

IBM WebSphere Application Server - Express version 5.1



Guide de migration

Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques» à la page 65.

Sixième édition - décembre 2003

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 2003. Tous droits réservés.

© Copyright International Business Machines Corporation 2000, 2003. All rights reserved.

Table des matières

Chapitre 1. Présentation du guide de migration de WebSphere Application Server - Express	1
---	----------

Chapitre 2. Migration du serveur de production	3
Migration	3
Migration et coexistence	4
Outils de migration	5
Commande WASPreUpgrade	5
Commande WASPostUpgrade	7
Mappage de la configuration lors de la migration	8
Migration manuelle des données de configuration	10
Migration à partir de la version 3.5.x vers la version 5.1	11
Migration de la version 3.5.x vers une version 5.1 sur une machine éloignée.	13
Migration à partir de la version 5.0 vers la version 5.1	16
Migration de la version 5.0.x vers une version 5.1 sur une machine éloignée.	16
Migration à partir d'un système d'exploitation non pris en charge	18

Chapitre 3. Migration à partir d'IBM WebSphere Studio Site Developer version 5.1	21
Migration des projets J2EE en vue de l'utilisation de la fonction de prise en charge du ciblage de serveur.	21
Compatibilité en amont avec fonction de prise en charge du ciblage de serveur activée	21
La génération de l'assistant requiert un package Java pour JDK 1.4	22

Chapitre 4. Migration à partir d'IBM WebSphere Studio Site Developer version 5.0.1	23
WebSphere Studio Workbench dans la version 5, la version 5.0.1 et la version 5.1	23
Utilisation d'IBM WebSphere Studio Site Developer version 5.1.1 avec l'espace de travail de la version 5.0	23
Migration de projets Java à partir de la version 5 ou de la version 5.0.1	24
Partage des projets entre la version 5 ou la version 5.0.1 et la version 5.1.1 à l'aide d'un système de migration SCM (Source Code Management)	24
Migration de projets Web	26
Conversion des projets Web vers Struts 1.1	26
Modifications apportées aux outils des services Web	27
Modifications apportées aux outils de profilage	27
Problèmes de compatibilité de l'assistant de modèle	28

Chapitre 5. Migration à partir d'IBM WebSphere Studio Site Developer version 4.0.x	29
---	-----------

Différences entre IBM WebSphere Studio Site Developer version 4.0.x et version 5	29
Modifications apportées à WebSphere Application Server et outils de conversion Servlet/JSP	30
Modifications internes depuis la version 4.0.3	31
Par défaut, la compilation ne peut pas s'effectuer s'il existe des dépendances circulaires dans un projet	31
Les répertoires source des projets Web de la version 5 sont compatibles avec la version 4.0.3	31
Structures des projets Web IBM WebSphere Studio Site Developer	31
Projets Web statiques et dynamiques	32
Distinctions entre HTML et JSP	32
Migration des projets à l'aide d'un système SCM (Software Configuration Management)	32
Migration des projets à l'aide de CVS ou Rational ClearCase	32
Suppression des références de chemin d'accès absolu dans les fichiers EAR et les fichiers de configuration serveur après la migration.	34
Migration des projets à l'aide d'autres systèmes SCM.	35
Migration par exportation et importation de projets	35
Migration de projets à l'aide d'un espace de travail existant en version 4.0.x	35
Suppression des références de chemin d'accès absolu dans les fichiers EAR et les fichiers de configuration serveur après la migration.	36
Incidents connus et limitations	36
Migration de données relationnelles dans les projets Web 4.0.3	37
Erreurs WSDL après importation d'un fichier de services Web à partir de 4.0.x	37
Migration des structures de projet J2EE et des niveaux de spécification J2EE	38

Chapitre 6. Migration à partir de WebSphere Studio "Classic" vers IBM WebSphere Studio Site Developer	39
--	-----------

Création d'une planification ne comportant qu'un seul serveur pour la migration	40
Création d'un fichier descripteur de configuration Web	40
Exportation d'un fichier JAR de migration	40
Importation du fichier JAR de migration dans IBM WebSphere Studio Site Developer	41
Test d'une application migrée sur un serveur de test	42

Chapitre 7. Migration à partir de VisualAge for Java vers IBM WebSphere Studio Site Developer 43

Différences entre VisualAge for Java et IBM WebSphere Studio Site Developer	43
Migration à partir de VisualAge for Java	44
Exportation des fichiers Java et des fichiers de ressources de projet à partir de VisualAge for Java	44
Démarrage d'IBM WebSphere Studio Site Developer et création de nouveaux projets destinés à contenir votre code	45
Importation des fichiers de ressources et des fichiers Java dans IBM WebSphere Studio Site Developer	45
Utilisation de l'éditeur web.xml pour vérifier que les servlets sont correctement définis (projet Web uniquement)	45
Migration des paramètres du projet et de l'espace de travail	46
Configuration de l'environnement de test WebSphere 4 et test des applications ayant fait l'objet d'une migration	47
Déploiement de vos applications à partir d'IBM WebSphere Studio Site Developer vers un serveur distant WebSphere Application Server	48
Partage des paramètres de projet IBM WebSphere Studio Site Developer entre plusieurs développeurs (post-migration)	48
Support de développement coopératif dans IBM WebSphere Studio Site Developer	48

Chapitre 8. Migration à partir de VisualAge for Java Visual Composition Editor vers Visual Editor for Java 49

Sauvegarde des informations avancées relatives aux métadonnées à partir de VisualAge for Java	49
Exécution de la migration (importation dans WebSphere Studio)	50

Chapitre 9. Configuration de la compilation (bibliothèque, fichiers JAR, fichiers JAR dépendant d'un projet, compilation Ant) 51

Fichiers JAR des bibliothèques Java et fichiers JAR tiers	51
---	----

Méthode recommandée en cas d'utilisation d'un fichier JAR tiers dans un projet Web	51
Méthode recommandée en cas d'utilisation d'un fichier JAR tiers dans plusieurs projets Web	51
Autre mode d'utilisation des fichiers JAR externes (chemin de compilation globale et chemin d'accès aux classes du serveur)	52
Optimisation de compilations multiprojets à l'aide de fichiers JAR dépendant d'un projet	53
Compilations de production automatisées avec Ant	53

Chapitre 10. Exemples de migration . . . 55

Exemple de servlet/JSP VisualAge for Java (LeapYear).	55
Exportation des fichiers à partir de VisualAge for Java	55
Création d'un projet Web IBM WebSphere Studio Site Developer	56
Importation des fichiers de ressources des projets et des fichiers Java dans le projet IBM WebSphere Studio Site Developer	56
Définition des servlets et mise en oeuvre des modifications d'application restructurée	57
Création d'un projet serveur IBM WebSphere Studio Site Developer	57
Test de l'application LeapYear migrée	58
Exemple d'application Web WebSphere Studio "Classic" version 4.0 (YourCo) (Windows)	58
Lancement de WebSphere Studio "Classic" version 4.0 et création d'une phase de migration	59
Création d'un fichier descripteur de configuration Web	59
Création d'un fichier de migration.	59
Démarrage d'IBM WebSphere Studio Site Developer et importation du fichier WAR	59
Création d'un projet serveur IBM WebSphere Studio Site Developer	60
Test de l'application YourCo migrée	61

Chapitre 11. Bibliographie 63

Remarques 65

Informations relatives à l'interface de programmation	67
Marques	67

Chapitre 1. Présentation du guide de migration de WebSphere Application Server - Express

La version IBM WebSphere Application Server - Express version 5.1 permet d'effectuer une migration de code à partir de :

- VisualAge for Java,
- IBM WebSphere Application Single Server Edition version 3.5
- IBM WebSphere Application Server - Express versions 5.0, 5.0.1, 5.0.2
- WebSphere Studio "Classic"
- IBM WebSphere Studio Site Developer
- IBM WebSphere Studio Site Developer version 5 bêta, à disponibilité anticipée ou à disponibilité générale
- IBM WebSphere Studio Site Developer version 5.0.1
- IBM WebSphere Studio Site Developer version 5.1

WebSphere Application Server - Express 5.1 comprend WebSphere Application Server 5.1 et WebSphere Studio Site Developer 5.1.1. Le premier chapitre traite de la migration de la fonctionnalité du serveur de WebSphere Application Server - Express. Les autres chapitres du guide de migration expliquent la migration du code à partir de différentes versions de WebSphere Studio Site Developer.

Remarque importante relative à la migration du serveur :

La migration de la configuration du serveur est judicieuse uniquement si vous gérez le serveur à l'aide de la console d'administration (en général dans un environnement de production). Dans ce mode de fonctionnement, la configuration du serveur et les applications déployées sont stockées dans le répertoire config sur le serveur. Le processus de migration effectue la migration de ces fichiers de configuration et d'applications à votre place. Cependant, si vous utilisez WebSphere Studio Site Developer pour configurer et déployer des applications sur votre serveur distant, il n'est pas nécessaire de procéder à la migration des fichiers de configuration du serveur, car les fichiers de configuration et d'applications sont tous conservés dans l'espace de travail de Studio Site Developer. Studio Site Developer fera migrer l'espace de travail. Vous pourrez ensuite définir une nouvelle instance de serveur WebSphere Application Server - Express 5.1 et continuer de configurer et de déployer vos applications à partir de Studio Site Developer.

Le présent guide se compose des chapitres suivants :

- Migration du serveur de production
- Migration à partir d'IBM WebSphere Studio Site Developer version 5.1
- Migration à partir d'IBM WebSphere Studio Site Developer version 5 ou version 5.0.1
- Migration d'IBM WebSphere Studio Site Developer à partir de la version 4.0.x
-  Migration à partir de WebSphere Studio "Classic" vers IBM WebSphere Studio Site Developer
- Migration à partir de VisualAge for Java vers IBM WebSphere Studio Site Developer

- Migration à partir de VisualAge for Java Visual Composition Editor vers Visual Editor for Java
- Création de la configuration (fichiers JAR de bibliothèque, fichiers JAR dépendant d'un projet, compilations Ant)
- Exemples de migration
- Bibliographie

Vous trouverez des informations concernant l'utilisation de WebSphere Application Server - Express dans le *Guide d'initiation* et l'aide en ligne du produit. Avant d'installer WebSphere Application Server - Express, lisez le *Guide d'installation*. Après avoir installé WebSphere Application Server - Express, lisez le *Guide d'initiation* et exécutez les tutoriels associés. Ces derniers présentent le plan de travail, le développement pour Java et les services Web. Après avoir exécuté les tutoriels, lisez le présent guide qui décrit la migration des ressources de l'application dans WebSphere Application Server - Express.

Le présent guide est disponible aux formats HTML et Acrobat PDF dans le répertoire /readme. Les deux formats contiennent des informations identiques. Le fichier migrate.html peut être lu dans n'importe quel navigateur Web. Pour ouvrir le fichier migrate.pdf, vous devez installer le logiciel Acrobat Reader, qu'il est possible de télécharger à l'adresse www.adobe.com/products/acrobat/readstep2.html.

Le présent *Guide de migration* utilise les conventions Windows. Par exemple, "RépInstall_WS\" sous Windows équivaut à "RépInstall_WS/" sous Linux.

Pour obtenir les prochaines mises à jour, connectez-vous au site www.ibm.com/websphere/developer/zones/studio/transition.html.

Chapitre 2. Migration du serveur de production

Migration

La migration est une activité qui permet de profiter du matériel existant. Les tâches et les outils de migration vous aident à mettre à niveau le produit et ses conditions requises, à réutiliser des composants d'application existants, si possible, et à transférer les configurations administratives à partir de votre version précédente vers la version actuelle.

La migration des produits WebSphere Application Server consiste à exploiter les applications et les environnements existants et à les modifier de sorte qu'ils soient compatibles avec la version du produit la plus récente.

Les fonctions de migration du produit sont fournies par les outils de migration dans IBM WebSphere Application Server - Express, version 5.1. Ces derniers prennent en charge la migration à partir de :

- IBM WebSphere Application Server - Express version 5.0, 5.0.1, 5.0.2
- IBM WebSphere Application Single Server Edition version 3.5

L'assistant d'installation du produit détecte les versions précédentes d'IBM WebSphere Application Server - Express et vous propose de lancer les outils de migration lors de l'installation de la version 5.1. Pour procéder à la migration à partir d'IBM WebSphere Application Server version 3.5, vous devez lancer ces outils directement.

Redbook - Migration

Le document *Migrating to WebSphere V5: An End-to-End Migration Guide* est disponible sur le site Web des redbooks à l'adresse suivante : <http://www.ibm.com/redbooks>. Pour trouver le redbook, recherchez le numéro de document SG24-6910-00. Il couvre plus largement la fonction de migration que cet article et contient des informations détaillées relatives à la planification de la migration d'applications et des exemples et outils de WebSphere Studio Application Developer.

Migration de la version 3.5 : Vers le modèle J2EE

Les utilisateurs de la version 3.5.x qui passent à la version 5 vont utiliser une plateforme compatible avec les spécifications J2EE. La technologie J2EE différencie clairement le développement et la création d'applications de l'administration, du déploiement et de la gestion de ces applications. La migration à partir de la version 3.5 implique la modification des structures, du développement et du déploiement des applications.

Les outils de migration aident l'utilisateur à passer de la version 3.5.x à la version 5 en effectuant la migration des configurations système et en créant des artefacts J2EE ainsi que le mappage des rôles de sécurité J2EE. Ils créent des applications d'entreprise J2EE initiales en fonction des configurations de la version 3.5.x. Toutefois, étant donné les modifications considérables apportées aux structures des applications, prévoyez de tester et d'optimiser soigneusement les applications

migrées à l'aide des outils de développement et de déploiement, afin de déterminer exactement le mode de fonctionnement des applications dans le nouvel environnement.

Le modèle J2EE permet de développer des applications indépendamment de leur environnement de déploiement final. Cette séparation des tâches simplifie le processus de promotion d'une application du développement initial à la production, ou le processus de déplacement d'une application d'un serveur à un autre. L'idée est de modifier uniquement les paramètres de déploiement de l'application et non son code.

Migration et coexistence

Avant de commencer, déterminez si une version de WebSphere Application Server est déjà installée sur la machine sur laquelle vous prévoyez d'installer la version 5.1 du produit. Si tel est le cas, vous devez envisager de copier la configuration et les applications de la version précédente vers la nouvelle version, c'est-à-dire de procéder à la *migration*. La migration ne désinstalle pas la version précédente. Cette dernière restera opérationnelle. Si vous l'exécutez en même temps que l'installation de la version 5.1, les deux versions *coexisteront*. Pour pouvoir exécuter les deux versions simultanément, vous devez configurer les ports de sorte qu'ils n'entrent pas en conflit. L'opération de migration effectuée la migration des définitions de port telles quelles, ce qui signifie qu'elles sont identiques pour les deux versions. WebSphere Application Server contient des outils de migration qui composent la fonction de migration. L'assistant de migration appelle les outils de migration. Vous pouvez également les appeler manuellement ultérieurement.

Globalement, la migration à partir d'IBM WebSphere Application Server - Express version 5.0.x vers la version 5.1 est très simple. Vous pouvez utiliser le programme d'installation pour procéder à la migration et avoir peu de réglages à effectuer après la migration, voire aucun. Vous pouvez aussi utiliser les outils de migration manuellement pour sauvegarder les données de configuration des versions 5.0.0, 5.0.1 ou 5.0.2, désinstaller les versions 5.0.0, 5.0.1 ou 5.0.2, installer la version 5.1 et restaurer les données de configuration.

En résumé, la migration à partir d'IBM WebSphere Application Server version 3.5 vers IBM WebSphere Application Server - Express version 5.1 implique des modifications significatives dans les structures, le développement et le déploiement des applications. Les outils de migration aident l'utilisateur à faire la transition en effectuant la migration des configurations système et en créant des artefacts J2EE ainsi qu'en mappant les paramètres de sécurité précédents aux rôles de sécurité J2EE. Ces mappages de sécurité permettent d'accéder aux éléments migrés lors de la transition. Les outils de migration créent des applications d'entreprise J2EE initiales en fonction des configurations de la version 3.5.X. Toutefois, étant donné les modifications considérables apportées aux structures des applications, prenez soin de tester et d'optimiser les applications migrées à l'aide des outils de développement et de déploiement.

La migration sauvegarde les fichiers énumérés ci-dessous dans le répertoire backup.

Pour la version 3.5.x

- bin/setupCmdLine.sh (ou bin/setupCmdLine.bat sur les plateformes Windows)
- classes (non sauvegardé pour iSeries)
- hosts

- properties
- servlets

Pour la version 5.0.x

- classes (non sauvegardé pour iSeries)
- config
- installableApps
- installedApps
- properties

Outils de migration

La présente section décrit les outils de migration fournis par WebSphere Application Server. Tous les outils de migration se trouvent dans le répertoire /migration sur le CD-ROM du produit. Il est important d'utiliser les outils de migration de la version d'Application Server que vous installez. Les outils sont modifiés au fur et à mesure des versions. Ceux qui figurent sur le CD-ROM du produit fournissent la fonction nécessaire à la migration d'une version précédente d'Application Server vers la version figurant sur le CD-ROM. Les outils du CD-ROM correspondent au produit figurant sur le CD-ROM. Si vous utilisez les outils de migration d'une version antérieure d'Application Server, vous risquez de rencontrer des erreurs lors de la migration.

WASPreUpgrade.sh (et WASPreUpgrade.bat)

Sauvegarde les applications et les données de configuration d'une installation précédente de WebSphere Application Server dans un répertoire de sauvegarde. Le script WASPostUpgrade restaure les données de configuration à partir de ce répertoire dans la nouvelle installation. Le programme d'installation appelle le script WASPreUpgrade.sh lors de l'installation, si vous choisissez de procéder à la migration. Vous pouvez également utiliser la commande permettant d'effectuer une migration manuelle, après l'installation de la nouvelle version.

WASPostUpgrade.sh (et WASPostUpgrade.bat)

Restaure les données de configuration d'une version précédente. WASPostUpgrade lit les données stockées par WASPreUpgrade dans le répertoire de sauvegarde. Le programme d'installation appelle le script WASPostUpgrade.sh lors de l'installation, si vous choisissez de procéder à la migration. Vous pouvez également utiliser la commande permettant d'effectuer une migration manuelle, après l'installation de la nouvelle version.

Commande WASPreUpgrade

Le script WASPreUpgrade.sh (ou WASPreUpgrade.bat) est un outil de migration de la configuration et des applications des versions ou des éditions précédentes vers la version 5.1 d'Application Server - Express.

Le fichier de la commande se trouve dans le sous-répertoire AppServer/bin du répertoire principal de l'installation après l'installation. Vous pouvez également y accéder directement à partir du sous-répertoire migration figurant sur le CD.

Syntaxe

```
WASPreUpgrade répertoireSauvegarde répertoireWasActuel  
[nomNoeudAdmin]  
[-nameServiceHost nom_hôte [-nameServicePort numéro_port ]]  
[-traceString spéc_trace [-traceFile nom_fichier ]]
```

Paramètres

Les deux premiers arguments sont requis et positionnels.

Certains arguments pris en charge sont décrits ci-dessous.

répertoireSauvegarde

Nom requis et positionnel du répertoire dans lequel l'outil WASPreUpgrade stocke les fichiers et la configuration sauvegardés et à partir duquel l'outil WASPostUpgrade lit ces données. L'outil WASPreUpgrade crée ce répertoire s'il n'existe pas.

répertoireWASActuel

Nom requis et positionnel du répertoire principal de l'installation pour l'installation de la version 3.5.x ou 5.0.x actuelle. Il peut s'agir de WebSphere Application Server Standard Edition version 3.5.x, WebSphere Application Server - Express version 5.0.x.

nomNoeudAdmin

Nom facultatif et positionnel du noeud contenant le serveur d'administration pour le produit installé. La valeur de cet argument doit correspondre au nom du noeud donné dans l'arborescence de topologie de l'onglet Topologie dans la console d'administration du produit installé. L'outil WASPreUpgrade appelle l'outil XMLConfig avec ce paramètre. Ce dernier est requis uniquement lors de la mise à niveau à partir de WebSphere Application Server Standard Edition version 3.5.x.

-nameServiceHost -nameServicePort

S'ils sont spécifiés, l'outil WASPreUpgrade transmet ces paramètres facultatifs à l'outil XMLConfig. Ils permettent de remplacer le nom d'hôte et le numéro de port par défaut utilisés par l'outil XMLConfig.

-traceString -traceFile

Paramètres facultatifs permettant de rassembler des informations de trace pour le service de maintenance IBM. Indiquez la spécification de trace (*spéc_trace*) `"*=all=enabled"` (avec les guillemets) pour regrouper toutes les informations de trace.

Journalisation

L'outil WASPreUpgrade affiche son statut sur l'écran lors de son exécution. Il sauvegarde également un ensemble d'informations de journal plus large dans le répertoire *backup*. Vous pouvez ouvrir le fichier `WASPreUpgrade.log` avec un éditeur de texte.

