

IBM® WebSphere® Commerce
for Linux



マイグレーション・ガイド

バージョン 5.4

IBM® WebSphere® Commerce
for Linux



マイグレーション・ガイド

バージョン 5.4

ご注意

本書の情報およびそれによってサポートされる製品を使用する前に、225 ページの『特記事項』に記載する一般情報をお読みください。

本書の内容は、新版で特に指定のない限り、IBM® WebSphere® Commerce Business Edition for Linux® バージョン 5.4 と、IBM WebSphere Commerce Professional Edition for Linux バージョン 5.4 以降のすべてのリリースおよびモディフィケーションに適用されます。製品のレベルにあった版を使用していることをご確認ください。

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

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

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

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

(URL は、変更になる場合があります)

原典： IBM WebSphere Commerce
for Linux
Migration Guide
Version 5.4

発行： 日本アイ・ビー・エム株式会社

担当： ナショナル・ランゲージ・サポート

第1刷 2002.7

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

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

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

© Copyright IBM Japan 2002

目次

まえがき	vii
WebSphere Commerce 5.4 の新機能	vii
マイグレーションを実行すべき担当者	viii
サポートされるマイグレーション・パス	x
前のバージョンからのマイグレーション	xi
オペレーティング・システム別のマイグレーション・パス	xi
サポートされないマイグレーション・パス	xi
その他の考慮事項	xii
本書の表記規則	xii

第 1 部 必要なマイグレーション・ステップ

第 1 章 マイグレーションの前に

マイグレーション前のアクション	4
マイグレーション前の考慮事項	5
ATP 在庫へのマイグレーション	5
オーダーおよびオーダー・アイテム	7
マスター・カタログ	8
デフォルト契約	9
アクセス・コントロール	10
メンバー・サブシステム	12
マイグレーション時のダウン時間の最小化	17
別個のマシンへの WebSphere Commerce 5.4 のインストール	18
別個のマシンへの WebSphere Commerce Suite 5.1 環境の複製	20
WebSphere Application Server 4.0 への遷移	22
遷移の概要	22
製品の前提条件のマイグレーション	25
DB2 [®] 、IBM HTTP Server、および他の無料の前提条件のマイグレーション	25
非 IBM 前提条件のマイグレーション	25
バージョン 4.0 への構成のマッピング	25
新しいインストール単位に以前の構成をリストアする	27

第 2 章 Commerce Suite 5.1 のバックアップ

Commerce Suite 5.1 システムのバックアップ	31
ディレクトリーおよびファイルのバックアップ	31
データベースのバックアップ	33
DB2 データベース	34

第 3 章 ソフトウェアのアップグレード

WebSphere Commerce Suite 5.1 および WebSphere Commerce 5.4 IBM ソフトウェアのマッピング	35
ハードウェアのアップグレード	36
オペレーティング・システムのアップグレード	36

IBM 以外のソフトウェアのアップグレード	37
Internet Explorer 5.5 以降	37
WebSphere Application Server 4.0.1 へのアップグレード	38
現在の構成のバックアップ	39
Commerce Server アプリケーション・サーバーの除去	39
WebSphere Application Server 自動マイグレーション・サポートの使用	40
WebSphere Application Server FixPak および eFix のインストール	57
WebSphere Commerce 5.4 へのアップグレード	59
WebSphere Commerce Suite の WebSphere Commerce 5.4 へのアップグレード	59
Payment Manager 3.1.2 へのアップグレード	60
決済カセット	61
データベース	61
インストール前の考慮事項	62
インストールおよびマイグレーション	63
インストール後の考慮事項	64
UDB のアップグレード	65
product.xml の更新	65

第 4 章 データベースのマイグレーション

STORE テーブルおよび populatedb スクリプトのセットアップ	67
リモート DB2 データベースのカタログ化 (DB2 のみ)	70
避けるべきシナリオ	70
データベース準備スクリプトの実行	71
DB2 データベース	71
必須のデータベース・プレマイグレーション項目	72
オプションのデータベース・プレマイグレーション項目	74
Commerce Suite 5.1 キャッシュ・トリガーの除去	76
DB2 データベース	76
カスタム制約の除去	77
DB2 データベース	77
データベース・スキーマのマイグレーション	78
データベース・マイグレーション・ログおよびトレース・ファイル	81
カスタム制約のリストア	82
DB2 データベース	82
識別名の更新	83
DB2 データベース	83
マスター・カタログの割り当て	84
DB2 データベース	85
オーダーおよびオーダー・アイテムの状況の変更	85
DB2 データベース	86
デフォルト・ストアのブートストラップ・データ	86
Payment Manager に関する考慮事項	86

データベース・マイグレーションの検証	87
migratedb.log ファイルの表示	87
追加のチェック	88
データベースの整合性チェッカーの実行	89
DB2 データベース	89
データベース整合性チェッカーの出力	91
LANGUAGE_ID のリセット	92
データベースの再マイグレーション	92

第 5 章 データ・マイグレーション後のアクション 95

Commerce Suite 5.1 インスタンス構成のマイグレーション	95
インスタンス構成をマイグレーションする前のステップ	96
インスタンス構成のマイグレーション	97
セキュリティ構成のマイグレーション	100
暗号化設定の確認	101
DB2 データベース	101
ストア・ファイル資産のマイグレーション	103
Commerce Suite 5.1 JavaServer Pages の更新	105
shipaddress.jsp に対する変更	107
register.jsp および account.jsp に対する変更	107
ストア・プロパティ・ファイルの更新	109
新規のサンプル・ストア・ディレクトリー	109
ストア・サービスで非公開のストア状況	109
すべてのキャッシングをオフに切り替える	110
割引、配送、または課税用のカスタム・コードの更新	110
マイグレーションの検証	113

第 6 章 データ・マイグレーション後の追加のアクション 115

ルール・サーバー構成のマイグレーション	115
ルール・サーバー管理コマンド	117
オークション	121
ビジネス・アカウント	121
カスタマイズしたデータベース・テーブル用のステージング・サーバーの再構成	122
使用されなくなった Commerce Suite 5.1 テーブルの除去	123
新しいキャッシュ・トリガーのロード	123
カスタマイズしたプロパティの改良	124
配送計算コード	125
商品および在庫の検索	125
カスタマイズ・コマンド	126
edit_registration ページにおけるログオン ID の形式	126
商品アドバイザーのマイグレーション考慮事項	127

第 2 部 追加のマイグレーション考慮事項 129

第 7 章 メンバー・サブシステムのマイグレーション考慮事項	131
マイグレーション手順の概説	131

既存のディレクトリー・サーバーを使用する既存の Commerce Suite 5.1 ユーザー	134
WebSphere Commerce 5.4 での 5.1 ディレクトリー・サーバーの継続使用	134

第 8 章 アクセス・コントロール・サブシステムの考慮事項 137

getResourceOwners() の使用例	142
------------------------------------	-----

第 9 章 Payment コンポーネントのマイグレーション考慮事項 143

PAYMTHD テーブルから支払ポリシーへのマイグレーション	143
ビジネス・ポリシーおよびビジネス・ポリシー・コマンド	144
支払い用の WebSphere Commerce 5.4 ビジネス・ポリシー・コマンドへのマイグレーション	146
DoCancelCmd の CMDREG エントリーの変更	146
サンプル JavaServer Pages ファイル - PayStatusPM.jsp	147

第 10 章 その他のマイグレーション考慮事項 149

デフォルトの通貨の動作	149
価格設定のための考慮事項	150

第 3 部 付録 151

付録 A. WebSphere Commerce Suite

5.1 コンポーネントの開始と停止 153

WebSphere Commerce Suite の開始と停止	153
WebSphere Application Server の開始と停止	153
IBM HTTP Server の開始と停止	154
DB2 ユニバーサル・データベースの開始と停止	155
WebSphere Payment Manager の開始と停止	155
WebSphere Payment Manager Engine の開始	155
WebSphere Payment Manager ユーザー・インターフェースの開始	156
WebSphere Payment Manager の停止	156

付録 B. データ・マイグレーション・スクリプトの拡張 159

付録 C. マイグレーション・スクリプトの概要 161

メンバーのマイグレーション	161
カタログのマイグレーション	163
ATP 在庫のマイグレーション	163
オーダー・アイテムのマイグレーション	164
配送計算コード	164
支払いのマイグレーション	164
割引データのマイグレーション	164
契約のマイグレーション	165
デフォルト契約	165
キャンペーンのマイグレーション	166

アクセス・コントロールのマイグレーション	166
付録 D. 後からの ATP インベントリ への変換.	169
DB2 データベースの場合	170
付録 E. データベース・スキーマの変更 点	171
データベース・スキーマの変更点.	171
付録 F. 変更されたプログラミング・イ ンターフェース.	191
使用すべきでないコマンド	191
変更されたコマンド	192
商品アドバイザー	192
UserRegistrationAddCmd および UserRegistrationUpdateCmd	192
AddressAdd コマンド	193

変更されたプログラミング・インターフェース	193
WCS_Order	194
WCS_Catalog	198
WCS_User	199
WCS_Databean	201
Enterprise JavaBeans	201

付録 G. サンプル JSP の更新	205
register.jsp.	206
account.jsp	211
infashiontext_en_US.properties	215

付録 H. トラブルシューティング	217
トレース情報の使用可能化	224

特記事項.	225
商標	227

まえがき

本書は、Linux 上で WebSphere Commerce Suite 5.1 から WebSphere Commerce 5.4 へのマイグレーションを行うためのステップを説明しています。 WebSphere Commerce Studio 5.1 から WebSphere Commerce Studio 5.4 へのマイグレーションを実行したい場合は、*WebSphere Commerce Studio* マイグレーション・ガイド バージョン 5.4 を参照してください。

本書での WebSphere Commerce 5.4 または WebSphere Commerce という表現は、現行リリースの WebSphere Commerce 5.4 を意味しています。 WebSphere Commerce Suite または Commerce Suite という表現は、前のリリースの Commerce Suite 5.1、Commerce Suite 5.1.1、Commerce Suite 5.1.1.1、および Commerce Suite 5.1.1.2 を意味しています。

重要

このマイグレーション・ガイドと、そのすべての更新版は、以下の WebSphere Commerce Web ページの Technical Library セクションから入手可能です。

- Business Edition:

http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html

- Professional Edition:

http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html

製品に加えられた最新の変更について知るには、WebSphere Commerce Disk 1 CD のルート・ディレクトリーにある README ファイルを参照してください。 WebSphere Commerce 5.4 を、前のバージョンの Commerce Suite とサポートされている製品がインストールされていない マシン上にインストールするには、*WebSphere Commerce* インストール・ガイド バージョン 5.4 を参照してください。

WebSphere Commerce 5.4 の新機能

重要な機能強化と新規機能が、前のリリースの WebSphere Commerce から WebSphere Commerce 5.4 に追加されています。このリリースで使用可能な機能強化と新規機能の詳細については、*IBM WebSphere Commerce* 新着情報 バージョン 5.4 の資料を参照してください。

以下の領域において機能強化がされています。

- アクセス・コントロール
- 拡張されたユーザー、メンバー、および組織管理
- 請求、送り状、およびクレジット管理
- ビジネス関係管理
- 購買サイドの購入
- Catalog Manager

- カタログ・サブシステム
- コラボレーション
- Commerce Accelerator 拡張機能
- Commerce モデル
- 構成マネージャー
- 契約ベースのコマース
- クーポン
- 在庫管理
- ローター・パッケージ
- マーケティング・サブシステム
- Commerce Integrator からのメッセージ拡張機能
- オーダー管理の機能強化
- 組織管理コンソール
- Payment Manager
- 商品管理
- 見積要求 (RFQ)
- RMA 機能を含むリターンおよびリファンド
- 検索の機能強化
- セキュリティーの機能強化
- ストア・サービス
- WebSphere Application Server 4.0.2
- WebSphere Commerce Analyzer の拡張機能
- HTTP を使用した XML

Commerce Suite 5.1 システムを WebSphere Commerce 5.4 に、マイグレーション・ガイドで説明されているとおりにマイグレーションした後は、「新着情報」資料と WebSphere Commerce 5.4 オンライン・ヘルプを参照して、マイグレーション先のシステムでこれらの新規機能を使用する方法を調べてください。

マイグレーションを実行すべき担当者

マイグレーション作業は特定の技術を要するので、その作業の大半はシステム管理者が実行すべきです。以下に示すのは、マイグレーション・プロセスにおける、各種のユーザーとその期待される役割の一覧です。

システム管理者

以下の知識と経験

- プログラミングの理解があること (たとえば、Java™、JavaServer Pages など)
- データベース管理の理解があること
- Web マスター
- システム体系に関する知識があること

以下の作業

- WebSphere Commerce のインストール、構成、および保守
- データベースの管理
- Web サーバーの管理
- アクセスの制御
- 大量インポートまたは他のメカニズムによるデータ更新の管理

マイグレーション・プロセスの期待事項

マイグレーションのプログラムおよび手順は、**現行のシステム資産**がダウンしている時間を最小限にとどめるマイグレーションができるものにすべきです。

ストア開発者

以下の知識と経験

- プログラミングの理解があること
- マルチメディア・ツールの理解があること

以下の作業

- ストアの作成およびカスタマイズ
- 決済、発送、税サポートのセットアップおよびカスタマイズ

マイグレーション・プロセスの期待事項

マイグレーションのプログラムおよび手順は、**現行のストア資産**がダウンしている時間を最小限にとどめるマイグレーションができるものにすべきです。

ストア管理者

以下の知識と経験

- ビジネス手順の理解があること
- Web の理解があること

以下の作業

- オーダーの管理
- 決済の処理
- ショッパーの援助
- オンライン・ストアの保守
- オンライン・ストアの変更

マイグレーション・プロセスの期待事項

オーダーやショッパーなどの、オンラインで入手した情報は、マイグレーション後にも使用可能です。

カタログ管理者

以下の知識と経験

- 商品のエキスパートであること
- Web とコンピューターの理解があること
- マルチメディア・ツールの理解があること

以下の作業

- ストア・カタログの作成
- 商品とカテゴリの作成および管理
- 価格設定体系の作成および管理
- レポートの作成および管理

マイグレーション・プロセスの期待事項

以前のバージョンの *WebSphere Commerce Suite* を使用した際の情報は、再作成する必要はありません。ツールは、カタログの拡張に対して適応性があります。

サポートされるマイグレーション・パス

注:

このガイドでは、Linux プラットフォーム上で WebSphere Commerce 5.4 Business Edition または Professional Edition へのマイグレーションを実行するためのステップを説明しています。

すべての言語のバージョンについて、以下のマイグレーション・パスがサポートされています。

- Commerce Suite 5.1 Pro Edition から WebSphere Commerce 5.4 Professional Edition へ
- Commerce Suite 5.1 Pro Edition から WebSphere Commerce 5.4 Business Edition へ
- Commerce Suite 5.1.1.1 から WebSphere Commerce 5.4 へ
- Commerce Suite 5.1.1.2 から WebSphere Commerce 5.4 へ

重要: このマイグレーション・ガイドは、上記のマイグレーション・パスについてのみテストされています。このガイドは、WebSphere Commerce 5.4 への、WebSphere Commerce Suite の上記以降のバージョンからのマイグレーションについては扱っていません。つまり将来の可能性として FixPak、eFix、または他の拡張機能の適用によってアップグレードされた Commerce Suite のいずれかのバージョンからのマイグレーションについては扱っていません。

将来の FixPak、eFix、または他の拡張機能のアプリケーションによって発生するマイグレーションの問題の情報については、以下の WebSphere Commerce Support Web ページを参照してください。

<http://www.ibm.com/software/webservers/commerce/support.html>

前のバージョンからのマイグレーション

このガイドは、前述のサポートされているマイグレーション・パスのマイグレーション・プロセスを説明しています。前のバージョンの Net.Commerce™ または Commerce Suite はサポートされていません。

前のバージョンの Net.Commerce または Commerce Suite から WebSphere Commerce 5.4 にマイグレーションするには、まず既存のシステムを WebSphere Commerce 5.1 レベルにマイグレーションし、それからこの資料を使用して WebSphere Commerce 5.4 にマイグレーションします。

- 前のバージョンの Net.Commerce から Commerce Suite 5.1 に遷移およびマイグレーションするには、*WebSphere Commerce Suite 5.1 Migration and Transition Guide* を参照してください。

オペレーティング・システム別のマイグレーション・パス

WebSphere Commerce 5.4 は、以下の同一のオペレーティング・システム上での Commerce Suite のマイグレーションをサポートしています。

- Linux から Linux

WebSphere Commerce 5.4 は、たとえば Windows NT® 上の Commerce Suite 5.1 から Linux 上の WebSphere Commerce 5.4 などへの、異なるオペレーティング・システム間のマイグレーションはサポートしていません。

サポートされないマイグレーション・パス

WebSphere Commerce 5.4 は、以下のマイグレーション・パスをサポートしていません。

- あるデータベース・ベンダーの製品から別のベンダーの製品へのマイグレーション

その他の考慮事項

- 複数の言語バージョンのデータベースを、複数の言語をサポートする 1 つのデータベースにマージしたい場合があります。これを実行したい場合は、IBM グローバル・サービスに相談して支援を受けてください。
- WebSphere Commerce Suite 5.1 がインストールされているマシンに WebSphere Commerce 5.4 システムをインストールする場合、WebSphere Commerce Suite 5.1 のディレクトリーは、古いストア資産の上にコピーする必要があるため、保存されて新しいディレクトリー・ツリーにコピーされます。WebSphere Commerce Suite 5.1 は、機能を停止します。
- WebSphere Commerce 5.1 または WebSphere Commerce Business Edition 5.1 の Java または Enterprise JavaBeans™ オブジェクトで作成されたコードまたはコマンドをカスタマイズした場合、それらを WebSphere Commerce 5.4 で求められるレベルに再デプロイする必要があります。WebSphere Commerce Studio マイグレーション・ガイド バージョン 5.4 の『カスタマイズ・コードの遷移』のセクションを参照してください。この遷移は、IBM WebSphere Application Server 3.5 から WebSphere Application Server 4.0.2 への移動を行うために必要です。

本書の表記規則

本書では以下の強調表示規則を使用します。

- **太文字**は、コマンドまたは、フィールド名、アイコン、メニュー選択などのグラフィカル・ユーザー・インターフェース (GUI) コントロールを示します。
- **モノスペース (Monospace)** は、ファイル名、ディレクトリーのパスや名前などの、示されたとおりに正確に入力する必要のあるテキストの例を示します。
- **イタリック** は、語を強調するために使用します。イタリックは、ご使用のシステムにおいて該当する値に置き換えなければならない名前も示します。以下のいずれかの名前が出てきたら、説明するとおりに、ご使用のシステムの値に置き換えてください。

host_name

WebSphere Commerce サーバーの完全修飾ホスト名 (たとえば、ibm.com は完全修飾名)。



このアイコンは、ヒント (作業を完了するために役立つ追加情報) を表すマークです。

注: 本書での Commerce Suite 5.1 という表現は、特に断りがない限り、Commerce Suite 5.1、Commerce Suite 5.1.1.1、または Commerce Suite 5.1.1.2 のいずれかを表していると理解してください。

第 1 部 必要なマイグレーション・ステップ

マイグレーション・ガイドのこの部の章では、Commerce Suite 5.1 から WebSphere Commerce Business Edition 5.4 へのマイグレーションに必要なタスクを説明しています。これには以下が含まれます。

- 3 ページの『第 1 章 マイグレーションの前に』
- 31 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』
- 35 ページの『第 3 章 ソフトウェアのアップグレード』
- 67 ページの『第 4 章 データベースのマイグレーション』
- 95 ページの『第 5 章 データ・マイグレーション後のアクション』

さらに、115 ページの『第 6 章 データ・マイグレーション後の追加のアクション』では、固有の要件に応じて実行できる、マイグレーション後のオプションのアクションを説明しています。

第 1 章 マイグレーションの前に

以降のいくつかのセクションでは、作動可能な Commerce Suite 5.1 システムがまだある間に実行すべき特定ステップを説明します。さらにこのセクションでは、WebSphere Commerce 5.4 へのマイグレーションを開始する前に、注意すべき考慮事項についても説明します。

重要

WebSphere Commerce 5.1 または WebSphere Commerce Business Edition 5.1 の Java または Enterprise JavaBeans で作成されたコードまたはコマンドをカスタマイズした場合、それらを WebSphere Commerce 5.4 で求められるレベルに再デプロイする必要があります。 *WebSphere Commerce Studio* マイグレーション・ガイド バージョン 5.4 の『カスタマイズ・コードの遷移』のセクションを参照してください。この遷移は、IBM WebSphere Application Server 3.5 から WebSphere Application Server 4.0.2 への移動を行うために必要です。

