œuvre de manière à prendre en charge différentes interfaces Web, telles que les services Web, HTTP Servlet, JavaServer Pages (JSP), etc. Cela permet une souplesse maximum dans la conception de l'interface utilisateur final. Pour mettre en œuvre les services Web, Video Central utilise le module d'exécution SOAP pour l'appel de service Web.

La couche BLL (logique applicative) est conçue en utilisant les services Web comme couche d'abstraction. Chaque service Web définit un ensemble de logique applicative et l'interface est décrite à l'aide de WSDL (Web Services Description Language). L'interface WSDL, quant à elle, peut être publiée dans un registre UDDI (Universal Description, Discovery, and Integration) public ou privé. Un composant clé de l'application Video Central implique l'utilisation de l'extension XML de DB2 pour gérer les enregistrements et les extractions de données dans le référentiel des infractions des clients. En outre, Video Central s'appuie sur le moteur de recherche de texte rapide de l'extension DB2 Net Search pour offrir son service Web de recherche de films.

La couche DAL (accès aux données) a été intégrée aux composants d'accès aux données (Data Access) en utilisant des beans IBM Data Access, qui encapsulent JDBC (Java Database Connectivity). Les JavaBeans qui mettent en œuvre la logique applicative utilisent les composants Data Access pour enregistrer et extraire les données. L'application Video Central utilise DB2 UDB comme serveur de base de données.

La Figure 2 illustre l'architecture de l'application Video Central.

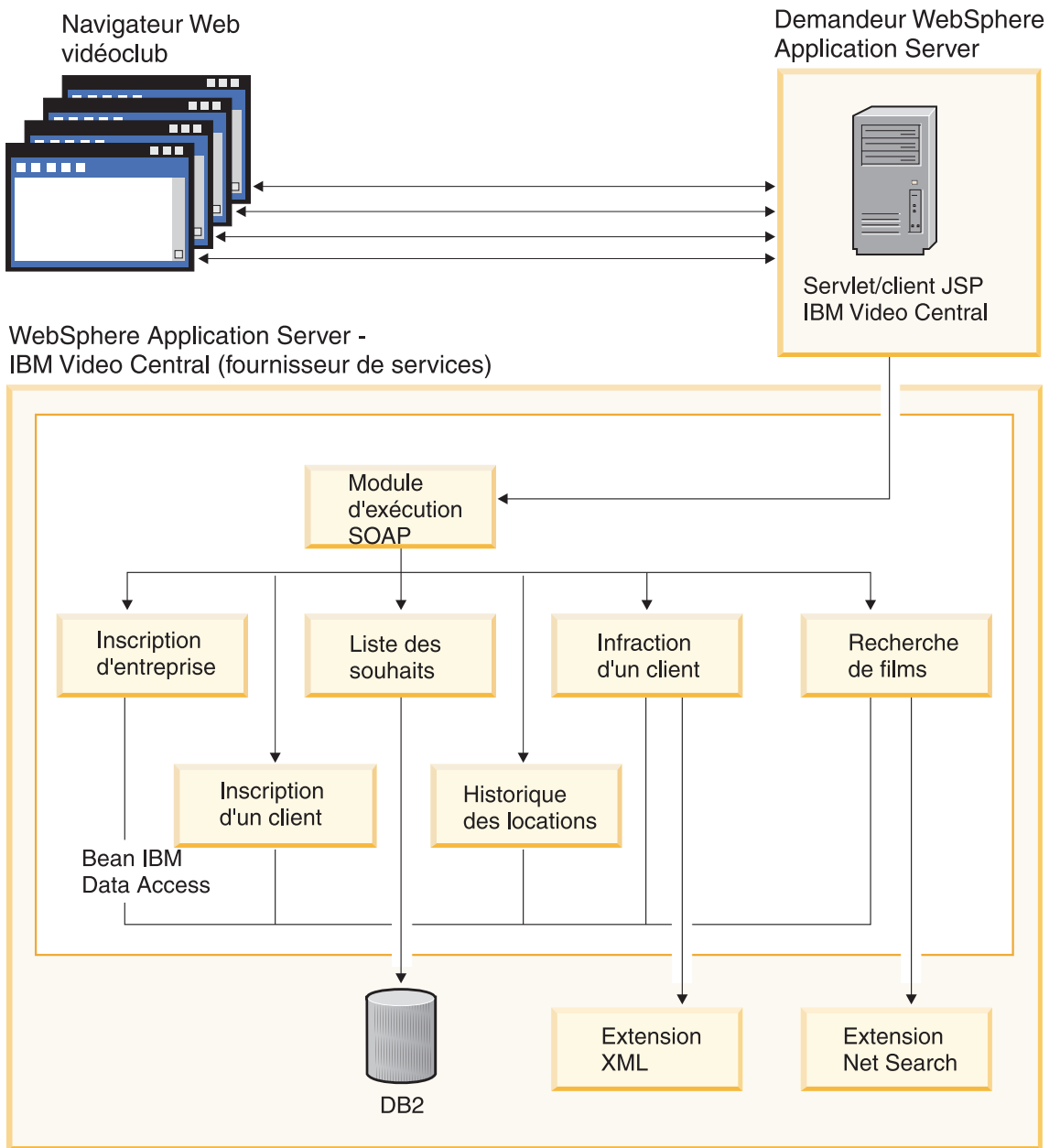


Figure 2. Architecture de l'application Video Central.

Résumé de la leçon

Dans ce chapitre, vous avez appris :

- ce que sont les services Web,
- les grandes lignes de la conception et de l'architecture de l'application Video Central.

Étape suivante

Maintenant, vous allez étudier l'environnement de développement DB2 requis et le préparer en vue du déploiement de l'application Video Central.

Chapitre 3. Préparation de l'environnement DB2 en vue de l'utilisation de Video Central

Durée

L'étude de ce chapitre nécessite environ 15 minutes.

Avant de commencer

Logiciels requis

- Microsoft Windows NT version 4.0 avec Service Pack 6a ou supérieur, ou Windows 2000
 - Java 2 SDK Standard Edition version 1.3
 - DB2 Universal Developer's Edition (DB2 UDE) version 8.1
 - DB2 Extension XML (cette extension est fournie avec DB2 UDE version 8.1, mais elle figure sur un CD distinct et doit être installée séparément)
 - DB2 Extension Net Search (cette extension est fournie avec DB2 UDE version v8.1, mais elle figure sur un CD distinct et doit être installée séparément)
-

Présentation

Dans ce chapitre, vous allez configurer l'environnement DB2 en vue de l'utilisation de Video Central. Cette opération implique les tâches suivantes :

- Préparation de la base de données de Video Central en vue de l'utilisation de DB2 Extension XML et de DB2 Extension Net Search.

Vous allez également :

- créer un utilisateur `ivc`,
 - créer les tables requises pour Video Central.
-

Tâches relatives à Video Central

Tâche 1 : Création de l'ID utilisateur "ivc"

Si vous ne l'avez pas déjà fait, créez un ID utilisateur `ivc` dans votre système d'exploitation.

Avec Windows 2000 :

1. Cliquez sur **Démarrer** —> **Paramètres** —> **Panneau de configuration**.

2. Sélectionnez **Outils d'administration** → **Gestion de l'ordinateur** → **Utilisateurs et groupes locaux**.
3. Dans la barre de menus, cliquez avec le bouton droit de la souris sur **Utilisateurs** → **Nouvel utilisateur**.
4. Créez un utilisateur nommé `ivc` avec le mot de passe `VideoCentral`. Désélectionnez la case **L'utilisateur doit changer de mot de passe à la prochaine ouverture de session**.

Avec Windows NT :

1. Cliquez sur **Démarrer** → **Outils d'administration (Commun)** → **Gestionnaire des utilisateurs**.
2. Dans la barre de menus, cliquez avec le bouton droit de la souris sur **Utilisateur** → **Nouvel utilisateur**.
3. Créez un utilisateur nommé `ivc` avec le mot de passe `VideoCentral`. Désélectionnez la case **L'utilisateur doit changer le mot de passe à la prochaine ouverture de session**.

Tâche 2 : Création de l'exemple de base de données

Dans ce tutoriel, nous utiliserons une base de données appelée `sample` comme base de données de Video Central. Pour créer cette base de données, effectuez l'une des opérations suivantes :

- Entrez `db2sample` sur une ligne de commande.
- Cliquez sur **Démarrer** → **Programmes** → **IBM DB2** → **Premiers pas**, puis cliquez sur **Création des bases de données exemples** dans la fenêtre **Premiers pas**.

Tâche 3 : Préparation de l'exemple de base de données en vue de l'utilisation de DB2 Extension XML

Avant de pouvoir enregistrer et extraire des documents XML dans DB2 avec Extension XML, vous devez préparer la base de données en vue de l'utilisation de XML.

Lorsque vous préparez une base de données pour XML, DB2 Extension XML :

- crée les types UDT (User-Defined Types, types définis par l'utilisateur) et les fonctions UDF (User-Defined Functions, fonctions définies par l'utilisateur),
- crée les tables de contrôle et les remplit avec les métadonnées nécessaires à Extension XML,
- crée le schéma `db2xml` et attribue les droits requis.

Pour préparer la base de données `sample` en vue de l'utilisation de DB2 Extension XML, procédez comme suit :

1. Dans une fenêtre de commande DB2, entrez
<rép_install_DB2>\tutorials\VideoCentral\db
\dbEnableXML.bat .

Ce script :

- a. établit une connexion à la base de données sample,
- b. lie les fichiers de liens Extension XML à la base de données sample,
- c. active Extension XML.

Remarque : Le script <rép_install_DB2>\tutorials\VideoCentral\db
\dbDisableXML.bat est fourni pour le cas où vous voudriez
que votre base de données ne prenne plus en charge XML.
Vous n'avez pas à l'utiliser dans le cadre de ce tutoriel.

2. Vérifiez que DB2 Extension XML est activé, en entrant les commandes
suivantes dans une fenêtre de commande DB2 :
 - a. db2 connect to sample.
 - b. db2 select procname from syscat.procedures where
procschema='DB2XML'.

La liste de procédures mémorisées suivante doit s'afficher :

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

L'application Video Central utilise deux de ces procédures mémorisées,
DXXGENXML et DXXSHREXML, dans le service Web Infractions d'un client.
DXXGENXML est utilisée pour interroger l'historique des infractions d'un
client et extraire celles-ci de DB2 sous forme de documents XML.
DXXSHREXML est utilisée pour ajouter une infraction d'un client, qui est
reçue dans DB2 sous forme de document XML.

Pour plus d'informations concernant Extension XML, consultez le manuel
XML Extender Administration and Programming, disponible sur le site Web
<http://www.ibm.com/software/data/db2/extenders/xmlext/library.html> .

Tâche 4 : Préparation de l'exemple de base de données en vue de l'utilisation de DB2 Extension Net Search

1. Dans une fenêtre de commande DB2, entrez
<rép_install_DB2>\tutorials\VideoCentral\db\dbEnableNX.bat .
Ce script :
 - a. démarre DB2,
 - b. établit une connexion à la base de données sample,
 - c. démarre le démon DB2 Extension Net Search,
 - d. prépare la base de données en vue de l'utilisation de Extension Net Search,
 - e. active les colonnes de texte (pour Video Central, Net Search est indexée sur les colonnes ivc.titleinfo.name et ivc.titleplot.summary).

Remarque : Le script <rép_install_DB2>\tutorials\VideoCentral\db\dbDisableNX.bat est fourni pour le cas où vous voudriez que votre base de données ne prenne plus en charge DB2 Extension Net Search. Vous n'avez pas à l'utiliser dans le cadre de ce tutoriel.

2. Vérifiez que DB2 Extension Net Search est activé, en ouvrant le fichier <rép_install_DB2>\tutorials\VideoCentral\db\movieSearch.log et en y recherchant d'éventuelles erreurs.

Remarque : Si ce fichier contient une ou plusieurs erreurs, reportez-vous au "Chapitre 8. Résolution des incidents courants".

Pour plus d'informations concernant Extension Net Search, consultez le manuel *Net Search Extender Administration and Programming*, disponible sur le site Web <http://www.ibm.com/software/data/db2/extenders/netsearch>, ainsi que l'"Annexe A — Informations de référence".

Tâche 5 : Création et remplissage des tables de Video Central

1. Ouvrez une fenêtre de commande DB2 pour pouvoir exécuter le fichier de commandes.
2. Dans la fenêtre de commande DB2, entrez
<rép_install_DB2>\tutorials\VideoCentral\db\tblCreate.bat .

Ce script :

- a. crée les tables de Video Central dans la base de données sample,
- b. crée les index et les contraintes requis,
- c. remplit les tables requises (tables de codes, table DAD, tables de recherche de films).

Remarque : Le script <rép_install_DB2>\tutorials\VideoCentral\db\tblClean.bat est fourni pour le cas où vous ne voudriez plus utiliser l'application Video Central et souhaiteriez supprimer les tables correspondantes. Vous n'avez pas à l'utiliser dans le cadre de ce tutoriel.

3. Dans une fenêtre de commande DB2, entrez les commandes suivantes :
 - a. DB2 connect to sample.
 - b. DB2 list tables for schema ivc.

La liste de dix tables suivante doit s'afficher :

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

Résumé de la leçon

Dans ce chapitre, vous avez :

- créé un utilisateur ivc,
- créé la base de données sample,
- préparé la base de données de Video Central en vue de l'utilisation de DB2 Extension XML et de DB2 Extension Net Search,
- créé et rempli les tables requises pour Video Central.

Étape suivante

Maintenant, vous allez étudier l'environnement de développement requis et le préparer en vue de l'exécution de l'application Video Central.

Chapitre 4. Initiation à WebSphere Studio Application Developer

Durée

L'étude de ce chapitre nécessite environ 20 minutes.

Présentation

Ce chapitre contient une brève introduction à IBM WebSphere Studio Application Developer. Vous y apprendrez comment configurer l'environnement de développement Java pour les services Web.

Logiciels requis

- WebSphere Application Server (WAS) version 4.0, Advanced Edition Single Server
- WebSphere Studio Application Developer (WSAD) version 4.0

Tâches relatives à Video Central

Tâche 1 : Définition de l'instance de serveur éloigné WAS

Cette tâche consiste à créer un projet de serveur Video Central et une définition de serveur éloigné WAS dans WebSphere Studio Application Developer.

Démarrage de WebSphere Studio Application Developer

Démarrez WebSphere Studio Application Developer en cliquant sur **Démarrer** → **Programmes** → **IBM WebSphere Studio Application Developer** → **IBM WebSphere Studio Application Developer**.

Création et configuration du serveur Video Central

1. Pour créer le serveur Video Central, procédez comme suit :
 - a. Dans la perspective Server, créez un projet Serveur en cliquant sur **Fichier** → **New** → **Server Project**.
 - b. Donnez au projet le nom VideoCentralServer. Cliquez sur **Finish**.
2. Pour créer et configurer une instance du serveur, procédez comme suit :
 - a. Dans la perspective Server, cliquez avec le bouton droit de la souris sur **Server Configurations** dans la sous-fenêtre Server Configuration, puis sélectionnez **New** → **Server Instance and Configuration**.
 - b. Donnez au serveur le nom VideoCentralServer et, dans la zone **Server instance type**, sélectionnez **WebSphere Servers** → **WebSphere v4.0 Remote Server**. La fenêtre doit maintenant se présenter comme suit (voir Figure 3) :

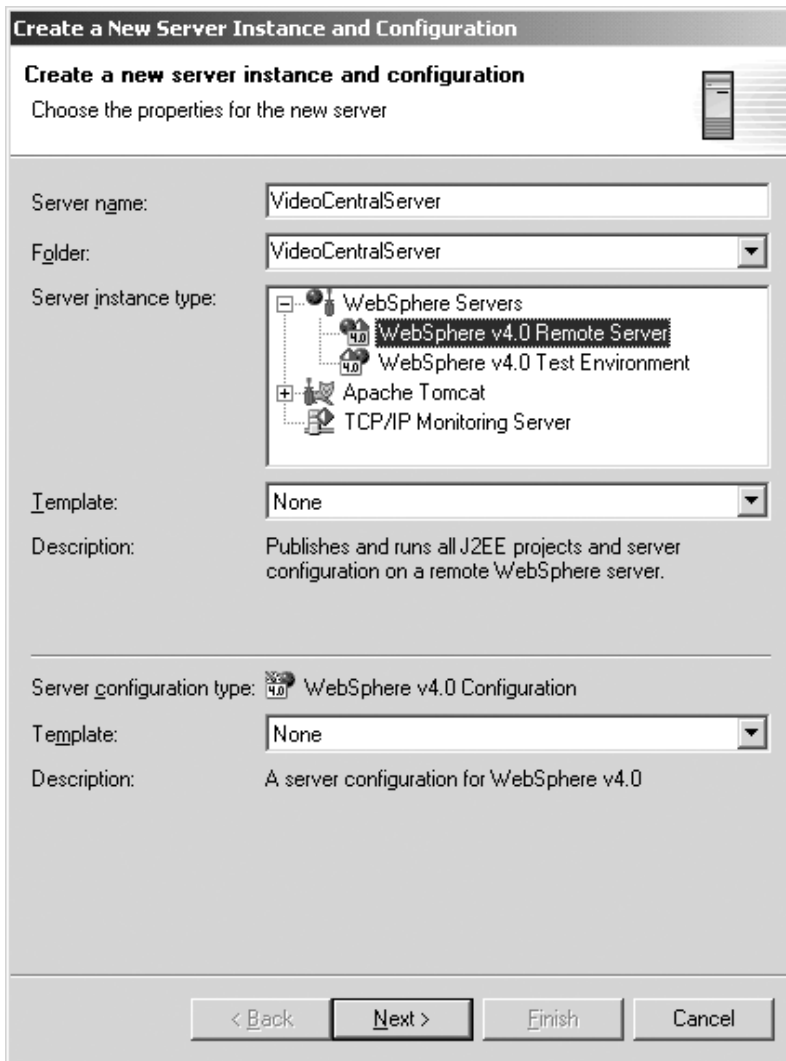


Figure 3. Création et configuration d'une instance de serveur.

- c. Cliquez sur **Suivant**.
- d. Dans la zone **Host Address**, remplacez 127.0.0.1 par localhost.
- e. Dans la zone **WebSphere installation directory**, indiquez l'emplacement où vous avez installé WebSphere Application Server 4.0 Advanced Edition Single Server (c'est-à-dire <rep_install_WAS>). Cliquez sur **Suivant**.
- f. Sélectionnez **Create a new remote file transfer instance** → **Copy file transfer mechanism**. Cliquez sur **Suivant**.