Commande WASPostUpgrade

Le script `WASPostUpgrade.sh` (ou `WASPostUpgrade.bat`) est un outil de migration de la configuration et des applications des versions ou des éditions précédentes vers la version 5.1 d'Application Server - Express.

Le fichier de la commande se trouve dans le répertoire `AppServer/bin` du répertoire principal de l'installation.

L'outil `WASPostUpgrade` installe toutes les applications migrées dans le répertoire `AppServer/installedApps` de l'installation de la version 5.1. Il inclut des applications provenant du répertoire de sauvegarde créé par l'outil `WASPreUpgrade`. L'outil `WASPreUpgrade` copie les applications du répertoire `installedApps` et d'autres répertoires de la version ou de l'édition précédente.

Syntaxe

```
WASPostUpgrade répertoireSauvegarde
                [-serverName nom_serveur]
                [-webModuleAdditionalClasspath chemin_classes]
                [-documentRootLimit nombre]
                [-substitute "clé1=valeur1[;clé2=valeur2;...]"]
                [-portBlock numéro_départ_port]
                [-backupConfig true | false]
                [-replacePorts true | false]
                [[-traceString spéc_trace [-traceFile nom_fichier]]]
```

Paramètres

Le premier argument est obligatoire. Certains arguments pris en charge sont décrits ci-dessous.

serverName

Nom de l'instance du serveur facultatif. Valeur par défaut : `server1`.

répertoireSauvegarde

Nom requis du répertoire dans lequel l'outil `WASPreUpgrade` stocke les fichiers et la configuration sauvegardés et à partir duquel l'outil `WASPostUpgrade` lit ces données. L'outil `WASPreUpgrade` crée ce répertoire s'il n'existe pas.

-backupConfig

Paramètre facultatif utilisé pour sauvegarder la configuration existante avant que les outils de migration ne la modifie. La valeur par défaut `true` permet de sauvegarder la configuration.

-documentRootLimit

Paramètre facultatif permettant de spécifier le nombre de fichiers que le programme copie à partir du répertoire principal des documents de l'application Web. Il est valable uniquement pour les mises à niveau de la version 3.5.x. S'il n'est pas spécifié, la valeur par défaut est 300.

-portBlock

Paramètre facultatif permettant de spécifier la valeur de départ à utiliser lors de la création des ports.

-substitute

Argument facultatif transmis à l'outil `XMLConfig`. Spécifiez des valeurs pour les variables de sécurité à substituer (exemple : `-substitute "NODE_NAME=noeud_admin;APP_SERVER=serveur_défaut"`).

Dans le fichier de données XML en entrée, chaque clé apparaît sous la forme *\$clé\$* pour la substitution. Cet argument remplace toute occurrence de *\$NODE_NAME\$* par *noeud-admin* et de *\$APP_SERVER\$* par *serveur_défaut* dans le fichier XML des entrées.

Si la chaîne de substitution contient des points-virgules, utilisez *\$semiColon\$* pour les distinguer du délimiteur ";". Sur les plateformes UNIX, ajoutez un caractère d'échappement devant chaque signe dollar (\$) dans la chaîne de substitution (exemple : *\\$semiColon\\$*).

Ce paramètre est valable pour les configurations sauvegardées à partir de Advanced Edition, version 3.5.x.

-traceString -traceFile

Paramètres facultatifs permettant de rassembler des informations de trace pour le service de maintenance IBM. Indiquez la spécification de trace (spéc_trace) *"*=all=enabled"* (avec les guillemets) pour regrouper toutes les informations de trace.

-webModuleAdditionalClasspath

Paramètre facultatif permettant d'indiquer le chemin ou le nom des chemins et des fichiers de répertoires ou de fichiers spécifiques à ne pas copier dans le fichier d'archive Web (WAR). A la place, le programme ajoute les chemins et les fichiers à l'attribut `additionalClassPath` de l'extension du module Web (`ibm-web-ext.xmi`). Ce paramètre est valide uniquement lors de la migration d'une installation de la version 3.5.x.

Journalisation

L'outil WASPostUpgrade affiche son statut sur l'écran lors de son exécution. Il sauvegarde également un ensemble plus large d'informations de journal dans le répertoire `logs`. Vous pouvez ouvrir le fichier `WASPostUpgrade.log` avec un éditeur de texte.

Mappage de la configuration lors de la migration

La présente section décrit les modifications apportées lors de la migration, qui implique toujours une machine unique, comme un environnement de développement sur une machine autonome.

Migration de la version 3.5 vers la version 5.x

Les outils de migration aident l'utilisateur à passer de la version 3.5.x à la version 5 en effectuant la migration des configurations système et en créant des artefacts J2EE ainsi que le mappage des rôles de sécurité J2EE. Ils créent des applications d'entreprise J2EE initiales en fonction des configurations de la version 3.5.x. Toutefois, étant donné les modifications considérables apportées aux structures des applications, prévoyez de tester et d'optimiser soigneusement les applications migrées à l'aide des outils de développement et de déploiement, afin de déterminer exactement le mode de fonctionnement des applications dans la version 5.

Consultez le fichier `WASPostUpgrade.log` pour obtenir des informations détaillées sur les beans enterprise migrés. Le modèle de programmation J2EE spécifie une architecture précisant le mode de création et de déploiement des applications. Etant donné que les applications de la version 3.5.x ne sont pas construites selon la même architecture, l'outil WASPostUpgrade les crée à nouveau. Il crée toutes les ressources Web et les beans enterprise migrés dans des applications J2EE. Il mappe toutes les

applications d'entreprise de l'installation de la version 3.5.x vers les applications J2EE du même nom, déployées sur le même serveur.

L'outil WASPostUpgrade mappe les ressources Web qui ne font pas partie d'une application d'entreprise vers une application J2EE par défaut qui comporte le nom du serveur. L'outil mappe les applications Web vers des fichiers WAR J2EE. Il combine les ressources dans un fichier EAR J2EE qu'il déploie dans la configuration de la version 5.

Détails relatifs au mappage pour la migration à partir de la version 3.5.x vers la version 5.x

- **datasources.xml**

Vous pouvez utiliser le fichier `datasources.xml` d'une version 3.5.x pour multiplier les paramètres de configuration de la source de données. La version 3.5.x stocke le fichier dans le répertoire `properties`. Les outils de migration procèdent à la migration d'un fichier `datasources.xml` existant en fusionnant les propriétés du fichier dans la configuration du pilote JDBC et de la source de données.

- **Applications d'entreprise**

Les entrées entreprise-application dans la version 3.5.x sont facultatives. Elles permettent, le plus souvent, d'assembler des ensembles d'objets pour les définitions de sécurité. Les parties web-application et bean entreprise de l'entrée entreprise-application renvoient à leurs entrées respectives dans d'autres parties du fichier xml. Chaque entrée entreprise-application est traitée en vue de créer une application J2EE du même nom. Les entrées de bean entreprise et Web-application sont utilisés comme pointeurs vers les définitions des beans entreprise et des applications Web. Les détails de ces entrées sont ensuite utilisés pour créer une application J2EE.

Dans les fichiers de beans entreprise, la définition JAR-file permet de trouver les fichiers JAR en vue de leur redéploiement et de leur ajout à l'application J2EE. Les entrées Web-application document-root permettent de trouver les ressources utilisées dans l'application Web (HTML, pages JSP, etc). Ces fichiers sont copiés dans le fichier WAR de l'application J2EE. Les entrées du chemin d'accès Web-application permettent de trouver les servlets et les fichiers JAR qui sont copiés dans le fichier WAR de l'application J2EE.

Les applications d'entreprise sont créées lors de la migration à partir de la version 3.5.x. Elles sont compatibles avec la spécification J2EE 1.2 et contiennent les modules de niveau JSP 1.1 et de niveau de servlet 2.2. La compatibilité qui en résulte est simple et permet l'interopérabilité avec des versions antérieures de WebSphere Application Server.

- **Sécurité J2EE**

Le modèle d'autorisation de sécurité de la version 3.5.x repose sur la notion d'application d'entreprise et de groupes de méthodes. Le résultat de l'association de l'application d'entreprise et des groupes de méthodes est un droit d'accès WebSphere Application Server. La spécification J2EE inclut un modèle d'autorisation fondé sur les rôles.

Pour convertir le modèle d'autorisation de WebSphere Application Server version 3.5.x en modèle d'autorisation fondé sur les rôles de la version 5, les outils de migration créent un mappage un-à-un entre l'autorisation WebSphere Application Server et un nouveau rôle associé à cette application. Ainsi, pour chaque application d'entreprise et pour chaque groupe de méthodes de la version 3.5.x, les outils de migration créent un rôle dans la version 5 défini dans le descripteur de

déploiement de l'application J2EE. Les sujets autorisés pour chaque rôle sont définis dans la table d'autorisations disponible dans la liaison de l'application J2EE.

La spécification J2EE inclut un modèle d'autorisation fondé sur les rôles. WebSphere Application Server interprète le rôle sous la forme d'une série d'autorisations permettant d'accéder à une ressource. Lors de l'appel d'une méthode de bean enterprise, l'autorisation d'accéder à la méthode sur un bean particulier est définie par un droit d'accès à la méthode. Ce droit d'accès à la méthode est associé à un ou plusieurs rôles dans le descripteur de déploiement du fichier JAR du bean.

Lors de l'accès aux ressources Web, le droit d'accéder à un URI Web et d'appeler une méthode HTTP sur cet URI est défini sous la forme de collections de ressources Web et de contraintes de sécurité dans la spécification J2EE. Le descripteur de déploiement du fichier WAR de l'application Web contient les contraintes de sécurité et les collections de ressources Web.

- **Niveaux JSP**

La version 5 exécute les objets JSP 1.0 et JSP 1.1 comme des objets JSP 1.2, qui est le seul niveau pris en charge.

- **Redirecteur de servlet**

La version 5 ne prend pas en charge le redirecteur de servlet des versions précédentes. Les outils de migration ignorent ces objets.

- **Le nom des modules de servlet est modifié lors de la migration de la version 3.5.x vers la version 5.**

Si la configuration de la version 3.5.x définit le servlet SimpleFileServlet, la migration du servlet n'est pas effectuée. Les outils de migration associent la valeur `true` à l'attribut `FileServingEnabled` dans le fichier `ibm-web-ext.xmi` du module Web.

Si la configuration de la version 3.5 définit le servlet `InvokerServlet`, la migration du servlet n'est pas effectuée. Les outils de migration associent la valeur `true` à l'attribut `ServeServletsByClassnameEnabled` dans le fichier `ibm-web-ext.xml` du module Web.

Si la configuration de la version 3.5 définit le servlet `DefaultErrorReporter`, la migration du servlet est effectuée dans le fichier `web.xml` du module Web. La migration utilise le nouveau module pour définir le nom de la classe.

- **Transferts**

Le type de transfert par défaut du moteur de servlet dans la version 3.5.x correspond à la technologie OSE (Open Servlet Engine). Dans la mesure où la version 5 ne prend plus en charge le transfert OSE, les outils de migration mappent ces transferts vers les transferts HTTP en affectant les mêmes ports. Vous devez ajouter manuellement les entrées d'alias `VirtualHost` pour chaque port.

Migration manuelle des données de configuration

Vous pouvez effectuer la migration des configurations d'administration à l'aide de l'assistant d'installation ou manuellement, conformément aux instructions ci-après. Si vous optez pour une migration manuelle, ne cochez pas la case de migration dans le panneau de migration de l'assistant d'installation.

Si vous utilisez une version antérieure de WebSphere Application Server, il se peut que l'administrateur système ait déjà optimisé plusieurs paramètres de serveur et

d'application pour votre environnement. Il est important d'élaborer une stratégie de migration de ces paramètres avec une efficacité maximale et une perte minimale.

Vous pouvez effectuer une migration manuelle progressive en appelant les outils de migration plusieurs fois et en spécifiant à chaque fois un fichier de configuration différent. Plusieurs raisons justifient l'existence de fichiers de configuration multiples. Quel que soit le cas, la migration des fichiers de configuration un par un permet de tester les applications de manière progressive avant de passer au fichier de configuration suivant.

Avant d'utiliser les outils de migration, consultez le document Notes sur l'édition version 5.x pour déterminer les correctifs à appliquer aux versions antérieures. L'application de ces correctifs à une version antérieure peut également affecter les fichiers ayant un rôle dans la migration. Appliquez tous les correctifs pour que la migration des configurations et des applications soit la plus efficace possible.

La migration manuelle permet une approche plus progressive de la migration que la migration totale effectuée à l'aide de l'assistant d'installation. IBM met à disposition un ensemble d'outils de migration pour la migration des configurations d'administration vers le produit WebSphere Application Server - Express à partir de n'importe quelle édition des versions 3.5.x ou 5.0.x. Le processus global de migration consiste à sauvegarder la configuration actuelle et les fichiers nécessaires avec l'outil de migration WASPreUpgrade, à désinstaller l'édition précédente, à installer la version 5 du produit sans sélectionner l'option de migration automatique et à restaurer la configuration de l'édition précédente avec l'outil de migration WASPostUpgrade.

Sélectionnez l'un des scénarios de migration suivants pour plus d'informations sur la migration des données de configuration vers un noeud de base WebSphere Application Server :

- «Migration à partir de la version 3.5.x vers la version 5.1»
- «Migration de la version 3.5.x vers une version 5.1 sur une machine éloignée» à la page 13
- «Migration à partir de la version 5.0 vers la version 5.1» à la page 16
- «Migration de la version 5.0.x vers une version 5.1 sur une machine éloignée» à la page 16
- «Migration à partir d'un système d'exploitation non pris en charge» à la page 18

Migration à partir de la version 3.5.x vers la version 5.1

Vous pouvez utiliser les outils de migration pour effectuer la migration des données de configuration à partir de la version 3.5 de WebSphere Application Server vers la version 5.1 de WebSphere Application Server - Express.

A priori, vous utilisez les outils de migration WASPreUpgrade et WASPostUpgrade de la version 5.1 de WebSphere Application Server pour procéder à la mise à niveau à partir de la version 3.5 vers la version 5.1 sur la *même* machine. Si votre scénario inclut la migration d'une configuration de version 3.5, se trouvant sur une certaine machine, vers WebSphere Application Server - Express version 5.1, sur une autre machine, suivez la procédure décrite à la section «Migration de la version 3.5.x vers une version 5.1 sur une machine éloignée» à la page 13.

La présente section décrit l'utilisation des outils de migration de la version 5.1 en vue de la migration de :

- WebSphere Application Server Single Server Edition version 3.5

L'outil WASPreUpgrade sauvegarde la configuration de la version 3.5 existante dans un répertoire *sauvegarde-spécifique-migration*. L'outil WASPostUpgrade utilise ce répertoire pour ajouter les anciens paramètres de configuration au nouvel environnement de la version 5.1.

Étapes de la procédure

1. Procurez-vous le CD-ROM du produit, version 5.1.
Sur ce CD se trouve le répertoire *migration/bin*. Il contient un environnement spécial que vous pouvez utiliser pour lancer l'outil WASPreUpgrade sans installer la version 5.1.
2. Sauvegardez la configuration actuelle à l'aide du script WASPreUpgrade à partir du répertoire */migration/bin* du CD-ROM de la version 5.1 du produit. Sauvegardez la configuration dans le répertoire *sauvegarde-spécifique-migration* :

```
WASPreUpgrade /usr/tmp/sauvegarde-spécifique-migration /usr/websphere/appserver  
nomDeVotrenomDeNoeud
```

Vérifiez que le serveur d'administration de l'environnement existant est en cours d'exécution. L'outil WASPreUpgrade indique son statut à l'écran et dans des fichiers journaux dans le répertoire *sauvegarde-spécifique-migration*. Les noms de fichiers journaux ASCII commencent par le texte WASPreUpgrade et comprennent un horodatage.

L'outil WASPreUpgrade enregistre tous les fichiers à partir des répertoires ci-dessous dans la configuration de la version 3.5.x existante dans le répertoire de sauvegarde.

Pour la version 3.5.x

- bin
- classes
- hosts
- properties
- servlets

L'outil WASPreUpgrade sauvegarde les fichiers sélectionnés dans le répertoire */bin* de la version 3.5.x. Il exporte également la configuration du serveur d'applications existante à partir du référentiel de la version 3.5.x. L'outil WASPreUpgrade appelle XMLConfig en vue de l'exportation du référentiel de la version 3.5 existant dans le fichier *websphere_backup.xml* du répertoire *sauvegarde-spécifique-migration*.

Si des erreurs surviennent lors de l'exécution de l'outil WASPreUpgrade, il se peut que vous deviez appliquer des correctifs à l'installation de la version 3.5 pour que l'étape d'exportation se termine correctement. Consultez la page de support IBM pour obtenir la liste des derniers correctifs applicables. Si vous affichez ces informations à partir de l'infocenter, cliquez sur **Support** pour ouvrir la page de support IBM.

3. Installez la version 5.1 du produit WebSphere Application Server - Express.
Ne sélectionnez pas l'option de migration, si elle apparaît.
Après chaque utilisation de WASPostUpgrade, vérifiez les paramètres du port de la version 5 dans les deux fichiers ci-dessous.

- Vérifiez l'attribution du port `BOOTSTRAP_ADDRESS` pour `server1` dans le fichier `serverindex.xml`.

Si le port `BOOTSTRAP_ADDRESS` de la version précédente est 900, le processus de migration le mappe à 7809. Si le port `BOOTSTRAP_ADDRESS` de la version précédente n'est pas 900, le processus de migration mappe la valeur à `server1` dans une migration Advanced Edition, ou au nom du serveur actuel dans une migration Advanced Single Server Edition.

- Vérifiez les attributions du port de transfert HTTP dans le fichier `server.xml`.

Le traitement `WASPostUpgrade` ajoute les ports de transfert HTTP de la version précédente au fichier `server.xml` de la version 5. Cela signifie que `server1` comporte des attributions de port du transfert HTTP doubles, provenant à la fois du panneau de coexistence et du *serveur par défaut* de la version précédente.

4. Procédez à la migration de la configuration précédente vers la nouvelle installation avec l'outil `WASPostUpgrade` dans le répertoire `AppServer/bin` du répertoire principal d'installation de la version 5.1.

L'outil `WASPostUpgrade` effectue la migration des informations de configuration de la version 3.5.x créées par l'outil `WASPreUpgrade` vers l'installation de la version 5.1. Etant donné que la version 5.1 est conforme au modèle de programmation J2EE, ce qui n'est pas le cas de la version 3.5, d'importantes modifications sont nécessaires pour appliquer la configuration de la version 3.5.x à une installation de la version 5.

L'outil `WASPostUpgrade` n'effectue pas la migration des exemples ou de l'application de la console d'administration car la version 5.1 en prévoit déjà.

L'outil `WASPostUpgrade` enregistre les informations détaillées spécifiques de chaque bean entreprise qu'il déploie, dans le fichier `WASPostUpgrade.log`.

5. Arrêtez le serveur d'administration de la version précédente s'il est en cours d'exécution, avant d'exécuter le noeud de la version 5.
6. La configuration de WebSphere Application Server après la migration permet de vérifier les résultats des outils de migration. Vous pouvez également consulter la rubrique Mappage de configuration lors de la migration pour vérifier les résultats de la migration. Elle comporte une description détaillée de la façon dont les outils de migration font migrer les objets et des points à vérifier.

Migration de la version 3.5.x vers une version 5.1 sur une machine éloignée

Vous pouvez utiliser les outils de migration pour effectuer une migration manuelle entre deux machines.

A priori, vous utilisez les outils de migration `WASPreUpgrade` et `WASPostUpgrade` de la version 5.1 de WebSphere Application Server - Express pour procéder à la mise à niveau à partir de la version 3.5 vers la version 5.1 sur la *même* machine.

Toutefois, dans certains cas, il est nécessaire d'effectuer la migration d'une configuration de version 3.5 à partir d'une machine, vers une version 5.1 située sur une autre machine. L'un des scénarios consiste à installer de nouvelles machines pour votre environnement de version 5.1 le plus récent et de procéder à la migration de votre configuration de version 3.5 existante sur d'autres machines.

La présente section décrit l'utilisation des outils de migration de la version 5.1 en vue de la migration de :

- WebSphere Application Server Single Server Edition version 3.5

L'outil WASPreUpgrade sauvegarde la configuration de la version 3.5 existante dans un répertoire *sauvegarde-spécifique-migration*. L'outil WASPostUpgrade utilise ce répertoire pour ajouter les anciens paramètres de configuration au nouvel environnement de la version 5.1.

Étapes de la procédure

1. Procurez-vous le CD-ROM du produit, version 5.1.

Sur ce CD se trouve le répertoire *migration/bin*. Il contient un environnement spécial que vous pouvez utiliser pour lancer l'outil WASPreUpgrade sans installer la version 5.1.