いくつかの EJB をカスタマイズした場合、JNDI 名が、マイグレーション後にカスタマイズされたコマンドで呼び出される名前と同じであることを確認してください。

ご使用の JSP またはカスタマイズ済みコードが以前に IBM WebSphere Application Server 3.5.x からの `com.ibm.util` パッケージを使用していた場合、コードまたは JSP を、WebSphere Application Server 4.0.2 に同梱されている IBM SDK for Java からの同等のクラスを使用して再作成する必要があります。 `com.ibm.util` パッケージは、現行バージョンの WebSphere Application Server からは除去されています。

マイグレーション前のアクション

本書の他の部分で記述しているマイグレーション・ステップを進める前に、システムが Commerce Suite 5.1 レベルで作動可能である間に、必ず以下のアクションを実行します。

- Commerce Suite 5.1 ストア・アーカイブを簡単にマイグレーションするために、それを WebSphere Commerce 5.4 にマイグレーションする前に、Commerce Suite 5.1 レベルで公開する必要があります。 Commerce Suite 5.1 レベルでストアを公開するための詳細ステップについては、Commerce Suite 5.1 オンライン・ヘルプの『ストア・アーカイブの発行』のセクションを参照してください。
- 新しい ATP 在庫表記を使用できるように、ご使用の在庫表記をマイグレーションする場合、WebSphere Commerce 5.4 にマイグレーションする前に、Commerce Suite 5.1 内の既存のオークションをすべてクローズする必要があります。オークションのクローズの詳細については、Commerce Suite 5.1 オンライン・ヘルプの『オークション入札のクローズ』のセクションを参照してください。

- Commerce Suite 5.1 に同梱しているプロパティ・ファイルをカスタマイズした場合、マイグレーションを完了した後にアクセスできるディレクトリー、たとえば、Commerce Suite 5.1 がインストールされているのは別のドライブにそのコピーを作成します。たとえば、UserRegistration_en_US.properties ファイルを /opt/WebSphere/CommerceSuite/properties ディレクトリーにバックアップします。
- Commerce Suite 5.1 から WebSphere Commerce 5.4 への変更では、次の列の長さを変更されています。マイグレーション・プロセスでこれらの列に含まれるデータが失われないようにするために、Commerce Suite 5.1 内のこれらの列に含まれるデータの長さが、列の新しい長さを超えないようにしてください。たとえば、MBRGRP.DESCRPTION は 512 文字を超えないようにします。

Table.Column	Commerce Suite 5.1	WebSphere Commerce 5.4
MBRGRP.DESCRPTION	LONG VARCHAR	VARCHAR(512)
ORGENCY.DESCRPTION	LONG VARCHAR	VARCHAR(512)
CONTRACT.NAME	VARCHAR(254)	VARCHAR(200)

- マイグレーション前に、Commerce Suite 5.1 テーブルに関する制約があれば除去し、WebSphere Commerce 5.4 へのマイグレーション後に、その制約をリストアします。出荷された Commerce Suite 5.1 テーブルへの外部キー・リンクを含むテーブルをカスタマイズした場合、データ・マイグレーション時に、参照保全制約(外部キー、基本キー、索引など)を除去しようとする、データ・マイグレーション・スクリプトは失敗する場合があります。これらの制約を除去するときには、77 ページの『カスタム制約の除去』のサンプル SQL ステートメントを使用できます。付属するデータ・マイグレーション・スクリプトを使用して、データをマイグレーションした後に、82 ページの『カスタム制約のリストア』の説明に従って、制約を追加して戻す必要があります。

何らかの Commerce Suite 5.1 参照保全制約を変更した場合、すなわち、すでに存在する索引または外部キーの関係に別の列を追加した場合、その列はデータ・マイグレーション・プロセスの一部として削除されます。

参照制約の詳細については、データベース・ベンダーが提供する資料を参照する必要があります。

- Commerce Suite 5.1 でデータベース・テーブルをカスタマイズしており、WebSphere Commerce 5.4 でステージング・サーバーを使用する場合、商品データベースとステージング・データベースの整合性を保つために、データ・マイグレーションに先だって、ステージ・コピー・ユーティリティー・コマンド (stagingcopy) を実行する必要があります。詳細については、Commerce Suite 5.1 オンライン・ヘルプの『ステージ・コピー・ユーティリティー・コマンド』のセクションを参照してください。

マイグレーション前の考慮事項

以降のいくつかのセクションでは、マイグレーション・プロセスを開始する前に考慮する必要がある、WebSphere Commerce 5.4 のいくつかの重要なアイテムに焦点を当てています。

重要

付属するデータベース・マイグレーション・スクリプトを使用したデータベース・マイグレーションは、Commerce Suite 5.1 から WebSphere Commerce 5.4 へのマイグレーション・プロセスの主要な部分です。Commerce Suite 5.1 データベースのマイグレーションを開始する前に、データベース・マイグレーション・スクリプトの記述を確認し、データベースのマイグレーション時に使用される設定を理解しておくことを強くお勧めします。

ATP 在庫へのマイグレーション

前のバージョンの WebSphere Commerce には、使用可能な在庫レベルを調べて更新するための、タスク・コマンド・インターフェースが備えられていました。デフォルトのタスク・コマンド・インプリメンテーションは、INVENTORY テーブルを使用して、使用可能な在庫レベルを記録していました。この前のレベルの機能のことを、*互換モード在庫* といいます。

表 1. 互換モード在庫

互換モード在庫のタスク・コマンド・インターフェース	説明	呼び出し元
ResolveFulfillmentCenterCmd	OrderItem の FulfillmentCenter を判別します。	OrderItemAdd、OrderItemUpdate、OrderPrepare
CheckInventoryCmd	アイテムに十分使用可能な在庫があるかどうかをチェックします。	ResolveFulfillmentCenterCmd
UpdateInventoryCmd	アイテムに使用可能な在庫を減らします。	OrderProcessCmd、PaySynchronizePM
ReverseUpdateInventory	アイテムに使用可能な在庫を増やします。	オーダーを取り消したときのオーダー管理ユーザー・インターフェース

WebSphere Commerce 5.4 は、新しいタスク・コマンド・インターフェースでこの機能を拡張し、使用可能なまたは予想される在庫アイテムをチェック、割り振り、またはバック・オーダーします。新しいデフォルトのタスク・コマンド・インプリメンテーションでは、RECEIPT、RADETAIL、および他の関連テーブルの情報を使用します。アイテムがチェックされるかバック・オーダーされると、おおよその入手可能時刻が得られます。この拡張機能は、*予定可能 (ATP) 在庫* といいます。割り振りやバック・オーダーは、支払いが時間どおりに開始されないと、失効する可能性があります。

表2. 予定可能 (ATP) 在庫

ATP 在庫タスク・コマンド・インターフェース	説明	呼び出し元
AllocateInventoryCmd	使用可能または予想される在庫をチェック、割り振り、またはバック・オーダーします。 FulfillmentCenters と、おおよその入手可能時刻を判別します。割り振りまたはバック・オーダーを取り消すときにも使用できます。	OrderItemAdd、 OrderItemUpdate、OrderPrepare、 OrderProcess、 ProcessBackOrders
GetEligibleFulfillmentCentersCmd	FulfillmentCenters の優先順位付けリストを判別します。	AllocateInventoryCmd
CheckInventoryAvailabilityCmd	おおよその入手可能時刻を得ます。	AllocateInventoryCmd
AllocateExistingInventoryCmd	使用可能な在庫を割り振ります。	AllocateInventoryCmd
DeallocateExistingInventoryCmd	割り振りを取り消します。	AllocateInventoryCmd、 ReleaseExpiredAllocations
AllocateExpectedInventoryCmd	バック・オーダーを作成します。	AllocateInventoryCmd
DeallocateExpectedInventoryCmd	バック・オーダーを取り消します。	AllocateInventoryCmd、 ReleaseExpiredAllocations

OrderItemAdd、OrderItemUpdate、および OrderPrepare コマンドには、新しいパラメーターが備えられており、それによって呼び出し側が在庫をチェック、割り振り、バック・オーダーしたり、割り振りやバック・オーダーを取り消すことができます。 OrderProcess は、割り振られてもバック・オーダーされてもいない OrderItems の在庫を、必ず割り振るか、割り振れない場合はバック・オーダーします。

表3. ATP 在庫コマンド

ATP 在庫使用可能コマンド	拡張された ATP 機能	デフォルトのアクション
OrderItemAdd、 OrderItemUpdate	チェック、割り振り、バック・オーダー、取り消し。	チェック。
OrderPrepare	チェック、割り振り、バック・オーダー、取り消し。	割り振りまたはバック・オーダー。
OrderProcess	割り振りまたはバック・オーダー。	該当しない。

STORE テーブル内の新しい列 ALLOCATIONGOODFOR は、ストアの ATP 在庫機能を使用可能にするために使用されます。この列の値がゼロであると、互換モード在庫が使用可能になります。値がゼロより大きいと、ATP 在庫が使用可能になります。支払いが開始されない場合、値には、割り振りとバック・オーダーの有効期限が切れた後の秒数が示されます。データベースを Commerce Suite 5.1 から WebSphere Commerce 5.4 にマイグレーションするときには、ATP 在庫へマイグレーションを選択できます。そうする場合、INVENTORY テーブルの情報は RECEIPT テーブルへ移動し、STORE.ALLOCATIONGOODFOR はデフォルト値 (43,200 秒または 12 時間) にセットされます。

WebSphere Commerce Suite 5.1 INVENTORY テーブル内の製品、DynamicKitBean、またはバンドルのいずれかであるカタログ・エントリは、データベースのマイグレーション時に WebSphere Commerce 5.4 RECEIPT テーブルに移動されません。

WebSphere Commerce 5.4 在庫機能はそれらをサポートしていないからです。
WebSphere Commerce Suite 5.1 INVENTORY テーブル内のパッケージまたはアイテムであるカタログ・エントリーだけが、データベースのマイグレーション時に WebSphere Commerce 5.4 RECEIPT テーブルに移動されます。新規の ATP 機能はこれらのタイプのカタログ・エントリーのみサポートします。

この時点では ATP 在庫にマイグレーションしないことを選択する場合、migrateATP スクリプトを使用して後でマイグレーションすることが可能です。このスクリプトは、169 ページの『付録 D. 後からの ATP インベントリーへの変換』で説明されています。

オーダーおよびオーダー・アイテム

Commerce Suite 5.1 オーダーまたはオーダー・アイテムに関して、注意する必要があるマイグレーション前の考慮事項は 2 つあります。

- オーダーまたはオーダー・アイテムは M 状態です (つまり、ショッパーが支払いを開始し、在庫更新が成功しましたが、オーダーまたはオーダー・アイテムが与信済みでない)。

この場合、マイグレーションの前に、オーダーまたはオーダー・アイテムの完了、削除、取り消しのいずれか適切なアクションを取る必要があります。通常、この状態のオーダーまたはオーダー・アイテムは、与信の進行を待つだけで、M 状態になるのは短期間だけですが、許可が失敗するか拒否されると、この状態が続きます。これらのオーダーまたはオーダー・アイテムが M 状態のままマイグレーションすると、WebSphere Commerce 5.4 は、PMClean コマンドをスケジュール・ジョブとして実行し、このようなオーダーおよびオーダー・アイテムをクリーンアップします。

- オーダーまたはオーダー・アイテムが C 状態です (つまり、支払いが与信済みである)。

アイテムが完全に完了し、出荷されている場合には、オーダーまたはオーダー・アイテムを、最終の S 状態 (つまり、オーダー・アイテムが出荷済み) にする必要があります。これにより、オーダー・アイテムが再び WebSphere Commerce 5.4 で割り振られることを避けられます。

データベース・マイグレーション・プロセスでは、ATP オプションを指定すると、データベース・マイグレーション・スクリプトによって ctos.sql スクリプトが生成されます。ctos.sql スクリプトは、状況が C であるオーダーまたはオーダー・アイテムがあれば、C から S へ変更します。ctos.sql スクリプトは、次の場合に実行します。

- ATP オプションを使用してデータベース・マイグレーション・スクリプトを実行する場合。

ctos.sql スクリプトは、マイグレーション後に Web サーバーと WebSphere Commerce Server — *instance_name* アプリケーション・サーバーを再始動する前に、実行する必要があります。

- ATP オプションを使用せずにデータベース・マイグレーション・スクリプトを実行し、マイグレーションを完了した場合、後でマイグレーション済みのシステムを実行して、ATP 在庫へマイグレーションするようにします。

WebSphere Commerce 5.4 へのマイグレーション後に migrateATP スクリプトを実行すると、ctos.sql スクリプトが生成されます。この場合、Web サーバ

ーと WebSphere Commerce Server - *instance_name* アプリケーション・サーバーを再始動する前に、これを実行する必要があります。

これは、独自のツールおよびコマンドをインプリメントするときのオプションのステップなので、状況を S に変更できないことに注意してください。

C 状態のマイグレーション済みのオーダーおよびオーダー・アイテムを、WebSphere Commerce 5.4 Commerce Accelerator ツールで表示したり編集することは可能ですが、このツールを使用して編集することはお勧めしません。このツールを使用して編集しようとする（すでに完了済みであれば編集すべきでない）、回復不能エラーになる可能性があります。このような場合、オーダーの状態は E 状態 (CSR による編集 - 顧客サービス担当者がオーダーを処理している) か T 状態 (一時的 - オーダー管理ユーザー・インターフェースによって使用され、オーダーが一時的にバックアップされる) のいずれかに変更されます。CSR は E 状態のオーダーの要約を表示することにより、T 状態のオーダーのオーダー番号を検出できます。T 状態のオーダーは、(ツールでの編集前の) 元のオーダーのバックアップ・コピーです。CSR は、この T 状態のオーダーを参照用に印刷し、Commerce Accelerator を使用して、顧客用に手動でオーダーを再作成できます。

マイグレーション・スクリプトは、C 状態のオーダー・アイテムを、指定したアイテム (ITEMSPC) に関連付けないことに注意してください。そのようにすると、完了したオーダー・アイテムが大量に存在する (数百万) 可能性があるため、パフォーマンスが低下することがあります。

- オーダーまたはオーダー・アイテムは P 状態 (保留 - 顧客はこのオーダーを変更できる)、または I 状態 (送信済み - 顧客はオーダーを送信したが、まだ支払いを開始していない) です。

マイグレーション・スクリプトでこれらのオーダーの `ORDERS.LOCKED` を 0 (ゼロ) にセットすると、これらのオーダーがアンロックされます。

マスター・カタログ

WebSphere Commerce Suite 5.1 では、カタログ・システムには構造化されたカタログ・データは必要ありませんでした。マスター・カタログを使用し始めるときには、WebSphere Commerce 5.4 では、特定の 방법으로カタログ・データを構造化する必要があります。

WebSphere Commerce 5.4 では、マスター・カタログは、ストアの商品取引を管理する中心です。ストアに必要なものはすべて、マスター・カタログに含まれます。これは、すべての商品、アイテム、関係、およびストアで販売されるものすべての標準価格を含む 1 つのカタログです。

WebSphere Commerce システムの全ストアに、マスター・カタログがなければなりません。マスター・カタログは複数のストアで共有することができ、また必要な数のストアを定義できます。カタログ管理用のマスター・カタログを作成することに加えて、表示の目的で 1 つ以上のナビゲーション・カタログを作成することもできます。ナビゲーション・カタログにはマスター・カタログと同じエントリーを含めることができますが、カスタマーに表示する目的で、ナビゲーション・カタログはマスター・カタログよりずっと柔軟な構造になっています。必要に応じていくつでもナビゲーション・カタログを作成することができます。しかし、オンラインの商

品取引を管理するためにマスター・カタログを使用するので、マスター・カタログをナビゲーション・カタログとして使用して、メンテナンスのオーバーヘッドを最小限に抑えるようお勧めします。

商品管理ツールを使用して、マスター・カタログを表示および管理することができます。

マスター・カタログの作成と管理の詳細については、マイグレーションを完了してから、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。

重要な構造上の考慮事項

商品管理ツールのような WebSphere Commerce 5.4 カタログ・ツールを使用するには、使用しているマスター・カタログは、以下の条件を満たしている必要があります。

- マスター・カタログは、適切なツリーであるべきです。つまり、サイクルであってはなりません。これは、以下のタイプのシナリオを避けなければならないことを暗に示します。親カテゴリ **A** にサブカテゴリ **B** があるとします。 **B** および **B** のサブカテゴリはどれも、 **A** の親カテゴリにならないようにすることが重要です。
- 複数のカテゴリの下で商品进行分类することはできません。複数のカテゴリに 1 つの商品を置くには、ナビゲーション・カタログを使用します。
- 商品に属するすべてのアイテムは、その商品が分類されているのと同じカテゴリで分類される必要があります。
- 商品管理ツールは、マスター・カタログでしか機能しません。

カタログ情報のマイグレーション

現在 Commerce Suite 5.1 ストアで 1 つのカタログを使用している場合、データベース・マイグレーション・スクリプトは、そのカタログをそのストアのマスター・カタログとして割り当てます。

現在 Commerce Suite 5.1 ストアで複数のカタログを使用している場合、マイグレーションされたストアのマスター・カタログとして、どのカタログを割り当てる必要があるかを考慮しなければなりません。この割り当ては、84 ページの『マスター・カタログの割り当て』の説明に従って、choosemc.sql スクリプトを使用して行われます。

デフォルト契約

Business

WebSphere Commerce 5.4 では、契約サポートを提供するための条件が導入されました。マイグレーション・プロセスでは、WebSphere Commerce Suite 5.1 ビジネス・フロー（たとえば配送料用）と同様の動作および特性を持つ、ご使用のシステムに対するデフォルトの契約が作成されます。

デフォルト契約は自動的に作成されるので、通常は、マイグレーション・プロセス中にユーザーがアクションを取る必要はありません。ビジネス・プロセスのために追加契約を作成する必要がある場合は、WebSphere Commerce 5.4 にマイグレーション

ョンした後に、WebSphere Commerce Accelerator を使用してそれを行います。
WebSphere Commerce 5.4 オンライン・ヘルプの『新規契約の作成』のセクション
を参照してください。

アクセス・コントロール

アクセス・コントロールは、商用サイトの全体的なセキュリティーとフロー制御に
不可欠な部分です。サイト管理者と参加組織は、参加者がシステム内のどのオブジ
ェクトでどのアクションを実行できるかを制御する必要があります。それで、
WebSphere Commerce 5.4 のリソース・レベルでのアクセス・コントロールは、ビ
ジネス・オブジェクトを扱うコードの外部でアクセス・コントロールの決定を行っ
てカスタマイズの可能性を広げられるようにするため、WebSphere Commerce Suite
5.1 の場合のようにプログラマチックなものではなく、ポリシー・ベースのものにな
っています。

WebSphere Commerce 5.4 では、アクセス・コントロールは、GUI と、アクセス・
コントロール・ポリシーを定義するために使用する XML ファイルを使用して管理
されます。これらのポリシーは、WebSphere Commerce 5.4 データベースに保管さ
れています。そして、WebSphere Commerce 5.4 システムの始動時にメモリーにロ
ードされます。

WebSphere Commerce 5.4 でアクセス・コントロールに加えられた主な改善点を次
に示します。

- 柔軟性を向上させるために、アクセス・コントロール・ポリシーは、ビジネス・
オブジェクトを扱うコードから外部化されました。
- 階層的なアクセス・コントロールが、アクセス・コントロール・モデルに組み込
まれました。
- すべてのアクセス・コントロール・ポリシーは、ActionGroups、ResourceGroups、
AccessGroups というグループに基づくようになりました。

Commerce Suite 5.1 に実装されているコマンド・レベルのアクセス・コントロール
で十分であり、コード変更を望まない場合は、以下のようになります。

- Commerce Suite 5.1 アクセス・コントロール・テーブルを適切なポリシーにマイ
グレーションする必要があります。これを適切に動作させるには、71 ページの
『データベース準備スクリプトの実行』で説明されているとおりに、データベ
ース準備スクリプトを実行する必要があります。これが実行されない場合、いくつ
かのカスタマイズ済みコマンドおよびビューに対するアクセス・コントロール・
ポリシーはマイグレーションされません。
- getResourceOwners() メソッドを上書きした場合、コマンド・レベルのアクセ
ス・コントロールでは、ここで戻される各リソース所有者が、保護可能なリソ
ース、すなわちコマンドの所有者として使用されます。
- getResourceOwners() メソッドを上書きしていない場合は、保護可能なリソ
ース、つまりコマンドのコンテキストに storeId が指定されていれば、そのコマ
ンドの所有者がストアの所有者になります。コマンドのコンテキストに storeId が
指定されていない場合は、ルート組織が使用されます。
- メソッド checkPermission() を上書きした場合、このメソッドは、コマンド・レ
ベルのアクセス・コントロールを実行してから呼び出されます。