- g. Dans la zone **Remote target directory**, indiquez l'emplacement où vous avez installé WebSphere Application Server 4.0 Advanced Edition Single Server (c'est-à-dire <rep_install_WAS>). Cliquez sur **Suivant**.
- h. Remplacez le numéro de port HTTP 8080 par 9080. Cliquez sur **Finish**. La fenêtre Navigator doit maintenant se présenter comme suit (voir Figure 4) :

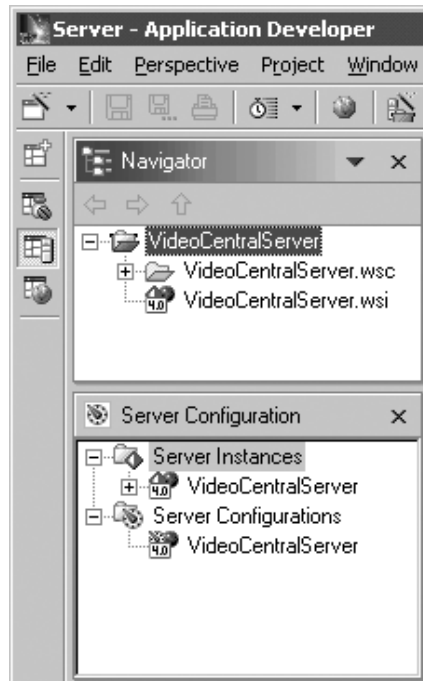


Figure 4. L'instance de serveur a été créée.

3. Sur l'écran Server Configuration, sous le dossier **Server Configurations**, cliquez avec le bouton droit de la souris sur **VideoCentralServer** → **Open**.
 - a. Sur la page General, sélectionnez **Enable Administration Client**.
 - b. Sur la page EJB, désélectionnez **Enable EJB test client**.
 - c. Cliquez sur **Fichier** → **Save VideoCentralServer**.
 - d. Fermez la fenêtre **VideoCentralServer**.
4. Configurez la source de données, de la manière suivante :
 - a. Dans la même fenêtre, cliquez sur l'onglet Source de données et, dans la liste des pilotes JDBC, sélectionnez **Db2JdbcDriver**.
 - b. Ajoutez la base de données sample, de la manière suivante :

- 1) Cliquez sur le bouton **Add** situé en regard de la case **Data source defined in the JDBC driver selected above**.
- 2) Dans la zone **Name**, indiquez `sample`.
- 3) Dans la zone **JNDI name**, indiquez `jdbc/Sample`. (Remarque : cette zone faisant la distinction entre les majuscules et les minuscules, n'oubliez pas d'indiquer en majuscule l'initiale du nom `Sample`.)
- 4) Dans la zone **Database name**, indiquez `sample`.
- 5) Dans la zone **Default User ID**, indiquez `ivc`.
- 6) Dans la zone **Default User password**, indiquez `VideoCentral`.
- 7) Cliquez sur **OK**.

Remarque : L'application Video Central se connecte à DB2 en utilisant l'ID utilisateur et le mot de passe enregistrés dans le fichier `VideoCentral.properties`.

La fenêtre doit maintenant se présenter comme suit (voir Figure 5) :

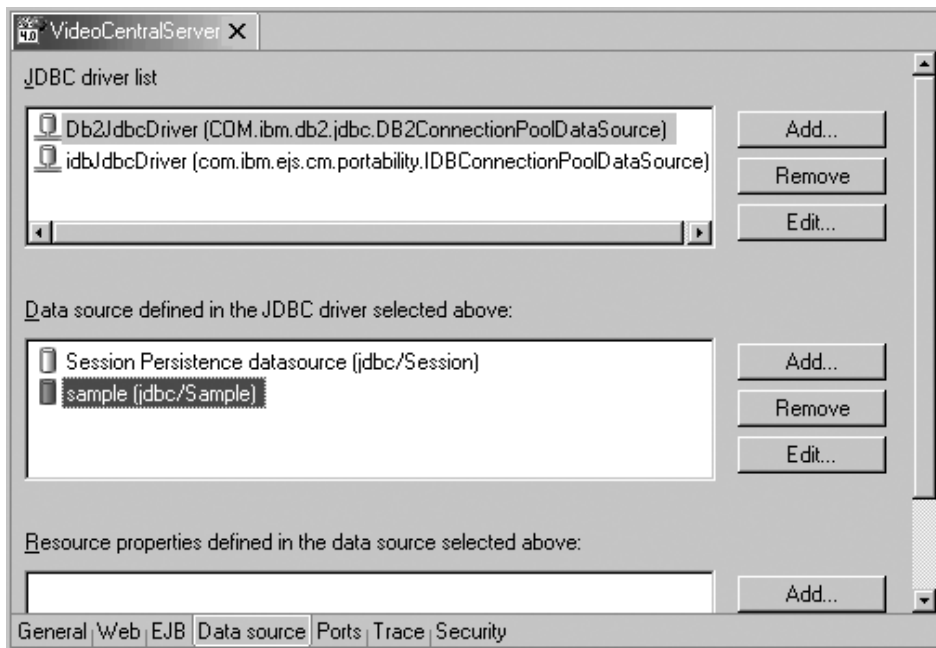


Figure 5. Configuration de la source de données.

- c. Fermez la fenêtre `VideoCentralServer` en cliquant sur **X** et en sélectionnant **Yes** lorsque le système vous demande si vous voulez enregistrer vos modifications.

5. Sur l'écran Server Configuration, sous le dossier **Server Instances**, cliquez avec le bouton droit de la souris sur **VideoCentralServer** → **Open**.
 - a. Sélectionnez la page Paths, puis cliquez sur **Add External JARs** et ajoutez les fichiers suivants :
 - <rép_install_DB2>\java\db2java.zip
 - <rép_install_DB2>\tools\databasean.jar
 - <rép_install_WAS>\lib\j2ee.jar
 - Remarque :** Le fichier db2java.zip est un ensemble de classes qui fournit la prise en charge JDBC 2.1 pour l'accès aux bases de données DB2. Ces classes communiqueront avec le serveur DB2 en utilisant une architecture de pilote JDBC de type 2.
 - b. Fermez la fenêtre VideoCentralServer en cliquant sur **X** et en sélectionnant **Yes** lorsque le système vous demande si vous voulez enregistrer vos modifications.

Tâche 2 : Définition du projet Web Video Central et importation du code source

Création du projet Web VideoCentral

1. Pour créer le projet Web VideoCentral, procédez comme suit :
 - a. Dans la perspective Web, cliquez sur **Fichier** → **New** → **Web project** pour créer un projet Web. Nommez le projet VideoCentral et nommez l'Enterprise Application Project VideoCentralEAR. Cliquez sur **Suivant**.
 - b. L'écran **Module Dependencies** s'affiche. Conservez les valeurs par défaut. Cliquez sur **Suivant**.
 - c. L'écran **Define Java Build Settings** doit maintenant s'afficher. Sélectionnez la page Libraries, cliquez sur **Add External JARs** et ajoutez les fichiers jars externes suivants dans la fenêtre en incrustation **JAR Selection** :
 - <rép_install_WSAD>\plugins\com.ibm.etools.webservices\runtime\soap.jar
 - <rép_install_DB2>\tools\databasean.jar

Remarque : Video Central utilise le fichier soap.jar pour effectuer l'analyse syntaxique des messages SOAP utilisés par les services Web. Le fichier databasean.jar est un ensemble de classes fournies par IBM pour faciliter l'accès à DB2 à partir de Java.

La fenêtre doit maintenant se présenter comme suit (voir Figure 6) :

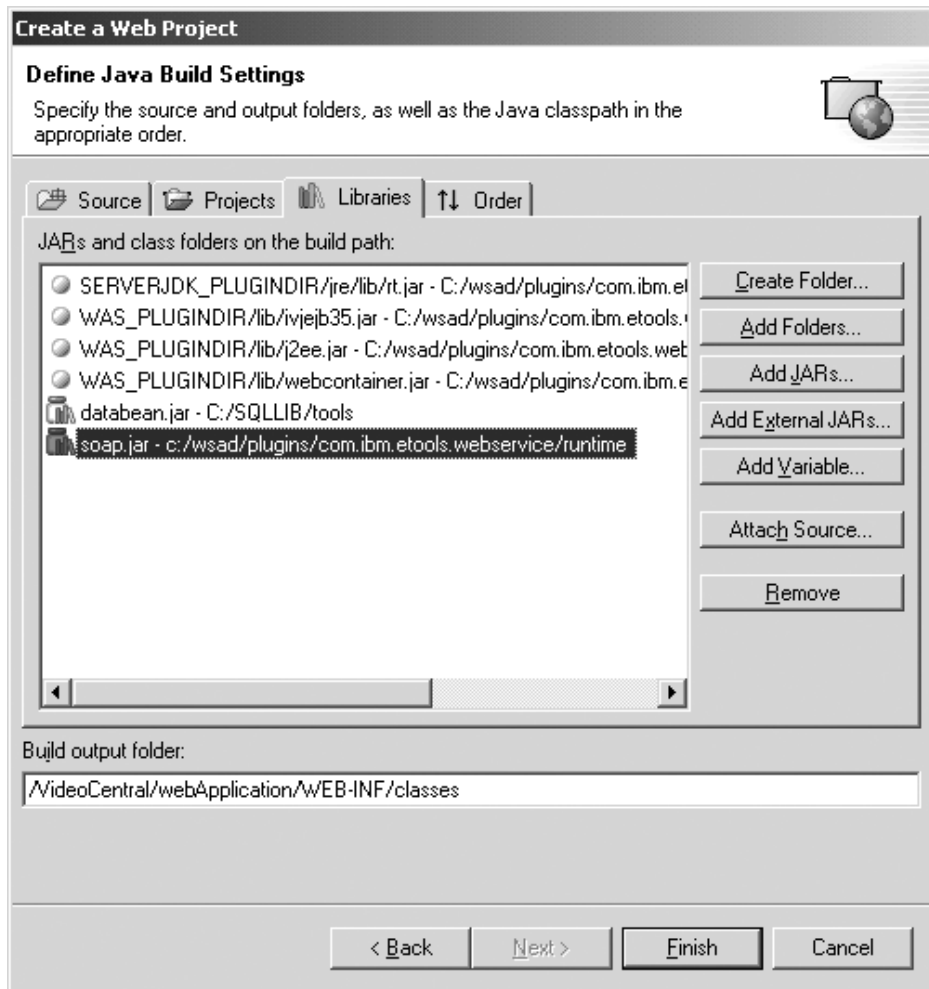


Figure 6. Ajout de fichiers JAR externes.

d. Cliquez sur **Finish**.

2. Ajoutez le projet Web VideoCentral au serveur, de la manière suivante :
 - a. Sur l'écran Server Configuration, développez **Server Configurations** et cliquez avec le bouton droit de la souris sur **VideoCentralServer**. Sélectionnez **Add Project** → **VideoCentralEAR**.
L'écran Server Configuration doit maintenant se présenter comme suit (voir Figure 7) :

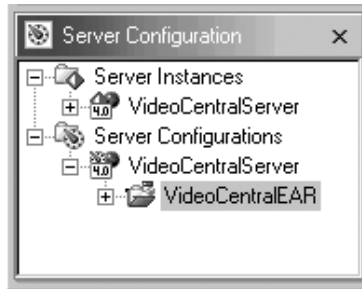


Figure 7. Ajout du projet Web au serveur.

- b. Sur l'écran Navigator, cliquez avec le bouton droit de la souris sur le projet Web VideoCentral et sélectionnez **Propriétés** → **Server Preferences**. Sélectionnez **VideoCentralServer** dans la zone, puis cliquez sur **Apply** et sur **OK** (voir Figure 8).

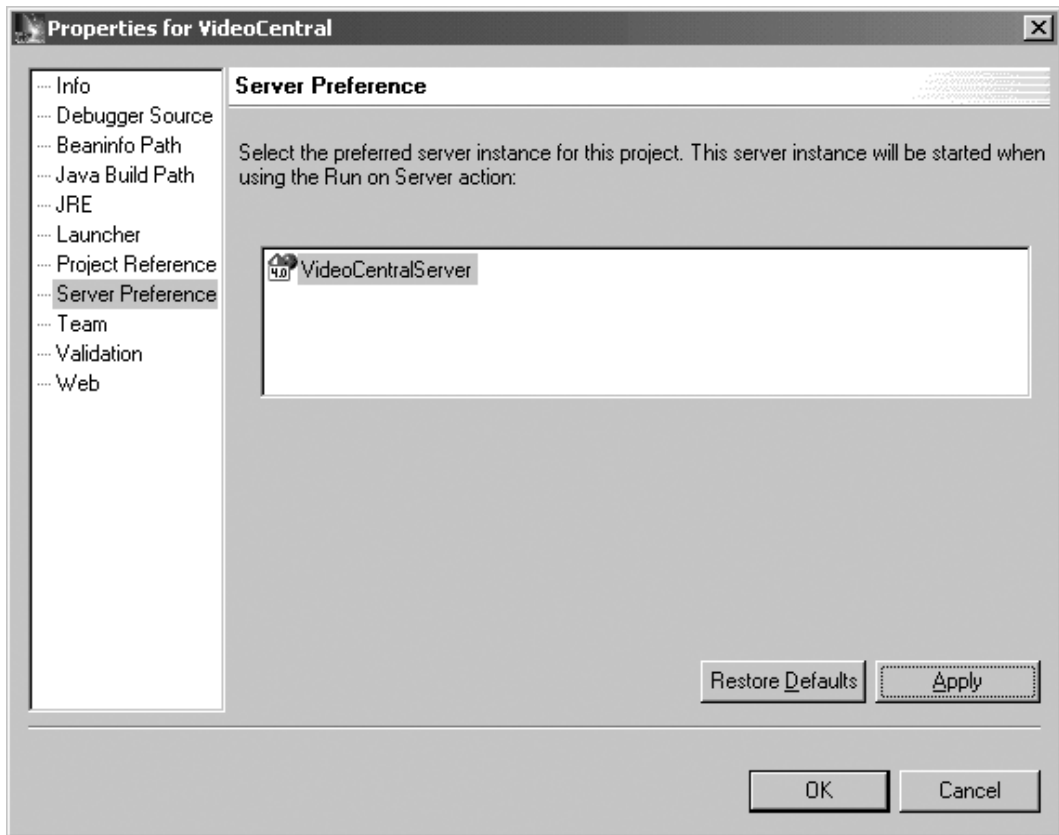


Figure 8. Ajout du projet Web au serveur (suite).

Importation de VideoCentral.jar dans WebSphere Studio Application Developer

1. Développez le projet Web **VideoCentral** que vous venez de créer et sélectionnez **source** (voir Figure 9).

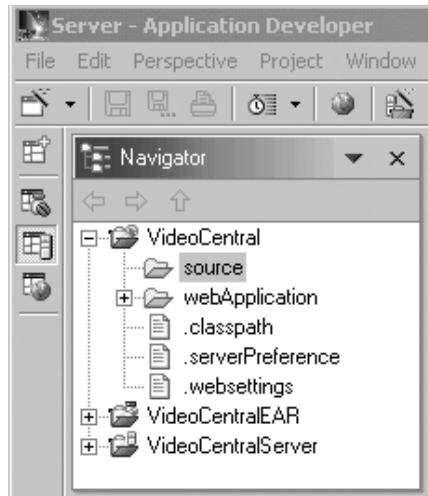


Figure 9. Avant d'importer les fichiers JAR.

2. Sélectionnez **Fichier** → **Import**.
3. Comme source d'importation, sélectionnez **Fichier ZIP**. Cliquez sur **Suivant**.
4. Parcourez l'arborescence jusqu'au fichier
<rép_install_DB2>\tutorials\VideoCentral\lib\VideoCentral.jar
5. Cliquez sur le bouton **Select All** (voir Figure 10).

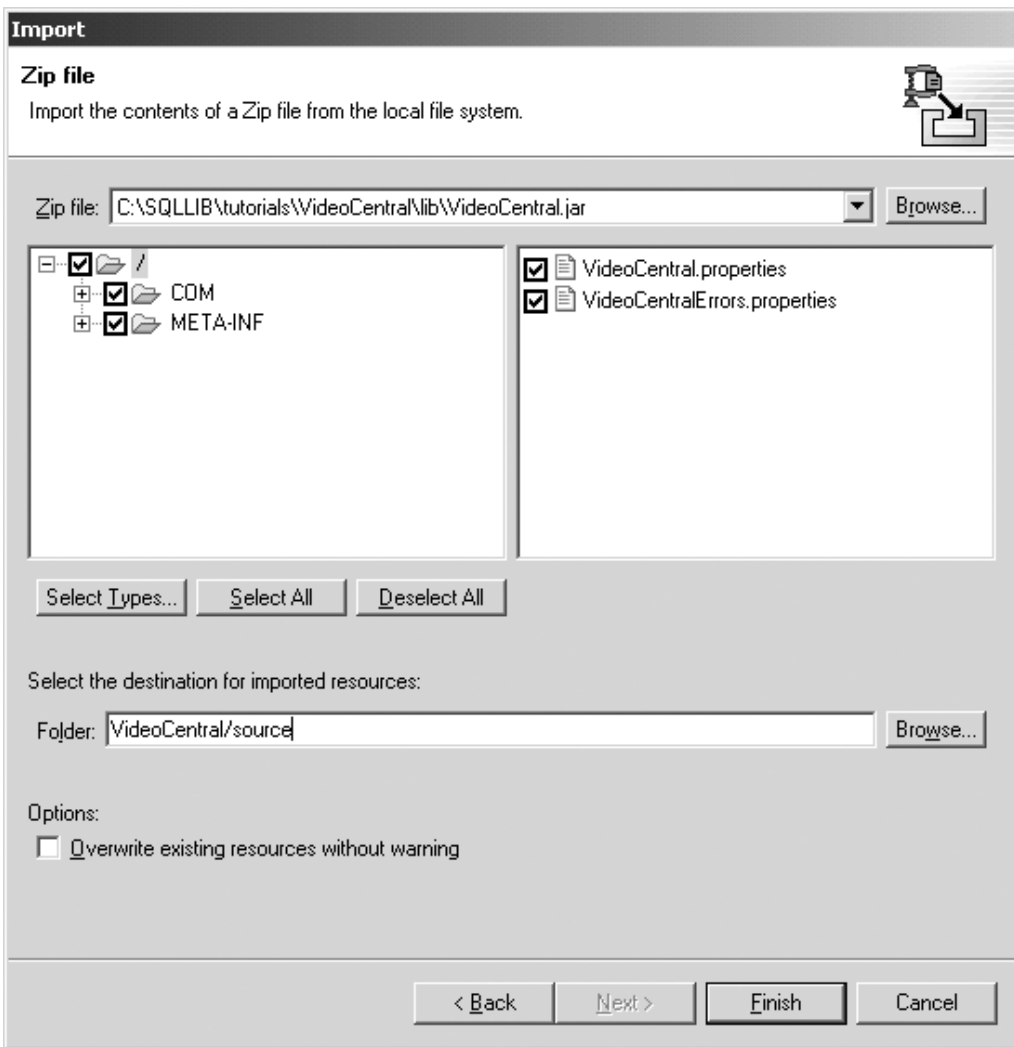


Figure 10. Sélection des ressources à importer.