2. Sauvegardez la configuration actuelle à l'aide du script WASPreUpgrade à partir du répertoire */migration/bin* du CD-ROM de la version 5.1 du produit, que vous devez monter sur la machine de la version 3.5.

Sauvegardez la configuration dans le répertoire *sauvegarde-spécifique-migration* sur la machine de la version 3.5 :

```
WASPreUpgrade /opt/tmp/sauvegarde-spécifique-migration /opt/websphere/appserver  
nomNoeudAdmin
```

Vérifiez que le serveur d'administration de l'environnement existant est en cours d'exécution.

L'outil WASPreUpgrade indique son statut à l'écran et dans des fichiers journaux dans le répertoire */sauvegarde-spécifique-migration*. Les noms de fichiers journaux ASCII commencent par le texte WASPreUpgrade et comprennent un horodatage.

L'outil WASPreUpgrade sauvegarde les fichiers sélectionnés dans le répertoire */bin* de la version 3.5.x. Il exporte également la configuration du serveur d'applications existante à partir du référentiel de la version 3.5.x. L'outil WASPreUpgrade appelle XMLConfig en vue de l'exportation du référentiel de la version 3.5 existant dans le fichier *websphere_backup.xml* du répertoire *sauvegarde-spécifique-migration*.

Si des erreurs surviennent lors de l'exécution de l'outil WASPreUpgrade, il se peut que vous deviez appliquer des correctifs à l'installation de la version 3.5 pour que l'étape d'exportation se termine correctement. Consultez la page de support IBM pour obtenir la liste des derniers correctifs applicables. Si vous affichez ces informations à partir de l'infocenter, cliquez sur **Support** pour ouvrir la page de support IBM.

3. Copiez le répertoire */sauvegarde-spécifique-migration* de la machine de la version 3.5 sur la machine de la version 5.1.

Utilisez ftp, un emplacement de stockage partagé ou un autre mécanisme pour copier le fichier sur la nouvelle machine.

Effectuez les opérations ci-dessous sur la machine sur laquelle est installé la version 5.1 de WebSphere Application Server - Express.

4. Copiez le fichier */sauvegarde-spécifique-migration/websphere_backup.xml* ou */sauvegarde-spécifique-migration/config/server-cfg.xml* et stockez-le à l'emplacement de votre choix pour conserver la copie en tant qu'archive.

Vous copiez le fichier afin de pouvoir modifier la version originale à l'étape suivante.

5. Modifiez le fichier `/sauvegarde-spécifique-migration/websphere_backup.xml` ou `/sauvegarde-spécifique-migration/config/server-cfg.xml` en vue de corriger les paramètres dépendants de la machine.

Apportez les modifications ci-dessous au fichier.

- a. **Modifiez le nom du noeud** dans le fichier `/sauvegarde-spécifique-migration/websphere_backup.xml`. Il n'y a pas de nom de noeud dans le fichier `/sauvegarde-spécifique-migration/config/server-cfg.xml`.

Si vous utilisez le nom de noeud de la configuration originale de la version 3.5 pour la machine de la version 5.1, ne modifiez pas le nom. Dans les autres cas, vous **devez** remplacer toutes les occurrences du nom de noeud de la version 3.5 par le nom de noeud que vous utilisez pour WebSphere Application Server version 5.1. Le nom de noeud apparaît dans plusieurs sections XML du fichier. Si vous ne modifiez pas toutes les occurrences, la migration des données sera incomplète.

- b. **Modifiez les noms des chemins** dans le fichier `/sauvegarde-spécifique-migration/websphere_backup.xml` ou `/sauvegarde-spécifique-migration/config/server-cfg.xml`.

Le fichier de configuration renvoie aux noms des chemins dans plusieurs sections XML du fichier. Mettez à jour toute référence à un fichier se trouvant hors de la structure de répertoires de la version 3.5 en indiquant le répertoire équivalent sur la nouvelle machine, même si vous devez créer ce dernier. Si vous configurez un environnement correspondant, vous devrez sans doute copier le répertoire d'origine sur la machine hébergeant la version 5.1. Il se peut aussi que vous deviez installer le logiciel approprié.

- c. **Corrigez les styles de spécifications** pour les noms des chemins qui dépendent du système d'exploitation.

Vous devez corriger les spécifications des chemins s'ils diffèrent de ceux qui fonctionnent sur la machine exécutant la version 5.1. Par exemple, si vous passez de la version 3.5 sur une plateforme Windows à la version 5.1 sur une plateforme Linux, remplacez les chemins spécifiques de Windows dans le fichier de configuration par le style de chemin de Linux. Remplacez `"c:\mon_répertoire\un_chemin"` par `"/opt/mon_répertoire/un_chemin"`.

- d. **Modifiez les ID utilisateur et les mots de passe pour qu'ils correspondent aux conditions de sécurité requises.**

Il se peut que vous deviez changer les ID utilisateur et les mots de passe s'ils ne sont pas identiques à ceux utilisés sur la machine hébergeant la version 5.1.

Pour remplacer un mot de passe codé par un mot de passe apparaissant en clair, remplacez `<password>{xor}LCoxayht</password>` par `<password>mon_mot_de_passe</password>`.

- e. **Modifiez les autres informations spécifiques de la machine.**

La configuration peut renvoyer à d'autres produits logiciels ou à d'autres configurations n'existant pas sur la nouvelle machine. L'ancienne machine, par exemple, peut disposer d'une base de données. La configuration de la version 5.1 doit, si possible, continuer de référencer la base de données sur l'ancienne machine. Modifiez la source de données de sorte qu'elle pointe vers la base de données située sur la machine hébergeant la version 3.5.

6. Installez la version 5.1 de WebSphere Application Server sans sélectionner l'option de migration.
7. Ajoutez la configuration de la version 3.5 à la configuration de la version 5.1.

Utilisez l'outil WASPostUpgrade situé dans le répertoire AppServer/bin du répertoire principal d'installation de la version 5.1 pour ajouter la configuration de la version 3.5 à la configuration de la version 5.1.

```
WASPostUpgrade /opt/tmp/sauvegarde-spécifique-migration
```

L'outil WASPostUpgrade enregistre les informations détaillées spécifiques de chaque bean enterprise qu'il déploie dans le fichier /migration-specific-backup/WASPostUpgrade.log.

Migration à partir de la version 5.0 vers la version 5.1

Vous pouvez utiliser le programme d'installation de la version 5.1 pour effectuer la migration à partir de la version 5.0.x de WebSphere Application Server - Express vers la version 5.1.

Étapes de la procédure

1. Arrêtez le serveur d'applications de la version 5.0.x.

Utilisez le script stopServer.sh (ou stopServer.bat) situé dans le répertoire AppServer/bin du répertoire principal de l'installation :

```
stopServer.sh server1
```

Vous pouvez procéder à la migration d'un noeud de version 5.0.x sans l'arrêter. Cependant, l'exécution du noeud lors de la migration de sa configuration n'est pas requise. Les outils de migration peuvent extraire toutes les données de configuration lorsque le noeud est arrêté. Vous devez arrêter le noeud pour pouvoir redémarrer le noeud de version 5.1 en cours d'installation. Vous pouvez donc arrêter le noeud maintenant.

2. Installez la version 5.1 du produit.

Sélectionnez l'option de migration, lorsqu'elle apparaît.

3. Vérifiez l'installation de la version 5.1 d'Application Server.

Utilisez l'outil Premiers pas lorsqu'il apparaît à la fin du processus d'installation du produit ou lancez le test de vérification de l'installation vous-même, si l'outil Premiers pas n'est pas disponible.

Vous pouvez désinstaller, si vous le voulez, la version 5.0.x du serveur.

Migration de la version 5.0.x vers une version 5.1 sur une machine éloignée

Vous pouvez utiliser les outils de migration pour effectuer une migration entre deux machines.

Étapes préalables

A priori, vous utilisez les outils de migration WASPreUpgrade et WASPostUpgrade de la version 5.1 de WebSphere Application Server pour procéder à la mise à niveau à partir de la version 5.0.x vers la version 5.1 sur la *même* machine.

Toutefois, dans certains cas, il est nécessaire d'effectuer la migration d'une configuration de version 5.0.x à partir d'une machine, vers une version 5.1 située sur une autre machine. L'un des scénarios consiste à installer de nouvelles machines pour votre environnement de version 5.1 le plus récent et de procéder à la migration de votre configuration de version 5.0.x existante sur d'autres machines.

La présente section décrit l'utilisation des outils de migration de la version 5.1 en vue de la migration.

L'outil WASPreUpgrade sauvegarde la configuration de la version 5.0.x existante dans un répertoire *sauvegarde-spécifique-migration*. L'outil WASPostUpgrade utilise ce répertoire pour ajouter les anciens paramètres de configuration au nouvel environnement de la version 5.1.

Etapes de la procédure

1. Procurez-vous le CD-ROM du produit, version 5.1.
Sur ce CD se trouve le répertoire *migration/bin*. Il contient un environnement spécial que vous pouvez utiliser pour lancer l'outil WASPreUpgrade sans installer la version 5.1.
2. Sauvegardez la configuration actuelle à l'aide du script WASPreUpgrade à partir du répertoire */migration/bin* du CD-ROM de la version 5.1 du produit, que vous devez monter sur la machine de la version 5.0.x.

Sauvegardez la configuration dans le répertoire *sauvegarde-spécifique-migration* sur la machine de la version 5.0.x :

```
WASPreUpgrade /opt/tmp/sauvegarde-spécifique-migration /opt/websphere/appserver
```

L'outil WASPreUpgrade indique son statut à l'écran et dans des fichiers journaux dans le répertoire */sauvegarde-spécifique-migration*. Les noms de fichiers journaux ASCII commencent par le texte "WASPreUpgrade" et comprennent un horodatage.

3. Copiez le répertoire */sauvegarde-spécifique-migration* de la machine de la version 5.0.x sur la machine de la version 5.1.
Utilisez ftp, un emplacement de stockage partagé ou un autre mécanisme pour copier le fichier sur la nouvelle machine.
4. Installez la version 5.1 de WebSphere Application Server sans sélectionner l'option de migration.
5. Ajoutez la configuration de la version 5.0.x à la configuration de la version 5.1.
Utilisez l'outil WASPostUpgrade situé dans le répertoire *AppServer/bin* du répertoire principal d'installation de la version 5.1 pour ajouter la configuration de la version 5.0.x à la configuration de la version 5.1.

```
WASPostUpgrade /opt/tmp/sauvegarde-spécifique-migration
```

L'outil WASPostUpgrade enregistre les informations détaillées spécifiques de chaque bean entreprise qu'il déploie dans le fichier */migration-specific-backup/WASPostUpgrade.log*.

6. Modifiez la configuration à l'aide des interfaces d'administration de WebSphere Application Server 5.1.

Apportez les modifications ci-dessous.

- a. **Modifiez les ID utilisateur et les mots de passe pour qu'ils correspondent aux conditions de sécurité requises.**

Il se peut que vous deviez changer les ID utilisateur et les mots de passe s'ils ne sont pas identiques à ceux utilisés sur la machine hébergeant la version 5.0.x.

- b. **Modifiez les autres informations spécifiques de la machine.**

La configuration peut renvoyer à d'autres produits logiciels ou à d'autres configurations n'existant pas sur la nouvelle machine. L'ancienne machine,

par exemple, peut disposer d'une base de données. Modifiez la source de données de sorte qu'elle pointe vers la base de données située sur l'ancienne machine.

7. Vous pouvez désinstaller, si vous le voulez, la version 5.0.x du serveur.

Migration à partir d'un système d'exploitation non pris en charge

Vous pouvez procéder à la migration d'une version antérieure de WebSphere Application Server version 3.5.x ou version 5.0.x qui fonctionne sur un système d'exploitation non pris en charge par la version 5.1.

Étapes de la procédure

1. Démarrez le serveur d'administration de WebSphere Application Server version 3.5.x ou version 5.0.x.
2. Exécutez l'outil de migration **WASPreUpgrade** à partir de la ligne de commande.

Deux options sont possibles. Vous pouvez exécuter cette commande depuis le répertoire `migration\bin` (ou `migration/bin`) dans le répertoire *répertoire_principal_plateforme* du CD-ROM de la version 5.1. Vous pouvez également copier les fichiers du répertoire du CD-ROM dans un répertoire que vous créez sur votre disque dur.

Identifiez votre version (3.5.x ou 5.0.x) et spécifiez un répertoire de sauvegarde dans lequel la commande stockera les fichiers de configuration et les applications à faire migrer à partir de la version antérieure. Reportez-vous à la section *WASPreUpgrade* pour connaître la syntaxe de la commande.

- a. Exécutez la commande depuis le répertoire `migration\bin` (ou `migration/bin`) dans le répertoire *répertoire_principal_plateforme* du CD-ROM de la version 5.1.

Identifiez le répertoire de sauvegarde et l'emplacement des fichiers de configuration.

```
unité_CD : \WASPreUpgrade répertoireSauvegarde cheminFichiers  
\WebSphere\AppServer nomDeVotreNoeud
```

Si cette procédure aboutit, passez à l'étape 4. Dans le cas contraire, suivez les instructions des étapes 2B à 2F.

- b. Créez un répertoire **migration** sur votre disque dur.
- c. Copiez les fichiers `WASPreUpgrade.bat` (ou `WASPreUpgrade.sh`) et `setupCmdLine.bat` (ou `setupCmdLine.sh`) à partir du répertoire `migration\bin\` (ou `migration/bin/`) situé dans le répertoire *répertoire_racine_plateforme* du CD-ROM de la version 5.1 dans le répertoire que vous avez créé sur votre disque dur.
- d. Editez le fichier `setupCmdLine.bat` (ou `setupCmdLine.sh`) dans le nouveau répertoire.

Modifiez les variables suivantes :

- **WAS_HOME** pour qu'elle pointe vers le chemin d'accès complet au répertoire de migration que vous avez créé.
 - **JAVA_HOME** de sorte qu'elle pointe vers le chemin d'accès complet au répertoire IBM Developer Kit ou Java.
- e. Vérifiez que le bit exécutable est activé pour les fichiers `setupCmdLine.sh` et `WASPreUpgrade.sh` du répertoire `migration/bin`, dans le répertoire *répertoire_principal_plateforme_UNIX* du CD-ROM de la version 5.1, si vous sauvegardez une installation UNIX.

- f. Exécutez la commande à partir du répertoire migration que vous avez créé. Identifiez le répertoire de sauvegarde et l'emplacement des fichiers de configuration.

```
votre_répertoire_de_migration\WASPreUpgrade répertoireSauvegarde  
cheminFichiers  
\WebSphere\AppServer nomDeVotreNoeud
```

3. Fermez WebSphere Application Server version 3.5.x ou version 5.0.x.
4. Effectuez une compression Tar ou zip du répertoire de sauvegarde et envoyez-le par FTP à un autre système.
5. Installez le nouveau système d'exploitation, en conservant le même nom d'hôte. Si possible, conservez le même nom et les mêmes mots de passe que pour l'ancien système. Placez les fichiers de base de données associés aux applications que vous migrez dans le même chemin que sous l'ancien système. En règle générale, essayez de conserver les mêmes chemins. Toutefois, n'installez pas la version 5.1 dans le même répertoire que la version précédente. Si vous modifiez les chemins et les noms, vous pouvez éditer les fichiers de configuration XML pour y répercuter les modifications. Apportez ces modifications avant d'exécuter la commande **WASPostUpgrade** ci-dessous.
6. Récupérez par FTP le répertoire de sauvegarde de l'autre système et décompressez-le.
7. Installez WebSphere Application Server- Express, version 5.1. Ne sélectionnez pas l'option de migration, si elle apparaît.
8. Exécutez l'outil de migration **WASPostUpgrade** sur la ligne de commande, à partir du répertoire bin du répertoire principal d'installation de la version 5.1. Spécifiez le répertoire de sauvegarde (et les noms des fichiers de configuration non standard du répertoire) créés par la commande **WASPreUpgrade**. Consultez la section *WASPostUpgrade* pour connaître la syntaxe de la commande.

```
répertoire_principal_installation\bin\WASPostUpgrade WAS_HOME\migration
```

Chapitre 3. Migration à partir d'IBM WebSphere Studio Site Developer version 5.1

Le présent chapitre se compose des sections suivantes :

- Migration des projets J2EE en vue de l'utilisation de la fonction de prise en charge du ciblage de serveur
- La génération de l'assistant requiert un package Java pour JDK 1.4

Migration des projets J2EE en vue de l'utilisation de la fonction de prise en charge du ciblage de serveur

IBM WebSphere Studio Site Developer version 5.1.1 comporte une nouvelle fonction de ciblage du serveur. Elle est désactivée par défaut mais vous pouvez l'activer dans la page des préférences J2EE en sélectionnant **Fenêtre > Préférences > J2EE**. Vous trouverez des détails relatifs à la fonction de ciblage du serveur dans la documentation en ligne de IBM WebSphere Studio Site Developer. Lorsque la fonction est activée, vous pouvez cibler un serveur d'applications particulier. Elle prend en charge JDK 1.4, qui correspond à l'environnement d'exécution Java (JRE) de WebSphere Application Server version 5.1 livré avec IBM WebSphere Studio Site Developer version 5.1.1. Les projets J2EE qui prennent en charge la fonction de ciblage du serveur ne sont pas compatibles en amont avec les versions antérieures d'IBM WebSphere Studio Site Developer, et par conséquent *ne peuvent pas* être partagés avec les membres d'une équipe travaillant sur des versions antérieures d'IBM WebSphere Studio Site Developer (exemple : IBM WebSphere Studio Site Developer version 5.1, IBM WebSphere Studio Site Developer version 5.0.1). IBM WebSphere Studio Site Developer permet la compatibilité en amont lorsque cette fonction, décrite dans la section «Compatibilité en amont avec fonction de prise en charge du ciblage de serveur activée» à la page 21 est activée. L'incompatibilité est due au fait que la fonction de ciblage de serveur modifie le fichier .classpath d'un projet J2EE et que les versions antérieures de WebSphere Application Server - Express *ne reconnaissent pas* les nouvelles entrées du fichier .classpath.

Compatibilité en amont avec fonction de prise en charge du ciblage de serveur activée

Lorsqu'elle est activée, vous pouvez faire en sorte que les projets J2EE ciblés sur un serveur cessent d'utiliser la fonction de prise en charge du ciblage de serveur en sélectionnant l'option de serveur cible **Aucun serveur cible spécifié** disponible dans la boîte de dialogue Modification du serveur cible. Vous pouvez ouvrir cette fenêtre à partir du menu contextuel (**Serveur cible > Modifier**) d'un projet J2EE dans le navigateur de ressources ou dans la perspective J2EE. Vous pouvez aussi modifier le serveur cible en sélectionnant l'option **Aucun serveur cible spécifié** à partir de la page des propriétés J2EE (**Propriétés > J2EE**) pour les projets EAR, EJB, de client d'application et de connecteur. Le paramètre du serveur cible d'un projet Web se trouve dans la page des propriétés Web (**Propriétés > Web**). Vous trouverez des détails relatifs à la modification du ciblage de serveur dans la documentation d'IBM WebSphere Studio Site Developer. Lorsque l'option **Aucun serveur cible spécifié** est sélectionnée, le fichier .classpath reprend le style compatible avec les versions antérieures d'IBM WebSphere Studio Site Developer et le fichier .server est supprimé du projet.

Remarque : Seuls les projets J2EE pour lesquels un serveur a été ciblé peuvent être déployés sur WebSphere Application Server version 5.1 et prendre en charge JDK 1.4.

La génération de l'assistant requiert un package Java pour JDK 1.4

En raison d'une modification apportée au JDK 1.4, l'utilisateur doit spécifier un package Java lors de l'utilisation des assistants Pages Web de base de données et Pages Web de bean Java pour générer des pages à exécuter sur IBM WebSphere Studio Site Developer version 5.1.1. Ce problème survient si le modèle de bean View est utilisé pour l'assistant Pages Web de bean Java ou le modèle Beans d'accès aux données IBM - Modèle maître de détail. Il se produit également pour les projets contenant des pages et des fichiers .java générés précédemment par ces assistants, et pour lesquels aucun package n'a été spécifié lors de la création. Pour le code précédemment généré, déplacez les fichiers .java dans un package. Ensuite, mettez à jour les fichiers .jsp ainsi que les instructions d'importation et les informations relatives aux classes. Dans le fichier web.xml du projet, mettez à jour l'entrée servlet-class.