上記のステップに加えて、WebSphere Commerce 5.4 のリソース・レベルのアクセス・コントロールを最大限に活用する場合、`getResources()` メソッドをインプリメントする必要があります。さらに、既存のコマンドをマイグレーションする場合、`getResourceOwners()` を独自にインプリメントしていたならば、そのインプリメンテーションを除去できます。ストア所有者またはサイト組織の使用は、コマンド・レベルのアクセス・コントロールでは適切であるはずですが、よりきめの細かいレベルのアクセス・コントロールは、リソース・レベルのアクセス・コントロールによって実行できます。

アクセス・コントロールの詳細については、137 ページの『第 8 章 アクセス・コントロール・サブシステムの考慮事項』を参照してください。

注:

1. Commerce Suite 5.1 から拡張されたコントローラー・コマンドを追加している場合、WebSphere Commerce 5.4 は、マイグレーション時にそれに対してコマンド・レベルのポリシーを追加するだけです。Commerce Suite 5.1 コマンドの WebSphere Commerce 5.4 バージョンで `getResources()` がインプリメントされている場合は、それによって戻されるリソースを判別してそのコマンドに適したリソース・レベルのポリシーを作成するか、あるいはリソース・レベルのアクセス・コントロールが必要ない場合は、コマンドがヌル値を戻すように `getResources()` でコマンドを指定変更する必要があります。

WebSphere Commerce 5.4 コマンドが、その `getResources()` に戻すものを判別するには、トレースを分析して `Action=WCBCommand` を探し、`getResources()` がチェックしているすべての保護可能リソースを見つけてください。上記のトレースでは、リソースは `Order` です。たとえば、`SERVER` トレースを使用可能にした場合のことを考慮してみましょう。ログ内には以下が示されています。

```
=====
TimeStamp:    2001-11-16 02:42:30.937
Thread ID:    <Worker#3>
Component:    SERVER
Class:        AccManager
Method:        isAllowed
Trace:        isAllowed? User=10012; Action=com.fvt.ACCOrderItemAddCmd;
Protectable=com.ibm.commerce.order.objects._Order_Stub;
Owner=7000000000000002000resource is Groupable
```

```
=====
TimeStamp:    2001-11-16 02:42:30.984
Thread ID:    <Worker#3> Component:
SERVER Class: AccManager
Method:        isAllowed
Trace:        PASSED? =false
```

上記のトレースの意味は、リソース・レベルのポリシーが失敗したということです。この場合、`ACCOrderItemAddCmd` は、`getResources()` をインプリメントするサーバー `OrderItemAdd` コマンドから拡張されます。したがって、デフォルトでは、`ACCOrderItemAdd` も、それに対する `getResources()` がヌルを戻すように変更されていない限り、リソース・レベルのポリシーを必要とします。このリソース・レベルのポリシーは、マイグレーション時には、どの WebSphere Commerce 5.4 コマンドを拡張するかが分からないので追加されません。

たいていの場合、コマンドはアクセス Bean を `getResources()` メソッドで戻します。たとえば、`getResources()` で

com.ibm.commerce.xyz.objects.XYZAccessBean が戻された場合、トレースにはこれが com.ibm.commerce.xyz.objects._XYZ_Stub と示されます。この違いは、WebSphere Commerce 5.4 がアクセス Bean をそのリモート・インターフェースに狭めなければならないからです (これは実際に保護可能インターフェースに拡張する EJB のリモート・インターフェースであるため)。

2. WebSphere Commerce Suite 5.1 では、リソース・レベルのアクセス・コントロールは、コマンド・ロジック内でプログラマチックに施行されていました。WebSphere Commerce 5.4 では、リソース・レベルのアクセス・コントロール・ポリシーは外部的に指定されます。これはコマンド・レベルのアクセス・コントロール・ポリシーの指定方法と似ています。マイグレーション時に、コマンド・レベルのアクセス・コントロール・ポリシーは Commerce Suite 5.1 から WebSphere Commerce 5.4 にマイグレーションされます。Commerce Suite 5.1 のデフォルトのアクセス・コントロール・ポリシーのカスタマイズによって必要とされるどのリソース・レベルのアクセス・コントロール・ポリシー (ACCMDGRP テーブルに保管される) も、手動で追加する必要があります。そうしない場合は、予期しないアクセス・コントロール違反例外を受け取ります。詳細については、217 ページの『付録 H. トラブルシューティング』の関連項目を参照してください。

メンバー・サブシステム

WebSphere Commerce 5.4 と WebSphere Commerce Suite 5.1 の主な違いは、WebSphere Commerce 5.4 では、各ユーザーおよび組織エンティティ・メンバーに、別の組織エンティティである親メンバーがなければならないという点です。これによって、ユーザーおよび組織エンティティは、メンバーシップ階層を形成できます。メンバー・グループは、メンバーシップ階層の一部ではないので、親メンバーはありません。

マイグレーション・プロセス時に、データベース・マイグレーション・スクリプトは、次の事柄に基づいて、ユーザーおよび組織エンティティの親と子孫を決定します。

- ユーザーが、BUSPROF テーブルにレコードを持ち、ORG_ID 列および ORGUNIT_ID 列の値を持っているか。
- 組織エンティティの ORGENTITY テーブルにある MEMBER_ID 列の値。

ユーザーと組織エンティティの親および子孫を判別したら、MBRREL テーブルに取り込みが行われ、メンバーシップ階層が取り込まれます。WebSphere Commerce 5.4 ビジネス論理では、このメンバーシップ階層を使用します。そのため、そのメンバーシップ階層を適切に判別できるようにするため、ご使用のデータベース内の特定の列に適切な値を入れる必要があります。ユーザーおよび組織エンティティの親と子孫は、次のようにして、データベース・マイグレーション・スクリプトによって判別されます。

- BUSPROF テーブルにレコードがあり、プロファイル・タイプが B (B2B ユーザー) に設定されているユーザーの場合:
 - ORGUNIT_ID は、ヌルでなければ、親メンバー ID として使用されます。
 - ORGUNIT_ID がヌルの場合、ORG_ID がヌルでなければ、それが親メンバー ID として使用されます。

- ORGUNIT_ID と ORG_ID の両方がヌルであれば、親メンバーとして、Default Organization 組織エンティティ (ORGENITY) が使用されます。

B2B ユーザーが、マイグレーション・プロセスで Default Organization が親として割り当てられることを防ぐため、マイグレーションの前に、Commerce Suite 5.1 BUSPROF テーブルをスキャンして ORGUNIT_ID および ORG_ID 列に記入するようにします。Default Organization を、B2B ユーザーの親組織エンティティにすることはお勧めしません。特定の登録ユーザーの BUSPROF テーブルの ORGUNIT_ID および ORG_ID 列に記入できない場合、そのような登録ユーザーのプロファイル・タイプを、B (B2B ユーザー) から C (B2C ユーザー) に変更する必要があります。

USERS テーブルには PROFILETYPE 列があり、有効な値として、ヌル、B、または C を入れることができます。

- B (登録済み B2B ユーザー)
- C (登録済み B2C ユーザー)
- ヌル (プロファイル・データなし)

Commerce Suite 5.1 コードをカスタマイズし、このコードがユーザーのプロファイル・タイプを設定していない場合、USERS テーブルの PROFILETYPE 列をクリーンアップする必要があります。WebSphere Commerce 5.4 の場合、次のようにすることをお勧めします。

- B2C ユーザーを Default Organization の下に置き、プロファイル・タイプを C に設定する。一般に、B2C ユーザーは BUSPROF テーブルにレコードを持っておらず、Default Organization を親にしています。
- B2B ユーザーのプロファイル・タイプを B に設定し、BUSPROF テーブルにレコードを入れ、適切な組織エンティティを親にする。B2B ユーザーを Default Organization の下に置くことはお勧めしません。

さらに、管理者 (つまり、USERS テーブルの登録タイプが A または S で、ACCMBRGRP テーブルにエントリーが入っているユーザー) のプロファイル・タイプを B に設定します。

- BUSPROF テーブルにレコードがないユーザーは、データベース・マイグレーション・スクリプトによって、親組織エンティティを Default Organization に設定します。
- ORGENITY テーブルの組織エンティティでは、MEMBER_ID 列がヌルでなければ、MEMBER_ID 列が親メンバー ID として使用されます。MEMBER_ID 列がヌルであれば、親メンバーは Root Organization に設定されます。

登録済みユーザーと組織エンティティのレコードを含む新しい MBRREL テーブルでは、データベース・マイグレーション・スクリプトは、MBRREL テーブルの内容を使用するだけで、メンバーシップ階層を判別します。MBRREL テーブルには、汎用ユーザー、ゲスト・ユーザー、およびメンバー・グループのレコードが含まれないことに注意してください。

マイグレーション時には、データベース・マイグレーション・スクリプトによって、次のような整合性検査が実行されます。BUSPROF にレコードを持つユーザーの場合、ORG_ID 列と ORGUNIT_ID 列がヌルでなければ、スクリプトは、ORGENITY テーブルを使用し、組織階層を ORGUNIT_ID から上方向に調べます。これは、最終的に MEMBER_ID にヌルが見つかるか、ORGENITY_ID と同じ値が見

つかるまで続きます。 ORGENTITY テーブルの MEMBER_ID 列の値が、 BUSPROF テーブルの ORG_ID 列の値と同じであることを確認します。同じでなければ、スクリプトは、不整合を訂正するようユーザーに通知します。

他のメンバー・サブシステムの考慮事項

- WebSphere Commerce 5.4 へのマイグレーション時に、MEMBER テーブルの STATE 列は、以下のように設定されます。
 - ゲスト・ユーザー (登録タイプは G) の場合、マイグレーション・スクリプトは STATE をヌルに設定します。
 - 登録済みユーザー (登録タイプは R) の場合、マイグレーション・スクリプトは STATE を approved に設定します。
 - 組織エンティティーの場合、マイグレーション・スクリプトは STATE を approved に設定します。
 - メンバー・グループの場合、マイグレーション・スクリプトは STATE をヌルに設定します。

組織エンティティーは、マイグレーションされ、承認グループを所有しなくなります。つまり、デフォルトでは、マイグレーション済みの組織エンティティーは、B2B ユーザー自己登録などのビジネス・プロセスの承認を必要としないということです。

- WebSphere Commerce 5.4 にデフォルトで付属している役割には、次の例外を除いて、Commerce Suite 5.1 に付属しているすべての役割が含まれており、さらにいくつかの新しい役割も加えられています。

- Order Clerk 役割は組み込まれていません。

Commerce Suite 5.1 は、Order Clerk 役割をサポートしていましたが、WebSphere Commerce 5.4 では不要になり使用されなくなりました。Order Clerk 役割で実行に使用されるタスクは、自動化されているか、または WebSphere Commerce 5.4 の顧客サービス・スーパーバイザーで実行できます。ユーザーに Commerce Suite 5.1 で Order Clerk 役割 (-5) があり、ACCCMDGRP テーブルにエントリーがある場合、そのユーザーは、アクセス・コントロール・マイグレーションの一部としてマイグレーションされ、その役割は、ユーザー定義の役割として扱われます。

まだ Order Clerk 役割が必要かどうかを確認してください。必要でなければ、WebSphere Commerce 5.4 ではサポートされなくなったので、除去してください。

- Customer 役割は組み込まれていません。

Commerce Suite 5.1 には、Customer というアクセス・グループが組み込まれていました。Commerce Suite 5.1 内の各アクセス・グループには、役割名の名前があります。通常は、次の 2 つの目的で Commerce Suite 5.1 のアクセス・グループが使用されます。

- コマンドをアクセス・グループに割り当てる (関連が ACCCMDGRP テーブルに保管される)
- ユーザーをアクセス・グループに割り当てる (関連が ACCMBRGRP テーブルに保管される)

アクセス・グループに割り当てられたユーザーは、アクセス・グループ名と同じ名前の役割を演じます。したがって、ユーザーを特定のアクセス・グループに割り当てることは、役割をそのユーザーに割り当てることと同じで、ユーザーは、そのアクセス・グループに関連したコマンドを実行できるようになります。Commerce Suite 5.1 の Customer アクセス・グループは、すべてのユーザーが実行できるコマンド群に関連付けられています。つまり、Customer アクセス・グループは、Commerce Suite 5.1 システム内の全ユーザーを表しており、各ユーザーに Customer 役割を割り当てる必要はありません。

WebSphere Commerce 5.4 では、Customer アクセス・グループの代わりに、AllUsers メンバー・グループが同梱されています。すべてのユーザーが実行できるコマンドのセットを AllUsers メンバー・グループに関連付けるために、アクセス・コントロール・ポリシーが作成されています。各ユーザーに Customer 役割を明示的に割り当てることは不必要なので、WebSphere Commerce 5.4 では、デフォルトで Customer 役割は同梱されなくなりました。特定の組織エンティティに対して、その組織エンティティの従業員であるユーザーのグループがあり、従業員以外は顧客と見なされます。Commerce Suite 5.1 で Customer アクセス・グループに明示的に割り当てられたユーザーは、WebSphere Commerce 5.4 へのデータ・マイグレーション時に、AllUsers メンバー・グループに明示的に割り当てられます。その場合、そのような明示的な割り当ては不要であるため、マイグレーション・スクリプトは警告メッセージを発行します。

- Merchant 役割は、Seller に名前変更されています (Merchant は B2C 用語であり、Seller は B2B 用語であるため)。
- Merchandising Manager 役割は、同じ理由で Product Manager に名前変更されています。
- Commerce Suite 5.1 では、USERS テーブルには、以下の 4 つの有効値を受け入れる、REGISTERTYPE 列が含まれています。
 - R — 登録済みユーザー
 - G — ゲスト・ユーザー
 - S — サイト管理者
 - A — 管理者

登録タイプ S と A は、役割に関連付けられています。WebSphere Commerce 5.4 では、USERS テーブルの REGISTERTYPE 列は、引き続き同じ値のセットをサポートします。しかし、WebSphere Commerce 5.4 からデフォルトで使用可能な役割のセットを指定する場合、登録タイプ A の意味をさらに正確にする必要があります。ここで、タイプ A には、特定の役割を演じる Seller 組織の従業員という、さらに具体的な意味が与えられます。登録タイプ A に対応する役割のセットは、管理コンソールを使用して、Administrators アクセス・グループの定義を変更することによって構成可能です。さらに、ユーザーの登録タイプの値は、役割の割り当てまたは割り当て解除時に、自動的に A または S に設定されるので、登録タイプの値は、ユーザーが演じる役割と整合していることが保証されます。

要約すると、次のようになります。

- Seller 組織のユーザーに Site Administrator 役割が割り当てられる場合、このユーザーの登録タイプ値は S になります。

- Seller 組織のユーザーに、Administrators アクセス・グループで定義された、Site Administrator 以外のいずれかの役割が割り当てられる場合、このユーザーの登録タイプ値は A になります。
 - マイグレーション時に、データベース・マイグレーション・スクリプトは以下を実行します。スクリプトは、登録タイプが A であり、どのアクセス・グループにも属さない Commerce Suite 5.1 のユーザーを、WebSphere Commerce 5.4 の Administrators アクセス・グループに明示的に割り当てます。ACCMBRGRP テーブルに何も入力されていない場合、スクリプトはこのステップを実行しないことに注意してください。マイグレーション・スクリプトを実行する前に、以下を実行する必要があります。
 - Administrators アクセス・グループの定義を調べ、必要であれば、役割のリストを変更します。たとえば、Commerce Suite 5.1 のユーザーを、REGISTERTYPE=A を指定した XXX というアクセス・グループに割り当てましたが、WebSphere Commerce 5.4 で、XXX は Administrators アクセス・グループにリストされたどの役割でもない場合、追加の基準として role=XXX を Administrators アクセス・グループに追加する必要があります。
 - S および A 値について USERS テーブルの REGISTERTYPE 列を直接検査する、カスタマイズした論理がある場合、そのコードを以下のように変更する必要があります。
 - ユーザーに付与された権限を判別するために、使用している論理が REGISTERTYPE 列を検査しようとする場合、その論理をアクセス・コントロール・ポリシーに置き換えます。WebSphere Commerce 5.4 では、新しいアクセス・コントロール設計を使用できるので、権限関連の論理をハードコーディングするのではなく、アクセス・コントロール・ポリシーを使用することをお勧めします。10 ページの『アクセス・コントロール』を参照してください。
 - ご使用の論理が REGISTERTYPE 列を検査しますが、アクセス・コントロールのためでなければ、「メンバー・サブシステム」から使用できる次のいずれかのプログラミング・インターフェースを使用する必要があります。
 - isAdministrator()
 - isSiteAdministrator()
 - isMemberInRole()
- これらのインターフェースの詳細については、マイグレーションを完了した後で、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。この変更によって、コードに含まれる、登録タイプの実際の値についての従属関係が除去されます。将来のバージョンの WebSphere Commerce では、登録タイプに有効な値のセットを変更できることに注意してください。
- Commerce Suite 5.1 から WebSphere Commerce 5.4 へのマイグレーション時には、メンバー・グループ内のマイグレーション済みユーザーのための、MBRGRP テーブルの EXCLUDE 列は、0 (ゼロ) に設定されます。値がゼロであるということは、そのユーザーはメンバー・グループに明示的に含まれているということです。

マイグレーション時のダウン時間の最小化



このセクションでは、2つのマシンを使用するマイグレーション時に、システムがオフラインである時間を最小限にするためのハイレベルな方法を記載しています。

Commerce システムをかなりの程度カスタマイズした場合は、この方法を使用するために IBM サポートに相談することが必要になることがあります。

このマイグレーション・ガイドで説明されているマイグレーション・プロセスは、一般にインプレース・マイグレーションする場合を対象にしています。つまり、同じマシン上で、使用している Commerce Suite 5.1 システムを WebSphere Commerce 5.4 にマイグレーションします。この場合、マイグレーション・プロセス時には、WebSphere Commerce 5.4 をオンラインにするまで、Commerce Suite システムをシャットダウンする必要があります。

以降のいくつかのセクションでは、マイグレーションの進行によってシステムがオフラインである時間を最小限にする 2 つの方法を示しています。どちらの方法も 2 台のマシンを必要とします。

注: これらのアプローチでは、Commerce Suite システムのダウン時間を最小限にできますが、WebSphere Commerce 5.4 システムを設定するための別のハードウェア・リソースが必要です。ただし、マイグレーションを完了した後に、その他の目的で使用する場合は Commerce Suite マシンを再デプロイできます。

別個のマシンへの WebSphere Commerce 5.4 のインストール

この方法では、WebSphere Commerce 5.4 を新規マシンにインストールして WebSphere Commerce Suite 5.1 資産をその新規マシンにコピーし、それらの資産をマイグレーションします。

1. 31 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』の説明に従って、Commerce Suite 5.1 システム、および Commerce Suite 5.1 と WAS のデータベースをバックアップします。
2. *WebSphere Commerce インストール・ガイド バージョン 5.4* の説明に従って、この製品の要件を満たす新しいマシンに WebSphere Commerce 5.4 をインストールします。



構成ファイルに記されているパス名を後から手動で更新しなくてもよいように、Commerce Suite 5.1 で使用したものと同一インストール・ディレクトリーにインストールしてください。(Commerce Suite 5.1 のデフォルトのインストール・ディレクトリーは /opt/WebSphere/CommerceSuite です。) WebSphere Commerce 5.4 のインスタンスは作成しないでください。

本書では、全体を通して WebSphere Commerce 5.4 のデフォルト・インストール・ディレクトリーが /opt/WebSphere/CommerceServer になっています。WebSphere Commerce 5.4 を WebSphere Commerce Suite 5.1 と同一ディレクトリーにインストールする場合は、本書で示されているものとは異なるインストール・ディレクトリーを使用することになりますので注意してください。