6. Cliquez sur **Finish**.

Les fichiers sont alors importés dans WebSphere Studio Application Developer et la barre d'actions indique qu'ils sont en cours de compilation.

7. Une fois les fichiers importés dans le projet Web VideoCentral, développez le dossier source. Vous devez obtenir une structure similaire à celle de la Figure 11 ci-après.

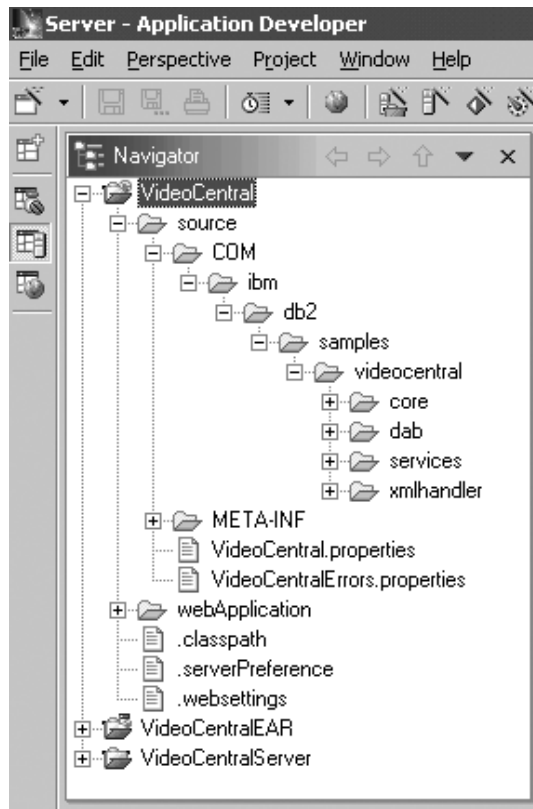


Figure 11. Après préparation du projet Web Video Central.

Résumé de la leçon

Dans ce chapitre, vous avez :

- appris à utiliser WebSphere Studio Application Developer,
- créé une référence de serveur WAS dans WebSphere Studio Application Developer afin de déployer et tester des applications,
- importé le code de Video Central dans WebSphere Studio Application Developer.

Étape suivante

Maintenant, vous allez étudier en détail la conception de l'application Video Central.

Chapitre 5. Conception de l'application Video Central

Durée

L'étude de ce chapitre nécessite environ 20 minutes.

Présentation

Ce chapitre décrit la conception et la mise en œuvre de l'application Video Central en termes des couches de composants logiciels suivantes :

1. couche WIL (Web Interface Layer, couche de l'interface Web)
 2. couche BLL (Business Logic Layer, couche de la logique applicative)
 3. couche DAL (Data Access Layer, couche d'accès aux données)
-

Conception de Video Central

Cette section décrit la conception de Video Central. Pour en faciliter la compréhension, elle comporte des schémas de classes et de séquences.

Couche WIL

La couche WIL (interface Web) expose les différents services Web fournis aux clients par Video Central. Les services Web peuvent être exposés à l'aide d'une couche d'abstraction de niveau supérieur. La conception de Video Central ne comporte pas cette couche d'abstraction pour ses services Web mais utilise le module d'exécution SOAP en lieu et place de cette dernière.

Développement de la couche BLL

Cette section présente les schémas de classes et de séquences des classes Java mises en œuvre dans Video Central. Les fichiers Java se trouvent dans les fichiers <rép_install_DB2>\tutorials\VideoCentral\lib\VideoCentral.jar et <rép_install_DB2>\tutorials\VideoCentral\lib\VideoClient.jar fournis avec ce tutoriel.

Mise en œuvre de l'application Video Central

Les sections qui suivent décrivent plus en détail la couche BLL (logique applicative) de l'application Video Central.

Schéma des classes de Video Central : La Figure 12 présente le schéma général des classes de l'application Video Central.

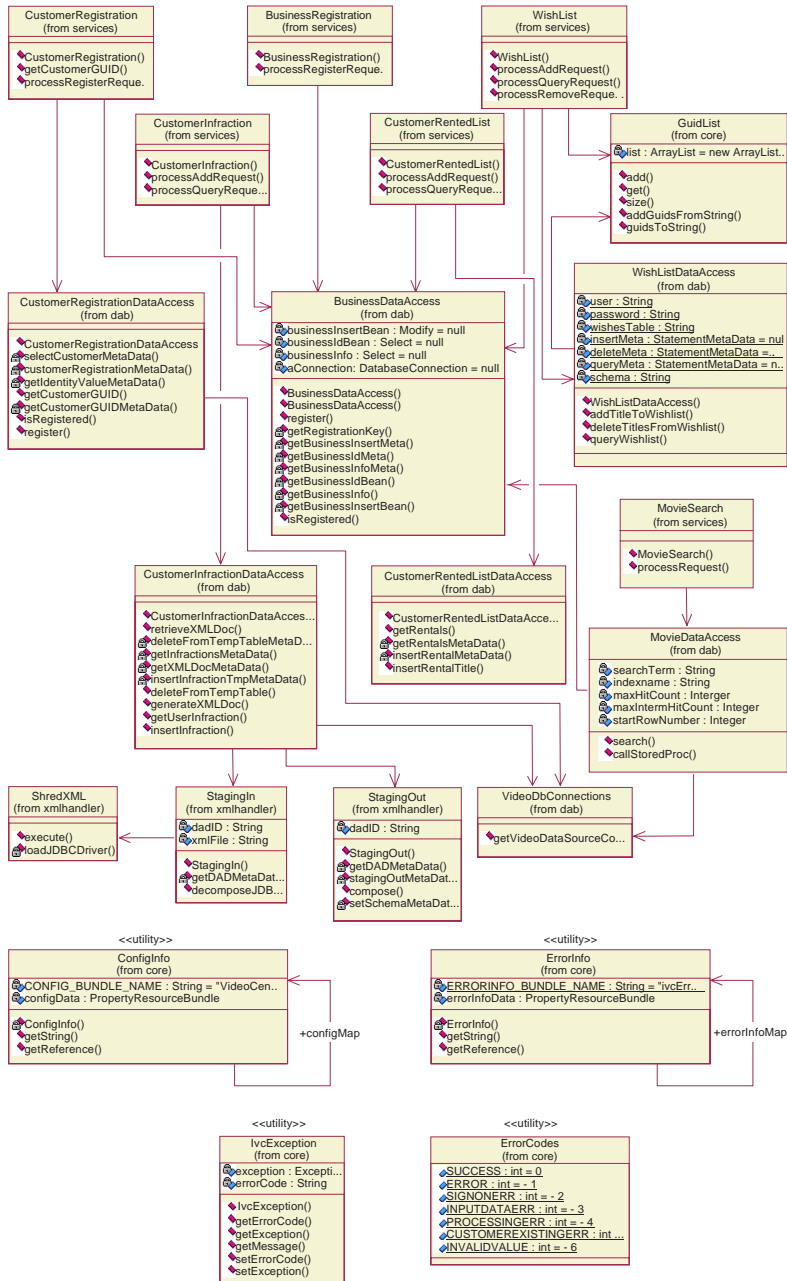


Figure 12. Schéma général des classes de l'application Video Central.

Schéma des classes du service Web Infractions d'un client : Chaque service Web est globalement mis en œuvre de la même manière. Cependant, le service Web Infractions d'un client présente la particularité unique de faire appel à DB2 Extension XML. Pour cette raison, il mérite un examen plus approfondi, qui permettra de comprendre comment il accède aux procédures mémorisées XML et les utilise. Le schéma des classes de la Figure 13 illustre sa mise en œuvre.

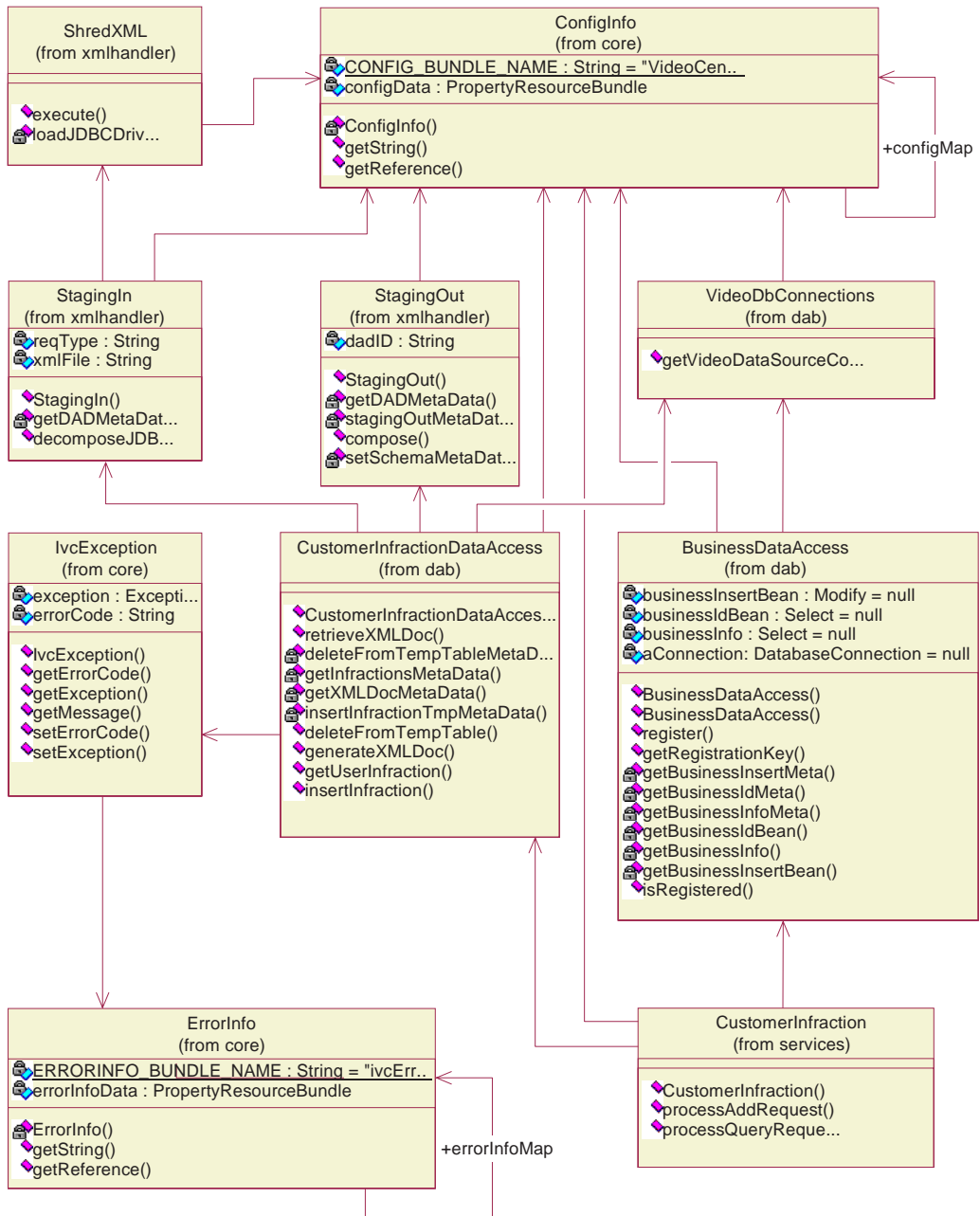


Figure 13. Schéma des classes du service Web Infractions d'un client.

Comme le montre la Figure 13, CustomerInfraction constitue la couche BLL (logique applicative) du service Web Infractions d'un client. CustomerInfractionDataAccess, StagingIn, ShredXML et StagingOut composent pour leur part la couche DAL (accès aux données), qui fait appel à Extension XML pour décomposer et composer les données XML. La fonction de VideoDbConnections est d'obtenir des connexions de base de données d'une source de données. BusinessDataAccess est utilisé par chaque service Web pour la validation des entreprises.

Schéma des classes du service Web Liste des souhaits d'un client : Le service Web Liste des souhaits d'un client est mis en œuvre à l'aide de JavaBeans Data Access (dans le package com.ibm.db.*). Les autres services sont mis en œuvre à l'aide de beans Data Access IBM (dans le package com.ibm.db.uibeans.*). Le fichier databean.jar fourni avec DB2 UDE contient les classes requises par la couche DAL (accès aux données) de Video Central. Le schéma des classes de la Figure 14 illustre la mise en œuvre du service Web Liste des souhaits d'un client.

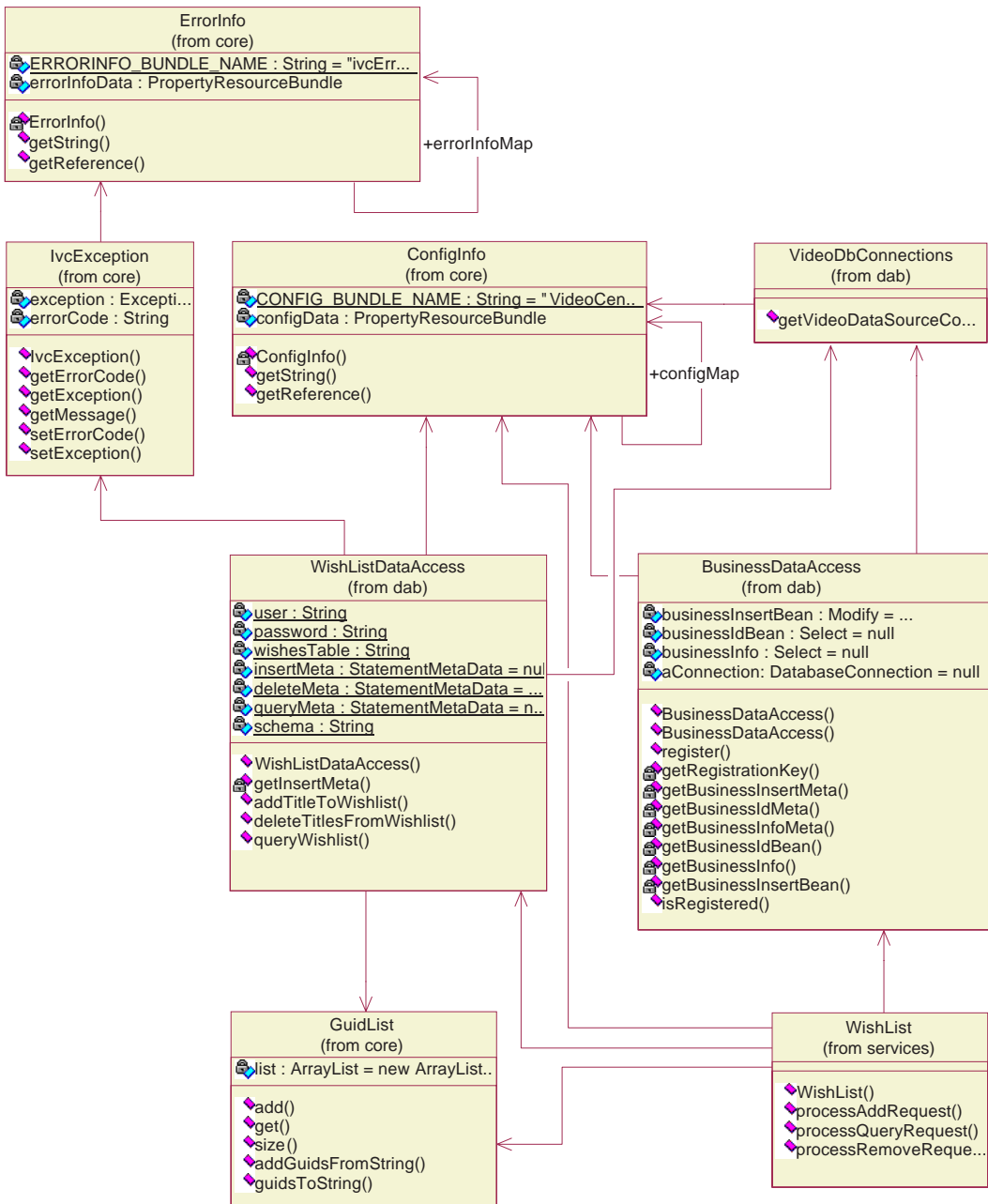


Figure 14. Schéma des classes du service Web Liste des souhaits d'un client.

Schéma de séquence du service Web Ajout d'une infraction d'un client :

Les six services Web étant globalement mis en œuvre de la même manière, il ne nous a pas paru nécessaire de fournir un schéma de séquence pour chacun d'eux. Seuls les schémas les plus intéressants figurent donc dans cette section. Celui de AddCustomerInfraction mérite une attention particulière car il fait appel à la procédure mémorisée de décomposition DXXSHREDXML d'Extension XML (voir Figure 15). Si vous examinez la méthode execute() de la classe ShredXML, vous verrez comment cette procédure mémorisée est utilisée.