Chapitre 4. Migration à partir d'IBM WebSphere Studio Site Developer version 5.0.1

Le présent chapitre se compose des sections suivantes :

- WebSphere Studio Workbench (WSWB) dans la version 5, la version 5.0.1 et la version 5.1
- Utilisation de WebSphere Studio version 5.1.1 avec un espace de travail de la version 5.0
- Migration de projets Java à partir de la version 5 ou de la version 5.0.1
- Partage des projets entre la version 5 ou la version 5.0.1 et la version 5.1 à l'aide d'un système SCM (Source Code Management)
- Migration des projets Web
- Conversion des projets Web vers Struts 1.1
- Fonction de migration ajoutée à la migration de projets J2EE
- Modifications apportées aux outils des services Web
- Modifications apportées aux outils de profilage
- Problèmes de compatibilité de l'assistant de modèle

WebSphere Studio Workbench dans la version 5, la version 5.0.1 et la version 5.1

IBM WebSphere Studio Site Developer version 5.1.1, repose sur le nouveau plan de travail Eclipse WebSphere Studio Workbench (WSWB) 2.1.2. Il existe des différences entre les versions 2.1.2 et 2.0.3 ou 2.0.2. Pour plus de détails sur ces différences, consultez le fichier *readme* situé dans le répertoire *RépInstall_WS\eclipse\readme* (où *RépInstall_WS* correspond au chemin d'installation d'IBM WebSphere Studio Site Developer).

IBM WebSphere Studio Site Developer version 5.0 reposait sur le plan de travail Eclipse WSWB 2.0.2 et IBM WebSphere Studio Site Developer version 5.0.1 sur le plan de travail Eclipse WSWB 2.0.3. Il n'existe pas de différence fondamentale entre les versions 2.0.2 et 2.0.3. L'édition d'IBM WebSphere Studio Site Developer version 5.0.1 était un correctif d'Update Manager installé sur IBM WebSphere Studio Site Developer version 5.0.

Utilisation d'IBM WebSphere Studio Site Developer version 5.1.1 avec l'espace de travail de la version 5.0

Lors du premier démarrage d'IBM WebSphere Studio Site Developer version 5.1.1 avec un espace de travail existant d'IBM WebSphere Studio Site Developer version 5.0, une boîte de dialogue apparaît et indique comment procéder à la migration à partir de la version 5.0. Cliquez sur **OK** pour effectuer la migration de l'espace de travail de la version 5.0 ou cliquez sur **Annuler** pour arrêter le lancement d'IBM WebSphere Studio Site Developer.

Même si vous avez procédé à la migration de l'espace de travail, vous pouvez l'utiliser dans la version 5.0 car les métadonnées des caractéristiques de nouveau projet sont ignorées et peuvent être lues par la version 5.0. Toutefois, dans la

version 5.0, vous ne pouvez apporter aucune modification aux projets de l'espace de travail affectant ou remplaçant les métadonnées des caractéristiques de nouveau projet des projets.

Migration de projets Java à partir de la version 5 ou de la version 5.0.1

La migration de projets Java à partir de la version 5 ou de la version 5.0.1 est simple et automatique. Une fois les projets chargés dans l'espace de travail de la version 5.1.1, aucune métadonnée n'est modifiée dans les fichiers .classpath ou .project, sauf si vous utilisez l'une des nouvelles fonctions disponibles dans la version 5.1.1.

Partage des projets entre la version 5 ou la version 5.0.1 et la version 5.1.1 à l'aide d'un système de migration SCM (Source Code Management)

Une attention particulière est requise lorsqu'un projet de référentiel coopératif est chargé et utilisé par des développeurs à l'aide d'IBM WebSphere Studio Site Developer version 5 et IBM WebSphere Studio Site Developer version 5.1.1. En effet, l'existence, le contenu et l'interprétation des fichiers de métadonnées dans les espaces de travail sont susceptibles d'être spécifiques à la version de la fonction ou du plug-in et peuvent varier d'une version à l'autre. La compatibilité des espaces de travail n'est garantie que dans les cas dans lesquels tous les développeurs mettent à niveau leur espace de travail IBM WebSphere Studio Site Developer pendant le verrouillage. Dans ces cas, les métadonnées ne génèrent aucun incident. Cependant, ce n'est pas le cas lorsque certains développeurs utilisent IBM WebSphere Studio Site Developer version 5.1.1 alors que d'autres utilisent IBM WebSphere Studio Site Developer version 5. Cette section indique quelles sont les actions recommandées et déconseillées.

En général, l'utilisateur d'IBM WebSphere Studio Site Developer version 5.1.1 remarque l'incompatibilité. Les métadonnées de la version 5.1.1 sont perdues lorsque l'utilisateur de la version 5 sauvegarde des modifications puis valide les fichiers de métadonnées mis à jour dans le référentiel. Voici certaines actions à ne pas effectuer :

- Un utilisateur d'IBM WebSphere Studio Site Developer version 5.1.1 crée ou modifie un projet de telle sorte qu'il entraîne des modifications dans un fichier partagé de métadonnées qui repose sur des informations spécifiques à la version 5.1.1. L'utilisateur valide ensuite les fichiers du projet mis à jour, dont le fichier partagé de métadonnées, dans le référentiel partagé.
- Un autre utilisateur utilisant IBM WebSphere Studio Site Developer version 5 partage ce projet à partir du même référentiel. Les informations spécifiques à la version 5.1.1 dans le fichier partagé de métadonnées ne sont pas interprétées par IBM WebSphere Studio Site Developer version 5 et sont généralement supprimées ou ignorées sans avertissement. L'utilisateur modifie le projet de telle sorte que le fichier partagé de métadonnées est modifié, ce qui entraîne la réécriture de ce fichier sans les informations spécifiques à la version 5.1.1. L'utilisateur valide ensuite les fichiers du projet mis à jour, dont le fichier partagé de métadonnées, dans le référentiel partagé. L'utilisateur ne sait alors généralement pas que des informations partagées ont été perdues.
- Un utilisateur utilisant IBM WebSphere Studio Site Developer version 5.1.1 extrait les modifications apportées à un projet à partir du référentiel partagé, dont le fichier partagé de métadonnées. Cet utilisateur ne se doute alors pas que le fichier a été rétrogradé.

Il convient d'apporter une attention particulière aux éléments ci-après lorsque le projet doit être partagé entre des utilisateurs d'IBM WebSphere Studio Site Developer version 5.1.1 et version 5 ou 5.0.1.

- **Fichiers et dossiers liés**

Cette prise en charge a été ajoutée dans IBM WebSphere Studio Site Developer version 5.1.1. Les informations concernant les ressources liées sont enregistrées dans le fichier .project.

Conseil : N'utilisez pas cette fonction. Il est conseillé de désactiver les ressources liées dans la page des préférences **Plan de travail > Ressources liées**.

- **Compilateurs d'outils externes (Ant)**

Les informations concernant les compilateurs d'outils externes sont enregistrées dans le fichier .project. Le format de ces informations a été modifié entre la version 5 et la version 5.1.1. Les compilateurs créés ou modifiés dans la version 5.1.1 utilisent le nouveau format, qui n'est pas interprété dans l'espace de travail de la version 5. Les compilateurs créés dans la version 5 utilisent l'ancien format et peuvent être utilisés dans la version 5.1.1.

Conseil : Créez ou éditez les compilateurs d'outils externes à partir d'un espace de travail de la version 5.

- **Modèles d'exclusion facultatifs des entrées de source Java dans le chemin d'accès aux classes de compilation**

Cette prise en charge a été ajoutée. Ces informations sont enregistrées dans le fichier .classpath.

Conseil : Ne spécifiez pas de modèles d'exclusion. Il est conseillé de désactiver les modèles d'exclusion dans la page des préférences **Java > Compilateur > Chemin de compilation**.

- **Dossiers de sortie associés aux entrées de source Java dans le chemin d'accès aux classes de compilation**

Cette prise en charge a été ajoutée. Ces informations sont enregistrées dans le fichier .classpath.

Conseil : Ne spécifiez pas d'autre dossier de sortie que celui par défaut (totalité du projet). Il est conseillé de désactiver les emplacements multiples de sortie dans la page des préférences **Java > Compilateur > Chemin de compilation**.

- **Chemin racine de pièce jointe source associé aux entrées de bibliothèque Java dans le chemin d'accès aux classes de compilation**

Lorsqu'un fichier ZIP source est associé à une bibliothèque JAR dans le chemin d'accès de compilation Java, le préfixe du chemin racine de la source est déduit automatiquement. Dans la version 5, il était possible de définir explicitement ce préfixe dans l'interface utilisateur et de l'enregistrer dans le fichier .classpath. Ainsi, un projet Java créé dans un espace de travail de la version 5.1.1 est susceptible de ne pas trouver la source jointe.

Utilisez la version 5 pour spécifier le chemin racine de la pièce jointe source. La version 5.1.1 permet une plus grande flexibilité concernant les pièces jointes source. Vous pouvez indiquer un dossier au lieu d'un fichier JAR ou ZIP comme pièce jointe source et associer la source à un dossier de fichier de classe. Cette fonctionnalité n'est pas disponible dans la version 5 (les informations de la version 5.1.1 sont ignorées).

- **PDE et conteneurs de chemin d'accès aux classes pour les plug-ins dépendants**

PDE utilisant des conteneurs de chemin d'accès aux classes a été ajouté. Les conteneurs de chemin d'accès aux classes sont enregistrés dans le fichier .classpath. Si les conteneurs de chemin d'accès aux classes PDE sont utilisés, les entrées du chemin d'accès aux classes de l'espace de travail de la version 5 ne

seront pas résolues et la majorité des fonctionnalités Java (compilation, recherche, exécution, débogage) ne généreront pas les résultats attendus.

Conseil : Assurez-vous que le paramètre d'utilisation des conteneurs de chemin d'accès aux classes sur la page des préférences **Développement de plug-in > Contrôle du chemin de compilation Java** est désactivé avant de créer de nouveaux projets de plug-in (ou de fragment).

Migration de projets Web

Les noms de dossier sont **Java Source** et **Web Content**. Il est possible de configurer les noms de dossier par défaut des nouveaux projets Web dans la page des préférences (**Fenêtre > Outils Web > Nouveau projet**). Les noms par défaut sont désormais **JavaSource** et **WebContent**. Ces noms par défaut seront utilisés uniquement pour les nouveaux projets Web. Les projets Web créés dans des versions antérieures à cette édition continueront à fonctionner avec les anciens noms. Ceci est également vrai pour les projets Web statiques.

Si vous souhaitez modifier les noms des dossiers source des projets des versions 4.0.x et 5.0 dans la version 5.1.1, utilisez l'option **Renommer** du menu en incrustation dans la vue Navigateur. Cette option permet de renommer les dossiers et de corriger le chemin de compilation Java des projets Web des versions 4.0.x et 5.0.x.

Dans le cas du dossier **JavaSource**, l'option **Renommer** fonctionne dans les vues Navigateur de projets et Packages. Dans le cas du dossier **WebContent**, l'option **Renommer** est disponible dans les vues du navigateur de ressources et du navigateur de projets.

Si un projet Web d'une version antérieure à cette édition est enregistré dans un référentiel SCM, puis chargé dans cette édition, il conservera l'ancienne structure composée des dossiers **source** et **webApplication**. L'une ou l'autre structure sera correctement compilée.

Remarque : Si les utilisateurs décident de renommer les dossiers **Java Source** et **Web Content**, ils doivent mettre à jour manuellement tous les scripts de compilation automatiques de telle sorte qu'ils utilisent les nouveaux noms de dossiers.

Conversion des projets Web vers Struts 1.1

Le contexte d'exécution des outils Struts est passé de la version 1.1 bêta 2 dans la version 5 à la version 1.1. Dans IBM WebSphere Studio Site Developer version 5 (disponibilité générale), lorsque vous créez un projet Web, vous pouvez choisir d'ajouter le support Struts à votre projet. Vous avez le choix entre Struts 1.0.2 et Struts 1.1 bêta 2. Dans IBM WebSphere Studio Site Developer version 5.1.1, le dernier choix est remplacé par Struts 1.1. Si vous avez créé des projets Web avec Struts 1.1 bêta 2 dans IBM WebSphere Studio Site Developer version 5, vous pouvez les convertir vers Struts 1.1, mais cette opération n'est pas obligatoire car Struts 1.1 bêta 2 demeure pris en charge. Pour convertir des projets Web créés avec Struts 1.1 bêta 2 vers Struts 1.1, procédez comme indiqué ci-après.

1. Chargez vos projets Struts 1.1 bêta 2 dans un espace de travail IBM WebSphere Studio Site Developer version 5.1.1.
2. Créez un nouveau projet Web Struts 1.1 appelé **Struts11**. Ainsi, vous pourrez accéder facilement aux objets Struts 1.1 indispensables lors de la conversion des projets réels. Vous pourrez supprimer ce projet lorsque vous aurez terminé.

3. Pour chaque projet Struts 1.1 bêta 2 à convertir vers Struts 1.1, procédez comme indiqué ci-après.
 - a. Supprimez les fichiers .jar suivants du répertoire Web Content/WEB-INF/lib de votre projet: commons-*.jar et struts.jar.
 - b. Copiez les fichiers .jar suivants du répertoire Struts11/WebContent/WEB-INF/lib vers le répertoire Web Content/WEB-INF/lib de votre projet : commons-*.jar et struts.jar.
 - c. Supprimez les fichiers .tld suivants du répertoire Web Content/WEB-INF du projet : struts-*.tld.
 - d. Copiez les fichiers .tld suivants du répertoire Struts11/WebContent/WEB-INF dans le répertoire Web Content/WEB-INF de votre projet : struts-*.tld.

Toutes les indications ci-dessus sont également valables si vous convertissez un projet Web Struts 1.1 bêta 3 de IBM WebSphere Studio Site Developer version 5.0.1 vers Struts 1.1.

Modifications apportées aux outils des services Web

Les outils des services Web disposent de deux nouveaux protocoles d'exécution d'IBM WebSphere Application Server version 5.0.2 qui s'exécutent uniquement sur WebSphere Application Server version 5.0.2. Aucune migration n'est nécessaire puisque IBM WebSphere Studio Site Developer version 5.1.1 et WebSphere Application Server version 5.0.2 prennent en charge les anciens et les nouveaux protocoles. IBM WebSphere Studio Site Developer génère et déploie trois protocoles d'exécution des artefacts de service Web : l'ancien protocole d'exécution "IBM SOAP" qui s'exécute sur WebSphere Application Server versions 4.x et 5.x ; le nouveau protocole d'exécution "IBM WebSphere 5.0.2 runtime" qui s'exécute uniquement sur WebSphere Application Server version 5.0.2 ; et le nouveau protocole d'exécution "Apache Axis 1.0" qui s'exécute uniquement sur WebSphere Application Server version 5.0.2.

Vous pouvez réutiliser les projets de la version 5 ainsi que les fichiers WAR et EAR associés aux services Web sans qu'il soit nécessaire de les modifier dans la version 5.1.1. Pour convertir les clients et les services Web vers le nouveau protocole d'exécution IBM WebSphere 5.0.2 et bénéficier des normes JSR 101, 109, WS-I et WS-Security, vous devez les régénérer via l'assistant de services Web. L'explorateur de services Web continue automatiquement à lire les préférences de l'utilisateur même si le fichier de données physique est déplacé automatiquement vers un nouvel emplacement.

Modifications apportées aux outils de profilage

Lorsque vous effectuez la migration d'un espace de travail à partir de la version 5, le message d'erreur "Erreurs lors de la restauration du plan de travail" s'affiche. Ce message apparaît si la perspective Profilage est ouverte lors de la migration et si la vue Mémoire dynamique ou Statistiques d'instances est visible dans la perspective Profilage. En effet, les vues Mémoire dynamique et Statistiques d'instances qui existaient dans la version 5 ont été supprimées. Ce message apparaît également si la perspective Profilage est ouverte dans l'espace de travail lors de la migration. Ignorez ce message d'erreur et cliquez sur **OK**.

Problèmes de compatibilité de l'assistant de modèle

Afin de pouvoir utiliser un modèle personnalisé créé dans la version 5, vous devez le charger, le reconnecter à la base de données et le sauvegarder. Lors du chargement suivant du modèle personnalisé sauvegardé, la connexion est vérifiée.

Il est possible que les artefacts J2EE générés dans cette édition ne puissent pas être lus par IBM WebSphere Studio Site Developer version 4.0.3, ni exécutés dans les environnements de test de la version 4.0.3. Puisque l'assistant de modèle n'était pas fourni avec la version 4.0.3, la compatibilité amont n'est pas conservée dans cette version.

Les applications modèles générées dans cette édition peuvent être exécutées dans la version 5 si, dans les préférences du projet Web, le dossier de la source Java est appelé "Java Source" et que le dossier du contenu Web est appelé "Web Content".

Chapitre 5. Migration à partir d'IBM WebSphere Studio Site Developer version 4.0.x

Le présent chapitre explique comment effectuer une migration à partir d'IBM WebSphere Studio Site Developer version 4.0.x vers la version 5.

Pour migrer un projet IBM WebSphere Studio Site Developer de la version 4.0.x vers la version 5, vous pouvez utiliser l'une des deux méthodes décrites en détail ci-après.

- Utilisation d'un système SCM (Software Configuration Management), tel que Concurrent versioning System (CVS) ou Rational ClearCase. Cette méthode est la plus recommandée.
- Exportation des projets à partir de la version 4.0.x, puis importation des projets dans la nouvelle version. Cette méthode permet de tout migrer, sauf les informations sur le chemin de compilation d'un projet.
- Utilisation de l'espace de travail existant de la version 4.0.x. Cette opération n'est pas prise en charge pour les raisons indiquées plus bas.

La migration de la version 4 à la version 5 ne modifie *pas* automatiquement le niveau J2EE des projets car la version 5 peut continuer à compiler et à déployer des applications dans WebSphere Application Server version 4. Il est possible de migrer tous les types de projet J2EE, dont les projets Web, à l'aide de l'assistant de migration J2EE, disponible dans IBM WebSphere Studio Site Developer. Pour accéder à l'assistant de migration J2EE, cliquez avec le bouton droit de la souris sur le projet de type J2EE et sélectionnez ensuite **Migrer > Assistant de migration J2EE**.

Différences entre IBM WebSphere Studio Site Developer version 4.0.x et version 5

Liste partielle des améliorations par rapport à la version 4.0.x :

- IBM WebSphere Studio Site Developer version 5 peut générer un code pour WebSphere Application Server version 4.0 ou version 5 et comporte les environnements de test WebSphere Application Server version 4.0.4 et version 5.
- Les archives EAR J2EE sont passées du niveau 1.2 au niveau 1.3 pour les projets WebSphere Application Server version 5.
 - Les EAR J2EE 1.2 peuvent s'exécuter sur WebSphere Application Server version 4.0.x ou WebSphere Application Server version 5.
- Le niveau de spécification d'EJB (Enterprise Java Beans) est passé de 1.1 à 1.2 pour les EAR WebSphere Application Server version 5.
 - sont toujours pris en charge et peuvent faire partie des EAR J2EE 1.2 dans WebSphere Application Server version 4.0.x ou des EAR J2EE 1.3 de la version 5.
- Les applications Web (WAR) J2EE sont passées du niveau 1.2 au niveau 1.3 pour les projets WebSphere Application Server version 5.
 - Les fichiers JSP sont passés du niveau 1.1 au niveau 1.2 et les fichiers Servlet du niveau 2.2 au niveau 2.3.

- Les projets Web (WAR) 1.2 dynamiques sont toujours pris en charge et peuvent faire partie des EAR J2EE 1.2 dans WebSphere Application Server version 4.0.x J2EE 1.2 ou des EAR J2EE 1.3 dans la version 5.
- La version 5 permet de créer des projets Web statiques, ainsi que des projets Web dynamiques. Dans un projet Web statique, vous ne pouvez créer que des données de contenu gérées par un serveur HTTP standard (HTML, JavaScript, images, texte, etc.).
- Le plan de travail sous-jacent, qui repose sur le projet à code ouvert Eclipse est passé de la version 1.0 à la version 2.0.
 - Un nouveau compilateur amélioré Java est inclus.
 - L'interface VCM (version Control Management) pour les fournisseurs SCM (Software Configuration Management) a été considérablement améliorée.

Modifications apportées à WebSphere Application Server et outils de conversion Servlet/JSP

L'InfoCenter WebSphere

[www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/index.html] contient les informations ci-après.

- Différences entre WebSphere Application Server version 3.5 et 4.0 [www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/was/03.html].
- Pour plus de détails sur la migration vers WebSphere Application Server version 5.0 à partir de la version 3.5 ou de la version 4.0, accédez à l'InfoCenter WebSphere à l'adresse www.ibm.com/software/webservers/appserv/infocenter.html et suivez les étapes ci-après pour accéder à la page concernant la migration.
 1. En regard de la première zone **Application Server for distributed operating systems**, sélectionnez la langue de votre choix.
 2. Cliquez sur le lien **All topics by activity** situé dans le coin supérieur gauche de la fenêtre.
 3. Cliquez sur **Migration**.

Le manuel *Migrating to WebSphere V5.0 An End-to-End Migration Guide* fournit des informations sur la migration à partir de la version 3.5 et de la version 4.0 vers la version 5 [www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246910.pdf].