3. Commerce Suite データベース (ステップ 1 でバックアップしたもの) を、WebSphere Commerce 5.4 システムにリストアします。データベースのリストア方法については、使用しているデータベース に付属の製品資料を参照してください。WAS データベースは IBM WebSphere Application Server 3.5 に適用され、WebSphere Application Server 4.0.2 のインストール時には新規の WAS40 データベースが作成されるため、WAS データベースをリストアする必要はありません。Commerce Suite 5.1 データベースをリストアした後、Commerce Suite 5.1 で使用されていたデータベース名でデータベース別名のカタログを作成します。
4. Commerce Suite インスタンス・ディレクトリーを WebSphere Commerce 5.4 マシンにコピーします。たとえば、Commerce Suite 5.1 でデフォルトのインスタンス位置を使用する場合は、すべての /opt/WebSphere/CommerceSuite/instances/*your_instance* ディレクトリーを Commerce Suite 5.1 マシンから WebSphere Commerce 5.4 マシンにコピーする必要があります。

注: WebSphere Commerce 5.4 を Commerce Suite 5.1 とは異なる位置にインストールした場合、構成ファイル内にパス名への参照があれば、それらをすべて突き止めて更新する必要があります。

5. *instance_name.xml* (たとえば、demo.xml)、およびその他の関連する .xml ファイルを変更し、古いホスト名をすべて WebSphere Commerce 5.4 マシンの新規ホスト名に変更します。

6. 次のファイルを Commerce Suite 5.1 マシンから WebSphere Commerce 5.4 マシンにコピーします。
 - /opt/WebSphere/CommerceSuite/instances ディレクトリーの wcs_instances ファイル。 WebSphere Commerce 5.4 をインストールした、対応する instances ディレクトリーにコピーします。
 - /opt/WebSphere/CommerceSuite/bin ディレクトリーの cfg.passwd ファイル。 WebSphere Commerce 5.4 をインストールした、対応する instances ディレクトリーにコピーします。
7. WebSphere Commerce 5.4 マシンでリストアした Commerce Suite および WAS データベース上で、67 ページの『第 4 章 データベースのマイグレーション』の説明に従ってデータ・マイグレーション手順を実行します。
8. 95 ページの『Commerce Suite 5.1 インスタンス構成のマイグレーション』の説明に従って、インスタンス・マイグレーション手順を実行します。
9. 100 ページの『セキュリティ構成のマイグレーション』の説明に従って、マーチャント・キーのマイグレーション手順を実行します。
10. 103 ページの『ストア・ファイル資産のマイグレーション』で説明されているように、ストア・データをマイグレーションします。
11. 必要に応じて、本書で説明されている他のすべてのマイグレーション手順を実行します。
12. WebSphere Commerce 5.4 ストアで公開およびショッピングが可能であること、およびシステムがおおむね作動可能であることを確認します。
13. 必要に応じて WebSphere Commerce 5.4 システムを拡張し、新しい機能を最大限に活用します。新機能を実装する方法の詳細については、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。
14. 以下のようにして、WebSphere Commerce データを、オンラインのままである実動 Commerce Suite 5.1 システムからの最新情報でリフレッシュします。
 - a. マイグレーション済みの WebSphere Commerce 5.4 データベースをバックアップします。
 - b. WebSphere Commerce 5.4 データベース内のマイグレーション済みの USERREG、PATRVALUE、ORDPAYINFO、および ORDPAYMTHD テーブルを、テキスト・ファイルにエクスポートします。このテーブルには、ご使用の WebSphere Commerce システムのユーザーの、マイグレーション済みのパスワードとクレジット・カード情報が入っています。
 - c. Commerce Suite 5.1 システムをオフラインにします。
 - d. もう一度 Commerce Suite 5.1 データベースをバックアップして WebSphere Commerce 5.4 マシンにリストアし、最初のバックアップおよびリストア以後の Commerce Suite 5.1 データベースに対する変更を取り込みます。
 - e. Commerce Suite 5.1 システムをシャットダウンします。
 - f. 最新の Commerce Suite 5.1 データベースで、67 ページの『第 4 章 データベースのマイグレーション』の説明に従ってデータ・マイグレーション手順を実行します。
 - g. データベースのリフレッシュ時に、システムに新規ユーザーまたは新規クレジット・カード情報が追加されていない場合は、ステップ 14b のテキスト・ファイルから USERREG、PATRVALUE、ORDPAYINFO、および ORDPAYMTHD

テーブルをインポートします。これらのテーブルには、ご使用の WebSphere Commerce システムのユーザーの、マイグレーション済みのパスワードとクレジット・カード情報が入っています。

前のいくつかのステップを使用したデータベースのリフレッシュ時に、システムに新規ユーザーまたは新規クレジット・カード情報が追加されている場合は、以下のようにします。

- Commerce Suite 5.1 *instance.xml* ファイルにあるオリジナルのマーチャント・キーを、マイグレーション済みの WebSphere Commerce 5.4 *instance.xml* ファイルにコピーします。
 - パスワードとクレジット・カード情報を、100 ページの『セキュリティー構成のマイグレーション』での説明に従って再マイグレーションします。
15. WebSphere Commerce 5.4 システムが作動可能であることを確認したら、オンラインにすることができます。

別個のマシンへの WebSphere Commerce Suite 5.1 環境の複製

この方法では、WebSphere Commerce Suite 5.1 環境を新規マシンに複製してこれらの資産を WebSphere Commerce 5.4 レベルにその新規マシン上でマイグレーションします。これは、以下のようにして行います。

1. 31 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』の説明に従って、実動 Commerce Suite 5.1 システム、および Commerce Suite 5.1 と WAS のデータベースをバックアップします。以下のアイテムにアクセス可能であることを確認します。
 - WebSphere Commerce Suite 5.1 データベース・イメージ。
 - WebSphere Commerce Suite 5.1 Web 資産ファイル (たとえば JSP および *.html ファイル)。
 - カスタマイズ済みのプロパティ・ファイル。
2. 最終的に WebSphere Commerce 5.4 マシンになる新規マシンに WebSphere Commerce Suite 5.1 をインストールします。この説明の目的に応じて、これは **ステージング・マシン** と呼びます。
3. 1 でバックアップしたデータベース・イメージと Web 資産ファイルをこのステージング・マシンにリストアします。
4. ステージング・マシンの環境と構成をセットアップして、可能な限り実動マシンに近いものにします。WebSphere Commerce Suite 5.1 システムおよびストアが、ステージング・マシン上で作動可能であることを確認します。
5. 31 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』の説明に従って、ステージング・サーバー・マシンをバックアップします。
6. このガイドの説明に従って、ステージング・マシン上でインプレース・マイグレーションを続行します。特に以下を実行します。
 - 35 ページの『第 3 章 ソフトウェアのアップグレード』の説明に従って、ソフトウェア・レベルを WebSphere Commerce 5.4 に必要なレベルにアップグレードします。
 - 67 ページの『第 4 章 データベースのマイグレーション』の説明に従って、データ・マイグレーション手順を実行します。

- 95 ページの『Commerce Suite 5.1 インスタンス構成のマイグレーション』の説明に従って、インスタンス・マイグレーション手順を実行します。
 - 100 ページの『セキュリティ構成のマイグレーション』の説明に従って、マーチャント・キーのマイグレーション手順を実行します。
 - 103 ページの『ストア・ファイル資産のマイグレーション』で説明されているように、ストア・データをマイグレーションします。
 - 必要に応じて、本書で説明されている他のすべてのマイグレーション手順を実行します。
7. マイグレーションしたストアで公開およびショッピングが可能であること、およびシステムがおおむね作動可能であることを確認します。
 8. これが WebSphere Commerce 5.4 レベルで作動可能であれば、必要に応じてステージング・システムを拡張し、新しい WebSphere Commerce 5.4 機能を最大限に活用します。新機能を実装する方法の詳細については、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。
 9. 以下のようにして、ステージング・マシン上の WebSphere Commerce データを、オンラインのままである実動 Commerce Suite 5.1 システムからの最新情報でリフレッシュします。
 - a. マイグレーション済みの WebSphere Commerce 5.4 データベースをステージング・マシンにバックアップします。
 - b. WebSphere Commerce 5.4 データベース内のマイグレーション済みの USERREG、ATTRVALUE、ORDPAYINFO、および ORDPAYMTHD テーブルを、テキスト・ファイルにエクスポートします。このテーブルには、ご使用の WebSphere Commerce システムのユーザーの、マイグレーション済みのパスワードとクレジット・カード情報が入っています。
 - c. Commerce Suite 5.1 システムをオフラインにします。
 - d. もう一度 Commerce Suite 5.1 データベースをバックアップして WebSphere Commerce 5.4 ステージング・マシンにリストアし、最初のバックアップおよびリストア以後の Commerce Suite 5.1 データベースに対する変更を取り込みます。
 - e. Commerce Suite 5.1 システムをシャットダウンします。
 - f. 最新の Commerce Suite 5.1 データベースで、67 ページの『第 4 章 データベースのマイグレーション』の説明に従ってデータ・マイグレーション手順を実行します。
 - g. データベースのリフレッシュ時に、システムに新規ユーザーまたは新規クレジット・カード情報が追加されていない場合は、ステップ 9b のテキスト・ファイルから USERREG、ATTRVALUE、ORDPAYINFO、および ORDPAYMTHD テーブルをインポートします。これらのテーブルには、ご使用の WebSphere Commerce システムのユーザーの、マイグレーション済みのパスワードとクレジット・カード情報が入っています。

前のいくつかのステップを使用したデータベースのリフレッシュ時に、システムに新規ユーザーまたは新規クレジット・カード情報が追加されている場合は、以下のようにします。

 - Commerce Suite 5.1 *instance.xml* ファイルにあるオリジナルのマーチャント・キーを、マイグレーション済みの WebSphere Commerce 5.4 *instance.xml* ファイルにコピーします。

- パスワードとクレジット・カード情報を、100 ページの『セキュリティー構成のマイグレーション』での説明に従って再マイグレーションします。
10. WebSphere Commerce 5.4 システムが作動可能であることを確認したら、オンラインにすることができます。

WebSphere Application Server 4.0 への遷移

WebSphere Commerce Suite 5.1 から WebSphere Commerce 5.4 で変わった大きな点の 1 つは、WebSphere Application Server 4.0 のサポートです。このセクションでは、WebSphere Application Server 4.0 へ移行する前に考慮しておく必要のある主要な点のいくつかを大まかに説明します。

バージョン 4.0 の WebSphere Application Server は Java 2 Platform, Enterprise Edition (J2EE) の仕様に完全に準拠しているため、IBM WebSphere Application Server 3.5 バージョンのときに比べて WebSphere Application Server 製品の編成が大きく変更されています。

このレベルの WebSphere Application Server にアップグレードするための詳しい手順については、38 ページの『WebSphere Application Server 4.0.1 へのアップグレード』で説明されています。

遷移の概要

このセクションでは、旧バージョンから WebSphere Application Server 4.0.1 で大きく変わった点を取り上げます。

- J2EE では、開発 (アプリケーションの作成) と管理 (アプリケーションのインストールと運用) の間に明確な区別があります。

この区別は、デプロイ環境に依存しないアプリケーションの開発を可能にします。加えて、J2EE がタスクを区別することによって、初期開発から実動までアプリケーションをプロモートしていくプロセスや、あるサーバーから別のサーバーへアプリケーションを移動するプロセスは単純化されます。これらのそれぞれのケースでは、アプリケーション・コードの変更は必要ありません。変更される可能性があるのはデプロイメントのパラメーターだけです。

バージョン 4.0 は、再編成されたインターフェースによって J2EE でのタスクの区別をサポートします。バージョン 3.x のときは、開発者はアプリケーションの作成、編集、および表示に管理コンソールを使用しましたが、バージョン 4.0 では Application Assembly Tool (AAT) が使用されます。

バージョン 4.0 では、各アプリケーションがサーバー・ドメインにインストールされ、インストールのときに環境にバインドされます。これによって、アプリケーション・レベルでの管理とモジュール・レベルでの管理が可能になり、管理者が個々のサーブレットや JSP や Bean を扱う必要はなくなります。

- J2EE では、アプリケーションとアプリケーション・サーバーの関係が変わりません。

エンタープライズ・アプリケーションには、たくさんの Web モジュールや EJB モジュールが含まれていることがあります。それぞれのモジュールは、異なるアプリケーション・サーバーやそのグループにインストールできます。これは、そのサーバーやサーバー・グループが複数のノードに分かれている場合でもそうです。結果として、1 つのアプリケーションの中に多くのアプリケーション・サー

パーやそのグループに広がるたくさんのモジュールを含めること、そして同様に、数多くの異なるアプリケーションのモジュールを 1 つのアプリケーション・サーバーやそのグループにインストールすることが可能になります。

J2EE アプリケーションを作成したら、管理コンソールを使用してそれをアプリケーション・サーバーにインストールできます。インストールしたモジュールは、管理コンソールで、属しているアプリケーション別に表示させたりインストール先のアプリケーション・サーバー別に表示させたりすることができます。モジュールの開始と停止は、個々のモジュール別にも、集合的にも実行できます。集合的なモジュールの開始は、それらのモジュールが属しているアプリケーションを開始することによっても行えますし、インストール先のアプリケーション・サーバーを開始することによっても行えます。モジュールの停止に関してもこれと同様です。

新しい J2EE アプリケーションのデプロイメント

J2EE アプリケーションの作成には、2 つのステップが関係します。1 つは該当するファイルをアーカイブ (クラス、JSP、HTML、イメージ・ファイルなど) にコピーすること、そしてもう 1 つはモジュールやアプリケーションごとにデプロイメント記述子ファイルを作成することです。バージョン 4.0 では、AAT が、ユーザーが該当する相対パスのファイルをアーカイブにコピーできるようにし、デプロイメント記述子を定義するための GUI 方式を提供することによって、この両方のステップをサポートしています。

開発者は、AAT を使用して、環境固有のバインディング情報も設定できます。これらのバインディングは、アプリケーションが管理コンソールでインストールされる時にデフォルトとして使用されます。加えてユーザーは、J2EE の仕様に対し、クラス名でサブレットを呼び出せるようにするなどの IBM 拡張を定義することもできます。他のアプリケーション・サーバーへの移植性を確保するため、これらの拡張は通常の J2EE デプロイメント記述子とは別の XML ファイルに保管されます。

役割ベースのセキュリティー

バージョン 4.0 のセキュリティーは、J2EE の役割ベースのセキュリティーの仕様に従っています。役割は、アプリケーションのデプロイメント記述子で指定され、アプリケーションがインストールされる時にユーザーかグループにバインドされます。管理コンソールでは、セキュリティー・センターを使用して、1 つのロケーションからすべてのセキュリティー・タスクを実行することができます。これには、アプリケーション内の役割に関するバインディング情報の変更から、セキュリティーを有効にする SSL プロパティーの設定に及ぶすべてが含まれます。また、アプリケーション固有のセキュリティー・タスクも、各アプリケーションのプロパティー・シートから実行できます。

以前インストールしたアプリケーションの再デプロイメント

バージョン 3.x では、すべてのタスクが管理コンソールから実行されました。しかしバージョン 4.0 では、アプリケーションの設定が AAT を通して J2EE デプロイメント記述子で定義されるようになります。

インストールされているアプリケーションのバインディングに影響を与える情報を変更する必要がない場合は、その場でデプロイメント記述子を編集して保管するこ

とができます。このようなアプリケーションを再デプロイするには、そのアプリケーションが保持されている `installedApps` フォルダで直接 AAT をオープンします。

アプリケーションを手動で作成したり編集したりすることも可能です。たとえば、JSP を追加する必要がある場合やサーブレット・クラスを変更する必要がある場合は、単純に新しい (または変更した) ファイルを `installedApps` フォルダ内の該当するロケーションに置くことができます。

バインディングに変更が必要なインストール済みのアプリケーションを再デプロイする場合は、管理コンソールを使用してアプリケーションをエクスポートし、それを AAT で編集した後、再び管理コンソールを使用してそれを再インストールします。既存のバインディング情報はエクスポートの段階で保管されるため、編集の際に追加されたコンポーネントやモジュールに必要な追加のバインディング情報だけが必要です。

注: セキュリティーと整合性を得るため、Web アプリケーションの URL については、すべてのオペレーティング・システムで大文字と小文字が区別されるようになりました。

J2EE リソース・タイプのサポート

バージョン 4.0 では、JDBC プロバイダーやデータ・ソースに加えて、URL や JMS、JavaMail といったいくつかのリソース・タイプが追加されました。どのタイプについても、リソース・プロバイダー (JDBC プロバイダー、URL プロバイダー、および JMS プロバイダー) を作成し、その各プロバイダーにリソース・ファクトリー (データ・ソース、URL、JavaMail セッション、JMS 宛先および JMS 接続) を作成することができます。ただし JavaMail については、構成不可能であり、追加の JavaMail を作成できないため、デフォルトの JavaMail プロバイダーは管理コンソールに表示されません。

J2EE がモデル作成や複製に与える影響

バージョン 3.x では、さまざまなタイプのオブジェクトをモデルとして用いたり複製したりすることができます。しかし、バージョン 4.0 では、J2EE に準拠するようになったため、複製できるのはアプリケーション・サーバーだけです。これらのモデルはサーバー・グループと呼ばれます。各サーバー・グループには、複数のアプリケーション・サーバーや複製を含めることができます。

詳細情報の入手方法

J2EE の詳細については、次の Web サイトを参照してください。

<http://java.sun.com>

構成サポートの変更に関する詳細は、WebSphere Application Server 4.0.1 のマイグレーション情報に含まれている関連情報をご覧ください。WebSphere Application Server 4.0.1 の情報は、次の Web サイトにある WebSphere Application Server Info Center で入手できます。

<http://www.ibm.com/software/webservers/appserv/infocenter.html>

バージョン 4.0 へのアップグレード方法については、以下のセクションを参照してください。

製品の前提条件のマイグレーション

前提条件や要件に関する最新の情報は、次の場所にある WebSphere Application Server 4.0.1 の前提条件の Web サイトで確認できます。

http://www.ibm.com/software/webservers/appserv/doc/v40/prereqs/ae_v401.htm

使用している JDBC プロバイダーが新しいインストールにふさわしいレベルのものであるかどうかを確認してください。このドライバーは、管理データベースに接続するために製品管理サーバーで必要とされます。

DB2[®]、IBM HTTP Server、および他の無料の前提条件のマイグレーション

WebSphere Application Server 4.0.1 では、製品の前提条件のマイグレーションを簡単にするため、サポートされているオペレーティング・システムに無料の Web サーバー、データベース、および IBM SDK for Java をインストールするオプションが用意されています。この IBM SDK for Java は、WebSphere Application Server 4.0.1 で必要とされるレベルとタイプに正確に適合するものです。詳細については、WebSphere Application Server 4.0.1 のインストール・ガイドを参照してください。

CD 版の製品には、無料の前提条件が組み込まれています。Web ダウンロード版はさまざまで、データベース付きのものやデータベース無しものなどがあり、ダウンロード・ファイルのサイズを選択できるようになっています。CD から製品をインストールするのではない場合は、製品の Web サイトで詳細を確認してください。使用したい機能が含まれているインストール・パッケージをダウンロードするようにしてください。

製品をインストールするときには、バックレベルの前提条件をアンインストールし、新しいバージョンをインストールできます。

非 IBM 前提条件のマイグレーション

それですでに最初に、先に言及した前提条件のページを調べて、どのソフトウェアにマイグレーションやアップグレードが必要かを確認してください。次いで、特定の製品に関する資料から、この製品でサポートされているバージョンへのマイグレーションの方法を確認してください。WebSphere Application Server 4.0.1 のインストール時に提供されない前提条件については、WebSphere Application Server 4.0.1 をインストールする前に前提条件をマイグレーションまたはアップグレードするのが、最も安全な方法です。

バージョン 4.0 への構成のマッピング

このセクションでは、前のバージョンの製品の構成をリストアする際に、オブジェクトや属性をバージョン 4.0 環境にマップする方法について詳しく説明します。

- ディレクトリー stdin、stdout、stderr パッシベーション・ディレクトリーおよび作業ディレクトリー

これらのディレクトリーの一般的なロケーションには、バージョン 3.x のインストール・ディレクトリーが含まれている場合がありますが、新しいバージョン 4.0 のインストールではこのロケーションが異なることがあるため、これらのエントリーが指定された場合は追加のチェックが行われます。バージョン 3.x と異なり、バージョン 4.0 のインストールでは、stdin、stdout、および stderr の

デフォルト・ロケーションが logs ディレクトリーになっています。ディレクトリーのマッピングの前には、パッシベーション・ディレクトリーと作業ディレクトリーの有無がチェックされます。これらのディレクトリーが存在する場合は、それが使用され、存在しない場合は、代わりに適当なデフォルトが使用されます。