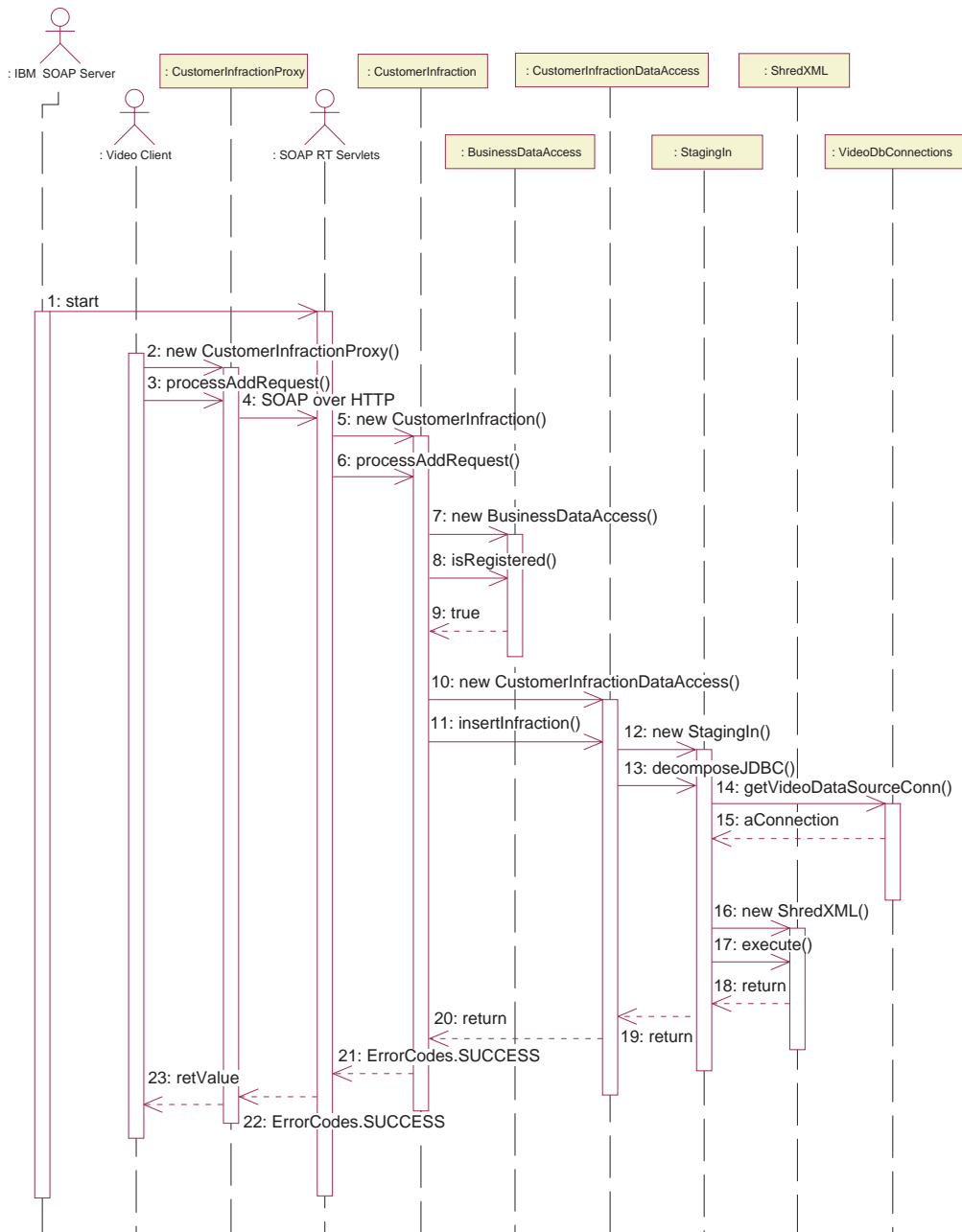


Figure 15. Schéma de séquence du service Web Ajout d'une infraction d'un client.

Schéma de séquence du service Web Interrogation des infractions d'un client : Le schéma de séquence de QueryCustomerInfraction mérite également une attention particulière car il fait appel à la procédure mémorisée de composition DXXGENXML d'Extension XML (voir Figure 16). Si vous examinez la méthode compose() de la classe StagingOut, vous verrez comment cette procédure mémorisée est utilisée.

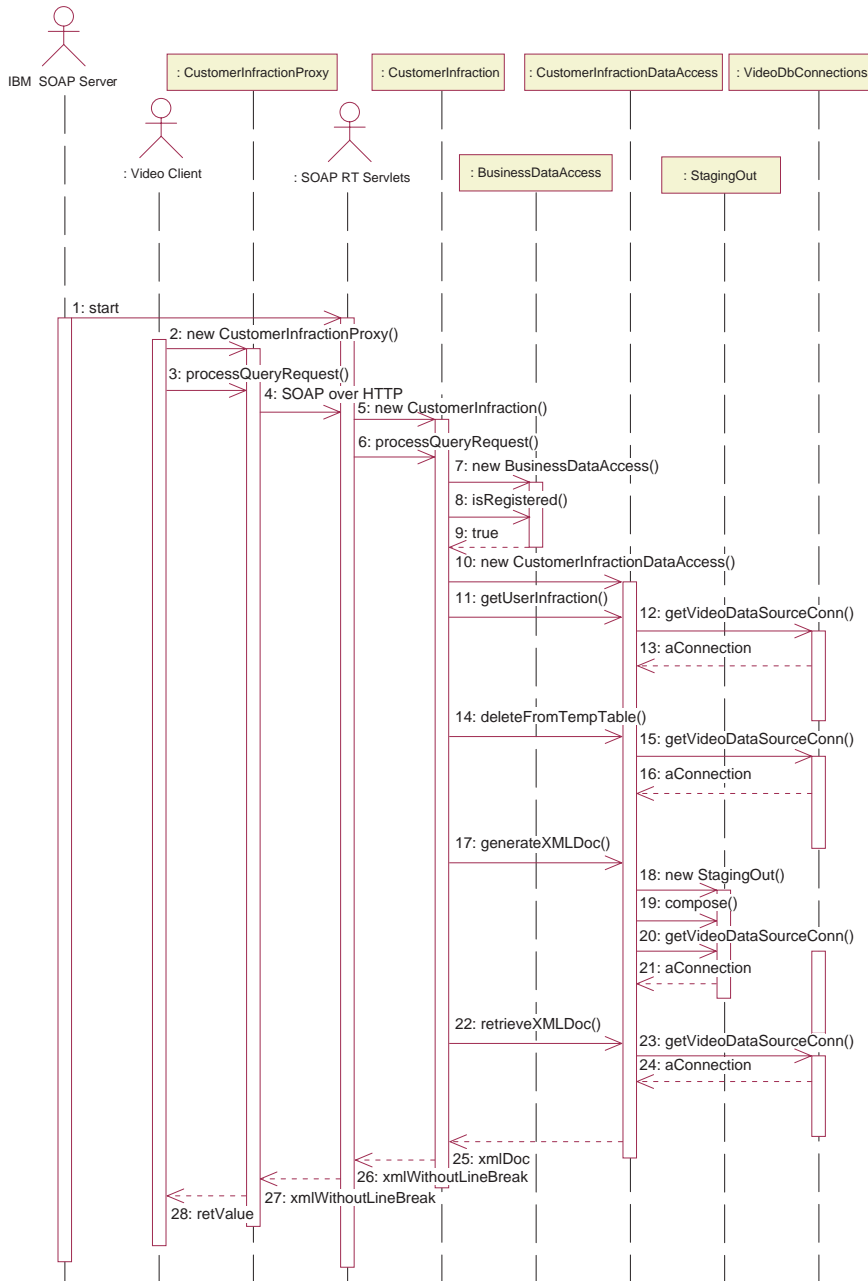


Figure 16. Schéma de séquence du service Web Interrogation des infractions d'un client.

Schéma de séquence du service Web Ajout d'un titre à la liste des souhaits d'un client : Le schéma de séquence du service Web Ajout d'un titre à la liste des souhaits d'un client est fourni pour illustrer l'enchaînement typique des opérations dans un service Web. Ce service a été mis en œuvre l'aide d'un niveau de classes d'accès aux données (Data Access) inférieur à celui des beans Data Access du package `com.ibm.db.*`, mais sa séquence est similaire. Elle est illustrée à la Figure 17. Les étapes 10 à 14 permettent d'insérer un titre dans la liste des souhaits d'un client à l'aide de la couche DAL (accès aux données).

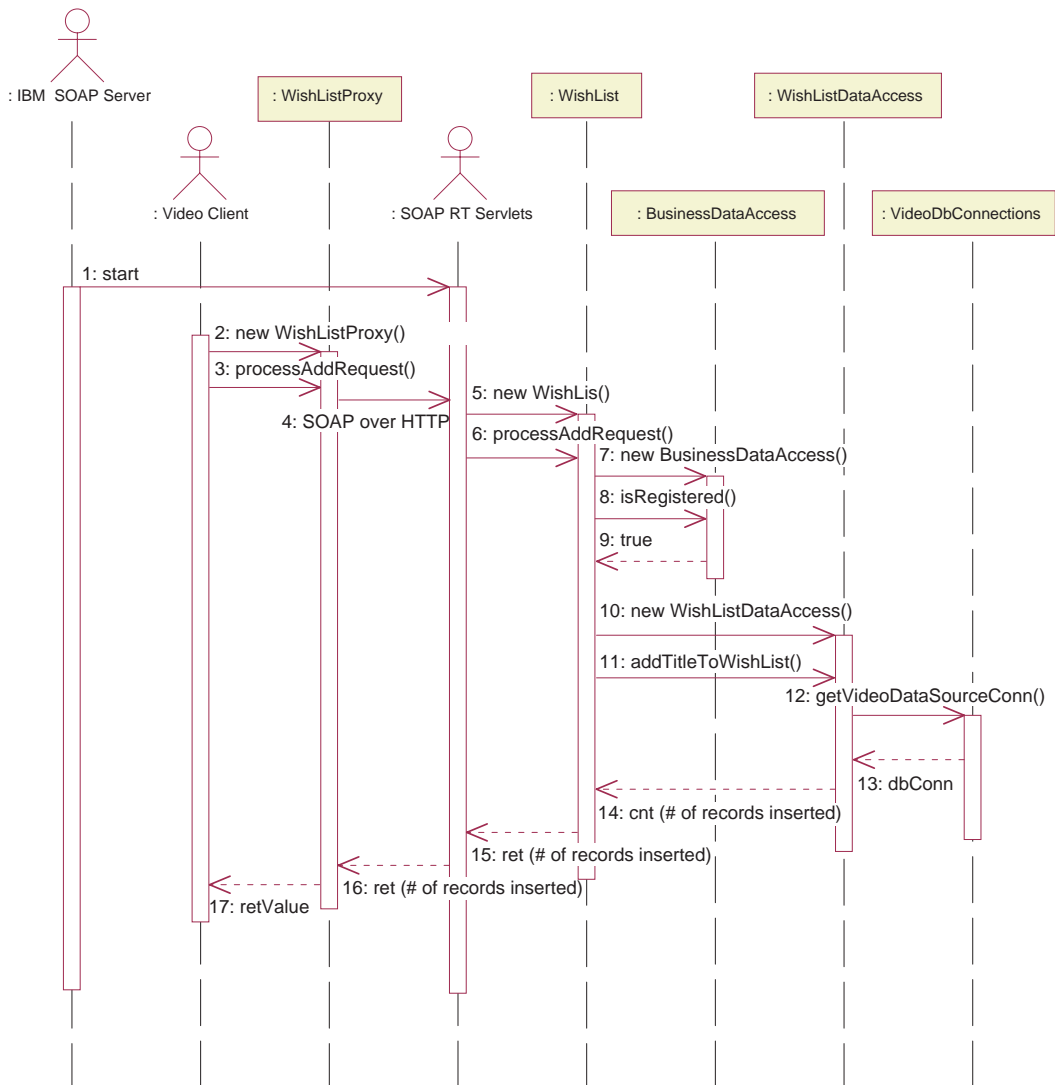


Figure 17. Schéma de séquence du service Web Ajout d'un titre à la liste des souhaits d'un client.

Modélisation des données

Cette section décrit le modèle de données des tables utilisées par Video Central. Pour plus de détails, reportez-vous au fichiers script du répertoire <rep_install_DB2>\tutorials\VideoCentral\db.

Présentation du modèle de données

Les Figure 18 et 19 présentent les tables utilisées par l'application Video Central.

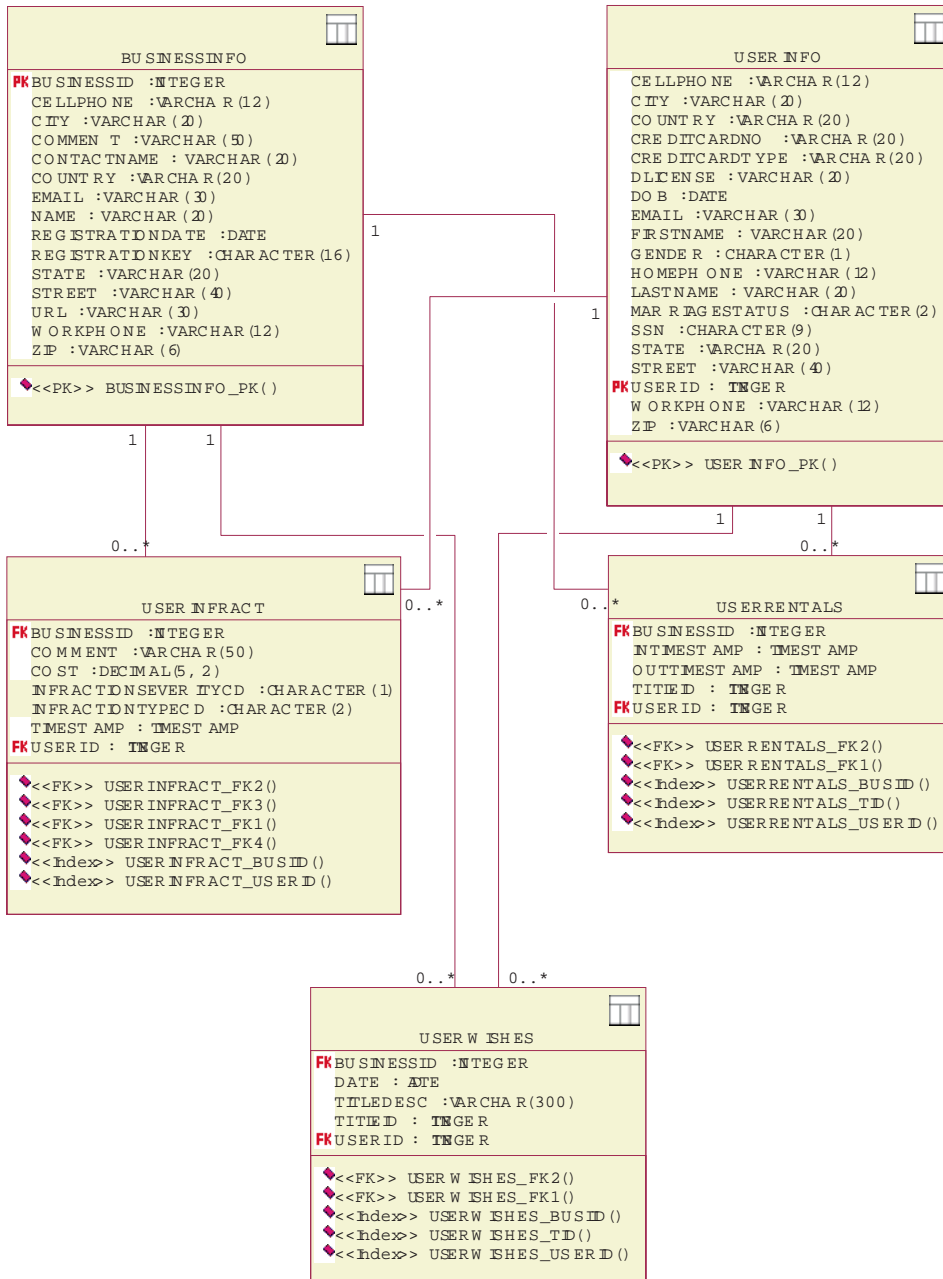


Figure 18. Schéma du modèle de données — première partie.

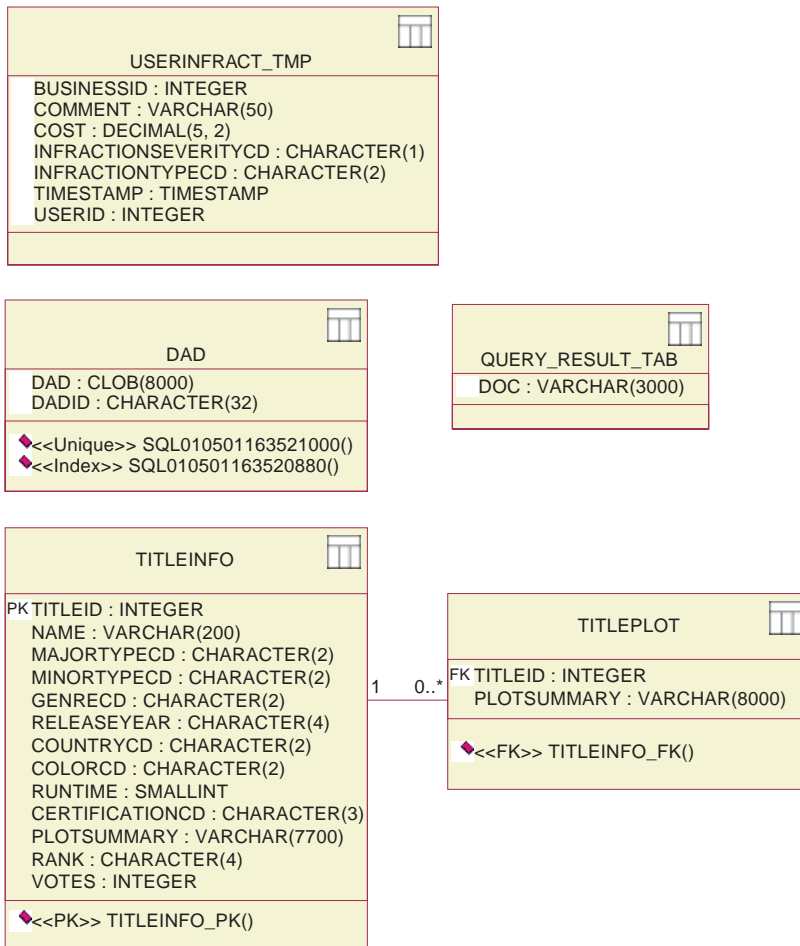


Figure 19. Schéma du modèle de données — seconde partie.

La table DAD contient les fichiers DAD (Document Access Definition, définition d'accès au document) utilisés pour la composition et la décomposition à l'aide d'Extension XML. Les tables USERINFRACT_TMP et QUERY_RESULT_TAB contiennent le rapport des infractions d'un client, généré en cas d'interrogation de ces infractions. Ces deux tables pourraient être supprimées si la fonction DxxGenXMLClob d'Extension XML était utilisée. Les tables TITLEINFO et TITLEPLOT contiennent le titre et le résumé du film, et les index des colonnes TITLEINFO.NAME et TITLEPLOT.PLOTSUMMARY sont utilisés par Extension Net Search dans le cadre du service Web Recherche de films.