La page de téléchargements de WebSphere Application Server

[www14.software.ibm.com/webapp/download/product.jsp?s=p&id=TDUN-49EVRT&type=s&dt=DIAGNOSTIC+TOOL] donne accès à des outils de conversion d'applications :

- MigrateWC convertit une page JSP .91 ou 1.0 au format 1.1. Il permet également de convertir un fichier servlet version 2.1 en version 2.2.
- XMLconvert convertit les fichiers de configuration XML de la version 3.02x ou 3.5x au format de la version 4.0.

Modifications internes depuis la version 4.0.3

Par défaut, la compilation ne peut pas s'effectuer s'il existe des dépendances circulaires dans un projet

Si votre projet contient des dépendances circulaires, la version 5 signale une erreur de compilation. Vous pouvez sélectionner **Fenêtre > Préférences > Java > Compilateur**, cliquer sur l'onglet **Chemin de compilation** et désélectionner l'option **Annuler la compilation en cas d'erreurs dans le chemin de compilation**. Cela ne provoque plus l'arrêt de la compilation mais entraîne l'affichage d'une ou plusieurs erreurs de type 'dépendance en boucle' dans la vue Tâches (même si la compilation est par ailleurs réussie). Dans ce cas, vous pouvez modifier ces erreurs en avertissements en sélectionnant l'onglet **Autre**, puis en modifiant la préférence dans la liste déroulante **Dépendances circulaires**.

Les répertoires source des projets Web de la version 5 sont compatibles avec la version 4.0.3

Dans IBM WebSphere Studio Site Developer version 5, la structure interne des projets a été modifiée par rapport à la version 4.0.3. Il est possible d'importer un fichier WAR Web J2EE version 5, exporté avec du code source Java, dans IBM WebSphere Studio Site Developer version 4. Le dossier du code source est alors automatiquement converti avec le nom approprié et la compilation se déroule normalement. De même, le projet Web s'exécute correctement sur WebSphere Application Server version 4 lorsqu'un projet de la version 4 est importé dans la version 5, car le dossier du code source est automatiquement converti avec le nom approprié. Pour plus d'informations sur les modifications du nom des dossiers, voir «Structures des projets Web IBM WebSphere Studio Site Developer» à la page 31.

Remarque : Ceci ne se vérifie pas si les projets Web sont partagés entre la version 5 et la version 4 via un système SCM (Software Configuration Management). Les projets de la version 4 doivent être migrés vers la structure de projet de la version 5 et ils ne peuvent pas être reconvertis vers la version 4 à partir d'un système SCM une fois qu'ils sont migrés.

Structures des projets Web IBM WebSphere Studio Site Developer

La structure interne des projets Web dans IBM WebSphere Studio Site Developer version 5 est différente de ce qu'elle était dans IBM WebSphere Studio Site Developer version 4.0.x. Cette différence n'est pas liée au passage de J2EE 1.2 en J2EE 1.3, mais plutôt à une modification d'utilisation de l'outil.

Dans la version 4, les projets Web sont par défaut des projets Web dynamiques et ils apparaissent dans la vue Navigateur avec un dossier **source** et un dossier **webApplication**. Dans la version 5, si vous créez un projet Web dynamique, celui-ci apparaît avec un dossier **Java Source** à la place du dossier **source** et un dossier **Web Content** à la place du dossier **webApplication**.

Toutefois, si un projet Web de la version 4 est enregistré dans un référentiel SCM, puis chargé dans la version 5, il conservera l'ancienne structure avec les dossiers **source** et **webApplication**. L'une ou l'autre structure sera correctement construite dans la version 5.

Projets Web statiques et dynamiques

La version 5 permet de créer des projets Web statiques, ainsi que des projets Web dynamiques.

Les projets Web statiques contiennent uniquement des ressources statiques comme des éléments HTML, des éléments Java Scripts, des images, du texte, etc., mais ne renferment aucun contenu dynamique. Les projets Web statiques peuvent être exécutés et servis par un serveur Web HTTP traditionnel et ne nécessitent pas de serveur d'applications Web.

Les projets Web dynamiques contiennent, outre des ressources statiques, des ressources J2EE dynamiques comme les servlets, les pages JSP, les filtres et les métadonnées associées. Lorsque vous créez des projets Web dynamiques, vous pouvez inclure des feuilles de style en cascade et des bibliothèques de balises JSP pour pouvoir commencer le développement avec un plus grand ensemble de ressources de projet. Les projets Web dynamiques sont toujours imbriqués dans les projets d'application entreprise et ne peuvent être exécutés que sur des serveurs d'applications Web.

Distinctions entre HTML et JSP

- Dans la version 4.0.x, les fichiers HTML et JSP étaient traités indifféremment par Page Designer. Un fichier HTML pouvait contenir des balises JSP par exemple. Ce n'est plus le cas dans cette nouvelle version qui distingue les fichiers JSP des fichiers HTML. Un fichier HTML ne peut donc pas contenir de balises JSP.
- La distinction précédente affecte l'encodage de fichiers JSP dans d'autres langues que l'anglais. Dans les versions précédentes, les règles d'encodage HTML ont été utilisées, même pour les fichiers JSP, pour déterminer l'encodage spécifié dans un fichier. L'attribut "content type" de la balise meta était vérifié (`<META http-equiv="Content-Type" content="text/html; charset=UTF-8">`). Dans cette nouvelle version, les règles d'encodage JSP sont utilisées pour déterminer l'encodage spécifié dans un fichier JSP. La directive de page du fichier JSP est ainsi définie par (`<%@page contentType="text/html; charset=UTF-8"%>`). Pour les fichiers HTML, l'encodage n'est pas modifié par rapport aux versions précédentes.

Migration des projets à l'aide d'un système SCM (Software Configuration Management)

Migration des projets à l'aide de CVS ou Rational ClearCase

Cette méthode est recommandée pour déplacer des espaces de travail de la version 4.0.x vers IBM WebSphere Studio Site Developer version 5. Cette méthode est la seule qui permette de migrer la totalité des informations, y compris celles relatives au chemin de compilation d'un projet.

1. Par mesure de sauvegarde, enregistrez tous vos projets en version 4 dans votre référentiel SCM. Validez ensuite toutes les modifications en attente.
2. Si vous souhaitez utiliser la version 4 et la version 5 d'IBM WebSphere Studio Site Developer, sauvegardez de nouveau votre travail dans une nouvelle branche (flux) de la version 5. C'est la branche que vous utiliserez lorsque vous passerez à la version 5.
3. Installez la version 5.
4. Fermez IBM WebSphere Studio Site Developer version 4 et démarrez IBM WebSphere Studio Site Developer version 5.

Conseil : Dans la version 4, le répertoire de l'espace de travail se trouve par défaut dans le répertoire d'installation. Dans la version 5, il correspond par défaut à un répertoire appelé **workspace** dans le répertoire My Documents. Pour modifier le répertoire de stockage de vos fichiers de travail, exécutez la commande avec l'option `-data` lorsque vous lancez le plan de travail.

Remarque : N'utilisez pas l'option `-data` pour indiquer un espace de travail de la version 4 car cette méthode de migration n'est pas prise en charge (Pour plus d'informations, voir «Migration de projets à l'aide d'un espace de travail existant en version 4.0.x» à la page 35.)

5. Désactivez **Fenêtres > Préférences > Plan de travail > Effectuer une compilation automatique lorsqu'une ressource est modifiées** (pour éviter les erreurs de compilation au fur et à mesure que les projets dépendants sont chargés).
6. **Pour CVS :** Chargez tous les projets requis à partir du référentiel SCM dans IBM WebSphere Studio Site Developer version 5.
Pour ClearCase : Utilisez un espace de travail version 5 nettoyé et, pour chaque projet à charger, sélectionnez **Fichier > Importer > Projet WebSphere Studio 4.x ClearCase existant dans l'espace de travail**.
7. Restaurez le paramètre de votre choix pour **Fenêtres > Préférences > Plan de travail > Effectuer une compilation automatique lorsqu'une ressource est modifiée**.
8. Si une compilation complète est nécessaire, modifiez le nom du dossier **source** en **Java Source** et celui du dossier **webApplication** en **Web Content** pour les projets Web. Si vous n'effectuez pas cette opération, l'ancienne structure dossier est conservée et les projets Web *ne sont pas* entièrement recompilés.
9. Effectuez une recompilation complète (**Projet > Régénérer tout**) et sauvegardez les projets générés dans le référentiel dans un nouveau flux version 5. (*Ne mélangez pas* ces ressources avec votre flot en version 4 en cours).

Remarque : Ces projets sont désormais des projets de version 5 qui ne peuvent *plus* être utilisés par IBM WebSphere Studio Site Developer version 4.0.x.

Remarques à prendre en compte après la migration :

- Dans IBM WebSphere Studio Site Developer version 4.0, les fichiers d'un projet sont stockés au format binaire dans CVS. Dans le format binaire, les retours chariot et les sauts de ligne dans les fichiers *ne sont pas* convertis en vue de leur utilisation sur d'autres plateformes et *ne peuvent pas* être reconnus dans les environnements des plateformes UNIX/Linux. Si vous travaillez dans des environnements de plateformes mixtes, comme DOS/Windows et UNIX/Linux, vous pouvez marquer les fichiers source comme étant de type texte (en sélectionnant **Equipe > CVS > Modifier la propriété ASCII/Binaire**) et les sauvegarder à nouveau au format CVS.
- Pour des projets Web en version 4 stockés dans un référentiel CVS, l'option **Fenêtre > Préférences > Equipe > CVS > Supprimer les anciennes entrées des répertoires vides** doit être désactivée (elle est activée par défaut). Si l'option n'est *pas* activée et que vous chargez un projet Web avec un dossier **source** vide (comme l'exemple Web MyHomePage), vous recevrez les messages d'erreur suivants lors de la restitution :

Le projet n'a pas été compilé en raison d'erreurs dans le chemin de classe (incomplet ou impliqué dans un cycle).

Dossier source requis manquant : /MyHomePageExample403/source.

- Pour les projets Web sauvegardés et chargés à partir du référentiel ClearCase, vous devez réserver le fichier pour pouvoir l'ouvrir dans l'éditeur. Si le fichier n'est pas réservé, vous recevez un message de type Erreur lors de l'activation de cette vue (exception pointeur Null dans les journaux de Page Designer). Lorsque vous modifiez un fichier web.xml à l'aide d'un éditeur xml, vous devez réserver les fichiers web.xml, ibm-web-bnd.xmi et ibm-web-ext.xmi (la ligne d'état vous indique que ces fichiers doivent être réservés car ils sont disponibles en lecture seule mais ces informations ne sont pas toujours très visibles).
- Si votre projet contient des dépendances circulaires, la version 5 signale une erreur de compilation. Vous pouvez sélectionner **Fenêtre > Préférences > Java > Compilateur**, cliquer sur l'onglet **Chemin de compilation** et désélectionner l'option **Annuler la compilation en cas d'erreurs dans le chemin de compilation**.
- Les fichiers .vcm_meta (ou .cc_meta) peuvent à présent être supprimés du projet de la version 5 car ils ne peuvent pas être utilisés par cette version (elle utilise un nouveau fichier de projet à la place) et vous utilisez une nouvelle branche de référentiel (flux) pour ces projets de la version 5. Ces fichiers sont toujours utilisés dans la branche en cours de la version 4.

Suppression des références de chemin d'accès absolu dans les fichiers EAR et les fichiers de configuration serveur après la migration

Les fichiers d'extension d'application EAR IBM et les fichiers de configuration du serveur version 4 contenaient des références de chemins d'accès absolus. Après avoir migré ces fichiers dans la version 5, vous devez les ouvrir à l'aide de l'éditeur approprié (qui remplace automatiquement les anciennes références de chemin d'accès absolu par des références de chemin relatif).

1. Dans la vue Navigateur et pour chaque projet EAR, cliquez à l'aide du bouton droit de la souris sur **META-INF/application.xml > Ouvrir avec > Editeur de descripteur de déploiement**.
 - a. Une boîte de dialogue affiche le message :

Le fichier d'extensions IBM contient des chemins absolus déconseillés.

Cette erreur peut être corrigée automatiquement et doit être sauvegardée.
Les chemins seront supprimés du fichier à la première opération.
Voulez-vous procéder à la correction automatique ?
 - b. Cliquez sur **Oui**.
 - c. Sauvegardez et fermez la fenêtre de l'éditeur.

Remarque : Vous pouvez également faire appel à l'assistant de migration J2EE pour effectuer la migration de la structure de projet, uniquement dans le cas d'un projet EAR. Pour accéder à l'assistant de migration J2EE, cliquez avec le bouton droit de la souris sur le projet EAR et sélectionnez ensuite **Migrer > Assistant de migration J2EE**.

2. Pour chaque configuration de serveur d'une perspective de serveur (vue Configuration de serveur) cliquez avec le bouton droit de la souris sur le serveur et sélectionnez **Ouvrir**.
 - a. Une boîte de dialogue s'affiche pour vous inviter à effectuer une correction automatique.

- b. Cliquez sur **Oui**.
- c. Sauvegardez et fermez la fenêtre de l'éditeur.

Migration des projets à l'aide d'autres systèmes SCM

D'autres fournisseurs SCM fournissent des plug-ins SCM pour IBM WebSphere Studio Site Developer. Vous pouvez consulter la liste des fournisseurs à l'adresse suivante : www.ibm.com/software/ad/studioappdev/partners/scm.html. Dans le cadre de la validation *Ready for IBM WebSphere Studio software* [www.developer.ibm.com/websphere/ready.html], tous les fournisseurs SCM de plug-in pour la version 4 se sont assurés que les étapes de migration précédentes (sauvegarde à partir de la version 4 vers le référentiel SCM, chargement à partir du référentiel dans la version 5) s'appliqueront également à leurs systèmes.

Migration par exportation et importation de projets

1. Dans IBM WebSphere Studio Site Developer version 4.0.x, exportez vos projets dans un fichier WAR, EAR ou JAR en sélectionnant **Fichier > Exporter**.
2. Dans IBM WebSphere Studio Site Developer version 5, importez le fichier WAR, EAR ou JAR en sélectionnant **Fichier > Importer**.

Remarque : Il ne s'agit pas d'une migration complète car aucune information relative au chemin de compilation du projet n'est conservée.

Migration de projets à l'aide d'un espace de travail existant en version 4.0.x

Cette approche n'est pas prise en charge entièrement et la migration effectuée n'est pas complète. Les paramètres de l'interface utilisateur, de débogage et la plupart des options de préférences sont perdus. Seuls le nom, les fichiers sources et le chemin de compilation Java de chaque projet (chemin d'accès aux classes) sont conservés. Aucune garantie n'est fournie pour les autres informations. Adoptez cette approche uniquement si aucun système SCM pris en charge n'est utilisé et s'il est indispensable de conserver les informations sur le chemin de compilation d'un projet (ce chemin est perdu lorsque vous importez un projet qui a été exporté à partir de la version 4). Vous pouvez utiliser l'espace de travail existant en version 4.0.x en procédant comme suit :

1. Validez toutes les modifications en attente dans le référentiel.
2. Fermez toutes les perspectives et quittez IBM WebSphere Studio Site Developer version 4.
3. Sauvegardez le contenu du *répertoire_espace_travail*, où *répertoire_espace_travail* est le chemin d'accès complet à l'espace de travail de la version 4.0.x. Le sous-répertoire de l'espace de travail de la version 4.0.x se trouve par défaut dans le répertoire d'installation du produit. Vous devez effectuer cette sauvegarde si vous voulez travailler à nouveau dans IBM WebSphere Studio Site Developer version 4.0.x. En effet, si vous pointez sur un espace de travail en version 4.0.x à partir d'un IDE en version 5, vous ne pourrez plus utiliser cet espace de travail dans IBM WebSphere Studio Site Developer version 4.0.x.
4. Installez IBM WebSphere Studio Site Developer version 5.
5. Lorsque vous démarrez IBM WebSphere Studio Site Developer version 5 avec un espace de travail version 4.0.x à partir d'une ligne de commande (utilisez l'option `-data` pour indiquer le chemin qualifié complet du répertoire de l'espace de travail en version 4.0.x avec la commande `), les informations .metadata` sont mises à jour.

6. Lorsque le système vous demande de confirmer la conversion dans le nouveau format d'interface utilisateur, cliquez sur **OK**.
7. Avant de recompiler ou de valider un projet qui se trouve dans l'espace de travail, sélectionnez tous les projets de la vue Navigateur dans la perspective Ressource, puis sélectionnez **Régénérer** dans le menu en incrustation. Tous les fichiers sont ainsi synchronisés avec les métadonnées appropriées.
8. Ouvrez tous les projets fermés (voir la liste des problèmes connus ci-dessous).
9. Vérifiez les variables de chemin d'accès aux classes (voir la liste des problèmes connus ci-dessous).
10. Certains compilateurs et valideurs ont été ajoutés, supprimés ou modifiés dans la version 5. Pour vous assurer que les messages d'erreur et d'avertissement corrects s'affichent, vous devez recompiler tous les projets en sélectionnant **Projet > Régénérer tout**, puis en choisissant **Exécuter la validation** pour chaque projet Java.
11. Certaines préférences peuvent être conservées mais beaucoup sont perdues. Vérifiez les paramètres des préférences de la version 5 pour vous assurer qu'elles correspondent à vos choix.

Suppression des références de chemin d'accès absolu dans les fichiers EAR et les fichiers de configuration serveur après la migration

Reportez-vous aux opérations à effectuer après la migration décrites à la section «Suppression des références de chemin d'accès absolu dans les fichiers EAR et les fichiers de configuration serveur après la migration» à la page 34.

Incidents connus et limitations

Les incidents suivants peuvent se produire si vous tentez d'effectuer une migration en ouvrant un espace de travail en version 4.0 dans IBM WebSphere Studio Site Developer version 5.

Valeur incorrecte dans la variable du chemin d'accès aux classes JRE_LIB

Pour redéfinir la variable du chemin d'accès aux classes JRE_LIB et indiquer un emplacement valide, procédez aux opérations ci-après. *Redéfinissez cette variable même si sa valeur vous paraît correcte* lorsque vous ouvrez la fenêtre Préférences pour la première fois.

1. Sélectionnez **Fenêtre > Préférences > Java > JRE installés**.
2. Dans la liste, cochez la case correspondant à l'emplacement de l'environnement JRE par défaut que vous souhaitez associer à la variable JRE_LIB.
3. Sélectionnez **Editer**, puis cliquez sur **OK** pour fermer la boîte de dialogue Edition de JRE.

Si vous n'effectuez pas cette opération, la valeur de JRE_LIB risque d'être erronée et de générer de nombreuses erreurs de compilation dans les fichiers Java.

Pour plus de sécurité, vérifiez également la valeur de toutes les autres variables de chemin d'accès aux classes.

Le menu Equipe contient l'option Partager le projet pour les projets précédemment partagés dans SCM

La prise en charge de l'environnement coopératif a été fortement modifiée entre Eclipse 1.0 et 2.0. La méthode de partage des projets au sein du référentiel a également changé.

- Si vous cliquez à l'aide du bouton droit de la souris sur le projet et que vous sélectionnez **Equipe > Partager le projet**, un assistant s'affiche et vous guide pour effectuer la procédure de migration. Une fois l'opération terminée, le projet est partagé et la vue Synchronisation s'affiche. Des conflits liés aux modifications apparaissent pour chaque fichier. Cela s'explique par le fait que les informations de synchronisation sont stockées différemment dans Eclipse 1.0 et 2.0.
- Si vous n'avez pas de modifications en cours (ce qui devrait être le cas si vous avez validé toutes les modifications en attente avant la mise en niveau conformément aux recommandations indiquées ci-dessus), il suffit de choisir le projet dans la vue Synchronisation et de sélectionner **Remplacer et mettre à jour** pour charger le contenu en cours à partir du serveur.
- Si vous avez des modifications à émettre, vous pouvez dérouler le menu triangle dans la vue Synchronisation et sélectionner **Comparer le contenu des fichiers**. Après un délai de traitement, la vue Synchronisation présente uniquement les fichiers qui sont réellement différents. Vous pouvez ensuite utiliser la vue Synchronisation pour résoudre les conflits indiqués.

Projets créés hors du répertoire de l'espace de travail

Par défaut, les projets sont créés dans le répertoire de l'espace de travail. Si vous avez redéfini la valeur par défaut pour créer les projets dans un autre répertoire, vous devez ouvrir tous les projets avant de fermer le plan de travail. Cette option permet de placer le fichier .project dans l'emplacement approprié. Si vous ne pouvez pas ouvrir un projet dont le répertoire se trouve hors de l'espace de travail, un projet masque le projet réel, avec un seul fichier .project.

Redéfinition obligatoire des points d'arrêt JSP

Vous devez supprimer tous les points d'arrêt JSP et les redéfinir dans l'espace de travail migré de la version 5.

Migration de données relationnelles dans les projets Web 4.0.3

Pour effectuer la migration des données relationnelles à partir des projets 4.0.3 d'IBM WebSphere Studio Site Developer, procédez aux opérations ci-après.