- Enterprise Bean

バージョン 3.x でサポートされていた仕様レベルは EJB 1.0 だけでしたが、バージョン 4.0 では、EJB 1.1 だけがサポートされています。とはいえ、多くの EJB 1.0 の Bean は、EJB 1.1 の Bean として正常にデプロイできます。Enterprise Bean は、アプリケーション・マイグレーションのフェーズの一部として自動的に再デプロイされます。必ず WASPostUpgrade.log を調べてこれらの Bean のデプロイメントの詳細を確認し、必要な変更を加え、再デプロイしてください。

- JDBC プロバイダーとデータ・ソース

バージョン 4.0 では、JDBC オブジェクトと DataSource オブジェクトが大幅に再定義されています。これらのオブジェクトは、バージョン 3.x の設定を入力変数として用い、新しい構成にマップされます。

バージョン 3.x からマップされたデータ・ソースと、サンプルで定義されているデータ・ソースの違いにお気付きになったかもしれません。この違いは、データ・ソースのユーザー ID とパスワードのフィールドにあります。サンプルではデフォルトのユーザー ID とパスワードが入っていますが、マイグレーションされたデータ・ソースはそうになっていません。これは、ユーザー ID とパスワードのデータが、データ・ソースではなく、Enterprise Bean のバインディングで定義されるからです。バージョン 3.x では、情報がコンテナ・レベルと EJB レベルで定義されているため、これを Enterprise Bean にマップする必要があります。

- JSP レベル

バージョン 4.0 は、JSP 0.91 をサポートしていません。それで、JSP 0.91 として実行するように構成された JSP オブジェクトはマイグレーションされませんが、これらは出力の中に入れられ、ログに記録されます。バージョン 4.0 では JSP 1.1 しかサポートされていないため、JSP 1.0 オブジェクトと 1.1 オブジェクトは JSP 1.1 として実行されます。

- モデルと複製

バージョン 4.0 では、モデルと複製の定義が大きく変わっています。バージョン 4.0 でモデルおよび複製としてサポートされるのは、アプリケーション・サーバーだけです。この点は、多くのオブジェクトがモデルや複製として使用できたバージョン 3.x とは大きく異なっています。バージョン 4.0 では、アプリケーション・サーバーに関係するすべてのモデルと複製がサーバー・グループにマップされます。

以前複製可能だった他のすべてのオブジェクトのマイグレーションでは、特別なマッピングが行われます。すべての複製は単なるオブジェクトとして扱われ、複製ではないかのようにマップされます。また、アプリケーション・サーバー・モデル以外のモデルは無視され、マップされません。

- 複数のアプリケーション・サーバー

バージョン 4.0 の Advanced Single Server エディションと Advanced Developer エディションでは、一度に 1 つのアプリケーション・サーバーしか構成できません

ん。バージョン 3.x では、一度にたくさんのアプリケーション・サーバーを定義することができました。これらのオブジェクトをバージョン 4.0 のどちらかのエディションにマイグレーションする際は、アプリケーション・サーバーの名前によって、行われるマイグレーションが判別されます。アプリケーション・サーバーの名前が適合する場合（たとえば、Default Server など）は、以前の構成に合わせてバージョン 4.0 オブジェクトの属性が更新され、子はすべてそのアプリケーション・サーバーにマイグレーションされます。一方名前が適合しなかった場合は、そのバージョン 3.x アプリケーション・サーバーの子だけが、バージョン 4.0 環境内の 1 つのアプリケーション・サーバーにマイグレーションされます。

- ノード名

バージョン 3.x のリポジトリには、複数のノード名とそれに関連付けられた子を入れることができます。しかし WASPostUpgrade ツールでは、マイグレーションされるノードに一致するオブジェクトとその子だけしか処理されません。この判断は、構成ファイル内のノードの名前と、マイグレーションされるマシンの完全修飾および非修飾ネットワーク名を調べることによって下されます。

- サーブレット・リダイレクター

バージョン 4.0 では、サーブレット・リダイレクターはサポートされていません。これらのオブジェクトは無視されます。

- トランスポート

バージョン 3.x のサーブレット・エンジンのデフォルト・トランスポート・タイプは、Open Servlet Engine (OSE) でした。しかし、バージョン 4.0 では OSE トランスポートがサポートされなくなったため、これらのトランスポートは同じポート割り当てで HTTP トランスポートにマップされます。

- datasources.xml

バージョン 3.x では、datasources.xml ファイルを使用してデータ・ソース構成の設定を補足できました。このファイルは properties ディレクトリに保管されました。このファイルが存在している場合は、ファイル内のプロパティがデータ・ソースの構成と JDBC プロバイダーの構成にマージされます。

新しいインストール単位に以前の構成をリストアする

バージョン 3.x 以降のインストールでは、システム構成のマイグレーションを助ける一連のマイグレーション・ツールが製品に装備されています。これらのツールは、自動マイグレーション・サポートの一環として、製品インストール・プログラムによって呼び出されますが、コマンド行から自分で呼び出すことも可能です。

バージョン 3.x の構成をリストアするツールは WASPostUpgrade といいます。このツールは、WASPreUpgrade ツールによって作成された情報を使用して、バージョン 4.0 のインストール単位に以前のバージョン 3.x の構成をリストアします。

バージョン 4.0 の製品は J2EE のプログラミング・モデルに従っていますが、それより前のバージョンはこれに従っていないため、バージョン 3.x の構成をバージョン 4.0 のインストールに適用するには、かなりの変更が必要になります。

J2EE アプリケーションの作成とデプロイメント

J2EE のプログラミング・モデルでは、アプリケーションを作成しデプロイする方法に関して、アーキテクチャーが指定されています。バージョン 3.x のアプリケーションにはこの同じアーキテクチャーが使用されていないため、マイグレーションの

プロセスでは、これらのアプリケーションを再作成します。マイグレーションされた Web リソースと Enterprise Bean は、J2EE アプリケーションで作成されます。バージョン 3.x で定義されたエンタープライズ・アプリケーションは、すべて、同じ名前でも J2EE アプリケーションにマップされ、デフォルトのサーバーにデプロイされます。マップされたもののエンタープライズ・アプリケーションに組み込まれていない、他のすべての Web リソースと Enterprise Bean は、*DefaultApplication* というデフォルトの J2EE アプリケーションにマップされます。

Web アプリケーションは J2EE WAR ファイルにマップされます。Enterprise Bean は EJB 1.1 Bean として J2EE JAR ファイルにデプロイされます。これらのリソースは J2EE EAR ファイルに結合されて、バージョン 4.0 の構成にデプロイされます。EJB 1.0 の仕様と EJB 1.1 の仕様との間にはいくつかの違いがありますが、ほとんどのケースで、EJB 1.0 Bean は EJB 1.1 Bean として正常に実行することが可能です。なお、WASPostUpgrade.log (このトピックの最後を参照) には、デプロイされた各 Bean に固有の詳細情報が保管されていますので、このログを注意深く分析してください。

セキュリティ

バージョン 3.x 環境に適用できたセキュリティ設定は、マイグレーション・プロセスの一環として J2EE のセキュリティ属性に適用されます。

サンプル

サンプルはマイグレーションされません。これは、バージョン 4.0 の J2EE に合わせて特別に変更されています。ですから、バージョン 3.x 製品で以前提供されたサンプルは使用せずに、新しいサンプルを使用してください。

マッピングに関する詳細

オブジェクトや属性をバージョン 4.0 の構成にマップする方法についての詳細は、関連情報を参照してください。

WASPostUpgrade パラメーター

Advanced Edition のインストールで構成をリストアする場合、管理サーバーは、以下のコマンドを正常に実行できる状態で稼働していなければなりません。

```
com.ibm.websphere.migration.postupgrade.WASPostUpgrade backupDirectoryName
[-import xml_data_file]
[-adminNodeName primary_node_name]
[-nameServiceHost host_name [-nameServicePort port_number]]
[-traceString trace_spec [-traceFile file_name]]
[-substitute "key1=value1[;key2=value2;[...]]"]
```

このコマンドの最初のパラメーターは必須です。その他のパラメーターは、特に注記がない限りオプションです。パラメーターを要約すると、次のようになります。

- *backupDirectoryName* - 保管されている構成と WASPreUpgrade コマンドで作成されたファイルがあるディレクトリーの名前。これは必要パラメーターです。
- [-import *xml_data_file*] - バージョン 3.x の XMLConfig ツールで作成した XML データ・ファイルを指定する際に使用する、オプション・パラメーター。このパラメーターが指定されない場合は、バックアップ・ディレクトリーにあるデフォルトの XML 構成ファイル (*websphere_3x_backup.xml*) が使用されます。

- [-adminNodeName *primary_node_name*] - 現行インストールの管理ノードの名前。このパラメーターは、Advanced Edition のインストールで構成をリストアする場合に必須です。XMLConfig は、このパラメーターを使用して呼び出されます。
- [-nameServiceHost *host_name* [-nameServicePort *port_number*]] - 指定しておく、XMLConfig 呼び出しに渡されるオプション・パラメーター。このパラメーターは、XMLConfig で使用されるデフォルトのホスト名やポート番号をオーバーライドする際に使用できます。このパラメーターは、Advanced Edition のインストールで構成をリストアする場合にのみ、使用します。
- [-traceString *trace_spec* [-traceFile *file_name*]] - IBM サービス技術員が使用するトレース情報の収集に使用されるオプション・パラメーター。traceString の値は "*=all=enabled" です。正しく処理されるように、必ず引用符を付けて指定してください。
- [-substitute "*key1=value1[;key2=value2;[...]]*"] - 指定しておく、XMLConfig 呼び出しに渡されるオプション・パラメーター。このパラメーターは、XML データ・ファイル内のセキュリティー値を置換するのに使用します。入力 XML ファイルでは、置換の対象となる各鍵を \$key\$ と表記します。

ロギング

WASPostUpgrade ツールは、稼働中、その状況を画面に表示します。さらに WASPostUpgrade は、より詳しい一連のロギング情報を logs ディレクトリーに保管します。ログは WASPostUpgrade.log というファイルに記録されます。このファイルはテキスト・エディターで表示できます。

第 2 章 Commerce Suite 5.1 のバックアップ

本書で説明しているマイグレーション・プロセスを進める前に、実動 Commerce Suite 5.1 システムの完全なシステム・バックアップを実行する必要があります。これにより、WebSphere Commerce 5.4 へのマイグレーション中に問題が生じて、以前のシステムにリカバリーすることができます。

同じ場所へマイグレーションをする場合には、WebSphere Commerce 5.4 へのマイグレーションが完了したら、以前のバージョンの Commerce Suite 5.1 に復帰することはできません。同じマシンでの WebSphere Commerce 5.4 と Commerce Suite 5.1 の共存はサポートされていません。

Commerce Suite 5.1 システムのバックアップ

使用している Commerce Suite 5.1 システムをバックアップするには、以下のようになります。

- 使用するオペレーティング・システムに付属する資料か、バックアップおよびリストア専用ソフトウェアに付属する資料に従って、Commerce Suite システムの完全なシステム・バックアップを実行します。通常は、磁気テープ装置、ZIP ドライブ、または他のファイル・システムにシステムをバックアップできます。
- バックアップには、Commerce Suite 5.1 で使用したカスタマイズ・ファイルおよびディレクトリー、さらにデータベース、Web サーバー、WebSphere Payment Manager、WebSphere Application Server、および IBM Developer Kit, Java 2 Technology Edition などの関連コンポーネントを含める必要があります。
- 特に、基礎となるすべてのサブディレクトリーとファイルを含む、主な Commerce Suite 5.1 インストール・ディレクトリーについては、マイグレーション・プロセス中にそれらのディレクトリーとファイルの参照が必要になることがあるので、マイグレーション・プロセス時に容易にアクセスできる場所にバックアップします。

ディレクトリーおよびファイルのバックアップ

以下のディレクトリーまたはファイルを手動でバックアップするには、以下のようになります。

1. コマンド・プロンプトで、一次バックアップ・ディレクトリーを作成します。
2. Commerce Suite 5.1 インストール・ディレクトリー (/opt/WebSphere/CommerceSuite) に移動します。
3. 適切なディレクトリーまたはファイルを選択し、一時バックアップ・ディレクトリーにコピーします。

特に、次のディレクトリーおよびファイルをバックアップする必要があります。

- Commerce Suite 5.1 インストール・ディレクトリー
/opt/WebSphere/CommerceSuite

- 次のような主な Commerce Suite 5.1 インストール・ディレクトリーのサブディレクトリー
 - stores
 - web
 - instances¥*your_instance*
(あるいは、作成したインスタンスが存在する場所。)
- 次のような重要な Commerce Suite 5.1 構成ファイル
 - /opt/WebSphere/CommerceSuite/bin ディレクトリーの `cfg.passwd` ファイル。
 - /opt/WebSphere/CommerceSuite/instances ディレクトリーの `wcs_instances` ファイル。
 - /opt/IBMHTTPServer/conf ディレクトリーの `httpd.conf` ファイル、またはそれに相当する他の Web サーバーの構成ファイル。
 - /opt/IBMHTTPServer/ssl パスの `keyfile.kdb` および `keyfile.sth` テスト鍵ファイル。
 - /opt/WebSphere/CommerceSuite/instances/*your_instance*/xml/rules ディレクトリーの `wcs.server` ファイル。
 - 静的 HTML ページや GIF ファイルなどのファイル・ベースのコンテンツ。
 - データベース `.sql` スクリプト。
 - JavaServer Pages (JSP ファイル)。
 - カスタマイズしたコマンドおよびファイル (たとえば、`.java`、`.class`、`.jar`、`.zip`、または `.properties` ファイル)。
 - カスタマイズした文書ファイル (たとえば、`.pdf` またはテキスト・ファイル)。
- キャンペーンのルール・プロジェクト。これらのファイルは、キャンペーンを公開すると生成されます。これらはキャンペーン後に名前が付けられますが、以下のようにさまざまな拡張子が付きます。
 - `campaign_name.adv`
 - `campaign_name.cdd`
 - `campaign_name.dbcp`
 - `campaign_name.flow0`
 - `campaign_name.flow1`
 - `campaign_name.jcp`
 - `campaign_name.rb`

WebSphere Commerce 5.4 でキャンペーン・コードの実行を開始すると、WebSphere Commerce はこれらのファイルを探し、新しい WebSphere Commerce スキーマ・テーブルへ永続的に保管します。データがこれらのファイルからスキーマへ転送されると、キャンペーン・コードは、ファイルではなくデータベースの探索を開始します。最終的に WebSphere Commerce 5.4 でキャンペーンが実行されて完成した時点で、これらのファイルは古くなります。

データベースのバックアップ

続くいくつかのセクションでは、データベースをバックアップする方法を説明します。

DB2 データベース

DB2 データベースをバックアップするには、2 層環境 (データベースが Commerce Suite からリモートにあるマシンにインストールされている環境) の DB2 サーバー・マシンか、単一層環境 (データベースが Commerce Suite と同じマシンにインストールされている環境) の Commerce Suite マシンから、以下のアクションを実行します。

1. DB2 データベースを作成したか、または DB2 データベースを所有する Windows® ユーザー ID で、Windows システムにログオンします。
2. ご使用の DB2 インスタンス所有者 ID (たとえば `db2inst1`) でログオンします。

```
su - db2_instance
```

3. すべてのデータベース・トランザクションを完了させます。
4. すべてのアプリケーションが各データベースから切断されていることを確認します。以下のコマンドを実行して、データベースに接続されているすべてのアプリケーションのリストを表示します。

```
db2 list applications
```

すべてのアプリケーションが切断されていれば、以下のメッセージが表示されません。

```
SQL1611W No data was returned by the Database System Monitor. SQLSTATE=00000
```

以下のコマンドを実行して、すべてのアプリケーションを強制的に切断します。

```
db2 force applications all
```

5. 以下のコマンドを再発行して、すべてのアプリケーションが停止したことを確認します。

```
db2 list applications
```

すべてのアプリケーションが停止していれば、次のメッセージが表示されます。

```
SQL1611W No data was returned by the Database System Monitor. SQLSTATE=00000
```

6. すべての Commerce Suite 5.1 データベース (MSER、MALL など) がカタログされていることを確認します。以下のコマンドを実行して、現在のインスタンス内でカタログされたすべてのデータベースのリストを表示します。

```
db2 list database directory
```

7. データベースをバックアップするディレクトリーを作成します。たとえば、`/db2_backup` などとします。
8. 以下のコマンドを入力して、すべての Commerce Suite 5.1 データベース (MSER、MALL など) をバックアップします。

```
db2 backup database db_name to backup_directory
```

`db_name` は、データベースの名前で、`backup_directory` は、データベースのバックアップ先の絶対パスです。 `backup_directory` は存在していなければなりません。バックアップが成功したことを示すメッセージが表示されるはずですが、たと

例えば、MALL データベースを上記のバックアップ・ディレクトリーにバックアップするには、次のコマンドを使用します。

```
db2 backup database MALL to /db2_backup
```

複数のデータベースがある場合、Commerce Suite 5.1 データベースごとにコマンドを繰り返します。



WAS データベースのような、Commerce Suite 以外のデータベースも、ここでバックアップすることを考慮する必要があります。

データベースのバックアップの詳細については、*DB2 管理の手引き* を参照してください。バックアップ・コマンドの構文の詳細については、*DB2 コマンド解説書* を参照してください。

第 3 章 ソフトウェアのアップグレード

この章では、WebSphere Commerce 5.4 で必要なレベルにソフトウェアをアップグレードする方法について説明します。ソフトウェアをアップグレードする前に、データベースなどの、ご使用の Commerce Suite 5.1 システムのバックアップを行ってください。システムのバックアップを実行する方法については、31 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』を参照してください。

Commerce Suite 5.1 のいずれかのソフトウェア・コンポーネントを停止する必要がある場合は、153 ページの『付録 A. WebSphere Commerce Suite 5.1 コンポーネントの開始と停止』を参照してください。

重要

この章の情報は、単一層の環境でのソフトウェアのアップグレード方法についてのみ説明しています。リモート Web サーバー (3 層環境) を使用している場合、リモート Web サーバーとリモート・データベースのインストールおよびセットアップ手順を説明している、*WebSphere Commerce* インストール・ガイド バージョン 5.4 のセクションを参照する必要があります。

WebSphere Commerce Suite 5.1 および WebSphere Commerce 5.4 IBM ソフトウェアのマッピング

以下の表は、Commerce Suite 5.1 または WebSphere Commerce 5.4 のどちらかがパッケージされているほとんどのソフトウェアについて、Commerce Suite 5.1 と WebSphere Commerce 5.4 との間のバージョン・レベルとデフォルト・インストール・パスの対応関係を示しています。

表 4. *WebSphere Commerce 5.4 for Linux* で提供されているソフトウェア・バージョンおよびインストール・パス

ソフトウェア	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
WebSphere Commerce	バージョン 5.1 /opt/WebSphere/CommerceSuite	バージョン 5.4 /opt/WebSphere/CommerceServer
DB2 Universal Database™ Enterprise Edition	バージョン 7.1.0.28 /usr/IBMDB2/V7.1	バージョン 7.1.0.60 (DB2 7.2 FixPak 6) /usr/IBMDB2/V7.1
IBM WebSphere Application Server — Advanced Edition	バージョン 3.5.4.1 /opt/IBMWebAS	バージョン 4.0.2 (バージョン 4.0.1 にバージョン 4.0.2 PTF および eFix を適用) /opt/WebSphere/AppServer
IBM HTTP Server	バージョン 1.3.12 /opt/IBMHTTPServer	バージョン 1.3.19.1 /opt/IBMHTTPServer
IBM SDK for Java	バージョン 1.2.2 /opt/IBMJava2-122	バージョン 1.3.1 /opt/WebSphere/AppServer/java

表 4. WebSphere Commerce 5.4 for Linux で提供されているソフトウェア・バージョンおよびインストール・パス (続き)

ソフトウェア	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
IBM WebSphere Payment Manager	バージョン 2.2 /opt/PaymentManager	バージョン 3.1.2 /opt/PaymentManager
IBM SecureWay [®] Directory Server	バージョン 3.2.1	バージョン 3.2.1
Blaze Advisor [™] ルール・サーバー	バージョン 3.1.2 /opt/WebSphere/CommerceSuite/Blaze	バージョン 3.1.3 /opt/WebSphere/CommerceServer/blaze
Macromedia LikeMinds Personalization Server	バージョン	バージョン 5.2.x