Résumé de la leçon

Dans ce chapitre, vous avez étudié la conception et le modèle de données de l'application Video Central.

Étape suivante

Maintenant, vous allez utiliser WebSphere Studio Application Developer pour déployer les services Web sur le serveur WebSphere Application Server.

Chapitre 6. Déploiement de l'application Video Central

Durée

L'étude de ce chapitre nécessite environ 20 minutes.

Logiciels requis

- WebSphere Application Server 4.0, Advanced Edition Single Server
 - IBM HTTP Server version 1.3.12, téléchargeable sur le site Web <http://www.ibm.com/software/webservers/htpservers/download.html>
 - WebSphere Studio Application Developer et le fichier VideoCentral.jar (l'importation de ce fichier dans WebSphere Studio Application Developer est décrite au Chapitre 4)
-

Présentation

Ce chapitre va vous permettre de déployer le code de l'application Video Central sur le serveur WebSphere Application Server (défini au Chapitre 4) à partir de l'environnement WebSphere Studio Application Developer.

Tâches relatives à Video Central

Vous allez maintenant apprendre à déployer un service Web sur un serveur WebSphere Application Server à partir de WebSphere Studio Application Developer.

Tâche 1 : Préparation de l'environnement au déploiement de Video Central à partir de WebSphere Studio Application Developer

1. Assurez-vous que que DB2 et HTTP Server sont actifs, de la manière suivante :
 - a. Affichez les services Windows de votre poste de travail et vérifiez que DB2 et IBM HTTP Server ont démarré. Si tel n'est pas le cas, démarrez-les.
 - b. Ouvrez un navigateur Web et entrez l'URL localhost. La fenêtre doit maintenant se présenter comme suit (voir Figure 20) :



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**



Figure 20. Vérification de HTTP server.

2. Sélectionnez l'écran Servers dans la perspective Servers de WebSphere Studio Application Developer. Sur cet écran, cliquez avec le bouton droit de la souris sur VideoCentralServer et sélectionnez **Start**. Sélectionnez l'écran Console et vérifiez que le message "Serveur Default server ouvert pour e-business " est présent (voir Figure 21).


```
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
```

Figure 21. Vérification du fonctionnement du serveur d'administration de WAS.

3. Sélectionnez l'écran Servers dans la perspective Servers de Websphere Studio Application Developer. Sur cet écran, cliquez avec le bouton droit de la souris sur VideoCentralServer et sélectionnez **Stop**.

Remarque : L'arrêt du serveur VideoCentralServer est nécessaire car ce serveur sera automatiquement démarré lors du déploiement des services Web et qu'une erreur de port en cours d'utilisation se produirait s'il était déjà démarré.

Tâche 2 : Déploiement des services Web sur le serveur WebSphere Application Server

Création et déploiement des services Web (utilisant Java Bean) à l'aide de WebSphere Studio Application Developer

1. Pour déployer les services Web sur le serveur WebSphere Application Server à l'aide de WebSphere Studio Application Developer, procédez comme suit :
 - a. Sélection du type de Service Web
Dans WebSphere Studio Application Developer, affichez la perspective Web, développez le projet Web VideoCentral, puis sélectionnez :
VideoCentral/source/COM/ibm/db2/samples/videocentral/services/
BusinessRegistration.java
comme premier service Web à déployer (voir Figure 22).

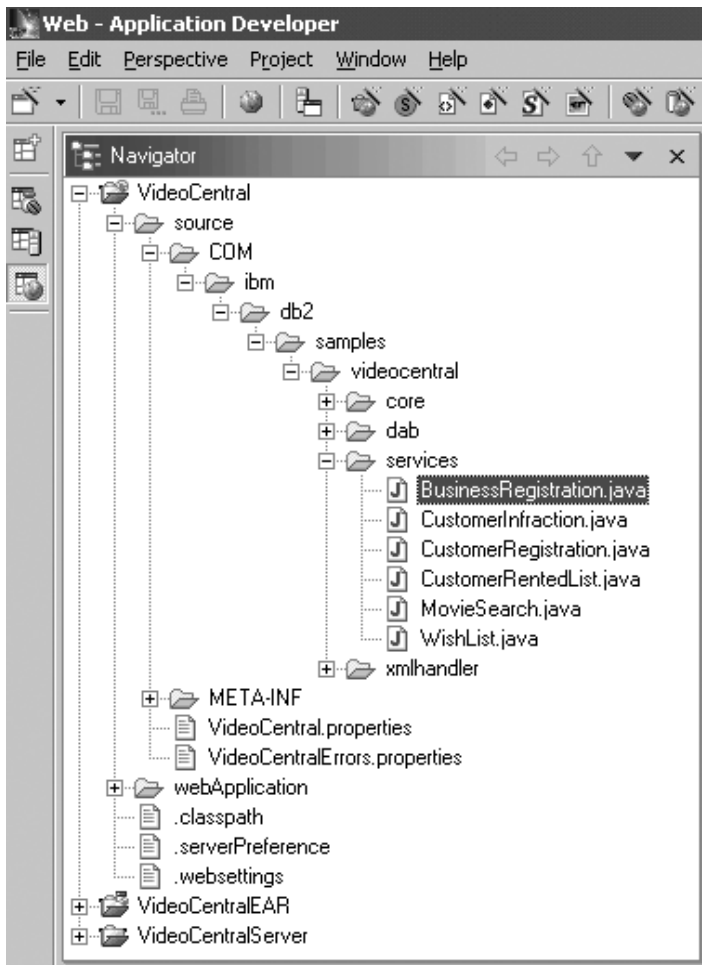


Figure 22. Choix du service Web à déployer.

- b. Cliquez avec le bouton droit de la souris sur **BusinessRegistration.java**, sélectionnez **New** —> **Other**, puis cliquez sur **Service Web** dans la fenêtre **New** qui s'affiche. Conservez les valeurs par défaut et cliquez sur **Suivant**.
2. Dans la fenêtre **Web Services Java Bean Selection**, vérifiez que le bean est bien celui que vous avez sélectionné à l'**Étape 1** (voir Figure 23).

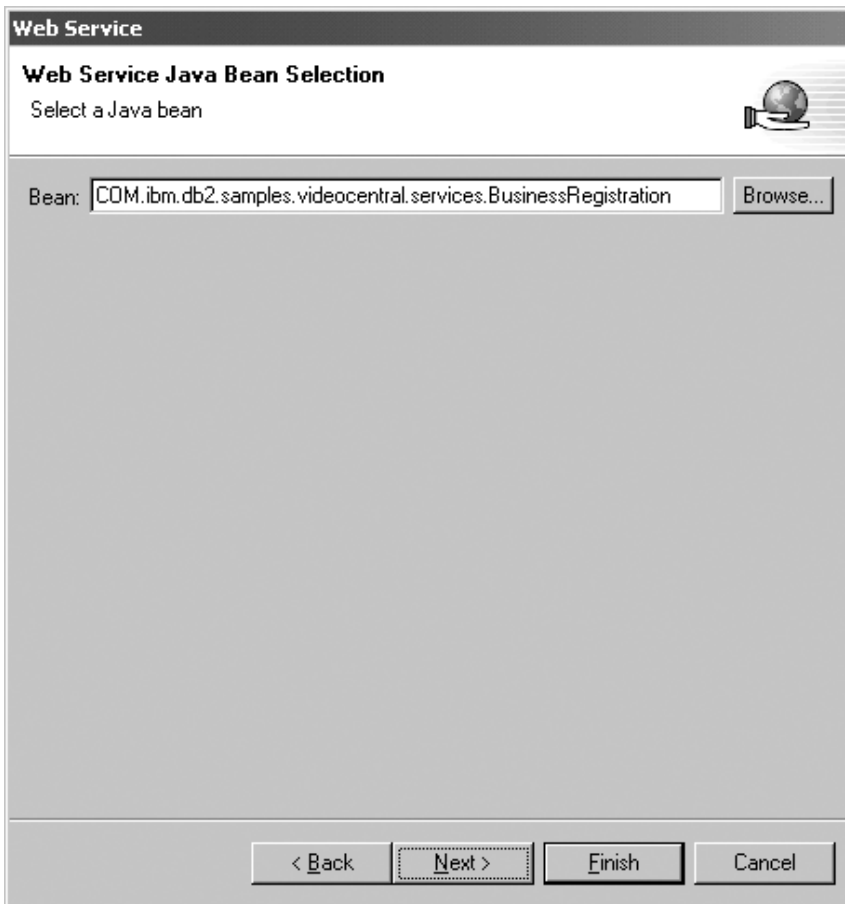


Figure 23. Java Bean à déployer.

Cliquez sur **Suivant**.

3. Configurez le Java Bean en tant que service Web. Conservez les valeurs par défaut et cliquez sur **Suivant** (voir Figure 24).

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

Figure 24. Configuration du Java Bean en tant que service Web.

4. Laissez les méthodes sélectionnées (voir Figure 25).



Figure 25. Sélection des méthodes publiques à déployer.

Cliquez sur **Suivant**.

5. Sur la page **Web Service Binding Proxy Generation**, dans la zone **Classe**, remplacez la classe :


```
proxy.soap.COM.ibm.db2.sample.videocentral.services.  
BusinessRegistrationProxy
```

 par :


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

 Reportez-vous à la Figure 26. Ainsi, la classe proxy peut être créée dans le package `COM.ibm.db2.samples.videoclient.proxies.*` du projet VideoCentral.



Figure 26. Génération du proxy de lien au service Web.

6. Cliquez sur **Finish** pour déployer le service.
7. Sélectionnez l'écran Servers dans la perspective Servers. Sur cet écran, cliquez avec le bouton droit de la souris sur VideoCentralServer et sélectionnez **Stop**.


Remarque : Si le déploiement a échoué, reportez-vous au "Chapitre 8. Résolution des incidents courants" pour connaître les causes probables de cet échec.

À ce stade, le fichier WSDL est créé. Vous le voyez apparaître dans votre projet Web VideoCentral :

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

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

Si vous voulez afficher le fichier WSDL dans l'éditeur par défaut, cliquez deux fois dessus et sélectionnez le bouton **Source** (voir le fichier BusinessRegistration-service.wsdl à la Figure 27). Le document de liaison définit le protocole et le format des opérations et des messages définis par un portType particulier. Le document du service est une collection de points d'extrémité connexes pour les liaisons du document de liaison.



```
<?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>
```

Figure 27. Contenu du fichier WSDL.

Répétez les étapes 1 à 7 de la Tâche 2 pour les cinq autres services Web. Les six services disponibles dans cette version de l'application Video Central sont les suivants :

- 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

Une fois tous ces services Web déployés, vous verrez deux fichiers WSDL pour chaque service dans votre projet Web (voir Figure 28).

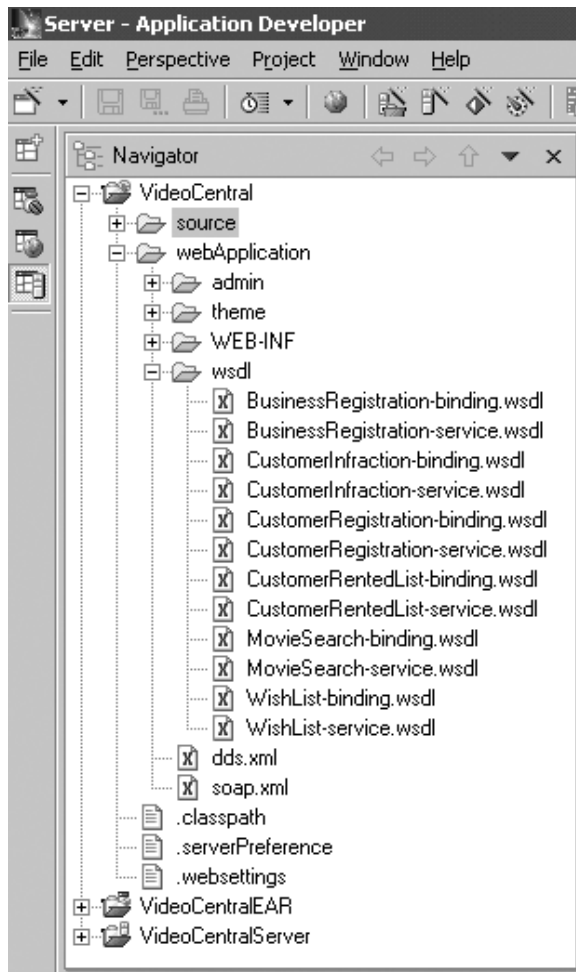


Figure 28. Tous les services Web déployés.

Résumé de la leçon

Dans ce chapitre, vous avez :

- vérifié l'environnement de déploiement de Video Central,
- déployé les six services Web de Video Central sur le serveur WebSphere Application Server à partir de WebSphere Studio Application Developer,
- généré les classes proxy à utiliser pour accéder aux services Web de Video Central,
- généré les fichiers WSDL pour chaque service. Ces fichiers pourraient être publiés dans le registre UDDI (voir le Chapitre 2) ou fournis à un demandeur de services.

Étape suivante

Maintenant, vous allez créer l'exemple de client Web pour appeler les services de Video Central afin de les voir fonctionner.

Chapitre 7. Création et exécution d'un exemple de client Web pour Video Central

Durée

L'étude de ce chapitre nécessite environ 25 minutes.

Avant de commencer

Vous devez avoir effectué avec succès les procédures décrites au "Chapitre 4. Initiation à WebSphere Studio Application Developer" et au "Chapitre 6. Déploiement de l'application Video Central".

Présentation

Ce chapitre décrit la conception d'un exemple de client Web (demandeur de services) destiné à l'application Video Central et la procédure à suivre pour utiliser les services Web fournis. Cette procédure comprend la régénération des classes proxy et le déploiement du client sur le serveur WebSphere Application Server. Au Chapitre 6, vous avez généré les classes proxy et les fichiers WSDL à partir des Java Beans. Vous allez maintenant régénérer les classes proxy à partir des fichiers WSDL.

Conception de l'exemple de client Video Central

La procédure décrite dans la section qui suit doit être effectuée sur le serveur d'applications du demandeur de services (Video Client). Le client crée les classes proxy correspondant aux services Web offerts par le fournisseur de services Web (voir Figure 29).

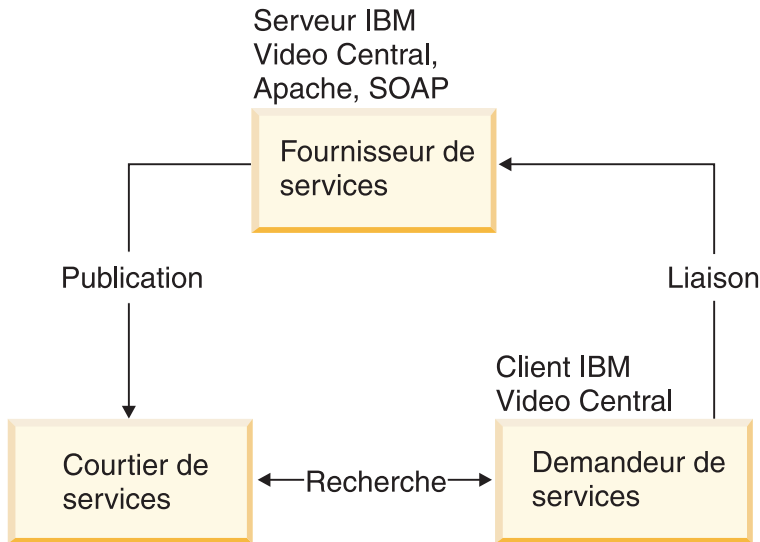


Figure 29. Composants des services Web.

Jusqu'ici, tous les fichiers utilisés ou créés l'ont été sur le serveur Video Central (le fournisseur de services). En revanche, la section qui suit concerne le serveur Video Client (le demandeur de services). Toutefois, pour les besoins de ce tutoriel, vous utiliserez le même serveur WebSphere Application Server, alors qu'en réalité Video Client résiderait sur un serveur d'applications distinct, situé dans chaque vidéoclub.

Schémas de classes et de séquences de l'exemple de client de Video Central

La Figure 30 présente le schéma des classes complet de l'exemple de client de Video Central, appelé Video Client.

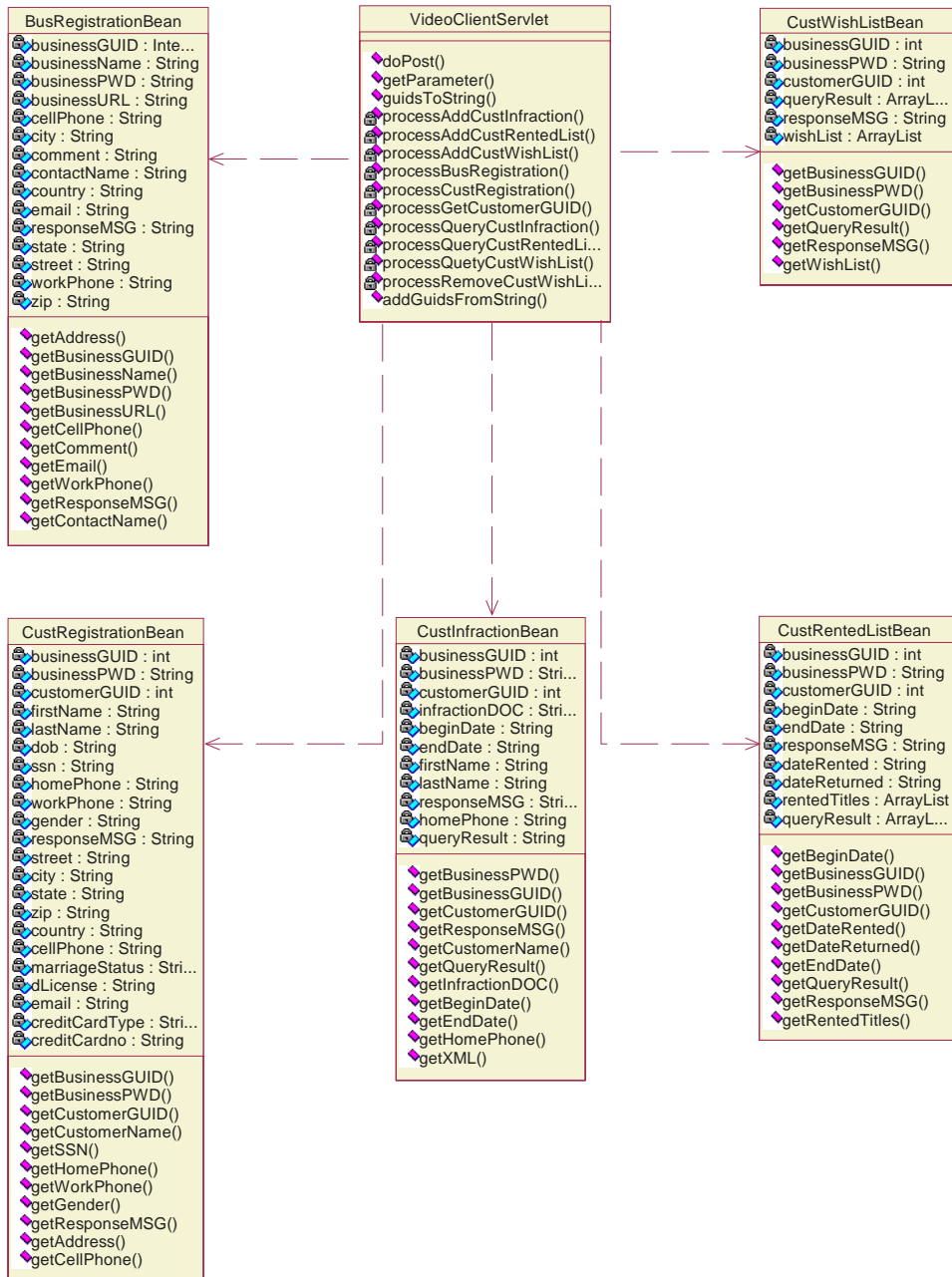


Figure 30. Schéma des classes de l'exemple de client de Video Central, appelé Video Client.