1. A partir d'un espace de travail IBM WebSphere Studio Site Developer 4.0.3, générez des fichiers DDL pour chaque base de données disponible.
2. Supprimez la base de données du dossier source/databases du projet Web (dans la vue Définition de données).
3. Ouvrez l'espace de travail 4.0.3 avec IBM WebSphere Studio Site Developer version 5.
4. Procédez à la migration des projets Web pour lesquels vous voulez restaurer les données relationnelles.
5. Sélectionnez **Fichier > Importer > Système de fichiers** et spécifiez les fichiers DDL de votre espace de travail 4.0.3.
6. Dans la vue Définition de données de la perspective Données, sélectionnez **Exécuter sur le serveur local** et spécifiez le projet Web cible.

Les artefacts de données relationnelles seront restaurés.

Erreurs WSDL après importation d'un fichier de services Web à partir de 4.0.x

Si vous avez importé un fichier de services Web à partir d'une version 4.0.x, les messages d'erreur suivants peuvent apparaître :

Erreur Le composant 'result' comporte une valeur incorrecte 'anyElement' définie pour son type. Les déclarations de type doivent désigner des valeurs valides définies dans un schéma.

Erreur Le composant 'return' comporte une valeur incorrecte 'findPatientResult' définie pour son élément. Les déclarations d'élément doivent désigner des valeurs valides définies dans un schéma.

Erreur Le composant 'response' comporte une valeur incorrecte 'findPatientResponse' définie pour son élément. Les déclarations d'élément doivent désigner des valeurs valides définies dans un schéma.

Pour corriger ces erreurs :

1. supprimez les fichiers WSDL,
2. régénérez vos services Web en exécutant à nouveau l'assistant des services Web.

Migration des structures de projet J2EE et des niveaux de spécification J2EE

Pour accéder à l'assistant de migration J2EE de la version 5, procédez comme indiqué ci-après.

1. Sélectionnez le projet.
2. Cliquez dessus avec le bouton droit de la souris et sélectionnez **Migrer > Assistant de migration J2EE**. Suivez les instructions de l'assistant qui vous guide dans la procédure de migration.
3. Si le projet est soumis au contrôle des sources, sauvegardez le projet restructuré dans votre système SCM.

Chapitre 6. Migration à partir de WebSphere Studio "Classic" vers IBM WebSphere Studio Site Developer

Le présent chapitre décrit la procédure de migration à partir de WebSphere Studio version 4.0 (Advanced et Professional Edition) vers IBM WebSphere Studio Site Developer. La migration à partir de WebSphere Studio "Classic" version 4.0 vers IBM WebSphere Studio Site Developer version 5.0 comprend les étapes suivantes :

1. création d'une planification de serveur unique pour la migration,
2. création d'un fichier de descripteur de configuration Web,
3. exportation d'un fichier JAR de migration,
4. importation du fichier JAR de migration dans IBM WebSphere Studio Site Developer,
5. configuration du serveur et test de l'application migrée.

Remarque : Les instructions suivantes permettent d'effectuer une migration à partir de WebSphere Studio version 4.0. Pour effectuer une migration à partir d'une version antérieure de WebSphere Studio, il est recommandé de migrer d'abord vers WebSphere Studio 4.0, puis vers IBM WebSphere Studio Site Developer.

La fonction avancée de publication (mappage de fichiers à différentes étapes de la publication) et la fonction Page Detailer (analyse des pages Web) de WebSphere Studio "Classic" n'est pas disponible dans IBM WebSphere Studio Site Developer. Certaines fonctions disponibles dans le coffret de CD-ROM de la version 4.0.x ne sont plus disponibles. Par exemple, la fonction Page Detailer pour l'analyse des pages Web, la fonction HotMedia pour les types de media enrichis, l'éditeur Voice XML (déplacé dans WebSphere Everyplace Toolkit et Portal Toolkit) et DataBaseWizard pour les unités mobiles.

Vous devez prendre en compte les limitations ci-après avant d'effectuer la migration de vos données WebSphere Studio.

- IBM WebSphere Studio Site Developer utilise un éditeur SQL de type XML qui n'autorise pas l'utilisation des fichiers .sql.
- Les informations de publication des projets et les données de test des sites ne peuvent pas être migrées.
- Les informations de configuration du serveur WebSphere Studio ne peuvent pas être migrées.
- Les informations de contrôle des versions ne peuvent pas être migrées.

Lors de la procédure de migration décrite ci-après, WebSphere Studio crée un fichier JAR contenant tous les fichiers des projets, les fichiers publiables et les fichiers source pour un serveur unique. Tous les fichiers visibles dans la vue Publication du serveur par défaut sont regroupés dans le fichier JAR. Vous pouvez ensuite importer le fichier JAR dans IBM WebSphere Studio Site Developer.

Toutes les informations de planification et de publication sont perdues lors de la migration des projets. Si la planification comporte plusieurs serveurs, seuls les fichiers publiés sur le serveur par défaut sont conservés. Pour la migration, il convient donc de créer une planification comportant un seul serveur.

Création d'une planification ne comportant qu'un seul serveur pour la migration

Si la planification en cours comporte plusieurs serveurs, créez une nouvelle planification, appelée Migration, qui ne comporte qu'un seul serveur. Pour cela, procédez comme indiqué ci-après.

1. Sélectionnez **Projet > Personnalisation des modèles de publication**.
2. Entrez Migration dans la zone **Nom du modèle**.
3. Cliquez sur **Ajout**.
4. Cliquez sur **OK**.
5. Sélectionnez **Projet > Modèle de publication** et sélectionnez **Migration** dans la liste des modèles disponibles.
6. Dans la vue de publication, sélectionnez **Insertion > Serveur**.
7. Entrez un nom de serveur tel que localhost.
8. La modification du serveur ou de la phase de publication ne propage pas les informations de mappage des servlets pour WebSphere Application Server version 4.0. Accédez à la vue de publication et pour chaque servlet, sélectionnez **Propriétés > Publication > Mappage du servlet**, puis copiez le mappage de servlet actuel.

Création d'un fichier descripteur de configuration Web

1. Dans la vue du fichier de projet, cliquez sur **Projet > Création d'un fichier descripteur de configuration Web**.
2. Sélectionnez tous les servlets requis.
3. Sélectionnez les fichiers TLD (Tag Library Descriptor) requis.
4. Cliquez sur **Création**.

Le nom par défaut du fichier descripteur de configuration Web est *nomServeur_web.xml* (localhost_web.xml dans notre scénario). A moins que vous n'ayez indiqué un autre répertoire, le fichier .xml est enregistré dans le répertoire WEB-INF.

Exportation d'un fichier JAR de migration

1. Dans la vue du fichier de projet, sélectionnez le serveur **localhost** et cliquez sur **Propriétés > Publication > Chemin d'accès à l'application Web**, puis entrez un chemin Web (racine du contexte) tel que monCheminWeb. Cette valeur sera également utilisée en tant que nom du projet WebSphere Application Server - Express.
2. Dans la vue présentant les fichiers des projets, sélectionnez **Projet > Création d'un fichier de migration**.
3. Vérifiez que **localhost** est le serveur sélectionné.
4. Vérifiez que **localhost_web.xml** est le descripteur de configuration Web sélectionné.
5. Cliquez sur **OK**.
6. Le nom du fichier JAR par défaut est *nomServeur.jar*, localhost.jar dans le présent exemple. Si nécessaire, renommez ce fichier.
7. Sauvegardez le fichier JAR.

Importation du fichier JAR de migration dans IBM WebSphere Studio Site Developer

1. Démarrez IBM WebSphere Studio Site Developer.
2. Créez un projet Web (**Fichier > Nouveau > Projet > Projet Web**).
3. Dans la zone **Nom du projet**, entrez le nom du projet Web. Vous devez indiquer le même nom entré lors de l'étape 1 de la section précédente «Exportation d'un fichier JAR de migration» à la page 40.
4. Indiquez le nom d'un projet EAR, nouveau ou existant, destiné à contenir le nouveau projet Web à des fins de déploiement.
5. Dans la zone **Racine du contexte**, entrez la valeur que vous avez indiquée dans la zone Chemin d'accès Web à l'application Web lors de la création du fichier JAR de migration dans WebSphere Studio. Cliquez sur **Terminer**.
6. Dans la vue Navigateur, sélectionnez le projet Web nouvellement créé.
7. Importez le fichier JAR.
 - a. Cliquez sur **Fichier > Importer**.
 - b. Cliquez sur **Fichier WAR**. Cliquez sur **Suivant**. Pour que le fichier JAR fonctionne correctement, vous *devez* l'importer à l'aide de l'option du même nom.
 - c. Entrez le chemin d'accès au fichier localhost.jar dans la zone **Fichier WAR** ou recherchez celui-ci en cliquant sur **Parcourir** (vous pouvez uniquement rechercher les fichiers.WAR, pas les fichiers.JAR).
 - d. Sélectionnez le projet Web existant que vous avez créé. La valeur que vous avez indiquée précédemment apparaît automatiquement dans la zone **Racine du contexte**.
 - e. Cliquez sur **Terminer**. Une boîte de dialogue affiche le message suivant :
"La ressource WEB-INF/web.xml existe déjà. Voulez-vous la remplacer ?"
 - f. Cliquez sur **Oui** pour qu'IBM WebSphere Studio Site Developer décompresse *localhost.jar*.
8. Il est possible que plusieurs références ne soient pas résolues ou que des fichiers d'importation soient manquants. Dans ce cas, ils apparaissent dans la vue Tâches. Pour résoudre ces erreurs, vous devez modifier le chemin de compilation Java du projet Web :
 - a. Cliquez à l'aide du bouton droit de la souris sur le projet et sélectionnez **Propriétés > Chemin de compilation Java**.
 - b. Cliquez sur l'onglet **Bibliothèques**. Cliquez sur **Ajouter des fichiers JAR externes**.
 - c. Importez les fichiers JAR dont vous avez besoin à partir des répertoires suivants :
 - répInstall_WS/runtimes/aes_v4/lib et
 - répInstall_WS/runtimes/base_v4/lib
9. Dans la vue Navigateur, cliquez, à l'aide du bouton droit de la souris, sur le projet et sélectionnez **Recompiler un projet**.

Test d'une application migrée sur un serveur de test

Vous pouvez à présent tester votre application. Pour effectuer cette opération sur le serveur de test par défaut, procédez comme indiqué ci-après.

1. A l'aide du bouton droit de la souris, cliquez sur le projet EAR.
2. Sélectionnez **Exécuter sur le serveur**.

Pour tester l'application dans d'autres environnements d'exécution de serveur, reportez-vous à l'aide en ligne de la fonction Outils de serveur.

Chapitre 7. Migration à partir de VisualAge for Java vers IBM WebSphere Studio Site Developer

Le présent chapitre décrit la procédure à suivre pour effectuer la migration à partir de VisualAge for Java Professional Edition ou VisualAge for Java Enterprise Edition vers IBM WebSphere Studio Site Developer.

Remarque : Les instructions suivantes concernent la migration de VisualAge for Java Version 3.5.3 ou 4.0 pour Windows. Pour effectuer une migration à partir d'une version antérieure de VisualAge for Java vers IBM WebSphere Studio Site Developer, vous devez d'abord procéder à la migration à partir de la version précédente de VisualAge for Java vers la version 3.5.3 ou 4.0 pour Windows, avant de procéder à la migration vers IBM WebSphere Studio Site Developer.

Remarque :  Instantiations, Inc., Partenaire Commercial IBM, distribue un produit, CodePro Studio, qui permet d'améliorer la productivité dans VisualAge for Java et WebSphere Application Server - Express et offre des fonctions de migration et de coexistence. Pour aider les clients VisualAge for Java à commencer la migration, Instantiations offre une fonction d'exportation VisualAge for Java vers IBM WebSphere Studio Site Developer gratuite et à usage illimité, avec la copie d'évaluation de CodePro Studio, utilisable pendant une durée limitée. Vous pouvez télécharger la copie d'évaluation à partir du site www.instantiations.com/vaj-migrate. Pour plus d'informations sur la prise en charge avancée des fonctions de migration et de coexistence, comprenant l'importation/exportation bidirectionnelle complète de fichiers, la création d'ensembles d'exportation/importation, la synchronisation de projets et l'automatisation des tâches, accédez au site Instantiations, Inc. à l'adresse <http://www.instantiations.com/codepro/ws>.

Différences entre VisualAge for Java et IBM WebSphere Studio Site Developer

Vous trouverez ci-après une liste succincte des améliorations apportées depuis VisualAge for Java.

- La spécification d'EJB est passée du niveau 1.0 au niveau 1.1 (EJB 2.0 est également pris en charge pour les applications déployées sur WebSphere Application Server version 5).
- Pour les applications Web, le niveau JSP reste 1.1 (1.2 pour les applications WebSphere Application Server version 5).
- Pour les applications Web, le niveau Servlet reste 2.2 (2.3 pour les applications WebSphere Application Server version 5).
- La plateforme Java 2 prise en charge est passée du niveau 1.2 au niveau 1.3. (Le compilateur peut cibler la génération de code 1.4, mais l'environnement d'exécution de WebSphere Application Server reste au niveau 1.3).
- Visual Composition Editor a été remplacé par Visual Editor for Java.
- Le contrôle de version et le référentiel de code source des propriétés VisualAge for Java ont été remplacés par la prise en charge de modules d'extension SCM.

- L'API VisualAge for Java Tools est remplacée par l'architecture de plug-in du plan de travail WebSphere Studio.
- Les outils XML VisualAge for Java ont été remplacés par les outils XML IBM WebSphere Studio Site Developer.
- Le concept de projet VisualAge for Java a été remplacé par plusieurs types de projet IBM WebSphere Studio Site Developer.

Migration à partir de VisualAge for Java

La procédure ci-après explique comment effectuer une migration à partir de VisualAge for Java. Chaque étape est détaillée plus loin.

1. Exportation des fichiers Java et des fichiers de ressources du projet à partir de VisualAge for Java.
2. Démarrage d'IBM WebSphere Studio Site Developer et création de nouveaux projets destinés à contenir votre code.
3. Importation des fichiers de ressources des projets et des fichiers Java dans IBM WebSphere Studio Site Developer.
4. Utilisation de l'éditeur web.xml pour s'assurer que les servlets sont correctement définis (projet Web uniquement).
5. Migration des paramètres du projet et de l'espace de travail.
6. Configuration du serveur et test des applications migrées.
7. Déploiement de vos applications à partir de IBM WebSphere Studio Site Developer vers WebSphere Application Server.
8. Partage des paramètres de projet IBM WebSphere Studio Site Developer entre plusieurs développeurs (post-migration).

Exportation des fichiers Java et des fichiers de ressources de projet à partir de VisualAge for Java

La migration en vrac des projets et des ressources versionnés à partir du référentiel VisualAge for Java n'est pas prise en charge. Vous pouvez uniquement migrer des projets et des ressources qui se trouvent dans votre espace de travail VisualAge for Java. Pour effectuer la migration d'une copie versionnée d'un projet ou d'une ressource dans IBM WebSphere Studio Site Developer, placez celle-ci dans l'espace de travail VisualAge for Java et procédez ensuite à la migration.

Remarque : Si le projet contient plusieurs types de données (des beans enterprise et des fichiers de code source Java, par exemple), vous devez fragmenter les données dans plusieurs fichiers JAR, selon leur type.

Exportez les projets dans un fichier JAR en procédant comme indiqué ci-après.

1. Si les projets que vous voulez exporter ne sont pas dans le plan de travail VisualAge for Java, ajoutez-les dans ce dernier.
2. Dans la fenêtre Plan de travail VisualAge for Java, sélectionnez le projet, appuyez sur le bouton droit de la souris et cliquez sur **Exportation**.
3. Sélectionnez le bouton d'option **Fichier JAR** et cliquez sur **Suivant**.
4. Indiquez le nom du fichier JAR.
5. Cochez la case **.java** pour exporter les fichiers Java et la case **ressources** pour exporter les fichiers de ressources.
6. Renseignez comme il convient les autres zones. Pour plus d'informations sur l'exécution de cette tâche, reportez-vous à l'aide en ligne VisualAge for Java.

Démarrage d'IBM WebSphere Studio Site Developer et création de nouveaux projets destinés à contenir votre code

Démarrez IBM WebSphere Studio Site Developer puis créez les projets appropriés. Les instructions générales concernant la migration présentées ci-dessous vous aident à déterminer le type de projet IBM WebSphere Studio Site Developer dans lequel vous devez importer les fichiers :

- Si le code fait partie d'une application Web, vous devez importer celui-ci dans un projet Web.
 - Importez tous les fichiers Java dans le répertoire **Java Source** du projet Web (l'arborescence appropriée est créée automatiquement par IBM WebSphere Studio Site Developer en fonction des instructions spécifiques du module).
 - Importez tous les fichiers de ressources dans le répertoire **Web Content** du projet Web.
- Si votre code est en langage Java pur (par exemple une application exécutée de manière autonome), vous devez importer le code dans un projet Java.

Remarque : Les informations fournies précédemment sont des instructions générales destinées à vous aider à déterminer le type de projets IBM WebSphere Studio Site Developer que vous devez utiliser. Nous vous conseillons de lire l'aide en ligne de IBM WebSphere Studio Site Developer et de vous familiariser avec les différents types de projet WebSphere Application Server - Express avant de créer un projet ou d'importer du code.

Importation des fichiers de ressources et des fichiers Java dans IBM WebSphere Studio Site Developer

1. Ouvrez IBM WebSphere Studio Site Developer et passez à la perspective Ressource.
2. Sélectionnez **Fichier > Importer > Fichier zip**. Cliquez sur **Suivant**.
3. Recherchez le fichier JAR approprié.
4. Sélectionnez les fichiers à importer, ainsi que le projet ou le dossier devant contenir les fichiers.

Remarque :

- Vous pouvez utiliser l'importation du système de fichiers à la place de l'importation de fichier ZIP, mais l'importation de fichier ZIP est plus couramment utilisée. Suivez les mêmes instructions si vous souhaitez importer un fichier et indiquer un dossier plutôt qu'un fichier ZIP dans l'assistant d'importation de fichier.
- Lorsque vous importez les fichiers dans IBM WebSphere Studio Site Developer, vérifiez que ces derniers sont placés dans le répertoire approprié. Nous vous conseillons de lire l'aide en ligne de IBM WebSphere Studio Site Developer et de vous familiariser avec les différents types de projet IBM WebSphere Studio Site Developer avant de créer un projet ou d'importer du code. Vous pourrez ainsi mieux déterminer les dossiers adaptés à chaque type de code.

Utilisation de l'éditeur web.xml pour vérifier que les servlets sont correctement définis (projet Web uniquement)

Si l'application utilise des servlets, vous devez définir les correspondances entre les servlets et les URL dans le fichier web.xml. Procédez comme indiqué ci-après.

1. Dans la perspective Web, ouvrez le fichier web.xml, disponible dans le sous-répertoire Web Content/WEB-INF de votre projet Web.
2. Cliquez sur l'onglet **Servlets**.
3. Cliquez sur **Ajouter** et sélectionnez le bouton d'option **Servlet**.
4. Entrez le nom du servlet et cliquez sur **OK**.
5. Cliquez sur **Parcourir** pour remplacer la valeur **Classe de servlet** par le nom de package approprié.
6. (Facultatif) Le nom affiché est un nom court permettant d'identifier le servlet. Dans la zone **Nom affiché**, entrez le nom court du servlet.
7. Un mappage des adresses URL définit un servlet et un modèle d'URL. Cliquez sur le bouton **Ajouter** situé en regard de la zone **Mappages des adresses URL**, puis entrez le nom du mappage correspondant.
8. Sauvegardez les modifications (**Fichier > Sauvegarder web.xml**) et fermez le fichier web.xml.

Migration des paramètres du projet et de l'espace de travail

Vous devez enregistrer les paramètres de VisualAge for Java et les définir dans IBM WebSphere Studio Site Developer :

- Chemin d'accès aux classes du projet
- Associations de ressources
- Formatage du code
- Configuration WTE
- Fichiers Java et fichiers de ressources du projet

Chemin d'accès à la classe du projet

Dans VisualAge for Java, vous définissez le chemin d'accès aux classes du projet dans la page Ressources de la fenêtre Options (**Fenêtre > Options > Ressources**). Après avoir migré les projets dans IBM WebSphere Studio Site Developer, vous pouvez définir le chemin d'accès à la classe du projet dans la fenêtre Propriétés du projet. (Cliquez avec le bouton droit sur le projet et sélectionnez **Propriétés > Chemin de compilation Java**. Cliquez sur l'onglet **Bibliothèques**.) Vous pouvez également définir les variables de chemin d'accès aux classes dans la fenêtre Préférences (**Fenêtre > Préférences > Java > Variables d'accès aux classes**.)

Associations de ressources

Si vous définissez une association entre un type de fichier et un exécutable, vous pouvez ouvrir un fichier dans le plan de travail même si celui-ci ne s'y trouve pas.