ハードウェアのアップグレード

現在のマシンが以下のハードウェア要件を 1 つでも満たしていないなら、ハードウェアに必要なアップグレードをすべて行い、要件を満たすようにする必要があります。

以下の要件を満たす、専用の Pentium[®] III 733 MHz を搭載する IBM 互換のパーソナル・コンピューター (実稼働環境を考慮すると 1 GHz 以上を推奨)。

- 1 プロセッサあたり 768 MB 以上のランダム・アクセス・メモリー (RAM)
- 4 GB 以上のフリー・ディスク・スペース
- 上記の要件に加え、マシン上に Commerce Suite 5.1 データベースのバックアップを収めるのに十分なフリー・ディスク・スペース。データベース・マイグレーション・スクリプトはデータベースのバックアップをマイグレーション時に 3 回実行するので、この追加のディスク・スペースは、少なくとも現在の WebSphere Commerce Suite 5.1 データベースの 3 倍のサイズが必要です。
- 1 プロセッサあたり 1 GB 以上のページング・スペース
- CD-ROM ドライブ
- グラフィックス表示可能モニター
- TCP/IP プロトコルをサポートするローカル・エリア・ネットワーク (LAN) アダプター。

詳細については、WebSphere Commerce インストール・ガイド バージョン 5.4 でのインストール前についてのセクションを参照してください。

オペレーティング・システムのアップグレード

WebSphere Commerce 5.4 は以下のものをサポートします。

- Red Hat Linux 7.2

Red Hat Linux 7.2 のセットアップで **upgrade** オプションを使用して、以前のバージョン (Red Hat Linux 7.1) からアップグレードすることができます。

注: Red Hat Linux 7.2 にアップグレードする前に、Red Hat Linux 7.1 の /etc/services ファイルをバックアップしておきます。Red Hat Linux 7.2 にアップグレードした後、DB2 のポート番号が /etc/services ファイルに

入っていないと DB2 を開始する際に問題が発生することがあります。オペレーティング・システムをアップグレードしたら、バックアップした `services` ファイルから新しい `/etc/services` ファイルに DB2 ポート番号をコピーします (番号がそのファイルに入っていない場合)。以下に例を示します。

```
db2cdb2inst1 50000/tcp # Connection port for DB2 instance db2inst1
db2idb2inst1 50001/tcp # Interrupt port for DB2 instance db2inst1
```

- SLES 7.0 (SuSE Linux Enterprise Server 7)

SuSE では SuSE Linux 7.2 から SLES 7.0 へのアップグレードをサポートしていないため、17 ページの『マイグレーション時のダウン時間の最小化』の説明に従って、2 台のマシンを使用して WebSphere Commerce Suite 5.1 を WebSphere Commerce 5.4 にマイグレーションする必要があります。つまり、SLES 7.0 を実行する新しいマシンに WebSphere Commerce 5.4 をインストールして、そのマシンに WebSphere Commerce Suite 5.1 資産をコピーしてから、17 ページの『マイグレーション時のダウン時間の最小化』の説明に従ってそれらの資産をマイグレーションする必要があります。

以下のパッケージがシステムで使用可能になっていることを確認します。次のコマンドを入力して、それらが使用可能かどうかを調べることができます。

```
rpm -qa | grep package_name
```

パッケージがインストールされていない場合、WebSphere Commerce のインストールを進める前に、それらをインストールする必要があります。

Red Hat Linux 7.2 の場合:

- `pdcksh-5.2.14-12`
- `ncurses4-5.0-2`

SLES 7.0 の場合:

- `pdcksh-5.2.14-12`

IBM 以外のソフトウェアのアップグレード

このセクションでは、Commerce Suite がサポートしている IBM 以外のソフトウェア・コンポーネント (Web ブラウザーなど) をアップグレードする方法について説明します。

Internet Explorer 5.5 以降

WebSphere Commerce のツールとオンライン・ヘルプには、WebSphere Commerce のマシンと同じネットワーク上において Windows オペレーティング・システムが実行されているマシンにおいて、Microsoft® Internet Explorer 5.5 を使用してのみアクセスできます。Internet Explorer は、5.50.4522.1800 のフル・バージョンのもの (Internet Explorer 5.5 Service Pack 1 およびインターネット・ツール)、あるいはそれ以降のものに対して Microsoft による最新の重要なセキュリティ更新を適用したものをを使用する必要があります。それより前のバージョンでは、WebSphere Commerce のツールの機能が完全にはサポートされません。

Internet Explorer は、以下の Microsoft のダウンロード・ページからダウンロードできます。

<http://www.microsoft.com/downloads/>

シヨッパーは、以下のいずれかの Web ブラウザーを使用して Web サイトにアクセスできます。これらは、すべて WebSphere Commerce でテスト済みです。

- Netscape Communicator 4.6 でサポートされている Netscape Navigator のすべてのバージョン (Netscape Navigator 4.04 および 4.5 を含む)
- Netscape Navigator for Macintosh 3.0 および 4.0 以上
- Microsoft Internet Explorer 4 および 5
- AOL 5 および 6

WebSphere Application Server 4.0.1 へのアップグレード

注:

始める前に、バージョン 3.x に対するバージョン 4.0 の再編成について説明している 22 ページの『遷移の概要』を必ず読んでください。製品のマイグレーション・プロセスの要約が続いて説明されています。このほとんどは、WebSphere Application Server 4.0.1 インストールおよびマイグレーション・プログラムがユーザーの代わりに行います。

1. 39 ページの『現在の構成のバックアップ』の説明に従って、現在の管理構成をバックアップします。
2. 39 ページの『Commerce Server アプリケーション・サーバーの除去』の説明に従って、WebSphere Application Server 4.0.1 にアップグレードする前に、**WebSphere Commerce Suite** アプリケーション・サーバーを WebSphere Application Server 管理コンソールから除去します。WebSphere Commerce 5.4 のインストールの後、新規の WebSphere Commerce アプリケーション・サーバーが作成されます。
3. 40 ページの『WebSphere Application Server 自動マイグレーション・サポートの使用』のステップに従って、残りの IBM WebSphere Application Server 3.5 管理構成とユーザー・データ・ファイルをバックアップするとともに、マイグレーション環境をセットアップします。このバックアップには WebSphere Commerce アプリケーション・サーバーの構成は含まれないことに注意してください。デフォルトの was3_backup または was_migration バックアップ・ディレクトリーを使用してください。最初にインストールおよびマイグレーション・プログラムを実行する時、バックアップを行うために、マイグレーションをする前の初期段階があります。
4. 42 ページの『IBM HTTP Server のアンインストール』および 42 ページの『IBM WebSphere Application Server のアンインストール』の説明に従って、現行バージョンの IBM HTTP Server、および IBM WebSphere Application Server を停止し、アンインストールします。
5. 43 ページの『WebSphere Application Server リポジトリーの除去』の説明に従って、以前の WebSphere Application Server インストールで使用されていた管理データベースを除去します。
6. 44 ページの『DB2 7.1 FixPak 3 (バージョン 7.1.0.41) へのアップグレード』と 46 ページの『DB2 7.2 FixPak 6 (DB2 7.1.0.60) へのアップグレード』の説明に従って、DB2 をアップグレードします。

7. 50 ページの『WebSphere Application Server 4.0.1 のインストールとインストール後の段階』の説明に従って、新しいバージョンの IBM WebSphere Application Server をインストールします。これには、DB2 のレベルなどのさまざまな前提条件をアップグレードすることも含まれます。
8. 新規の管理サーバーを始動します。
9. was3_backup ディレクトリーを使用して、構成を新規のインストールにリストアします。

現在の構成のバックアップ

WebSphere Application Server 4.0.1 への切り替えを始める前に、既存の WebSphere Application Server 構成をバックアップする必要があります。

1. WebSphere Application Server 管理サーバーが稼働中であることを確認してください。
2. 構成をエクスポートできるバックアップ・ディレクトリー (たとえば、/was_backup_all) を作成します。
3. WebSphere Application Server XMLConfig ユーティリティーを使用し、-export フラグを付けて現在の管理構成をバックアップします。コマンド行から /opt/IBMWebAS/bin のパスに切り替え、次のコマンドを実行します。

```
./XMLConfig.sh -export back_up_file_name.xml -adminNodeName your_host_name
```

この中で、back_up_file_name.xml は、現在使用中の WebSphere Application Server 構成のバックアップ・ファイルの完全修飾名です。たとえば、以下のようになります。

```
./XMLConfig.sh -export /was_backup_all/was35_backup_all.xml -adminNodeName montreal
```

エクスポートされたすべての項目はスクリーンに表示されます。

Commerce Server アプリケーション・サーバーの除去

以下のようにして WebSphere Commerce アプリケーション・サーバーを WebSphere Application Server 管理コンソールから除去する必要があります。

1. WebSphere Application Server が停止しているなら、始動します。
2. WebSphere Application Server 管理コンソールで、以下のようになります。
 - a. ご使用のホスト名まで、ノードを展開します。
 - b. ご使用のインスタンスに対応する WebSphere Commerce Suite Application Server を右マウス・ボタン・クリックして、「**Stop (停止)**」を選択します。
 - c. ご使用のインスタンスのアプリケーション・サーバーが停止したら、そのインスタンスをもう一度右マウス・ボタン・クリックして「**Remove (除去)**」を選択します。
 - d. データ・ソース・エントリーを右マウス・ボタン・クリックして、「**Remove (除去)**」を選択することにより WebSphere Commerce Suite のデータ・ソースを除去します。
 - e. 「WebSphere Commerce Suite JDBC Driver (WebSphere Commerce Suite JDBC ドライバー)」上で右マウス・ボタン・クリックして、「**Uninstall (アンインストール)**」を選択します。表示されるウィンドウでご使用のホスト名を選択し、「**Uninstall (アンインストール)**」をクリックします。

- f. 定義された JDBC ドライバーを右マウス・ボタン・クリックして、「**Remove (除去)**」を選択することにより、その JDBC ドライバーを除去します。

WebSphere Application Server 3.5.4 管理コンソールで、以下のエントリが WebSphere Application Server から除去されたことを確かめてください。

- WebSphere Commerce Application Server
- WebSphere Commerce DataSource
- WebSphere Commerce JDBC ドライバー
- 仮想ホスト — `VH_instance_name_tools` (たとえば、`VH_demo_tools`)

WebSphere Application Server 自動マイグレーション・サポートの使用

IBM WebSphere Application Server バージョン 4.0 では、自動マイグレーション・サポートがインストール・プログラムに含まれています。インストール・プログラムはインストール前の段階で、今までインストールされていたバージョンを検出し、マイグレーションしたインストールの表示方法に関する情報を集め、さらに現在の管理構成をエクスポートします。製品前提条件の更新の後、製品の新しいバージョンをインストールし、バックアップしてある管理構成をインポートするという、2 番目の段階を実行します。

注: 開始する前に、WebSphere Application Server 管理サーバーと Web サーバーが稼働していることを確認してください。

インストール前の段階

インストール前の段階で、以下のようになります。

1. インストール・プログラムを始動すると、それは以前にインストールされたバージョンの製品を自動的に検出し、それらをリストに表示します。インストール・プログラムが、選択したバージョンからのマイグレーションをサポートしている場合、「**Perform migration (マイグレーションを実行)**」チェック・ボックスがリストの上方に表示されます。
2. インストール・プログラムによって、以下の情報を求められます。
 - バックアップ・ディレクトリー。
デフォルト・ディレクトリー (`/opt/IBMWebAS/migration/backup`) は後続の段階で削除されるので、`/was3_backup` などの別のディレクトリーでオーバーライドしてください。
 - 一時ステージング用ディレクトリー。
デフォルト・ディレクトリー (`/opt/IBMWebAS/migration/tmp`) は後続の段階で削除されるので、`/was3_migration` などの別のディレクトリーでオーバーライドしてください。
 - マイグレーション・ログ用ディレクトリー。
デフォルト・ディレクトリー (`/opt/IBMWebAS/migration/logs`) は後続の段階で削除されるので、`/was_migration/logs` などの別のディレクトリーでオーバーライドしてください。

3. マイグレーション・プロセスを開始するための確認プロンプトで、「はい」をクリックして続けます。
4. まだ WebSphere Application Server 管理サーバーを始動していないなら、インストール・プログラムはそれを始動するよう促します。
5. 「**Start Migration (マイグレーションを開始)**」をクリックします。インストール・プログラムは現在の構成をエクスポートし、マイグレーション・ログ・ファイルを表示します。マイグレーション・ログはマイグレーション終了後のみ表示されます。
6. インストール・プログラムがマイグレーション・ログ・ファイルをチェックするように促す時、以下のうちの 1 つを行ってください。
 - マイグレーション・ログ・ファイルがマイグレーションの成功を示している場合は、「終了」をクリックします。
次に 50 ページの『WebSphere Application Server 4.0.1 のインストールとインストール後の段階』でインストール・プログラムを実行する時、マイグレーションにおける次の段階 (インストールとマイグレーション後処理) が開始します。
 - マイグレーション・ログ・ファイルがマイグレーションの成功を示していない場合は、「キャンセル」をクリックし、ログに記録されているエラーを訂正します。
このケースでは、次にインストール・プログラムを始動する時に、マイグレーションは最初から開始します。

注: マイグレーション・ログ・ファイルがマイグレーションに問題があることを示しているのに「完了」をクリックした場合には、ファイル `/tmp/WAS_Migration_temp.properties` を削除するまでマイグレーションのこの段階を再実行することはできません。この段階が正常に完了したら、以下のようになります。

1. 前提条件をマイグレーションします。
2. バージョン 4.0 をバージョン 3.x と同じディレクトリー構造にインストールしようとする場合には、次のいずれかを行います。
 - マイグレーション・バックアップ・ディレクトリーがバージョン 3.x ディレクトリー構造内にある場合は、そのマイグレーション・バックアップ・ディレクトリーを保存して、バージョン 3.x のディレクトリー構造の残りの部分を削除します。
 - そうでなければ、バージョン 3.x のディレクトリー構造全体を削除します。
3. インストール・プログラムを再始動します。

IBM HTTP Server および WebSphere Application Server のアンインストール

マイグレーション前の段階を完了した後、前のバージョンの IBM HTTP Server および WebSphere Application Server をアンインストールします。DB2 データベースでは、以前の WAS リポジトリーの必要性がなくなった場合、それを WebSphere Application Server 4.0.1 のインストール後の WAS40 リポジトリーの作成時に除去することができます。WebSphere Application Server 4.0.1 をインストールする前に、DB2 を正しいレベルまでアップグレードする必要もあります。

IBM HTTP Server のアンインストール:

注:

1. IBM HTTP Server をアンインストールする場合には、WebSphere Application Server もアンインストールしなければなりません。
2. IBM HTTP Server および WebSphere Application Server を再インストールする場合には、WebSphere Application Server を再インストールする前に IBM HTTP Server をインストールしなければなりません。なぜなら、WebSphere Application Server はご使用の Web サーバーの構成ファイルを変更するからです。

IBM HTTP Server をインストールするには、以下のようになります。

1. IBM HTTP Server を停止します。ユーザー ID root でログオンして、以下を行います。

- a. WebSphere Commerce Suite と WebSphere Application Server が停止していることを確認します。
- b. 端末ウィンドウから以下のコマンドを入力します。

```
cd /opt/IBMHTTPServer/bin
./apachectl stop
```

2. /opt/IBMHTTPServer ディレクトリーまたはそのサブディレクトリーでファイルを作成またはカスタマイズした場合、それらを保存しておきたいければ、WebSphere Commerce Suite コンポーネントによって使用されないディレクトリーにそれらをバックアップしてください。

3. /opt/IBMHTTPServer/conf のコピーを保管しておきたい場合は、異なる名前で作成してください。

4. ユーザー ID root でログオンして、次のように入力します。

```
rpm -e IBM_Apache_Source-1.3.12-2
rpm -e IBM_ADMIN_EN-1.3.12-2
rpm -e IBM_ADMIN_Server-1.3.12-2
rpm -e IBM_SSL_128-1.3.12-2
rpm -e IBM_SSL_Base-1.3.12-2
rpm -e IBM_HTTP_Server-1.3.12-2
rpm -e gsk4bas-4.0-3.159
```

5. 残っているすべての IBM HTTP Server ファイル、およびディレクトリーを次のようにして除去します。

```
cd /opt
rm -rf IBMHTTPServer
```

即時に WebSphere Application Server をアンインストールする必要があります。

IBM WebSphere Application Server のアンインストール: 以前のバージョンの WebSphere Application Server をアンインストールするには、以下のようになります。

1. ユーザー ID root でログインしているときに、以下のようにして WebSphere Application Server を停止します。

- a. 端末ウィンドウで以下のように入力して、WebSphere Application Server 管理クライアントを開始します。

```
cd /opt/WebSphere/AppServer/bin
./adminclient.sh
```


- b. WebSphere Application Server 管理コンソールで、使用しているホストの非修飾ホスト名が付いたノードを選択します。
- c. 「停止」 ボタンをクリックします。次のようなメッセージが戻されます。

You are trying to stop the node that the console is connected to. This will cause the console to exit after the node is stopped. Do you want to continue?
(コンソールの接続先のノードを停止させようとしています。このまま処理を続けると、ノードが停止した後コンソールは終了します。処理を続けてもよろしいですか?)

「はい」 をクリックして先に進みます。

- d. WebSphere Application Server 管理コンソールの処理が終わった後、端末ウィンドウから以下のコマンドを実行して、 WebSphere Application Server 関連のプロセスがすべて停止したことを確認してください。

```
ps -ef | grep IBMWebAS
ps -ef | grep startupServer
```

- e. このコマンドが何らかの Java プロセスを戻した場合は、検出されたプロセスごとに kill コマンドを発行して、それを停止させてください。
2. /opt/IBMWebAS ディレクトリーまたはそのサブディレクトリーでファイルを作成またはカスタマイズした場合、それらを保存しておきたければ、異なるディレクトリーにそれらをバックアップしてください。
 3. 以下のコマンドを実行して、RPM パッケージをアンインストールします。

```
rpm -e IBMWebAS-3.5ADV-1
rpm -e IBMJava2-SDK-1.2.2-9.0
```

4. /opt/IBMWebAS ディレクトリーがまだ存在する場合、次のように入力してそれを除去します。

```
cd /opt
rm -rf IBMWebAS
```

WebSphere Application Server リポジトリーの除去: DB2 を使用しており、以前のインストールで使用していた WebSphere Application Server 管理データベース (たとえば、WAS データベース) がなくなった場合は、以下のようにしてデータベースを除去できます。

1. データベースをバックアップします。データベース管理システムを使用することもできますし、単にデータベースを名前変更することもできます。データベースのバックアップに関する詳細情報は 33 ページの『データベースのバックアップ』を参照してください。
2. コマンド・ウィンドウで以下のように入力して、WebSphere Application Server データベースを除去します。

```
db2 drop db db_name
```

IBM WebSphere Application Server 3.5.x 管理データベースの除去は必須ではないことに注意してください。後で作成される WebSphere Application Server 4.0.x リポジトリー・データベースの名前は、 IBM WebSphere Application Server 3.5.x 管理データベースの名前とは異なるからです。

DB2 7.2 FixPak 6 へのアップグレード

このセクションでは、WebSphere Commerce 5.4 が必要とするレベルの DB2 にアップグレードする方法を説明しています。

DB2 7.1 FixPak 3 (バージョン 7.1.0.41) へのアップグレード: DB2 7.1 FixPak 2 から DB2 7.2 FixPak 6 にアップグレードするには、DB2 のレベルをバージョン 7.2 にアップグレードするために、まず DB2 7.1 FixPak 3 を適用する必要があります。

DB2 7.1 FixPak 3 は、ユーザーの便宜のために、WebSphere Commerce Disk 3 CD の /DB2_FP3 ディレクトリーに用意されています。

英語バージョンの FixPak 3 も、以下の DB2 サポート Web サイトからダウンロードして入手できます。

ftp://ftp.software.ibm.com/ps/products/db2/fixes/english-us/db2linuxv7/FP3_U475381/

各国語バージョンの FixPak 3 については、以下の Web サイトを参照してください。

<http://www14.software.ibm.com/webapp/download/postconfig.jsp?id=4000604&pf;=Platform+Independent&v;=Version+7&e;=&cat;=&s;=p>