La Figure 31 présente le schéma de séquence complet de l'exemple de client de Video Central, appelé Video Client.

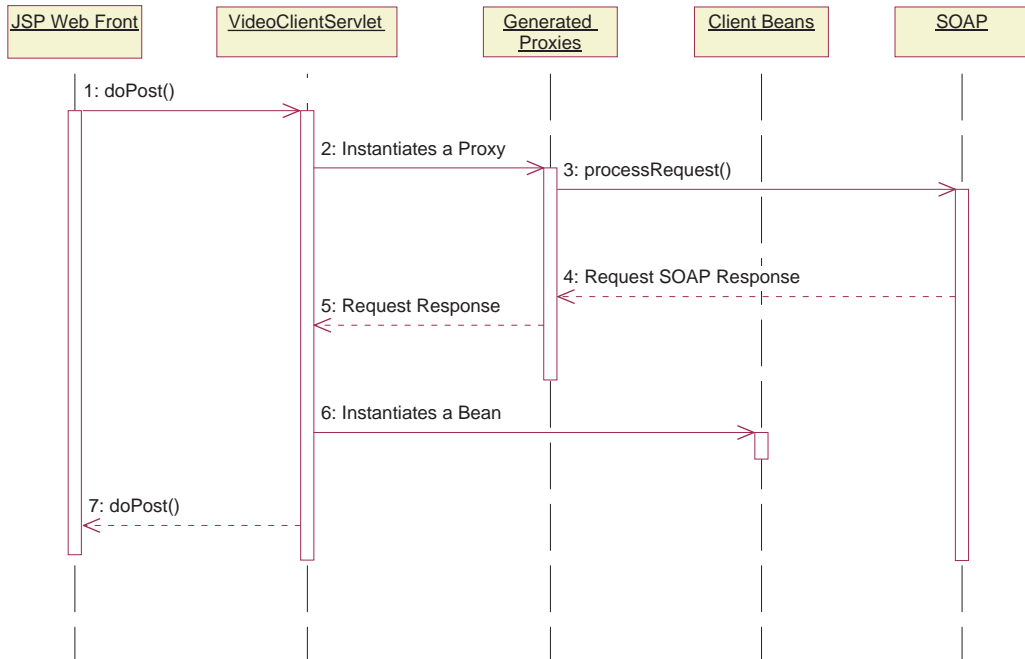


Figure 31. Schéma de séquence de l'exemple de client de Video Central, appelé Video Client.

Dans Video Client, toutes les demandes sont transmises au servlet VideoClientServlet. Lorsqu'il traite chaque demande, ce servlet appelle la demande proxy correspondante, qui appelle alors le service Web éloigné. À l'aide des informations contenues dans la réponse, VideoClientServlet instancie un bean client. Enfin, les résultats sont affichés sur le poste de l'utilisateur final à l'aide de JavaServer Page.

Tâches relatives à Video Central

Vous allez maintenant apprendre à :

- créer les classes proxy requises,
- déployer le code client sur le serveur WebSphere Application Server.

Tâche 1 : Préparation du projet client

Dans WebSphere Studio Application Developer, effectuez les opérations suivantes :

1. Créez le projet Web VideoClient, de la manière suivante :
 - a. Dans la perspective Web, sélectionnez le projet Web **VideoCentral** (il s'agit du niveau supérieur de l'arborescence de WebSphere Studio Application Developer).
 - b. Créez un projet Web en cliquant sur **Fichier** → **New** → **Web Project**.
 - c. Nommez le projet VideoClient et nommez l'Enterprise Application Project VideoClientEAR.
 - d. Cliquez sur **Suivant**.
 - e. L'écran **Web Settings** s'affiche. Conservez les valeurs par défaut et cliquez sur **Suivant**.
 - f. L'écran **Java Settings** doit maintenant s'afficher. Sélectionnez la page Libraries, cliquez sur **Add External JARs** et ajoutez les fichiers jars externes suivants, dans l'ordre indiqué, dans la fenêtre en incrustation **JAR Selection** :
 - <WAS_install_dir>\lib\xerces.jar
 - <rép_install_WSAD>\plugins\com.ibm.etools.webservice\runtime\soap.jar

Le fichier xerces.jar est requis par le demandeur du service Web Infractions d'un client car les infractions sont manipulées sous forme d'objets XML en mémoire et xerces est un analyseur syntaxique de modèle DOM (Document Object Model) XML fourni par IBM.

La fenêtre doit maintenant se présenter comme suit (voir Figure 32) :

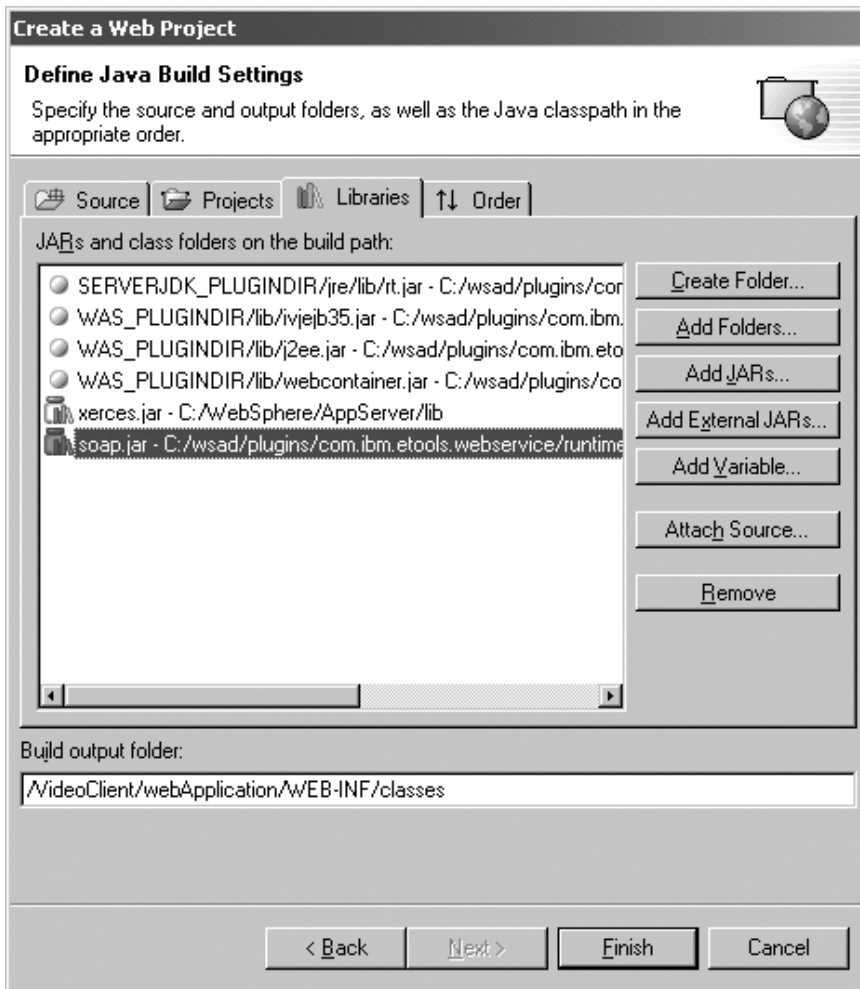


Figure 32. Ajout de fichiers JAR externes.

- g. Cliquez sur **Finish**.
2. Ajoutez le projet Web VideoClient au serveur, de la manière suivante :
 - a. Sur l'écran Server Configuration, développez **Server Configurations** et cliquez avec le bouton droit de la souris sur **VideoCentralServer**. Sélectionnez **Add Project** → **VideoClientEAR** (voir Figure 33).

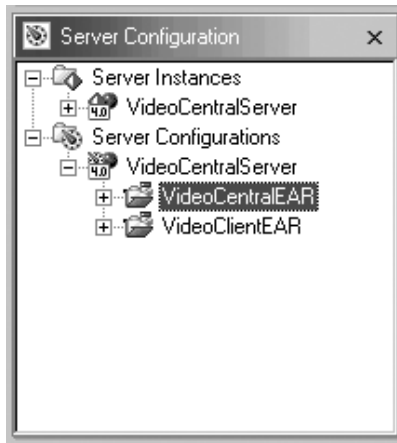


Figure 33. Ajout du projet Web client au serveur.

- b. Sur l'écran Navigator, cliquez avec le bouton droit de la souris sur le projet Web **VideoClient** et sélectionnez **Propriétés** → **Server Preferences**. Sélectionnez **VideoCentralServer** dans la zone.
 - c. Cliquez sur **Apply**, puis sur **OK**.
 3. Importez le code du client dans **VideoClient**, de la manière suivante :
 - a. Développez le projet Web **VideoClient** que vous venez de créer et sélectionnez **Source** (voir Figure 34).

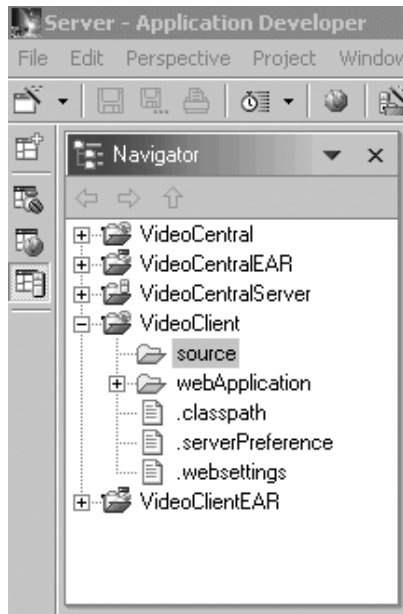


Figure 34. Sélectionnez Source

- b. Sélectionnez **Fichier** → **Import**.
- c. Comme Source d'importation, sélectionnez **Fichier ZIP**. Cliquez sur **Suivant**.
- d. Parcourez l'arborescence jusqu'au fichier
<rép_install_DB2>\tutorials\VideoCentral\lib\VideoClient.jar.
Cliquez sur **OK**. L'écran **Fichier ZIP** s'affiche (voir Figure 35).

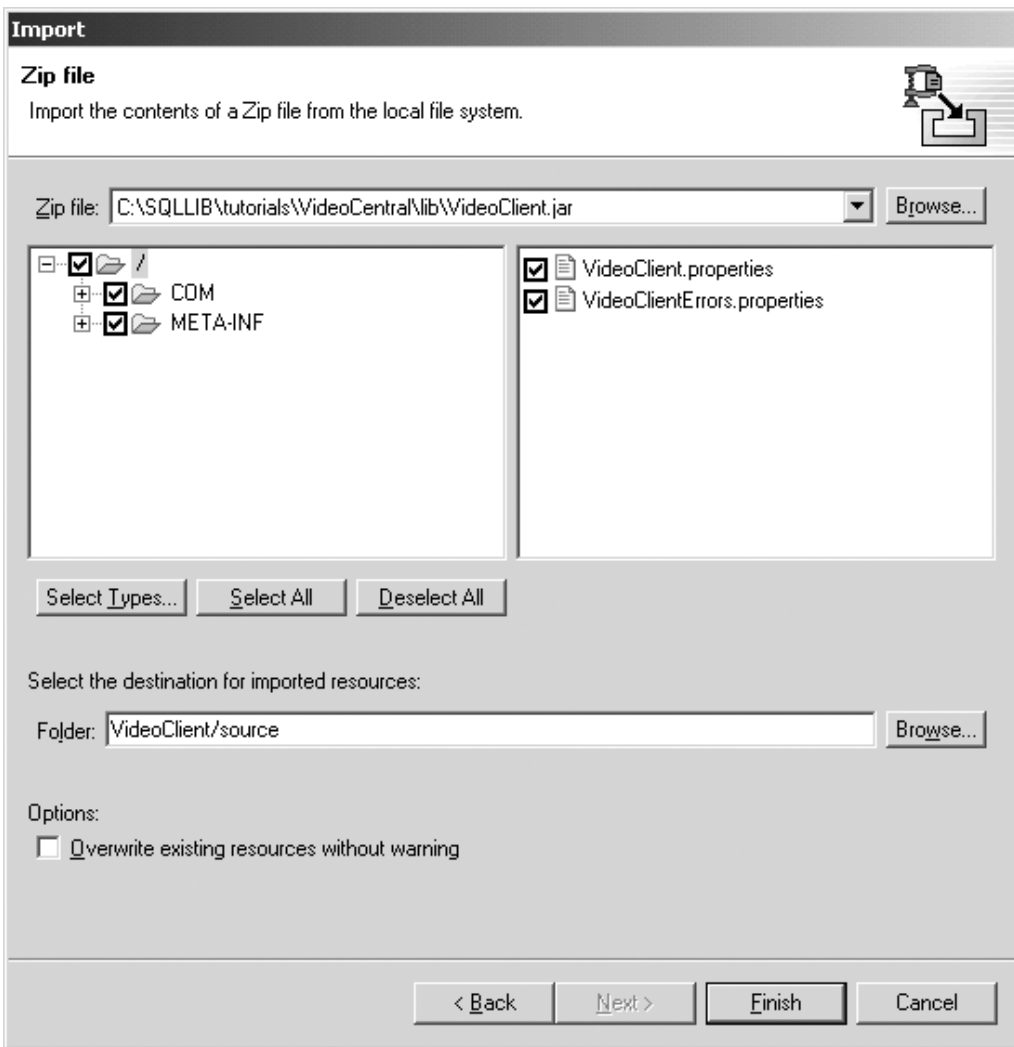


Figure 35. Importation du code du client dans le projet VideoClient à partir d'un fichier JAR.

- e. Cliquez sur **Finish**. Les fichiers sont alors importés dans WebSphere Studio Application Developer et la barre d'actions indique qu'ils sont en cours de compilation.

4. Importez les fichiers JSP du client, de la manière suivante :
 - a. Développez le projet Web VideoClient et sélectionnez **webApplication** (voir Figure 36).

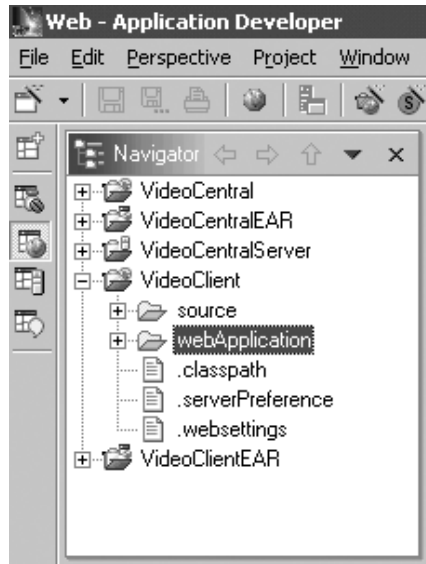


Figure 36. Sélection de webApplication.

- b. Sélectionnez **Fichier** → **Import**.
- c. Comme Source d'importation, sélectionnez **File System**. Cliquez sur **Suivant**.
- d. Parcourez l'arborescence jusqu'au fichier <rep_install_DB2>\tutorials\VideoCentral\jsp, puis cliquez sur **OK**. L'écran **File System** s'affiche. Cliquez sur **Select All** (voir Figure 37).