Dans VisualAge for Java, vous définissez les associations de ressources dans la fenêtre Options (**Fenêtre > Options > Ressources > Associations de ressources**). Après avoir migré les fichiers de ressources dans IBM WebSphere Studio Site Developer, vous pouvez définir les associations de ressources dans la fenêtre Préférences (**Fenêtre > Préférences > Plan de travail > Association de fichier**).

Outil de formatage du code

Dans VisualAge for Java, vous définissez les options de formatage du code dans la page Module de formatage de la fenêtre Options (**Fenêtre > Options > Codage > Module de formatage**). Après avoir migré le code vers IBM WebSphere Studio Site

Developer, vous pouvez définir le formatage du code dans la fenêtre Préférences (**Fenêtre > Préférences > Java > Outil de formatage du code**).

Configuration WTE

Dans VisualAge for Java, les paramètres d'exécution de l'environnement de test WebSphere et WebSphere Application Server sont définis dans plusieurs fichiers de propriétés figurant dans le répertoire suivant :

RépInstallVisualAge\ide\project_resources\IBM WebSphere Test Environment\properties, où *RépInstallVisualAge* est le répertoire d'installation du produit.

Si, par exemple, vous avez activé la réécriture des URL dans le fichier de propriétés *session.xml* en attribuant la valeur *true* à la propriété comme indiqué ci-après, `<url-rewriting-enabled>true</url-rewriting-enabled>`, vous pouvez configurer cette propriété dans l'environnement de test d'IBM WebSphere Studio Site Developer, version 4.0. (Dans la perspective du serveur, ouvrez la vue Configuration de serveur, cliquez avec le bouton droit de la souris sur le serveur que vous voulez utiliser et sélectionnez **Ouvrir**. Cliquez sur l'onglet **Web** et cochez la case **Activer la réécriture des URL**).

Fichiers Java et fichiers de ressources du projet

Le fichier de propriétés *default.servlet_engine* contient la racine du contexte `<root-uri>` des applications Web VisualAge for Java. Lors de la création d'un projet Web dans IBM WebSphere Studio Site Developer, la boîte de dialogue **Créer un projet Web** contient une zone **Racine du contexte** réservée à ces données.

Les paramètres de l'application Web figurant dans des fichiers, tels que *RépInstallVisualAge\ide\project_resources\IBM WebSphere Test Environment\hosts\default_host\default_app\servlets\default_app.webapp* que vous avez personnalisés doivent être migrés dans le fichier *Votre_projet_Web\Web Content\WEB-INF\web.xml* dans IBM WebSphere Studio Site Developer. Par exemple, si vous avez modifié les noms de servlet et les chemins d'accès aux servlets dans le fichier *default_app.webapp*, vous devez effectuer les modifications correspondantes dans le fichier *web.xml*.

Configuration de l'environnement de test WebSphere 4 et test des applications ayant fait l'objet d'une migration

Si l'application est un projet Java, vous pouvez utiliser le support normal d'**exécution** ou de **débogage** d'IBM WebSphere Studio Site Developer des projets Java pour tester l'application.

Si l'application utilise WebSphere Application Server, vous devez la tester à l'aide du système WebSphere Application Server intégré. Il faut pour cela créer et démarrer un serveur de test par défaut. Dans le cas d'un projet Web, cliquez avec le bouton droit sur la page HTML principale et sélectionnez **Exécuter sur le serveur** pour lancer le navigateur web.

Pour plus d'informations sur le test d'autres types de projet, consultez l'aide en ligne.

Déploiement de vos applications à partir d'IBM WebSphere Studio Site Developer vers un serveur distant WebSphere Application Server

Si vous utilisez WebSphere Application Server en tant qu'environnement d'exécution, déployez les applications à l'aide de la fonction Outils de serveur de IBM WebSphere Studio Site Developer.

Partage des paramètres de projet IBM WebSphere Studio Site Developer entre plusieurs développeurs (post-migration)

Les projets d'IBM WebSphere Studio Site Developer (et leurs paramètres) peuvent être partagés entre les développeurs. Pour ce faire, enregistrez un projet sur le serveur SCM (Software Configuration Management) d'IBM WebSphere Studio Site Developer et procédez à son extraction sur le système d'un membre d'une autre équipe IBM WebSphere Studio Site Developer.

Support de développement coopératif dans IBM WebSphere Studio Site Developer

Pour plus d'informations sur le support de développement coopératif dans IBM WebSphere Studio Site Developer version 4.0, consultez l'article suivant : www.ibm.com/websphere/developer/library/techarticles/0108_karasiuk/0108_karasiuk.html

Le guide d'installation et l'aide en ligne contiennent également des informations sur le support de développement coopératif dans IBM WebSphere Studio Site Developer.

Chapitre 8. Migration à partir de VisualAge for Java Visual Composition Editor vers Visual Editor for Java

Le présent chapitre décrit les instructions à suivre pour migrer des applications créées dans l'éditeur de composition visuelle VisualAge for Java vers Visual Editor for Java dans WebSphere Application Server - Express :

- Sauvegarde des métadonnées de la phase de conception avancée à partir de VisualAge for Java
- Exécution de la migration (importation dans WebSphere Studio)

Sauvegarde des informations avancées relatives aux métadonnées à partir de VisualAge for Java

Cette étape n'est pas obligatoire mais il est fortement conseillé de l'exécuter (en particulier si l'application comporte des connexions) pour les raisons ci-après.

- VisualAge for Java n'a pas sauvegardé les informations relatives au placement des beans de niveau supérieur (beans contenus dans d'autres beans) sur la surface de format libre. A l'opposé, WebSphere Studio sauvegarde toujours ces informations sous forme de commentaire sur la ligne de code qui déclare le bean et restaure ce dernier à cet emplacement sur le canevas à chaque ouverture de l'éditeur visuel. Vous avez la possibilité de sauvegarder les informations relatives à la phase de conception dans VisualAge for Java avant la migration. Si ces données ne sont pas sauvegardées, l'éditeur calcule l'emplacement par défaut des beans de niveau supérieur (également appelés composants au format libre) lors de l'ouverture initiale des fichiers .java ; cet emplacement peut être modifié aisément par le biais d'une opération glisser-déposer.
- Bien que le nouvel éditeur Visual Editor for Java dans WebSphere Application Server - Express ne prenne pas en charge la conception de connexion, cette fonction pourra être ajoutée ultérieurement. (**Remarque** : Il ne s'agit pas d'un engagement mais d'une simple éventualité.)

Pour sauvegarder les informations avancées relatives aux métadonnées avant la migration, procédez comme indiqué ci-après.

1. Accédez à VisualAge for Java Developer Domain [www.software.ibm.com/vad/data/document4293] et téléchargez l'utilitaire *IBM VCE Code Generation and Export Utility*.
2. Conformément aux instructions fournies dans le readme, ajoutez l'outil dans VisualAge for Java, puis arrêtez et relancez VisualAge for Java.
3. Versionnez le code de l'application en cours dans le référentiel VisualAge for Java (pour vous permettre de revenir à cette version lors d'une phase de développement VisualAge for Java).
4. Pour chaque application graphique dans VisualAge for Java, sélectionnez un ou plusieurs programmes graphiques (XxxxxView en général), cliquez dessus avec le bouton droit de la souris et effectuez les opérations ci-après.
 - a. Cliquez sur **Exportation/Génération du code VCE** et laissez l'option d'exportation dans un répertoire après la génération du code sélectionnée.
 - b. Cliquez sur **Fin**.
 - c. Conservez **Répertoire** en tant que destination sélectionnée pour l'exportation et cliquez sur **Suivant**.

- d. Sélectionnez le répertoire cible en désélectionnant l'option **.class** et en sélectionnant **.java** (puisque vous recherchez le code source) et cliquez ensuite sur **Fin**.
- e. Si vous le souhaitez, vous pouvez également recharger le code VisualAge for Java avec la version précédente (cliquez à l'aide du bouton droit de la souris et sélectionnez **Remplacement par > Edition précédente**).

Exécution de la migration (importation dans WebSphere Studio)

Vous pouvez à présent importer vos classes dans WebSphere Application Server - Express. Voir Chapitre 7, «Migration à partir de VisualAge for Java vers IBM WebSphere Studio Site Developer», à la page 43. Après avoir importé vos programmes sources Visual Composition Editor dans WebSphere Application Server - Express, vous pouvez les mettre à jour dans Visual Editor for Java.

Chapitre 9. Configuration de la compilation (bibliothèque, fichiers JAR, fichiers JAR dépendant d'un projet, compilation Ant)

La structure du présent chapitre est la suivante :

- Fichiers JAR de bibliothèque Java et fichiers JAR tiers externes
- Optimisation de compilations multiprojets avec des fichiers JAR dépendant d'un projet
- Compilations de production automatisées à l'aide de Ant

Fichiers JAR des bibliothèques Java et fichiers JAR tiers

Pour obtenir des explications détaillées concernant les éléments impliqués, reportez-vous à l'article relatif au chargement des classes J2EE *J2EE Class Loading Demystified*

(www.ibm.com/websphere/developer/library/techarticles/0112_deboer/deboer.html) (modules J2EE et chemin d'accès aux classes) et au développement de fichiers JAR d'utilitaire J2EE

(www.ibm.com/websphere/developer/library/techarticles/0112_deboer/deboer2.html) (fichiers JAR Java dans les modules J2EE). Vous y trouverez des informations techniques et des conseils très intéressants.

Méthode recommandée en cas d'utilisation d'un fichier JAR tiers dans un projet Web

Pour utiliser un fichier JAR tiers dans un projet Web, il convient de l'importer dans le dossier des bibliothèques du projet (en le conservant en tant que fichier JAR). Cette méthode d'utilisation d'un fichier JAR est la seule méthode J2EE compatible qui évite d'effectuer des modifications lors du déploiement sur un autre serveur.

Pour utiliser un fichier JAR externe dans un seul projet Web, suivez la procédure ci-dessous. Si vous devez utiliser le fichier JAR dans plusieurs projets Web, suivez plutôt la procédure présentée à la section «Méthode recommandée en cas d'utilisation d'un fichier JAR tiers dans plusieurs projets Web» à la page 51.

1. Sélectionnez **Fichier > Importer > Système de fichiers**. Cliquez sur **Suivant**. Vous devez sélectionner **Système de fichiers** et non **Fichier Zip** pour que le fichier JAR ne soit pas décompressé lors de l'importation.
2. Cliquez sur **Parcourir** et recherchez le répertoire contenant le fichier JAR.
3. Importez-le dans le dossier *ProjetWeb/WebContent/WEB-INF/lib*, où *ProjetWeb* représente le nom du projet Web.
4. Cliquez sur **Terminer**. Le fichier JAR est automatiquement ajouté dans le chemin de compilation Java et aucune modification supplémentaire n'est requise lors de l'exécution.

Méthode recommandée en cas d'utilisation d'un fichier JAR tiers dans plusieurs projets Web

Pour utiliser un fichier JAR tiers dans plusieurs projets Web, il convient de l'importer dans le projet EAR (en le conservant en tant que fichier JAR). Cette méthode d'utilisation d'un fichier JAR est la seule méthode J2EE compatible qui évite d'effectuer des modifications lors du déploiement sur un autre serveur.

Pour utiliser un fichier JAR externe avec plusieurs projets Web, suivez la procédure ci-dessous. Si vous devez uniquement utiliser le fichier JAR dans un seul projet Web, suivez les étapes présentées à la section précédente.

1. Sélectionnez **Fichier > Importer > Système de fichiers**. Cliquez sur **Suivant**. Vous devez sélectionner **Système de fichiers** et non **Fichier Zip** pour que le fichier JAR ne soit pas décompressé lors de l'importation.
2. Cliquez sur **Parcourir** et recherchez le répertoire contenant le fichier JAR.
3. Importez le fichier JAR dans le projet d'application d'entreprise qui contient les projets Web.
4. Cliquez sur **Terminer**. Le fichier JAR est automatiquement ajouté dans le chemin de compilation Java et aucune modification supplémentaire n'est requise lors de l'exécution.
5. Suivez la procédure décrite à la section suivante pour ajouter le fichier JAR à la page Dépendances du projet Web.

Autre mode d'utilisation des fichiers JAR externes (chemin de compilation globale et chemin d'accès aux classes du serveur)

Vous pouvez laisser le fichier JAR en dehors de WebSphere Application Server - Express et l'ajouter dans le chemin de compilation Java et dans le chemin d'accès aux classes de l'instance du serveur. Cette solution n'est pas conseillée car elle rend difficile la portabilité de votre application. Lorsque vous devez effectuer un transfert vers un autre serveur, il est nécessaire de mettre à jour le chemin d'accès aux classes du serveur. De même, vous devez vérifier que vos fichiers de classes n'entrent pas en conflit avec d'autres versions de fichiers de classes similaires déjà définis dans le chemin d'accès aux classes du serveur (ces fichiers sont requis par le serveur et d'autres applications de ce dernier). Si vous optez pour cette procédure, procédez comme indiqué ci-après.

1. Ajoutez le fichier JAR externe dans le chemin d'accès aux classes de compilation Java du projet pour lequel le fichier JAR est requis.
 - a. Sélectionnez le projet, cliquez dessus avec le bouton droit de la souris et sélectionnez **Propriétés** dans le menu en incrustation.
 - b. Cliquez sur **Chemin de compilation Java**.
 - c. Cliquez sur l'onglet **Bibliothèques**.
 - d. Cliquez sur **Ajouter des fichiers JAR externes**. Sélectionnez le fichier JAR et cliquez sur **Ouvrir**.
 - e. Cliquez sur **OK**.
2. Ajoutez le fichier JAR externe dans le chemin d'accès aux classes de l'instance du serveur
 - a. Ouvrez la vue Configuration de serveur et développez le dossier **Serveur**.
 - b. Sélectionnez l'instance de serveur sur laquelle le projet est déployé. Cliquez sur le serveur avec le bouton droit de la souris et sélectionnez **Ouvrir**.
 - c. Cliquez sur l'onglet **Chemins d'accès**.
 - d. Dans `ws.ext.dirs`, cliquez sur **Ajouter des fichiers JAR externes**. Sélectionnez le fichier JAR et cliquez sur **Ouvrir**. `ws.ext.dirs` est utilisé pour les fichiers JAR d'application, et `CLASSPATH` est utilisé pour les fichiers JAR du serveur.
 - e. Fermez l'instance de serveur et enregistrez les modifications.

Optimisation de compilations multiprojets à l'aide de fichiers JAR dépendant d'un projet

La fonction d'auto-compilation de WebSphere Application Server - Express peut ralentir la vitesse d'exécution de compilations de plusieurs projets complexes. Vous pouvez contrôler ces auto-compilations à l'aide de fichiers dépendants, en activant ou en désactivant des projets, ou en utilisant des projets au format source ou JAR. Ces options peuvent toutefois être complexes à mettre en oeuvre. Pour plus d'informations sur ces options et sur l'optimisation de la compilation, reportez-vous à l'article WebSphere Developer Domain "Optimizing Multi-Project Builds Using dependent Project JARs in WebSphere Studio Application Developer" (www.ibm.com/websphere/developer/library/techarticles/0204_searle/searle.html).

Compilations de production automatisées avec Ant

Vous pouvez utiliser Ant avec WebSphere Application Server - Express pour automatiser des compilations de production. Un article complet sur ce sujet explique les points suivants :

- Présentation de Ant
- Comment exécuter Ant à partir et en dehors de WebSphere Application Server - Express
- Comment utiliser Ant pour des compilations de production d'éléments J2EE (EJB, WAR, EAR, etc.)
- Comment étendre Ant à l'aide de nouvelles tâches de compilation WebSphere Application Server - Express

Consultez l'article "Using Ant with WebSphere Studio Application Developer" dans le domaine WebSphere Developer (www.ibm.com/websphere/developer/library/techarticles/0203_searle/searle1.html).

Chapitre 10. Exemples de migration

Le présent chapitre contient des exemples de migration destinés à vous aider à en savoir plus sur la migration de VisualAge for Java et WebSphere Studio "Classic" vers WebSphere Application Server - ExpressIBM WebSphere Studio Site Developer.

- Exemple de servlet/JSP VisualAge for Java (LeapYear)
- Exemple d'application WebSphere Studio "Classic" (YourCo)

Exemple de servlet/JSP VisualAge for Java (LeapYear)

Description

Il s'agit de l'exemple FindTheLeapYears fourni avec VisualAge for Java version 4.0. Vous trouverez des informations le concernant dans l'aide en ligne de VisualAge for Java (**Exemples > Environnement de développement JSP/Servlet**).

Migration - Généralités

Vous pouvez exécuter la procédure suivante pour migrer l'exemple de VisualAge for Java vers IBM WebSphere Studio Site Developer. Ces étapes sont présentées en détail ci-dessous :

1. Exportation des fichiers de ressources de projet et des fichiers Java à partir de VisualAge for Java.
2. Création d'un projet Web dans IBM WebSphere Studio Site Developer.
3. Importation des fichiers de ressources des projets et des fichiers Java dans le projet IBM WebSphere Studio Site Developer.
4. Définition des servlets et mise en oeuvre des modifications de restructuration d'applications requises.
5. Création d'un projet serveur IBM WebSphere Studio Site Developer.
6. Test de l'application migrée.

Exportation des fichiers à partir de VisualAge for Java

1. Ouvrez VisualAge for Java.
2. Sélectionnez le projet **IBM JSP Examples**.
3. A l'aide du bouton droit de la souris, cliquez sur le projet et sélectionnez **Exportation**. Sélectionnez le bouton d'option **Répertoire** et cliquez sur **Suivant**.
4. Entrez le nom du répertoire dans lequel vous voulez exporter les fichiers.
5. Décochez la case **.class**. Il n'est pas nécessaire d'exporter ces fichiers dans la mesure où vous allez recompiler le projet dans WebSphere Application Server - Express et recréer ces fichiers.
6. Cochez la case **.java** et cliquez sur **Détails**. Sélectionnez uniquement les fichiers LeapYear et cliquez sur **OK**.
7. Cochez la case **ressource** et cliquez sur **Détails**.
8. Sélectionnez les fichiers **LeapYearInput.html** et **LeapYearResults.jsp** qui se trouvent dans le répertoire suivant : **IBM WebSphere Test Environment\hosts\default_host\default_app\web\JSP\sample3**.

9. Cliquez sur **OK**.
10. Décochez la case **Création d'un fichier manifest** (il n'est pas nécessaire de créer un fichier de manifeste).
11. Cliquez sur **Fin**.
12. Fermez VisualAge for Java.

Création d'un projet Web IBM WebSphere Studio Site Developer

1. Démarrez IBM WebSphere Studio Site Developer.
2. Créez un projet Web (**Fichier > Nouveau > Projet > Web > WebProject**) appelé LeapYear.
3. Assurez-vous que le **projet Web dynamique** est sélectionné, puis cliquez sur **Suivant**.
4. Sélectionnez **Nouveau**.
5. Remplacez le **nom du projet d'application d'entreprise** par LeapYearEAR et sélectionnez **Niveau J2EE 1.2**. Vous pouvez placer le projet Web dans un projet EAR (Enterprise Application) existant, mais dans cet exemple vous devez le placer dans LeapYearEAR.
6. Conservez LeapYear dans la zone **Racine du contexte**.
7. Cliquez sur **Terminer**.

Importation des fichiers de ressources des projets et des fichiers Java dans le projet IBM WebSphere Studio Site Developer

Importez les fichiers source Java dans le répertoire source du projet LeapYear en procédant comme indiqué ci-après.

1. Dans la perspective Web, développez LeapYear et sélectionnez le répertoire **JavaSource**.
2. Sélectionnez **Fichier > Importer > Système de fichiers** et cliquez sur **Suivant**. Recherchez le répertoire dans lequel vous avez exporté les fichiers et cliquez sur **OK**.
3. Vous voulez uniquement importer les fichiers source Java dans le répertoire **JavaSource**. Dans la fenêtre d'importation, développez le répertoire d'exportation et sélectionnez uniquement le sous-répertoire **com**. (Il contient les trois fichiers source Java).
4. Cliquez sur **Terminer**. Cette opération crée les fichiers LeapYear\JavaSource\com\ibm\ivj\wte\samples\leapyear\LeapYearXXXX.java. Les classes Java sont automatiquement compilées dans LeapYear\WebContent\WEB-INF\classes.

Importez les fichiers de ressources dans le projet LeapYear situé dans le répertoire **WebContent** en procédant comme indiqué ci-après.

1. Dans la perspective Web en cours, développez le projet LeapYear et sélectionnez le répertoire **WebContent**.
2. Sélectionnez **Fichier > Importer > Système de fichiers** et cliquez sur **Suivant**. Recherchez le répertoire dans lequel vous avez exporté les fichiers, développez le répertoire d'exportation dans le sous-répertoire **sample3** et cliquez sur **OK**.
3. Etant donné que vous devez importer uniquement les fichiers de ressources dans le répertoire **WebContent**, sélectionnez le sous-répertoire **sample3** qui contient les fichiers .jsp et .html dans la boîte de dialogue d'importation.