前提条件: Linux 上に DB2 をインストールする前に、Korn シェルをインストールしておく必要があります。この FixPak をインストールする前に、DB2 UDB V7.1 をインストールしておく必要があります。必須の前提条件がインストールされているかどうかを判別する場合、以下のコマンドを発行します。

```
rpm -qa | grep db2
```

コマンドの出力には以下が含まれているはずですが、

```
db2cliv71-7.1.0-x
```

ここで、*x* はフィックス・レベルです。

インストール前: root としてログオンし、それぞれのインスタンスに対して以下のコマンドを発行します。

```
su - ioname
. $HOME/sql1lib/db2profile
db2 force applications all
db2 terminate
db2stop
db2licd end
exit
```

ここで *ioname* は、インスタンス所有者名です (たとえば db2inst1)。

次に、root としてログオンしている間に、以下のコマンドを入力します。

```
su - aname
. $HOME/sql1lib/db2profile
db2admin stop
exit
```

ここで *aname* は、DB2 管理サーバー名です。

DB2 7.2 FixPak 3 のインストール: WebSphere Commerce Disk 3 CD をマシンにマウントし、CD の DB2_FP3 ディレクトリーに移動します。

FixPak をインストールするには、以下のようにします。

```
su - root
cd patches_path
./installpatch
```

インストール後: 以下に示すように、FixPak のインストール後も実行する必要があるいくつかの項目があります。これには以下に説明するバインド・ファイルの再バインドなどがあります。追加の考慮事項については、DB2 7.2 FixPak 3 の README 資料を参照してください。

インスタンスの更新:

注: root としてログオンしてインスタンスを更新する必要があります。

それぞれのインスタンスに対して、以下のコマンドを発行します。

```
/usr/IBMDB2/V7.1/instance/db2iupdt iname
```

ここで *iname* は、インスタンス名です (たとえば db2inst1)。

DAS インスタンスの場合は、以下のコマンドを発行します。

```
/usr/IBMDB2/V7.1/instance/dasiupdt dasname
```

ここで *dasname* は、DAS インスタンス名です。db2iupdt または dasiupdt コマンドの詳細を参照するには、db2iupdt または dasiupdt に "-h" オプションを指定して実行します。

バインド・ファイルの再バインド: この FixPak に同梱されている新規のバインド (.bnd) ファイルが原因で、FixPak のインストール後に、ご使用のすべてのデータベースに対して DB2 ユーティリティーを再バインドする必要があります。このステップは、この FixPak の修正点を有効にするために必要です。以下に概説している再バインドの手順は、データベースごとに一度だけ実行する必要があります。

注: ランタイム・クライアントは、データベース・ユーティリティー (import、export、reorg、Command Line Processor) および DB2 CLI バインド・ファイルがデータベースで使用可能になるまでは、それらを各データベースにバインドするために使用できません。代わりに、DB2 アドミニストレーション・クライアントまたは DB2 Application Development Client を使用する必要があります。

非 DRDA DB2 データベースへの再バインド: フィックスを適用した後、DB2 ユーティリティーをコマンド・プロンプトから以下のように再バインドする必要があります。

```
db2 terminate
db2 CONNECT TO dbname
```

```
db2 BIND path/@db2ubind.lst GRANT PUBLIC
db2 BIND path/@db2cli.lst GRANT PUBLIC
db2 terminate
```

ここで *dbname* は、ユーティリティーのバインド先のデータベースの名前です。
path は、バインド・ファイルが置かれるディレクトリーの絶対パス名です。たとえば `/sql1lib/bnd` などとします。DB2UBIND.LST および DB2CLI.LST には、DB2 に使用する共通の必須バインド・ファイルのリストが含まれています。

特定のパッケージの再バインド: 特定の `.bnd` ファイルだけをデータベースにバインドしたい場合は、それぞれのデータベースに対して以下のコマンドをコマンド・プロンプトから発行します。

```
db2 terminate
db2 CONNECT TO dbname

db2 BIND path/bind_file BLOCKING ALL GRANT PUBLIC
db2 terminate
```

ここで、*bind_file* はバインド・ファイルの名前で、*dbname* はご使用のデータベースの名前です。*path* バインド・ファイルが置かれるディレクトリーの絶対パス名です。

DB2 7.2 FixPak 6 (DB2 7.1.0.60) へのアップグレード: DB2 7.1 FixPak 2 から DB2 7.2 FixPak 6 にアップグレードするには、DB2 7.2 FixPak 6 を適用する前に、前のセクションで説明しているとおりにまず DB2 7.1 FixPak 3 を適用する必要があります。

DB2 7.2 FixPak 6 は、ユーザーの便宜のために、WebSphere Commerce Disk 2 CD の `/Software_Patches/DB2_FP6` ディレクトリーに備えられています。

英語バージョンの FixPak 6 の、DB2 サポート Web サイトからダウンロードして入手できます。

ftp://ftp.software.ibm.com/ps/products/db2/fixes/english/db2linuxv7/FP6_U481413/

各国語バージョンの FixPak については、以下の Web サイトを参照してください。

<http://www14.software.ibm.com/webapp/download/postconfig.jsp?id=4000604&pf=Platform+Independent&v=Version+7&e=&cat=&s=p>

前提条件: この FixPak をインストールする前に、DB2 UDB V7.2 または DB2 UDB V7.1 (FixPak 3 適用済み) をインストールしておく必要があります。

Linux 上に DB2 をインストールする前に、Korn シェルをインストールしておく必要があります。この FixPak をインストールする前に、DB2 UDB V7.2 または DB2 UDB V7.1 (FixPak 3 適用済み) をインストールしておく必要があります。必須の前提条件がインストールされているかどうかを判別する場合、以下のコマンドを発行します。

```
rpm -qa | grep db2
```

コマンドの出力には以下が含まれているはずですが、

```
db2c1iv71-7.1.0-x
```

ここで、*x* はフィックス・レベルです。

インストール前: root としてログオンし、それぞれのインスタンスに対して以下のコマンドを発行します。

```
su - ioname
. $HOME/sql1lib/db2profile
db2 force applications all
db2 terminate
db2stop
db2licd end      # run at each physical node
exit
```

ここで *ioname* は、インスタンス所有者名です (たとえば db2inst1)。

それから、root としてログオンしている間に、以下のコマンドを入力します。

```
su - aname
. $HOME/sql1lib/db2profile
db2admin stop
exit
```

ここで *aname* は、DB2 管理サーバー名です。

DB2 7.2 FixPak 6 のインストール: WebSphere Commerce Disk 2 CD をマシンにマウントして、/Software_Patches/DB2_FP6 ディレクトリーに移動します。

FixPak をインストールするには、以下のようにします。

```
su - root
cd patchespath
./installFixPak
```

インストール後: 以下に示すように、FixPak のインストール後も実行する必要があるいくつかの項目があります。これには以下に説明するバインド・ファイルの再バインドなどがあります。追加の考慮事項については、DB2 7.2 FixPak 6 の README 資料を参照してください。

インスタンスの更新:

注: root としてログオンしてインスタンスを更新する必要があります。

それぞれのインスタンスに対して、以下のコマンドを発行します。

```
/usr/IBMDB2/V7.1/instance/db2iupdt iname
```

ここで *iname* は、インスタンス名です (たとえば db2inst1)。

注: 同じマシン上で Enterprise Extended Edition (EEE) と Enterprise Edition (EE) の両方を実行している場合、db2iupdt に "-k" オプションを指定して実行する必要があります。こうしないと、それは EEE ファイル・セットが存在するかどうかをチェックします。存在する場合は、それが EEE インスタンスであるかのように更新が実行されます。

DAS インスタンスの場合は、以下のコマンドを発行します。

```
/usr/IBMDB2/V7.1/instance/dasiupdt dasname
```

ここで *dasname* は、DAS インスタンス名です。db2iupdt または dasiupdt コマンドの詳細を参照するには、db2iupdt または dasiupdt に "-h" オプションを指定して実行します。

バインド・ファイルの再バインド: この FixPak に同梱されている新規のバインド (.bnd) ファイルが原因で、FixPak のインストール後に、ご使用のすべてのデータベースに対して DB2 ユーティリティーを再バインドする必要があります。このステップは、この FixPak の修正点を有効にするために必要です。以下に概説している再バインドの手順は、データベースごとに一度だけ実行する必要があります。

注: ランタイム・クライアントは、データベース・ユーティリティー (import、export、reorg、Command Line Processor) および DB2 CLI バインド・ファイルがデータベースで使用可能になるまでは、それらを各データベースにバインドするために使用できません。代わりに、DB2 アドミニストレーション・クライアントまたは DB2 Application Development Client を使用する必要があります。

非 DRDA DB2 データベースへの再バインド: フィックスを適用した後、DB2 ユーティリティーをコマンド・プロンプトから以下のように再バインドする必要があります。

```
db2 terminate
db2 CONNECT TO dbname
```

```
db2 BIND path@db2ubind.lst GRANT PUBLIC
db2 BIND path@db2cli.lst GRANT PUBLIC
db2 terminate
```

ここで *dbname* は、ユーティリティーのバインド先のデータベースの名前です。*path* は、バインド・ファイルが置かれるディレクトリーの絶対パス名です。たとえば /sql1lib/bnd などとします。DB2UBIND.LST および DB2CLI.LST には、DB2 に使用する共通の必須バインド・ファイルのリストが含まれています。

特定のパッケージの再バインド: 特定の .bnd ファイルだけをデータベースにバインドしたい場合は、それぞれのデータベースに対して以下のコマンドをコマンド・プロンプトから発行します。

```
db2 terminate
db2 CONNECT TO dbname
```

```
db2 BIND path/bind_file BLOCKING ALL GRANT PUBLIC
db2 terminate
```

ここで、*bind_file* はバインド・ファイルの名前で、*dbname* はご使用のデータベースの名前です。*path* バインド・ファイルが置かれるディレクトリーの絶対パス名です。

DB2 Application Development Client のインストール: 以前の DB2 のインストールに DB2 Application Development Client コンポーネントが組み込まれていなかった場合、WebSphere Application Server 4.0.1 にアップグレードする前に、このコンポーネントを DB2 7.2 FixPak 6 レベルでインストールすることが必要になります。

このコンポーネントは、WebSphere Commerce 5.4 で提供されている完全インストール可能 DB2 7.2 FixPak 6 CD を使用してインストールします。このコンポーネントのインストールについては、*WebSphere Commerce* インストール・ガイド バージョン 5.4 の DB2 のインストールについての章を参照してください。

jdbc2 を使うようにデータベース・マネージャーを構成する: DB2 でデータベースを作成する場合は、事前に DB2 サーバー・マシンで以下のことを実行しておかなければなりません。

1. db2inst1 ユーザーに変更します。

```
su - db2inst1
```

2. 以下のファイルの最後に、次の行を追加します。

- Red Hat Linux の場合: db2inst1 /home/db2inst1/.bashrc
- SuSE Linux Enterprise Server の場合: db2inst1 /home/db2inst1/.profile

```
./sqllib/java12/usejdbc2
EXTSHM=ON
export EXTSHM
db2set DB2ENVLIST=EXTSHM
```

3. ファイルを保管します。
4. db2stop と入力します。
5. db2start と入力します。
6. exit と入力します。

WebSphere Application Server 用データベースの作成: was40 という名のデータベースを作成し、DB2 アプリケーション・ヒープ・サイズを設定するために、以下のステップを実行してください。

1. 必ず、DB2 インスタンス所有者 db2inst1 でログインします。
2. DB2 が稼働中であることを確認してください。
3. 以下のコマンドを使用して、was40 という名のデータベースを作成します。

```
db2 create database was40
```

このプロセスを完了するには数分かかることがあります。

4. 以下のコマンドを使用して、アプリケーション・ヒープ・サイズを設定します。
5. データベースの作成が完了した場合、変更を有効するために DB2 を停止してから始動します。

```
db2 update db config for WAS40 using applheapsz 256
```

```
db2stop
db2start
```

ご使用のシステムにおいて、アプリケーション・ヒープ・サイズが 256 でうまく作動しない場合には、512 に増やしてください。

データベースへの接続の検証: 以下のステップを実行し、was40 という名のデータベースへの接続を検証してください。

1. 必ず、DB2 インスタンス所有者 db2inst1 でログインします。
2. 以下のコマンドを使用して、was40 という名のデータベースに接続します。

```
db2 connect to was40
```

正しい出力は、次のようなものになります。

```
Database Connection Information
```

```
Database server = DB2/Linux 7.2.4
SQL authorization ID = DB2INST1
Local database alias = WAS40
```

3. DB2 インスタンス所有者としてデータベースの接続を切断してログアウトするには、コマンド・プロンプトで `exit` と入力します。

WebSphere Application Server 4.0.1 のインストールとインストール後の段階

以下のステップを実行して WebSphere Application Server をインストールする前に、WebSphere Application Server をサポートするハードウェア、ソフトウェア、および API について、次の Web サイトを参照してください。

<http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html>

ここで、ご使用のオペレーティング・システムで WebSphere Application Server を使用するために、どの製品、およびどの修正レベルがサポートされているかを確認することができます。

続くステップでは、以下のものを使用する WebSphere Application Server Advanced Edition の構成をインストールする方法を説明しています。

- Red Hat Linux 7.2
- SLES 7.0
- IBM Developer Kit for Linux、Java 2 Technology Edition 1.3.1
- IBM HTTP Server 1.3.19
- DB2 Universal Database (UDB) 7.2 FixPak 6

以下の前提条件を満たしていることを確認してください。

1. 必ず、スーパーユーザー (root) 特権でマシンにログインしてください。
2. システム上にある既存の Web サーバーが稼働中の場合は、Web サーバーを停止します。IBM HTTP Server 1.3.19 を WebSphere Application Server のインストールの一部としてインストールする計画で、IBM HTTP Server 1.3.19 以前のバージョンがすでにインストールされている場合、WebSphere Application Server のインストール・プログラムによる IBM HTTP Server 1.3.19 のインストールが正常に行われるようにするために、IBM HTTP Server 1.3.19 以前のバージョンはアンインストールしなければなりません。
3. 別の Web サーバーがシステム上で稼働している場合は、その Web サーバーを停止します。
4. CD-ROM ドライブに WebSphere Application Server CD-ROM を挿入します。
5. 必要なら、`mkdir` コマンドを使用して CD-ROM のマウント・ポイントを作成してください。次のコマンドはディレクトリ `/cdrom` にマウント・ポイントを作成します。マシンのローカル・ファイル・システムのどのロケーションでも CD-ROM をマウントすることができます。

```
mkdir /cdrom
```

以上のステップのコマンドは、CD-ROM ドライブが `/cdrom` にマウントされていることが前提になっています。CD-ROM を異なるロケーションにマウントした場合には、コマンドを実行する時にそのロケーションを使用してください。

6. 以下のコマンドを入力して、CD-ROM ドライブをマウントします。

```
mount -t iso9660 -r /dev/cdrom /cdrom
```


注: ウィンドウ・マネージャーによっては、CD-ROM を自動的にマウントするものもあります。詳細については、オペレーティング・システムの資料を参照してください。

7. サポートされている Web サーバー、あるいは WebSphere とともに使用する計画のデータベースが、現在 WebSphere Application Server で必要とされるバージョンより新しい場合、prereq.properties ファイルを更新するか、前提条件チェック機能を WebSphere Application Server のインストールの前に使用不可にします。更新済みの prereq.properties ファイルを入手するには、以下の WebSphere Application Server Tools の Web サイトから最新バージョンをダウンロードします。

<http://www.ibm.com/software/webservers/appserv/tools.html>

更新済みの prereq.properties ファイルが、確実にローカルの /tmp ディレクトリにダウンロードまたはコピーされるようにしてください。

前提条件チェック機能を使用不可にするには、以下のステップを実行します。

- a. prereq.properties ファイルを、/cdrom ディレクトリから、WebSphere Application Server をインストールする予定のマシンの /tmp ディレクトリにコピーする。
- b. テキスト・エディターで prereq.properties ファイルをオープンし、個々のコンポーネントの前提条件チェックを、その特定のキーの値を 1 から 0 に変更することにより使用不可にします。
- c. 編集した prereq.properties ファイルを保管します。

WebSphere Application Server 4.0.1 のすべての製品前提条件を満たしたら、次の段階を開始することができます。

1. WebSphere Application Server 4.0.1 インストール・プログラムを再始動する際には、以下のような状態が検出される場合があります。
 - 適切にアップグレードされなかった前提条件が残っている。インストール・プログラムを終了し、それらの前提条件が確実に満たされるようにしなければなりません。
 - 新規のインストール・ディレクトリが現行のものと同じである場合に、アプリケーション・サーバーが稼働中。インストール・プログラムが適切にファイルをオーバーレイするように、WebSphere Application Server を停止する必要があります。
2. この後の 52 ページの『WebSphere Application Server 4.0.1 のインストール』で説明されているように、インストール・プログラムは標準インストール・プロセスをガイドしていきます。

インストールの完了後には以下が実行されます。

- インストール後のマイグレーションが実行されます。インストール・プログラムは最初の段階でエクスポートされた構成をインポートし、マイグレーション・ログ・ファイルを表示します。
- 最初に管理サーバーを始動する時には、マイグレーション・プロセスの間に作成された構成ファイルをロードするため、始動に余分の時間がかかります。

WebSphere Application Server 4.0.1 のインストール

ここでの説明は、以下のことを前提としています。

- マシンに、インストールのための十分なメモリーとディスク・スペースがあること。要件については、以下の "WebSphere Application Server Supported Hardware, Software, and APIs" の Web サイトを参照してください。

<http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html>

- IBM HTTP Server の使用を計画している場合は、 WebSphere Application Server のインストール・プロセス時にそのインストールを選択すること。サポートされる別の Web サーバーを WebSphere Application Server とともに使用する計画の場合は、そのサーバーがすでに WebSphere Application Server と同じノードにインストールされていること。
- WebSphere Application Server で使用するための、サポートされるデータベースがインストールおよび構成されていること。

注: WebSphere Application Server をインストールする前に、 IBM HTTP Server 以外のいずれかの Web サーバーを購入し、インストールしなければなりません。IBM HTTP Server は WebSphere Application Server に同梱されており、WebSphere Application Server のインストール時に自動的にインストールすることができます。

WebSphere Application Server をインストールするには、以下のステップを実行します。

1. /cdrom ディレクトリーにナビゲートします。
2. 更新済みの `prereq.properties` ファイルをダウンロードしていない場合、あるいは前提条件チェック機能を使用不可にしていない場合、`install.sh` コマンドを以下のように使用して WebSphere Application Server インストール・プログラムを開始します。

```
./install.sh
```

ステップ 7 (51 ページ) に従って、更新済みの `prereq.properties` ファイルをダウンロードした場合、あるいは前提条件チェック機能を使用不可にした場合は、`install.sh` コマンドを以下のように使用して、 WebSphere Application Server インストール・プログラムを開始します。

```
./install.sh -prereqfile /tmp/prereq.properties
```

3. 「Welcome to the IBM WebSphere Application Server Setup program (IBM WebSphere Application Server のセットアップ・プログラムへようこそ)」ダイアログ・ボックスがオープンします。「次へ」をクリックして続きます。
4. 「Install Options (インストール・オプション)」ダイアログ・ボックスがオープンします。「**Custom Installation (カスタム・インストール)**」を選択してから、「次へ」をクリックします。
5. 「Choose Application Server Components (Application Server コンポーネントの選択)」ダイアログ・ボックスがオープンします。インストールするコンポーネントを選択し、インストールしないコンポーネントを選択解除します。以下の点に注意してください。
 - Java 2 Software Developer's Kit (SDK) はデフォルトでインストールされます。

- コンポーネントの Server、Admin、Samples、Application Assembly and Deployment Tools、 IBM HTTP Server 1.3.19、 および Web Server Plugins は、デフォルトでインストールが選択されています。
- WebSphere Application Server とともに IBM HTTP Server を使用する計画の場合、 IBM HTTP Server 1.3.19 と Web Server Plugins のオプションを必ず選択してください。
- WebSphere Application Server とともに別のサポート Web サーバーを使用する計画の場合、 Web Server Plugins オプションを必ず選択してください。

注: Application Server または管理コンソールを立ち上げる際にプラグインは必要ありませんが、実動アプリケーションの場合、サポートされる Web サーバーとそれに対応する Web サーバー・プラグインがインストールされていないと、サーブレットを機能させることができません。

非実動アプリケーションの場合、内部 HTTP トランスポート・ポート 9080 を使用して Web サーバー・プラグインをインストールしなくても、内部 HTTP トランスポート・システムを使用してサーブレットを機能させることができます。たとえば、内部 HTTP トランスポートを使用してサンプル・スヌープ・サーブレットを機能させるには、次の URL を入力します。