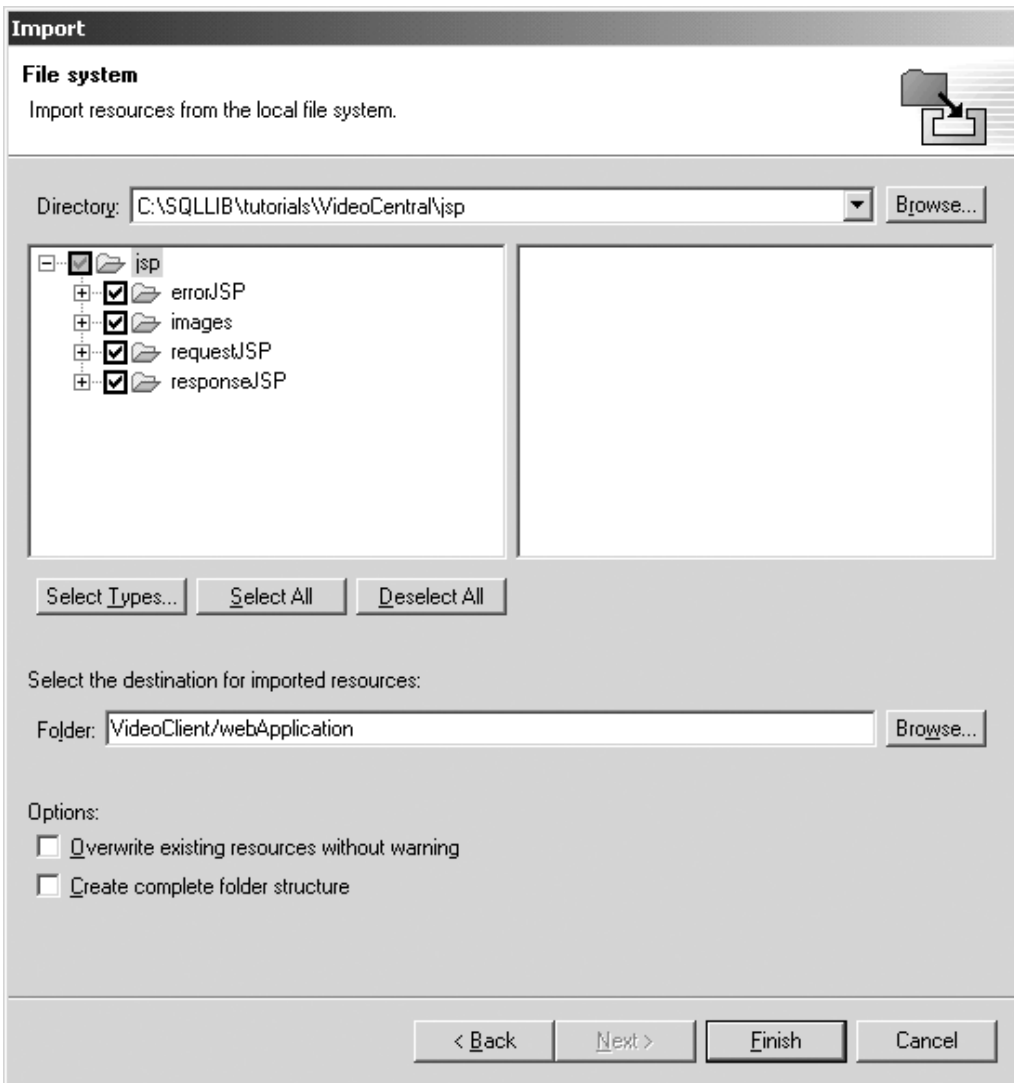


Figure 37. Écran File system.

- e. Cliquez sur **Finish**.

Les fichiers sont alors importés dans WebSphere Studio Application Developer et la barre d'actions indique qu'ils sont en cours de compilation.

Une fois les fichiers importés dans le projet Web VideoClient, vous devez obtenir une structure similaire à celle de la Figure 38 ci-après.

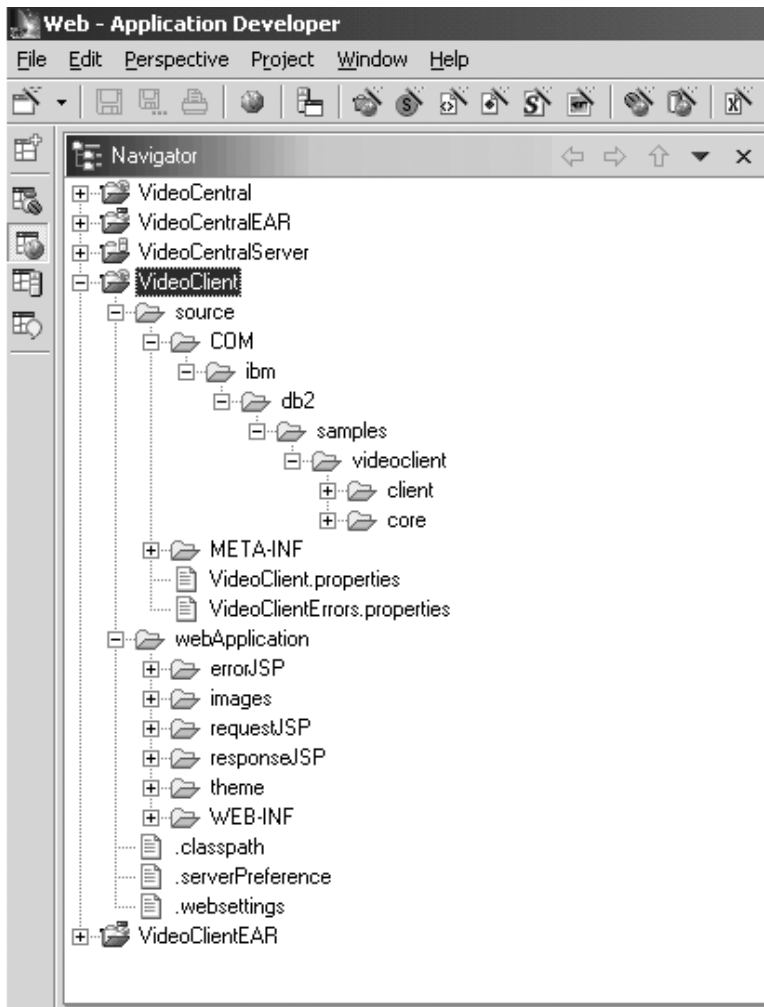


Figure 38. Après l'importation du client de Video Central, appelé VideoClient.

Tâche 2 : Création des proxy à l'aide de WebSphere Studio Application Developer

Pour pouvoir effectuer la procédure décrite dans cette section, vous devez avoir effectué celle décrite au Chapitre 4. En outre, les fichiers WSDL créés dans la section précédente sont requis pour cette tâche.

1. Dans la perspective Web, sous le projet Web VideoClient, cliquez avec le bouton droit de la souris sur **source/COM/ibm/db2/samples/videoclient**. Sélectionnez **New** → **Folder**, puis crez un dossier appelé proxies. Le conteneur doit indiquer
VideoClient/source/COM/ibm/db2/samples/videoclient
/proxies. Cliquez sur **Finish**.
2. Générez le proxy pour le service BusinessRegistration.
 - a. Dans la perspective Web, développez **webApplication\wsdl** sous le projet Web VideoCentral, cliquez avec le bouton droit de la souris sur **BusinessRegistration-service.wsdl**, puis sélectionnez **New** → **Other**. Dans la fenêtre en incrustation, sélectionnez **Services Web** → **Web Service Client**. Cliquez sur **Suivant**.
 - b. Cliquez de nouveau sur **Suivant** jusqu'à ce que la fenêtre en incrustation **Web Service Binding Proxy Generation** s'affiche.
 - c. Dans la zone **Dossier**, remplacez /VideoCentral/source par /VideoClient/source.
 - d. Dans la zone **Classe**, remplacez
proxy.soap.BusinessRegistrationProxy par :
COM.ibm.db2.samples.videoclient.proxies.BusinessRegistrationProxy

Remarque : Pour vous faciliter la tâche lors de la création des autres proxy, copiez ce chemin d'accès dans le presse-papiers. Vous pourrez le coller dans le chemin d'accès des proxy suivants.

La fenêtre doit maintenant se présenter comme suit (voir Figure 39) :

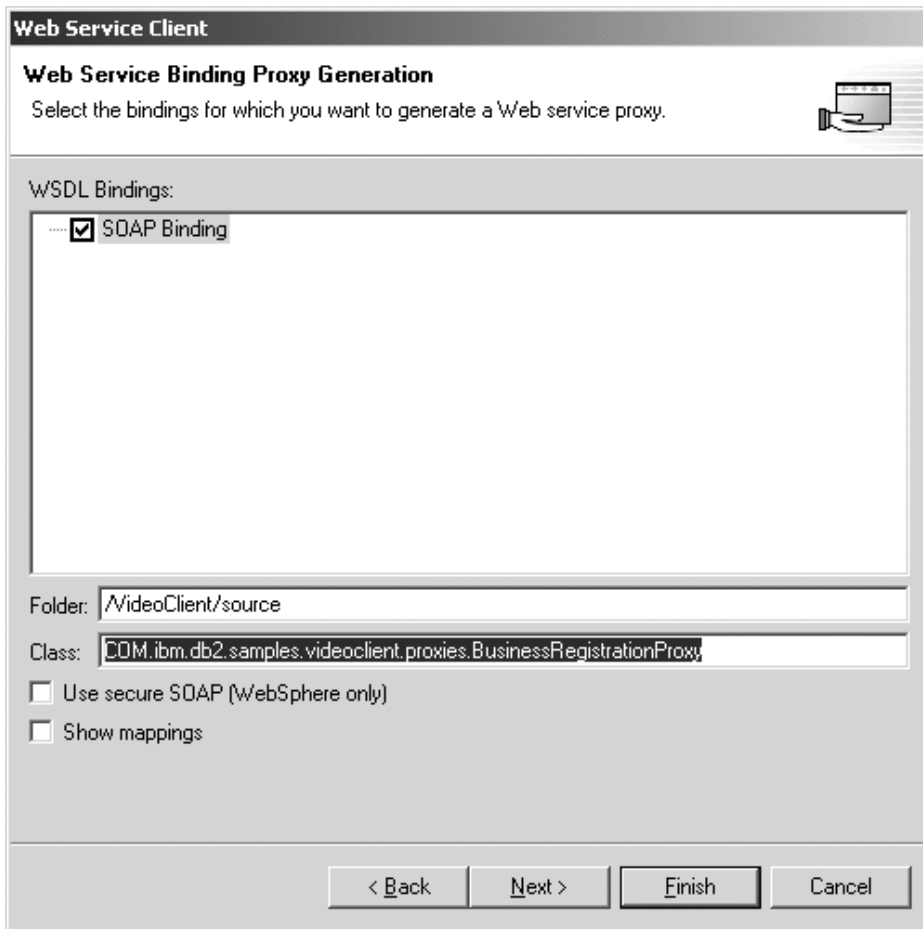


Figure 39. Génération du proxy BusinessRegistration.

e. Cliquez sur **Finish**.

Répétez l'**étape 2 de la Tâche 2** pour les cinq autres fichiers WSDL :

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

Les fichiers Java proxy doivent maintenant apparaître dans votre projet Web VideoClient. WebSphere Studio Application Developer ne doit plus afficher d'erreur de compilation, mais tout au plus quelques avertissements, lesquels sont acceptables (voir Figure 40).

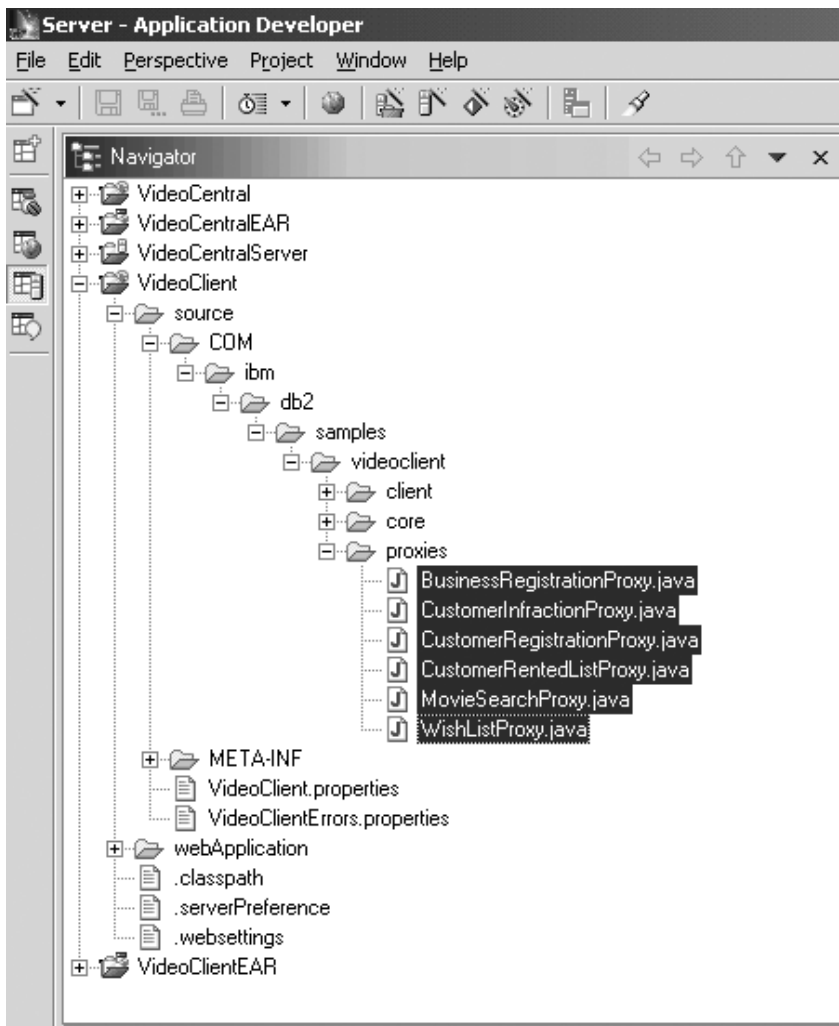


Figure 40. Proxy créés pour tous les fichiers WSDL.

Tâche 3 : Déploiement de l'exemple de client sur le serveur WebSphere Application Server

1. Dans la perspective Web, développez le dossier **VideoClient\webApplication\WEB-INF**.
2. Dans le dossier **Web-INF**, ouvrez le fichier **web.xml**.
3. Sélectionnez la page **Servlets**.
4. Ajoutez le servlet **VideoClientServlet**, de la manière suivante :
 - a. Cliquez sur le bouton **Add**, sélectionnez **VideoClientServlet**, puis cliquez sur **OK**.
 - b. Sélectionnez **VideoClientServlet**, puis, dans la zone **URL mappings**, cliquez sur **Add** et tapez **servlet/VideoClient** là où est indiqué (New URL). La fenêtre doit maintenant se présenter comme suit (voir Figure 41) :

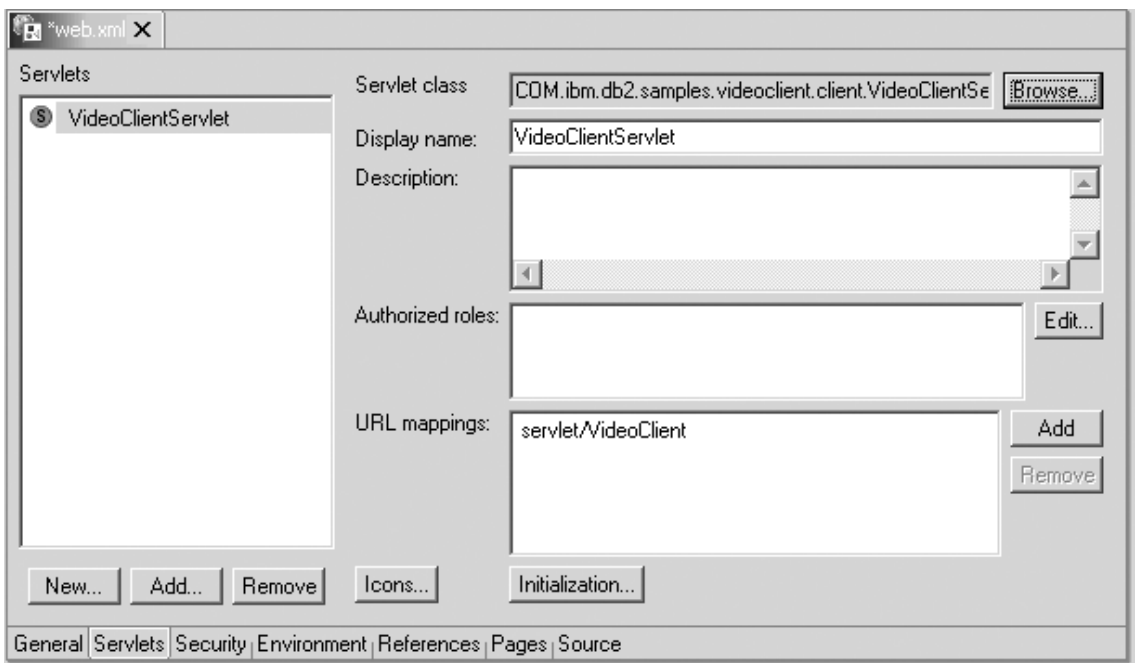


Figure 41. Ajout des servlets client.

5. Cliquez sur **Fichier** → **Save web.xml**.
6. Fermez la fenêtre **web.xml**.

Tâche 4 : Exécution de l'exemple de client

1. Pour accéder au client, ouvrez un navigateur Web et entrez l'URL `http://localhost:9080/VideoClient/requestJSP/VideoClientRequest.jsp`.
2. Vous pouvez maintenant utiliser le client. Pour commencer à utiliser les services de Video Central, inscrivez une entreprise. Ensuite, inscrivez des clients de cette entreprise. Vous pouvez alors ajouter une infraction pour un client à l'aide d'extension XML. Cela fait, vous pouvez demander la liste des infractions de ce client. Si vous utilisez un navigateur Web XML, le document XML contenant les infractions s'y affichera. Pour finir, testez les autres services comme vous le souhaitez.

Remarque : Un fichier `info.vc` sera créé dans votre répertoire `c:\`. En effet, la version actuelle de l'application Video Central lit et écrit dans ce fichier pour enregistrer les entreprises clientes. Nous reconnaissons que cette approche n'est pas la meilleure et nous modifierons la conception de l'application dans les versions à venir. Toutefois, Video Central étant une application business-to-business, aucun contexte n'est disponible entre les appels de services Web. L'utilisation du fichier `info.vc` permet d'effectuer des demandes de services (via le client) sans qu'il soit nécessaire de fournir à chaque fois un ID et un mot de passe. Idéalement, une clé d'accès de sécurité digne de confiance devrait être codée pour chaque demande de service Web.

Résumé de la leçon

Dans ce chapitre, vous avez :

- étudié les grandes lignes de la conception de l'exemple de client Video Central,
- créé les classes proxy,
- déployé le code du client,
- exécuté le client.

Chapitre 8. Résolution des incidents courants

1. Lorsque le déploiement du code sur le serveur WebSphere Application Server à partir de WebSphere Studio Application Developer échoue, vérifiez les points suivants : le chemin d'installation de WebSphere peut être incorrect. La valeur par défaut indiquée sur l'écran **Java bean Web Service Deployment** de WebSphere Studio Application Developer est `C:\WebSphere\AppServer`. Remplacez-la par le répertoire d'installation de WebSphere Application Server dans votre environnement.
2. Si vous obtenez le message d'erreur "Launching the server failed: Trace service port 7000 is in use. ORB bootstrap port 900 is in use. Change each used port number to another unused port on the ports page of the server configuration editor. In case you have another WebSphere server running you can try to increase each used port number by one and try again", procédez comme suit :
Dans WebSphere Studio Application Developer, sélectionnez l'écran **Servers** dans la perspective **Servers**. Sur cet écran, cliquez avec le bouton droit de la souris sur **VideoCentralServer** et sélectionnez **Stop**.
3. Si une erreur se produit lorsque vous tentez d'accéder à partir de votre navigateur aux services Web fournis par Video Central, la première chose à faire est de vérifier si l'état de l'instance du serveur est "Server is synchronized". Il se peut que vous deviez publier à nouveau les projets Web dans WebSphere Studio Application Developer.
4. Si vous obtenez le message d'erreur "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]" lorsque vous accédez aux services Web de Video Central, vérifiez que le numéro du port utilisé par la classe proxy correspondante pour communiquer avec le service est identique au numéro du port sur lequel le module d'exécution SOAP est à l'écoute (dans ce tutoriel, il s'agit du port 9080).
5. Si vous obtenez le message d'erreur "DES7300N L'index n'a pas été activé, SQL0438N", vérifiez l'entrée qui définit l'emplacement des index Net Search, `mvnets.inxdir`, dans le fichier `VideoCentral.properties` pour vous assurer qu'elle contient le chemin d'accès correct. Vous devrez peut-être aussi exécuter les commandes suivantes :
 - `db2nx update index name database sample`
 - `db2nx update index plot database sample`

6. Pour vous aider à déboguer les erreurs éventuelles de l'application dans WebSphere Studio Application Developer, le journal de la console du serveur, accessible sur l'écran Servers, fournit des informations utiles pour la résolution des incidents.