4. Cliquez sur **Terminer**. Les fichiers sont importés dans le répertoire **WebContent**.

Définition des servlets et mise en oeuvre des modifications d'application restructurée

1. Vous devez maintenant créer un servlet. Sélectionnez le projet LeapYear et développez-le (**Leap Year > WebContent > WEB-INF**) afin d'afficher le fichier web.xml. Ouvrez le fichier web.xml.
2. Cliquez sur l'onglet **Servlets** situé au bas de la page.
3. Cliquez sur **Ajouter**.
4. Vérifiez que le bouton d'option **Servlet** est sélectionné.
5. Sélectionnez la classe **LeapYear**, puis cliquez sur **OK**.
6. Sélectionnez **Mappage d'URL > Ajouter** et entrez ensuite LeapYear.
7. Sauvegardez les modifications (**Fichier > Sauvegarder web.xml**) et fermez le fichier web.xml.

Vous devez maintenant apporter les modifications nécessaires aux applications en raison d'un léger changement de structure de l'application/source.

1. Deux erreurs doivent apparaître dans la vue Tâches. L'une concerne le fichier LeapYearInput.html et l'autre le fichier LeapYearResults.jsp.
2. Ouvrez le fichier LeapYearResults.jsp. Remplacez `/JSP/index.html` par `LeapYearInput.html`.
3. Ouvrez le fichier LeapYearInput.html. Remplacez `/servlet/com.ibm.ivj.wte.samples.leapyear.LeapYear` par `LeapYear`.
4. Sauvegardez les modifications et fermez les fichiers LeapYearResults.jsp et LeapYearInput.html.
5. Pour éviter de générer une erreur d'exécution, ouvrez le fichier LeapYear.java, disponible dans le sous-répertoire `JavaSource\com\ibm\ivj\wte\samples\leapyear`.
6. Accédez à la ligne 118 et modifiez `getRequestDispatcher` en remplaçant `"/JSP/Sample3/LeapYearResults.jsp"` par `"LeapYearResults.jsp"`.
7. Sauvegardez les modifications et fermez le fichier LeapYear.java.

A ce stade, l'exemple a été migré vers IBM WebSphere Studio Site Developer. Il suffit maintenant de créer un projet serveur IBM WebSphere Studio Site Developer et de tester l'exemple dans l'environnement de test WebSphere.

Création d'un projet serveur IBM WebSphere Studio Site Developer

1. Sélectionnez **Fichier > Nouveau > Projet > Serveur > Projet serveur**. Cliquez sur **Suivant**. Dans la zone **Nom du projet**, entrez `newServer` et cliquez sur **Terminer**. Vous accédez directement à la perspective **Serveur**.
2. A l'aide du bouton droit de la souris, cliquez sur **newServer** et sélectionnez **Nouveau > Serveur et configuration de serveur**.
3. Dans la zone **Nom du serveur**, entrez `WSTestEnv`. Dans la zone **Type de serveur**, sélectionnez **Environnement de test WebSphere V4.0**. Cliquez sur **Terminer**.

Vous devez maintenant définir le projet EAR dans la configuration du serveur.

1. Dans la zone Configuration de serveur, développez les options disponibles et cliquez sur **WSTestEnv**.
2. Cliquez sur cette entrée à l'aide du bouton droit de la souris et sélectionnez **Ajouter > LeapYearEAR**.

Test de l'application LeapYear migrée

1. Sélectionnez le fichier LeapYearInput.html.
2. A l'aide du bouton droit de la souris, cliquez sur le fichier HTML. Dans le menu en incrustation, sélectionnez **Exécuter sur le serveur**.
3. Patientez pendant le démarrage du serveur. Affichez la page **Console** (en cliquant sur l'onglet **Console** de la vue Serveurs) jusqu'à ce que le message "Serveur par défaut ouvert pour l'e-business" s'affiche.
4. Lorsque le navigateur apparaît, entrez 2001 dans la zone **Start Year** et cliquez sur **Soumettre**.
5. Le message LeapYear: init apparaît dans la vue Console. Attendez que la liste des années bissextiles apparaisse à l'écran et sélectionnez **WSTestEnv** dans la vue Serveurs. Cliquez dessus avec le bouton droit de la souris et sélectionnez **Ouvrir**.

Exemple d'application Web WebSphere Studio "Classic" version 4.0 (YourCo) (Windows)

Description

Pour cet exemple, vous devez utiliser WebSphere Studio "Classic" version 4.0.x.

L'exemple que vous allez utiliser s'appelle **YourCo**. Pour y accéder, ouvrez l'aide en ligne (**Aide > WebSphere Studio 4.0 > Procédures > Utilisation des exemples > Généralités**). Pour charger cet exemple, suivez les instructions fournies à la rubrique **Ouverture des exemples Studio** (pour WebSphere Application Server 4.0) et chargez YourCo.war.

Remarque : L'application migrée sera exécutée dans IBM WebSphere Studio Site Developer, mais ce produit n'offre pas actuellement toutes les fonctions de conception et de développement de page Web de WebSphere Studio, Professional ou Advanced Edition.

Etapes préalables

- Assurez-vous que l'application exemple YourCo est chargée dans WebSphere Studio "Classic".
- Arrêtez les instances de WebSphere Application Server (pour éviter tout risque de conflit avec IBM WebSphere Studio Site Developer).

Etapes de la migration

Pour migrer cet exemple à partir de WebSphere Studio "Classic" vers IBM WebSphere Studio Site Developer, suivez la procédure ci-dessous. Chaque étape est décrite en détail ci-dessous.

1. (Facultatif) Lancement de WebSphere Studio "Classic" et création d'une phase de migration.
2. Création d'un fichier de descripteur de configuration Web.
3. Création d'un fichier WAR de migration (contenant un chemin d'accès Web).

4. Démarrage d'IBM WebSphere Studio Site Developer et importation du fichier WAR.
5. Création d'un projet serveur IBM WebSphere Studio Site Developer.
6. Test de l'application migrée.

Lancement de WebSphere Studio "Classic" version 4.0 et création d'une phase de migration

(Facultatif) Il convient normalement de créer une phase de migration mais, pour les besoins de cet exemple, vous devez utiliser la phase de test incluse dans WebSphere Studio "Classic". En procédant ainsi, il n'est pas nécessaire de modifier manuellement les mappages de servlets comme indiqué à l'étape 8.

Pour plus d'informations sur la création d'une phase de migration, reportez-vous à la section "Migration à partir de WebSphere Studio "Classic" vers IBM WebSphere Studio Site Developer"

Création d'un fichier descripteur de configuration Web

1. Dans la vue du fichier de projet, sélectionnez **Projet > Création d'un fichier descripteur de configuration Web** et validez la valeur par défaut WEB-INF\localhost_web.xml.
2. Sélectionnez tous les servlets requis (tout fichier n'ayant pas pour nom xxxxBean).
3. Il n'existe aucun fichier TLD (Tag Library Descriptor) pour cet exemple.
4. Cliquez sur **Création**.

Création d'un fichier de migration

1. Dans la vue du fichier de projet, sélectionnez le serveur *localhost*, cliquez sur **Propriétés > Publication > Chemin d'accès Web à l'application Web** et entrez un chemin Web (racine du contexte), "newStudioSample". (La procédure qui consiste à définir un chemin Web dans le produit final IBM WebSphere Studio Site Developer est conseillée).
2. Dans la vue présentant les fichiers des projets, sélectionnez **Projet > Création d'un fichier de migration**.
3. Vérifiez que **localhost** est le serveur sélectionné.
4. Vérifiez que **localhost_web.xml** est le descripteur de configuration Web sélectionné.
5. Cliquez sur **OK**.
6. Le fichier JAR par défaut s'appelle `X:\Studio40\projects\YourCo\localhost.jar`, où X est le répertoire d'installation de WebSphere Studio.
7. Cliquez sur **Sauvegarder**.
8. Fermez WebSphere Studio "Classic".
9. Renommez le fichier localhost.jar en localhost.war.

Démarrage d'IBM WebSphere Studio Site Developer et importation du fichier WAR

1. Démarrez IBM WebSphere Studio Site Developer.
2. Sélectionnez **Fichier > Importer > Fichier WAR > Suivant**.

Remarque : Pour que le fichier JAR fonctionne correctement, vous devez *obligatoirement* l'importer à l'aide de l'option de fichier WAR.

3. Entrez le chemin d'accès au fichier localhost.war dans la zone **Fichier WAR** ou recherchez celui-ci en cliquant sur **Parcourir**.
4. Dans la zone **Projet Web**, sélectionnez **Nouveau** et entrez newStudioSample.
5. Dans la zone **Nom du projet EAR**, sélectionnez **Nouveau** et entrez newStudioSampleEAR.
6. Cliquez sur **Terminer**. IBM WebSphere Studio Site Developer décompresse localhost.war.
7. Vous allez obtenir un grand nombre de références non résolues ou de fichiers manquants. Ces éléments apparaissent dans la vue des tâches.
 - a. com.ibm.db requiert le fichier databeans.jar,
 - b. com.ibm.webtools.runtime requiert le fichier webtlsrn.jar,
 - c. com.ibm.ejs.ns.jndi requiert le fichier ns.jar,
 - d. com.ibm.websphere.advanced.cm.factory requiert le fichier cm.jar,
 - e. com.ibm.ejs.models.base.extensions.webappext.ServletExtensions requiert le fichier ws-base-extensions.jar

Pour résoudre cet incident, vous devez modifier le chemin de compilation Java du projet Web.

1. Cliquez à l'aide du bouton droit de la souris sur le projet et sélectionnez **Propriétés > Chemin de compilation Java**.
2. Cliquez sur l'onglet **Bibliothèques**. Cliquez sur **Ajouter des fichiers JAR externes**.
3. Importez les fichiers JAR suivants : **databeans.jar**, **webtlsrn.jar**, **ns.jar**, **cm.jar** et **ws-base-extensions.jar** à partir du répertoire MyInstall\runtimes\aes_v4\lib
4. Vingt-quatre avertissements sont encore générés. Vous pouvez les ignorer.
1. Cliquez avec le bouton droit de la souris sur le projet **newStudioSample** et sélectionnez **Regénérer un projet**.

A ce stade, l'exemple a été migré vers IBM WebSphere Studio Site Developer. Il suffit maintenant de créer un projet serveur IBM WebSphere Studio Site Developer et de tester l'exemple dans l'environnement de test WebSphere.

Création d'un projet serveur IBM WebSphere Studio Site Developer

1. Accédez à la perspective du serveur.
2. Sélectionnez **Fichier > Nouveau > Projet > Serveur > Projet serveur**. Cliquez sur **Suivant**. Dans la zone **Nom du projet**, entrez newServer et cliquez sur **Terminer**.
3. Dans la vue Navigateur, cliquez avec le bouton droit de la souris sur **newServer** et sélectionnez **Nouveau > Serveur et configuration de serveur**.
4. Dans la zone **Nom du serveur**, entrez WSTestEnv. Dans la zone **Type de serveur**, sélectionnez **Environnement de test WebSphere V4.0**. Cliquez sur **Terminer**.

Vous devez maintenant définir le projet EAR dans la configuration du serveur.

1. Dans la vue Configuration de serveur, cliquez sur **Serveurs > WSTestEnv**.
2. Cliquez dessus avec le bouton droit de la souris et sélectionnez **Ajouter > newStudioSampleEAR**.

Remarque : (Facultatif) Cliquez avec le bouton droit de la souris sur le projet newStudioSample, sélectionnez **Propriétés > Préférences du serveur >**

Toujours exécuter sur le serveur suivant et **WSTestEnv** puis **Appliquer > OK**. (Cette étape est nécessaire uniquement si d'autres serveurs sont installés.)

Test de l'application YourCo migrée

1. Sélectionnez le fichier **YourCoIntro.html** stocké dans le répertoire `WebContent\StudioSamples` du projet `NewStudioSample`.
2. Cliquez avec le bouton droit de la souris sur le fichier **YourCoIntro.html** et sélectionnez **Exécuter sur le serveur**, puis **WSTestEnv** dans le menu en incrustation.
3. Patientez pendant le démarrage du serveur. Affichez la page **Console** (en cliquant sur l'onglet **Console** de la vue **Serveurs**) jusqu'à ce que le message **Serveur par défaut ouvert pour l'e-business s'affiche**.
4. Si vous n'avez pas encore exécuté cet exemple dans **WebSphere Studio "Classic"**, vous devez configurer la base de données en cliquant sur **Configuration de la base de données**.
5. Lorsqu'un afficheur s'ouvre, faites défiler son contenu et cliquez sur **Run This Sample**.
6. Patientez jusqu'à ce que la page **Bienvenue** de l'afficheur apparaisse à l'écran et cliquez ensuite sur **Employee Center**.

Remarque : Les erreurs suivantes s'afficheront sur la page de la console lors de la première exécution de cette application : `DataSource not found. Try to construct a new datasource name: jdbc/yourco DataSource not found. Try to construct a new datasource name: jdbc/studio`. Ces erreurs sont corrigées automatiquement. Vous pouvez les ignorer.

7. Lorsque vous avez terminé, fermez la fenêtre de l'afficheur et la vue du navigateur Web puis, dans le **Panneau de configuration Serveur**, cliquez avec le bouton droit de la souris sur **WSTestEnv** et sélectionnez **Arrêter**.
8. (Facultatif) Arrêtez **IBM WebSphere Studio Site Developer**.

Chapitre 11. Bibliographie

Vous pouvez consulter les informations les plus récentes sur la migration et d'autres sujets sur le site suivant :

www.ibm.com/websphere/developer/zones/studio/transition.html

Les publications et les pages Web suivantes fournissent des informations utiles concernant l'utilisation de WebSphere Application Server - Express :

- *JSR-000053 Java Servlet 2.3 and JavaServer Pages 1.2 Specifications* :
java.sun.com/aboutjava/communityprocess/first/jsr053/index.html
- InfoCenter IBM WebSphere Application Server Version 4 :
www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/index.html
- *WebSphere Version 4 Application Development Handbook*:
www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246134.pdf
- *Programming J2EE APIs with WebSphere Advanced*:
www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246124.pdf
- *WebSphere Application Server Version 3.5 to 4.x - Migration Hints & Tips* :
www7b.software.ibm.com/wsdd/library/techarticles/0208_wright/wright.html
- *WebSphere Studio Application Developer Service Portal* :
www.ibm.com/software/ad/studioappdev/support/
- *WebSphere Studio Application Developer FAQ Frequently Asked Questions* :
www.ibm.com/support/search.wss?rs=457&tc=SSBRLP&dc=D800
- *WebSphere Application Server Service Portal* :
www.ibm.com/software/software/webservers/appserv/support.html
- *WebSphere Application Server FAQ Frequently Asked Questions* :
www.ibm.com/support/search.wss?rs=180&tc=SSEQTP&dc=D800

Autres références bibliographiques :

- Article sur l'utilisation de Ant avec WebSphere Studio Application Developer (y compris compilations et exportations de projets JZEE) :
www.ibm.com/websphere/developer/library/techarticles/0203_searle/searle1.html
- Article sur l'optimisation de compilations complexes dans WebSphere Studio Application Developer :
www.ibm.com/websphere/developer/library/techarticles/0204_searle/searle.html
- Article sur le chargement de classes J2EE (modules J2EE et chemins d'accès aux classes) dans WebSphere Studio Application Developer :
www.ibm.com/websphere/developer/library/techarticles/0112_deboer/deboer.html
- Article sur le développement de fichiers JAR d'utilitaire J2EE (fichiers JAR Java dans des modules J2EE) dans WebSphere Studio Application Developer :
www.ibm.com/websphere/developer/library/techarticles/0112_deboer/deboer2.html
- Article sur le support de développement coopératif dans WebSphere Studio Application Developer :
www.ibm.com/websphere/developer/library/techarticles/0108_karasiuk/0108_karasiuk.html
- Article sur la migration des composants Enterprise Access Builder à partir de VisualAge for Java vers WebSphere Studio Application Developer :
www.ibm.com/websphere/developer/techjournal/0201_minocha/minocha.html

- Article sur la conception d'une application d'EJB à l'aide de Session Facade permettant de communiquer avec des CMP :
www.ibm.com/websphere/developer/library/techarticles/0106_brown/sessionfacades.html
- Articles sur les procédures à suivre dans WebSphere Application Server :
www.ibm.com/software/webservers/appserv/ws_bestpractices.pdf
- Article sur le site dédié aux procédures à suivre :
www.ibm.com/websphere/developer/zones/bp/
- Page Web principale du domaine WebSphere Developer :
www.ibm.com/websphere/developer
- Articles techniques sur le domaine WebSphere Developer :
www.ibm.com/websphere/developer/techjournal/
- Informations concernant la gamme de produits WebSphere Studio, les fonctionnalités et les orientations de WebSphere Studio Application Developer :
www.ibm.com/websphere/developer/zones/studio/transition.html
- Tout sur la gamme d'outils de développement IBM WebSphere Studio :
www.ibm.com/websphere/developer/library/techarticles/0108_studio/studio_beta.html
- Forum de discussion externe sur Application Developer :
[news://news.software.ibm.com/ibm.software.websphere.studio.application-site-developer](http://news.software.ibm.com/ibm.software.websphere.studio.application-site-developer)
- Forum de discussion sur WorkBench (Eclipse) :
[news://news.software.ibm.com/ibm.software.websphere.studio.workbench](http://news.software.ibm.com/ibm.software.websphere.studio.workbench)
- Groupe de discussion externe WebSphere Application Server :
[news://news.software.ibm.com/ibm.software.websphere.application-server](http://news.software.ibm.com/ibm.software.websphere.application-server)
- Article sur le déploiement d'une application J2EE à partir de WebSphere Studio Application Developer vers WebSphere Application Server :
www.software.ibm.com/vad.nsf/Data/Document3584
- Fournisseurs du module Application Developer Software Configuration Management (gestion du code source) :
www.ibm.com/software/ad/studioappdev/partners/scm.html
- Migration des applications vers Application Developer à partir d'outils de développement tiers :
www.ibm.com/websphere/developer/zones/studio/migration.html
- Migration des applications VisualCafé WebLogic vers Application Developer (déploiement vers WebLogic) :
www.ibm.com/websphere/developer/library/techarticles/0209_searle/searle1.html
- Eclipse.org : www.eclipse.org
- WebSphere Developer Domain Plugin Central :
www.ibm.com/websphere/developer/downloads/plugin/
- Plug-in Eclipse Workbench (ne faisant pas partie d'Eclipse.org) :
www.eclipse-workbench.com/jsp/plugins.jsp
- Plug-in Eclipse (ne faisant pas partie d'Eclipse.org) : www.eclipse-plugins.2y.net/eclipse/plugins.jsp

Remarques

Note to U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, reportez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent Document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM EMEA Director of Licensing
IBM Europe Middle-East Africa
Tour Descartes
La Défense 5
2, avenue Gambetta
92066 - Paris-La Défense CEDEX
France

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit à l'adresse suivante :

*IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan*

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales : LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT"; IBM DECLINE TOUTE RESPONSABILITE, EXPLICITE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE VALEUR 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.

Le présent document peut contenir des inexactitudes ou des coquilles. Il est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut modifier sans préavis les produits et logiciels décrits dans ce document.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

*Lab Director
IBM Canada Ltd. Laboratory
8200 Warden Avenue
Markham, Ontario, Canada L6G 1C7*

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux termes du Contrat sur les produits et services IBM, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

LICENCE SUR LES DROITS D'AUTEURS :

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation IBM.

Toute copie totale ou partielle de ces programmes exemples et des oeuvres qui en sont dérivées doit comprendre une notice de copyright, libellée comme suit :

(C) (nom de la société) (année). Des segments de code sont dérivés des Programmes exemples d'IBM Corp. (C) Copyright IBM Corp. 2000, 2003. All rights reserved.

Informations relatives à l'interface de programmation

Les informations relatives aux interfaces de programmation sont destinées à vous aider à créer des applications à l'aide de ce programme.

L'interface de programmation générique permet à l'utilisateur d'écrire des applications qui obtiennent des services de ce programme.

Toutefois, cette interface de programmation générique peut également contenir des aides au diagnostic, à la modification et à l'optimisation. Ces dernières ont pour but de vous assister dans le débogage de votre application.

Avvertissement : N'utilisez pas ces aides comme une interface de programmation car elles sont susceptibles d'être modifiées.

Marques

Les termes qui suivent sont des marques d'International Business Machines Corporation aux Etats-Unis et/ou dans certains autres pays :

- AIX
- CICS
- Cloudscape
- DB2
- DB2 Extenders
- DB2 Universal Database
- e-business
- IBM
- iSeries
- OS/390
- S/390
- VisualAge
- WebSphere
- z/OS

Java et toutes les marques et logos incluant Java sont des marques de Sun Microsystems, Inc. aux Etats-Unis et/ou dans certains autres pays.

ActiveX, Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.

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