Chapitre 9. Sources d'informations complémentaires

- Sur IBM DB2 : <http://www.ibm.com/software/data/db2/>
- Documentation Extension XML :
<http://www.ibm.com/software/data/db2/extenders/xml/ext/library.html>
- Documentation Extension Net Search :
<http://www.ibm.com/software/data/db2/extenders/netsearch/download.html>
- Sur SOAP : <http://www.w3.org/TR/SOAP/>
- Sur la programmation en Java : <http://www.javasoft.com>
- Sur WSDL : <http://www.w3.org/TR/wsdl>
- Sur les services Web : <http://www.ibm.com/developerworks/webservices/>
- Sur IBM DeveloperWorks - XML :
<http://www.ibm.com/developerworks/xml/>
- Sur WebSphere Developer Domain : <http://www7b.boulder.ibm.com/wsdd/>
- Sur DB2 Developer Domain : <http://www7b.boulder.ibm.com/dmdd/>
- Sur les services Web DB2 :
<http://www.ibm.com/software/data/webservices/>
- Sur l'intégration WebSphere/DB2 :
<http://www.ibm.com/software/data/integration/WebSphere>

Annexe A. Informations de référence

Qu'est-ce que DB2 Extension XML ?

Extension XML vous aide à intégrer la puissance d'IBM DB2 Universal Database et la souplesse de XML. DB2 Extension XML vous permet d'enregistrer des documents XML et d'y accéder, de générer des documents XML à partir de données relationnelles existantes, et de décomposer des documents XML en données relationnelles (en enregistrant le contenu des attributs ou des éléments sans balises).

Extension XML fournit de nouveaux types de données, ainsi que des fonctions et des procédures mémorisées nouvelles, pour vous aider à gérer vos données XML sous DB2. Il vous permet également de décomposer des documents XML en leurs composants constitutifs et d'enregistrer ceux-ci sous forme de colonnes dans plusieurs tables. Il est possible de définir des index sur un élément ou un attribut d'un document XML pour accélérer son extraction. En outre, la recherche de texte et de section dans une colonne XML ou composant à l'aide des extensions Texte.

Vous pouvez également formuler un document XML à partir de vos tables DB2 existantes afin d'échanger des données dans un environnement business-to-business. Net.Data et Extension XML peuvent être utilisés pour générer des documents XML à partir de DB2. Vous pouvez ensuite diffuser ces documents aux clients, qui peuvent les afficher dans leur navigateur.

Extension XML fournit les fonctions suivantes pour vous aider à gérer et exploiter les données XML sous DB2 :

- Outils d'administration conçus pour vous aider à gérer l'intégration de données XML dans des tables relationnelles.
- Méthodes d'enregistrement et d'utilisation de vos données XML : colonne XML et collection XML.
- Référentiel de DTD (Document Type Definition, définition de type de document) (DTD_REF) dans lequel vous pouvez enregistrer les DTD utilisées pour valider les données XML.
- Schéma de mappage, appelé fichier DAD (Document Access Definition, définition d'accès au document), qui vous permet de mapper des documents XML vers des données relationnelles. Les fichiers DAD sont gérés à l'aide de la table XML_USAGE, créée lorsque vous préparez une base de données en vue de l'utilisation de XML.
- Puissantes fonctions définies par l'utilisateur (fonctions UDF), permettant d'enregistrer et d'extraire des documents XML dans des colonnes XML ainsi que d'extraire des valeurs d'éléments ou d'attributs XML. Une fonction UDF est une fonction définie dans le système de gestion de base de données et que vous pouvez utiliser dans des requêtes SQL. Extension XML fournit les types de fonctions UDF suivants :
 - Enregistrement : enregistrement de documents XML intacts dans des colonnes rendues compatibles avec XML, sous forme de types de données XML.
 - Extraction : extraction de documents XML, ou des éléments et attributs spécifiés sous forme de types de données de base.
 - Mise à jour : mise à jour de documents XML complets ou des valeurs d'éléments et d'attributs spécifiées.

Toutes les fonctions UDF d'Extension XML possèdent le préfixe db2xml, qui est le nom de schéma de ces fonctions.

Qu'est-ce que DB2 Extension Net Search ?

DB2 Extension Net Search ajoute la puissance d'un moteur d'extraction rapide en texte intégral aux applications Net.Data, Java ou DB2 CLI intégrées à DB2 Universal Database.

Extension Net Search permet une indexation rapide de volumes de données importants, ainsi qu'une recherche à grande vitesse même avec un grand nombre d'utilisateurs simultanés. Elle place des colonnes de table prétriées en mémoire principale lors de l'indexation pour éviter des accès à la base de données et des opérations de pagination coûteux lors de la recherche.

Extension Net Search fournit les fonctions suivantes, qui complètent celles de DB2 Extension Texte et DB2 Extension Text Information :

- Indexation :
 - Type d'index très rapide. (Ngram)
 - Possibilité de définir plusieurs index sur une même colonne de texte.
 - Indexation sans verrouillage des données.
 - Mise à jour dynamique des index, reflétant les modifications dans la base de données.
- Recherche :
 - Fournie sous forme de procédure mémorisée et non de fonctions UDF.
 - Possibilité de rechercher des mots, des groupes de mots, des radicaux ou des correspondances partielles.
 - Possibilité de restreindre la recherche aux sections des documents identifiées par des balises spécifiques.
 - Possibilité de recherche numérique sur une plage de valeurs.
 - Prise en charge des opérations booléenne et des recherches avec caractères génériques.

- Résultats de la recherche :
 - Possibilité de spécifier comment les résultats de la recherche doivent être triés lors de l'indexation, ou d'indiquer des valeurs de rang pour le tri.
 - Possibilité de définir des sous-ensembles des résultats de recherche lorsque celle-ci porte sur des volumes de données importants et que des listes de résultats longues sont prévisibles.
 - Possibilité de définir une limite pour les termes recherchés, en définissant un nombre maximum de correspondances.
 - Possibilité de définir l'accès (définition de curseur) à partir des résultats de la recherche.

Qu'est-ce qu'IBM WebSphere Studio Application Developer ?

IBM WebSphere Studio Application Developer (également appelé WSAD dans ce document) s'inscrit dans la lignée de WebSphere Studio, Professional et Advanced Editions et de VisualAge for Java, Enterprise Edition. Ces produits permettent de développer, tester et déployer des applications d'e-business.

Les nouveaux produits WebSphere Studio sont conçus de manière ascendante afin de répondre aux exigences de tous les nouveaux types d'applications. Parmi ces exigences on peut citer : les normes ouvertes, Java, XML, les services Web, les tests, différents niveaux d'intégration avec d'autres composants et produits ISV, la modularité, l'expansibilité, le développement à base de rôles, une utilisabilité accrue pour tous les utilisateurs, une meilleure prise en charge du travail en équipe, et un cycle de développement plus court. WebSphere Studio fournit des outils de développement intégrés pour tous les rôles de développement d'applications d'e-business, des développeurs Web aux développeurs Java et des analystes métier aux architectes et aux programmeurs d'entreprise.

WebSphere Studio regroupe les fonctions les plus populaires de WebSphere Studio "Classic" et de VisualAge for Java et les combine aux avantages de nos technologies les plus récentes, ce qui lui permet d'offrir des normes ouvertes, une intégration des outils, une souplesse accrue et la possibilité d'intégrer les applications existantes.

Environnement des services Web

WebSphere Studio comporte des assistants et d'autres outils qui permettent le développement rapide de services Web.

Ces outils sont basés sur les normes ouvertes et multiplateformes suivantes :

- *Universal Description Discovery and Integration (UDDI)* : permet aux entreprises de se présenter, de publier des spécifications techniques décrivant comment elles souhaitent traiter par e-business avec les autres

entreprises, et de rechercher des entreprises qui fournissent les biens et services dont elles ont besoin, le tout via des registres UDDI en ligne.

- **Simple Object Access Protocol (SOAP)** : ce protocole est une norme qui permet à des applications métier d'échanger de façon fiable des messages commerciaux électroniques via Internet.

SOAP est un protocole simple qui permet l'échange d'informations dans un environnement décentralisé réparti. Basé sur XML, il se compose de trois parties : une enveloppe qui définit un cadre pour la description du contenu d'un message et la manière de le traiter, un ensemble de règles de codage pour l'expression des instances des types de données définis par les applications, et une convention de représentation des appels de procédure éloignées et de leurs réponses.

SOAP peut être utilisé en combinaison avec un grand nombre d'autres protocoles. Toutefois, les seules liaisons définies dans ce document décrivent comment l'utiliser avec HTTP et HTTP Extension Framework.

Pour connaître la spécification SOAP version 1.1 et obtenir d'autres informations de référence concernant SOAP, consultez le site Web <http://www.w3.org/TR/SOAP/>

- **Web Services Description Language (WSDL)** : décrit les programmes accessibles via Internet (ou d'autres réseaux) et les formats de messages et les protocoles utilisés pour communiquer avec eux.

WebSphere Studio facilite les processus suivants pour vous aider à construire et déployer des applications capables de tirer parti des services Web :

- **Découverte.** Exploration du registre commercial UDDI à la recherche de services Web intégrables.
- **Création ou Transformation.** Création de services Web à partir d'éléments existants, tels que des beans, des URL qui acceptent et renvoient des données, des appels DB2 Extension XML, des procédures mémorisées DB2 et des requêtes SQL.
- **Construction.** Encapsulation d'éléments existants sous forme de services SOAP et HTTP accessibles par GET/POST et description de ces éléments à l'aide de WSDL. Les assistants de services Web vous aident à générer un proxy SOAP pour des services Web décrits en WSDL et à générer des squelettes de beans à partir de WSDL.
- **Déploiement.** Déploiement de services Web dans les environnements de test de WebSphere Application Server ou Tomcat à l'aide de Server Tools.
- **Test.** Test de services Web s'exécutant localement ou à distance pour obtenir une évaluation immédiate.
- **Développement.** Génération d'exemples d'applications pour vous aider à créer votre propre application client de service Web.

- **Publication.** Publication de services Web dans le registre commercial UDDI, ce qui vous permet de faire connaître vos services Web afin que d'autres entreprises puissent y accéder.

Pour plus d'informations sur les outils relatifs aux services Web, consultez l'aide en ligne de WebSphere Studio Application Developer.

Environnement de développement XML

WebSphere Studio fournit un environnement de développement XML complet qui comprend des outils permettant de construire des DTD ainsi que des schémas et des fichiers XML. Il prend également en charge l'intégration des données relationnelles et de XML.

Les outils XML suivants sont disponibles :

L'éditeur XML permet de créer, visualiser et valider des fichiers XML. Vous pouvez l'utiliser pour créer des fichiers XML à partir de rien, de DTD existantes ou de schémas XML existants. Vous pouvez aussi l'utiliser pour modifier des fichiers XML, les associer à des DTD ou des schémas, et les valider.

L'éditeur DTD permet de créer, visualiser et valider des DTD (Document Type Definition, définition de type de document). Vous pouvez l'utiliser pour créer et valider des éléments, des attributs, des entités et des notations de DTD. À l'aide de cet outil, vous pouvez aussi générer des fichiers schémas XML, ainsi que des beans Java afin de créer des instances XML d'un schéma XML. Il vous permet également de générer un formulaire HTML par défaut basé sur les DTD que vous avez créées.

L'éditeur de schéma XML permet de créer, visualiser et valider des schémas XML. Il vous permet de créer des composants de schéma XML, d'importer et de visualiser des schémas XML, de générer des DTD et des définitions de tables relationnelles à partir de schémas XML, de générer des beans Java afin de créer des instances XML d'un schéma XML, et de générer des DDL à partir d'un schéma XML.

L'éditeur de trace XSL permet d'appliquer une feuille de style XSL à un document XML afin de créer un document résultant (HTML ou XML). À l'aide de cet outil, vous pouvez transformer des documents XML en documents HTML, texte ou d'un autre type XML. L'éditeur affiche les trois documents (résultat, source XML et source XSL) et vous permet d'exécuter visuellement le script de transformation XSL en examinant les relations entre ces trois documents.

L'éditeur de mappage XML vers XML permet de mapper un ou plusieurs documents XML source vers un seul document XML cible. Après avoir fourni un fichier source (DTD ou XML) et un fichier cible, vous pouvez définir les mappages entre la source et la cible. Chaque mappage se compose d'une zone cible, d'une fonction de conversion et d'une ou plusieurs zones source. Vous pouvez modifier et supprimer les mappages, et les enregistrer en vue d'une utilisation ultérieure. Une fois les mappages définis, vous pouvez générer un script XSLT qui peut ensuite être utilisé pour combiner et transformer des documents XML conformes aux DTD source.

L'assistant de requête SQL et XML permet de créer un fichier XML à partir des résultats d'une requête SQL. Il vous permet également de créer un schéma XML ou un fichier DTD décrivant la structure du fichier XML, que vous pourrez utiliser dans d'autres applications. Vous pouvez également l'utiliser pour créer un fichier DADX utilisable avec l'outil de services Web. Ce fichier DADX contiendra votre requête SQL.

L'éditeur de mappages RDB vers XML facilite la définition d'un mappage entre des tables relationnelles et un fichier DTD. Vous pouvez mapper des colonnes d'une ou plusieurs tables relationnelles vers des éléments et des attributs d'un document XML. Vous pouvez également générer un script DAD (Document Access Definition, définition d'accès au document), utilisé par IBM DB2 Extension, pour composer des documents XML à partir de données DB2 existantes ou décomposer des documents XML en données DB2. Cet outil vous permet aussi de créer un sous-programme de test afin de tester le fichier DAD généré.

Pour plus d'informations sur les outils XML, consultez l'aide en ligne de WebSphere Studio Application Developer.

Annexe B. Glossaire

DAD (Document Access Definition, définition d'accès au document) : Permet de définir le schéma d'indexation d'une colonne XML ou le schéma de mappage d'une collection XML.

DTD (Document Type Definition, définition de type de document) : Fichier (ou fichiers à utiliser ensemble) écrit en respectant la syntaxe de déclaration XML et qui contient une description formelle d'un type de document particulier. Il définit les noms pouvant être utilisés pour les types d'éléments, les emplacements où ils peuvent apparaître, et comment ils s'articulent.

JDBC (Java Database Connectivity) : API (interface de programme d'application) qui présente les mêmes caractéristiques qu'ODBC (Open Database Connectivity) mais qui est spécifiquement conçue pour être utilisée par des applications de base de données Java.

SOAP (Simple Open Access Protocol) : Spécification de protocole qui définit une manière uniforme de transmettre des données codées en XML.

UDDI (Universal Description, Discovery and Integration) : Mécanisme de registre permettant aux clients de rechercher de façon dynamique d'autres services Web.

WSDL (Web Services Description Language) : Format XML qui décrit les possibilités offertes par un service Web, l'emplacement où il réside et la manière de l'appeler.

XML (eXtensible Markup Language) : Format universel des documents et des données structurés sur le Web. Il est extensible car il ne s'agit pas d'un format fixe, contrairement à HTML. Il s'agit d'un sous-ensemble de SGML (Standard Generalized Markup Language, langage général de mise en forme).

XSL (eXtensible Style Language) : Définit le langage standard des feuilles de style utilisées en XML.

Annexe C. Remarques

Les termes qui suivent sont des marques d'International Business Machines Corporation dans certains pays :

- DB2
- IBM
- VisualAge
- WebSphere

Microsoft et Windows NT sont des marques de Microsoft Corporation dans certains pays.

Java et toutes les marques et logos incluant Java sont des marques de Sun Microsystems, Inc. dans certains pays.

D'autres sociétés sont propriétaires des autres marques, noms de produits ou logos qui pourraient apparaître dans ce document.

Comment contacter IBM

En cas d'incident technique, lisez et effectuez les instructions présentées au Chapitre 8, Résolution des incidents courants, et dans le guide d'identification et de résolution des incidents "DB2 Troubleshooting Guide" avant de prendre contact avec le Service Clients DB2. Ce dernier manuel décrit les informations que vous pouvez collecter afin d'aider le Service Clients DB2 dans sa recherche d'une solution à votre problème.

Pour savoir comment commander un produit DB2 Universal Database, prenez contact avec votre partenaire commercial IBM.

Aux États-Unis, vous pouvez appeler l'un des numéros suivants :

- service clients : 1-800-237-5511
- informations sur les contrats de maintenance : 1-888-426-4343

Informations produit

Vous pouvez obtenir des informations produit concernant DB2 Universal Database par téléphone, ou via le World Wide Web à l'adresse suivante : <http://www.ibm.com/software/data/db2/udb/winos2unix/support>

Ce site Web contient les informations les plus récentes concernant la bibliothèque technique, la commande de manuels, le téléchargement de clients, les newsgroups, les FixPaks, les nouveautés, ainsi que des liens à des ressources Web.

Aux États-Unis, vous pouvez appeler l'un des numéros suivants :

- commande de produits et informations générales : 1-800-IBM-CALL (1-800-426-2255)
- commande de manuels : 1-800-879-2755

Pour savoir comment prendre contact avec IBM en dehors des États-Unis, consultez le site Web général IBM Worldwide à l'adresse suivante : <http://www.ibm.com/planetwide>. Pour la France, consultez le site Web <http://www.ibm.com/contact/fr>.

Dans certains pays, les revendeurs agréés IBM doivent prendre contact avec leur structure d'assistance pour obtenir des informations.

IBM