`http://local_host:9080/servlet/snoop`

内部 HTTP トランスポートのメカニズムは、実稼働環境で使用できるように設計されていません。

- IBM HTTP Server の Web サーバー・プラグインを、それがマシン上にインストールされていない場合は、 IBM HTTP Server 1.3.19.1 オプションを選択してインストールします。
- ここでの説明は、すべてのコンポーネントをインストールすることを想定しています。

「次へ」をクリックして続きます。

6. Web Server Plugins オプションを選択した場合、「Choose Application Server Components (Application Server コンポーネントの選択)」ダイアログ・ボックスがオープンします。 Web サーバーに適したプラグインを選択し、「次へ」をクリックします。
7. 「Database Options (データベース・オプション)」ダイアログがオープンします。
 - 「Database Options (データベース・オプション)」ダイアログで以下のステップを実行してください。
 - a. 「Database Type (データベース・タイプ)」フィールドで、プルダウン・メニューから **DB2** を選択します。
 - b. 「**Remote DB (リモート DB)**」が選択されていないことを確認します。この例では、データベースと WebSphere Application Server が同一ノードにインストールされています。
 - c. 「Database Name (Database SID) (データベース名 (データベース SID))」フィールドに、データベースの名前 **WAS40** を入力します。

- d. 「DB Home (DB ホーム)」フィールドで、DB2 インスタンス所有者のホーム・ディレクトリーの絶対パス名 /home/db2inst1 を入力するか、または「参照」ボタンを使ってホーム・ディレクトリーの絶対パス名を指定します。
 - e. 「DB URL」フィールドは編集できません。
 - f. 「Server Name (サーバー名)」フィールドは編集できません。
 - g. 「Port Number (ポート番号)」フィールドは編集できません。
 - h. 「Database User ID (データベース・ユーザー ID)」フィールドで、データベース・インスタンス所有者の名前 db2inst1 を入力します。
 - i. 「Database Password (データベース・パスワード)」フィールドに、データベース・インスタンス所有者の現在のパスワードを入力します。
 - j. 「次へ」をクリックして続きます。
8. 「Select Destination Directory (宛先ディレクトリーの選択)」ダイアログがオープンします。WebSphere Application Server のインストール先ディレクトリーを指定します。デフォルトの宛先ディレクトリーを受け入れることができます。または、絶対パス名を入力するか「参照」をクリックして別のディレクトリーを指定することもできます。注意すべきこととして、インストールの対象として IBM HTTP Server を選択した場合、宛先ディレクトリーを変更することはできません。「次へ」をクリックして続きます。
 9. 「Install Options Selected (選択済みのインストール・オプション)」ダイアログ・ボックスがオープンします。表示内容が正しいことを確認してから「**Install (インストール)**」をクリックし、インストールを完了させます。
 10. マシンの構成に応じて、「Location of Configuration files (構成ファイルのロケーション)」ダイアログ・ボックスがオープンすることがあります。ここで、指定された Web サーバー構成ファイルの保管先ディレクトリーの絶対パス名を入力するように求められます。フィールドに入力するか、「参照」をクリックして、ファイルの絶対パス名を指定します。
 11. 「Setup Complete (セットアップの完了)」ダイアログ・ボックスがオープンします。README ファイルを見るには、「**Yes, I want to view the ReadMe File (はい。README ファイルを読みます)**」が選択されていることを確認してから、「**Finish (終了)**」をクリックします。これで、デフォルトのブラウザ・ウィンドウに README ファイルが表示されます。後で ReadMe ファイルを見るには、「**Yes, I want to view the ReadMe File (はい。README ファイルを読みます)**」を選択解除してから、「**Finish (終了)**」をクリックして、インストール・プログラムを終了します。
 12. 「WebSphere Application Server - First Steps (WebSphere Application Server - 最初のステップ)」ダイアログ・ボックスがオープンします。この GUI を使って、インフォセンター内の製品情報へのアクセス、管理サーバーの始動、管理コンソールの立ち上げ、またはアプリケーション・アセンブリー・ツールの立ち上げを行うことができます。WebSphere を使用するには、まず Web サーバーを開始してから構成する必要があるため、ひとまずこのダイアログをクローズします。後で「First Steps (最初のステップ)」GUI を立ち上げるには、opt/WebSphere/AppServer/bin ディレクトリーに置かれている firststeps.sh スクリプトを実行します。

13. CD-ROM を CD-ROM ドライブから除去する前に、以下のように `umount` コマンドを使用して CD-ROM をアンマウントします。

```
umount /cdrom
```

14. IBM HTTP Server 以外の Web サーバーを使用している場合、サーバーを始動します。IBM HTTP Server を使用しており、これを WebSphere Application Server のインストール時にインストールした場合、これが正常に実行されるように Web サーバーを構成することが必要になる場合があります。

以下のステップを実行し、IBM HTTP Server が正常にインストールされ、構成されているか確認します。

- a. Web サーバーが稼働中であることを確認してください。稼働していない場合、以下のコマンドを入力して Web サーバーを始動します。

```
/opt/IBMHTTPServer/bin/apachectl start
```

- b. ブラウザーを開始してから、ローカル・マシンの名前を URL で入力します。IBM HTTP Server の Web ページが表示されたら、サーバーのインストールと構成は正しく完了しているということです。

IBM HTTP Server の構成についての詳細は、以下の Web サイトにある IBM HTTP Server の資料を参照してください。

<http://www.ibm.com/software/webservers/httpservers/library.html>

IBM HTTP Server で Secure Sockets Layer (SSL) を使用可能にするには、以下の Web サイトにある IBM HTTP Server の資料を参照してください。

<http://www.ibm.com/software/webservers/httpservers/doc/v1319/index.html>

15. 『インストール内容のテスト』に進みます。

インストール内容のテスト: このセクションでは、WebSphere Application Server システムのインストールと構成をテストする方法について説明します。以下の解説では、サポートされている Web サーバー、データベース、および WebSphere Application Server コンポーネントがインストールされていることを前提としています。

以下のステップを実行し、WebSphere のインストール内容をテストします。

1. 必ず、スーパーユーザー (root) 特権でマシンにログインしてください。
2. 次のように `startupServer` スクリプトを実行して WebSphere 管理サーバーを開始します。

```
cd /opt/WebSphere/AppServer/bin
./startupServer.sh
```

`/opt/WebSphere/AppServer/logs` ディレクトリー内に置かれている `tracefile` という名前のファイルを調べて、管理サーバーの始動が正常に完了したことを確認します。

サーバーの始動が正常に完了していれば、メッセージ「Server open for e-business (e-business 用の Server がオープンします)」がそのファイルに示されます。

3. 次のように `adminclient` スクリプトを実行して、管理コンソールを始動します。

```
cd /opt/WebSphere/AppServer/bin
./adminclient.sh
```

4. メッセージ **Console Ready** (コンソールは作動可能) が表示されたら、次のようなステップを行ってアプリケーション・サーバーを管理します。
 - a. 管理コンソールがオープンすると、ツリー・ビューが表示されます。
「**WebSphere Administrative Domain (WebSphere 管理可能ドメイン)**」エントリーの横の正符号 (+) をクリックし、ビューを展開します。
 - b. 「**Nodes (ノード)**」エントリーの横の正符号 (+) をクリックし、ビューを展開します。
 - c. ご使用のホスト・マシンの名前を確認してから、エントリーのビューを展開します。
 - d. 「**Application Servers (アプリケーション・サーバー)**」エントリーの横の正符号 (+) をクリックし、ビューを展開します。
 - e. 「**Default Server (デフォルト・サーバー)**」エントリーを選択してから、ツールバー上にある「**Start (始動)**」をクリックします。通知ウィンドウがオープンし、サーバーは始動済みであることが知らされます。「**OK**」をクリックして通知ウィンドウをクローズします。
いったんデフォルト・サーバーが始動されると、これが停止するかまたはマシンを再始動する場合に自動的に始動されるようになります。管理サーバーで障害が起きて、デフォルト・サーバーは稼働し続けます。
5. Web サーバーが稼働中であることを確認してください。 Web サーバーが稼働していなければ、始動してください。
6. ブラウザーを始動してから、デフォルトでインストールされているサンプル・サブレットである次のようなスヌープ・サブレットの URL を入力します。

```
http://machine_name/servlet/snoop
```

ここで、*machine_name* は、WebSphere Application Server が稼働しているマシンの名前を表しています。 /servlet/snoop に関する解説が表示されます。

IBM HTTP Server が正常に動作しない場合は、*WebSphere Commerce* インストール・ガイド バージョン 5.4 の『IBM HTTP Server での実行のための SSL の使用可能化』の章を参照し、SSL を手動で使用可能にしてください。SSL が適切に使用可能になっていない場合は、ストアを正常に立ち上げることができません。

Enterprise Bean を使ったテスト: このセクションでは、Enterprise Bean と Increment サンプルを使った WebSphere 構成のテスト方法について説明します。以下の解説では、前のセクションに従って WebSphere Application Server がインストールされており、インストール内容をテスト済みであることを前提としています。

以下のステップを実行し、Enterprise Bean を使って WebSphere 構成をテストします。

1. 必ず、スーパーユーザー (root) 特権でマシンにログインしてください。
2. 管理コンソールが稼働中であることを確認してください。
3. デフォルト・サーバー (「**WebSphere 管理可能ドメイン**」 > 「**ノード**」 > **node_name** > 「**アプリケーション・サーバー**」にある) が稼働中であることを確認してください。

4. Web サーバーが稼働中であることを確認してください。 Web サーバーが稼働していなければ、始動してください。
5. Web ブラウザーを開始して、以下の URL を指定します。

```
http://machine_name/webapp/examples/HitCount
```

ここで、*machine_name* は、WebSphere が稼働しているマシンの名前を表しています。 Web ページがオープンすると、いくつかの選択オプションが表示されます。

6. 「**Generate hit count using (ヒット数の生成)**」ヘッディングの下で、「**Enterprise Java Bean**」のラジオ・ボタンをクリックします。
7. 「**Transaction Type (トランザクション・タイプ)**」ヘッディングの下で、「**None (なし)**」オプションのラジオ・ボタンをクリックします。
8. 「増分」をクリックします。

ヒット数が表示されれば、WebSphere は適切に機能していることになります。

SSL の使用可能化: テスト目的で IBM HTTP Server の Secure Sockets Layer (SSL) を使用可能にするには、テスト目的でご使用の IBM HTTP Server 構成ファイルに応じて `httpd.conf.sample` ファイルを使用します。

1. `/opt/IBMHTTPServer/conf` ディレクトリーに移動します。
2. 既存の `httpd.conf` ファイルを `httpd.conf.bak` に名前変更し、バックアップします。
3. `httpd.conf.sample` を `httpd.conf` に名前変更します。
4. 以下の行が `httpd.conf` ファイルから欠落している場合には、ファイルの末尾にそれらを追加するか、`httpd.conf.bak` ファイルから上書きコピーしてください。

```
LoadModule ibm_app_server_http_module /opt/WebSphere/AppServer/bin/mod_ibm_app_server_http.so
WebSpherePluginConfig /opt/WebSphere/AppServer/config/plugin-cfg.xml
AddModule mod_app_server_http.c
```

実動目的で SSL を使用可能にするには、*WebSphere Commerce* インストール・ガイド バージョン 5.4 の『IBM HTTP Server での実行のための SSL の使用可能化』の章を参照してください。

WebSphere Application Server FixPak および eFix のインストール

FixPak または eFix をインストールする前に、以下を行う必要があります。

- WebSphere Application Server が停止していることを確認します。 WebSphere Application Server を停止するには、以下のようにします。
 1. 端末ウィンドウで以下のように入力して、WebSphere Application Server コンソールを開始します。

```
export DISPLAY=fully_qualified_host_name:0.0
cd /opt/WebSphere/AppServer/bin
./adminclient.sh
```
 2. WebSphere Application Server コンソールで、ショート・ホスト名が付いたノードを選択します。
 3. 「停止」ボタンをクリックします。次のようなメッセージが戻されます。

You are trying to stop the node that the console is connected to. This will cause the console to exit after the node is stopped. Do you want to continue?
(コンソールの接続先のノードを停止させようとしています。このまま処理を続けると、ノードが停止した後コンソールは終了します。処理を続けてもよろしいですか?)

「はい」をクリックして先に進みます。

4. WebSphere Application Server 管理コンソールの処理が終わった後、端末ウィンドウから以下のコマンドを実行して、WebSphere Application Server 関連のプロセスがすべて停止したことを確認します。

```
ps -ef | grep AppServer  
ps -ef | grep startupServer
```

5. このコマンドが何らかの Java プロセスを戻した場合は、kill コマンドを出して、それを停止させます。

- IBM HTTP Server が停止していることを確認します。

```
cd /opt/IBMHTTPServer/bin  
./apachectl stop
```

- 必要であれば、WebSphere Commerce Disk 2 CD をマウントします。その場合、次のように入力します。

```
mount CDROM_dir
```

CDROM_dir は CD のマウント先に指定しようとしているディレクトリーです。

WebSphere Application Server FixPak をインストールするには、以下のステップを完了します。

1. CD の Software_Patches/WAS_PTF2 ディレクトリーに移動し、以下のコマンドを実行します。

```
cp was40_ae_ptf_2_linux.intel.tar /tmp  
cd /tmp  
tar -xvf was40_ae_ptf_2_linux.intel.tar  
./install.sh
```

2. WebSphere Application Server ルート・ディレクトリーを入力するプロンプトが出されたら、/opt/WebSphere/AppServer と入力します。すべての質問に **yes** と答えます。

WebSphere Application Server eFix をインストールするには、以下のステップを完了します。

1. WebSphere Commerce Disk 2 CD の /Software_Patches/WAS_efixes ディレクトリーに移動します。
2. 以下のコマンドを使用して、それぞれの eFix をインストールします。

```
cp eFix_jar_file_name.jar /tmp  
cd /tmp  
java -jar eFix_jar_file_name.jar -target /opt/WebSphere/AppServer
```

eFix_jar_file_name.jar はそれぞれの .jar ファイルの名前です。

スクリプトがエラーなしでソフトウェアをアップグレードしたことを検証するには、以下のエラー・ログを調べます。

- /tmp/WC54efixunix.log

- /opt/WebSphere/AppServer/eFix/PQ54291/Extractor.log
- /opt/WebSphere/AppServer/eFix/PQ57814/Extractor.Log
- /opt/WebSphere/AppServer/eFix/PQ58443/Extractor.Log

WebSphere Commerce 5.4 へのアップグレード

このセクションでは、WebSphere Commerce 5.4 インストール・プログラムを使用して WebSphere Commerce 5.4 にアップグレードする方法について説明します。

WebSphere Commerce Suite の WebSphere Commerce 5.4 へのアップグレード

このセクションでは、WebSphere Commerce バージョン 5.4 のインストール方法について説明します。このセクションのステップを完了するには、WebSphere Commerce Disk 1 CD が必要です。

WebSphere Commerce バージョン 5.4 をインストールする前に、Web サーバー、データベース、IBM SDK for Java、および WebSphere Application Server がインストールされていることを確認してください。

WebSphere Commerce をインストールする前に、36 ページの『ハードウェアのアップグレード』および 36 ページの『オペレーティング・システムのアップグレード』のセクションで説明されている前提条件をすべて満たしていることを確認してください。

インストール手順

WebSphere Commerce バージョン 5.4 をインストールするには、以下のようにします。

1. ユーザー ID root でログオンします。
2. 必要であれば、WebSphere Commerce Disk 1 CD をマウントします。その場合、次のように入力します。

```
mount CDROM_dir
```

CDROM_dir は CD のマウント先に指定しようとしているディレクトリーです。

3. DB2 サーバーが開始済みであることを確認します。DB2 の開始については、155 ページの『DB2 ユニバーサル・データベースの開始と停止』を参照してください。
4. WebSphere Application Server が開始されていなければなりません。153 ページの『WebSphere Application Server の開始と停止』を参照してください。
5. コマンド行で、次のように入力して CD に切り替えます。

```
cd /CDROM_dir
```

CDROM_dir は、CD をマウントしているディレクトリーです。

6. コマンド行で、次のように入力します。

```
install.sh
```

7. インストール・ウィンドウがオープンした後、「次へ」をクリックしてインストールを続行します。
8. プログラムのご使用条件が表示されたら、それを受諾して、「次へ」をクリックします。
9. WebSphere Commerce のインストール場所およびサイズが表示されます。「次へ」をクリックして、インストールを開始します。
10. インストールが完了したら、「終了」をクリックします。
11. 「終了」をクリックすると、インストーラーはポスト・インストール・スクリプトの `wcb.sh` を呼び出します。画面のプロンプトに従います。非 root ユーザーとして実行するかどうかを尋ねられたら、`yes` と入力するようお勧めします。

注:

- a. root ユーザーで WebSphere Application Server を実行することを選ぶ場合、root ユーザーは WebSphere Application Server と WebSphere Commerce の両方のインストール・ツリーを所有することになります。非 root ユーザーで WebSphere Application Server を実行することを選ぶと、非 root ユーザー (デフォルトでは `wasuser`) は、WebSphere Application Server、WebSphere Payment Manager、および WebSphere Commerce のインストール・ツリーを所有することになります。

root ユーザーで WebSphere Application Server を実行することを選ぶ場合、マイグレーション・プロセスのすべてのステップは root ユーザーとして実行する必要があります。非 root ユーザー (`wasuser`) で WebSphere Application Server を実行することを選ぶ場合、マイグレーション・プロセスのすべてのステップは非 root ユーザーとして実行する必要があります。
 - b. `wcb.sh` スクリプトで使用するデフォルトのグループ名とユーザー名を使用することもお勧めします。別の名前を選択する場合には、この資料全体にわたって、それらの名前置き換える必要があります。
 - c. グループ名とユーザー名には、大文字は使用できません。
12. 以下を入力して、CD をアンマウントします。

```
cd /  
umount CDROM_dir
```

CDROM_dir は、CD をマウントしているディレクトリーです。

13. WebSphere Commerce Disk 1 CD を取り出します。

Payment Manager 3.1.2 へのアップグレード

Payment Manager 3.1.2 にアップグレードする前に、以下で入手できる最新の Payment Manager 製品情報を参照してください。

<http://www.ibm.com/software/webservers/commerce/payment>

Payment Manager 3.1.2 にマイグレーションする前に、まだ実行していなければ、既存の Payment Manager データベースをバックアップしてください。データベースをバックアップする方法については、33 ページの『データベースのバックアップ』を参照してください。データベースをバックアップした後、以下のようになります。

- 最新の README ファイル `readme.framework.html` をお読みください。これには、以下の Payment Manager の Web サイト上の資料ライブラリー・リンクからアクセスします。

<http://www.ibm.com/software/webservers/commerce/payment>

- この Payment Manager の新バージョンをインストールする前に、マイグレーションしないカセットをすべて除去します。

Payment Manager 3.1.2 のインストールおよびマイグレーション・プログラムでは、Payment Manager バージョン 2.1.4.0 以降からのデータがマイグレーションされます。既存のデータベースが引き続き使用され、インストール中に必要なすべてのデータ・マイグレーションが行われます。

マイグレーションを行えるのは、Payment Manager バージョン 2.1.4.0 以降の作業バージョンからだけです。つまり、Payment Manager バージョン 2.1.3 またはそれより前の修正レベルからのマイグレーションはサポートされていません。Payment Manager バージョン 2.1.3 またはそれより前のものを使用している場合、バージョン 2.1.4.0 にアップグレードしてからでなければ、Payment Manager バージョン 3.1 にアップグレードすることはできません。

決済カセット

Payment Manager 3.1.2 がインストールされた後、Payment Manager バージョン 2.1 以降ですでにインストールされている (IBM またはサード・パーティーによって提供される) すべてのカセットが適切に機能するという保証はありません。

Payment Manager をインストールする前に、以下の点に注意してください。

- 現在サード・パーティーの決済カセットを使用している場合、ご使用のシステム上に Payment Manager をインストールする前に、既存のカセットを Payment Manager 3.1.2 にマイグレーション済みできるか、カセット・プロバイダーにまず確認する必要があります。
- サード・パーティーと IBM 提供のどちらの決済カセットも、カセットが正常に機能するように Payment Manager 3.1.2 へマイグレーションしなければなりません。IBM 提供のカセット (Cassette for VisaNet や Cassette for SET™ など) を使用している場合、マイグレーションを行うためにバージョン 3.1.2 のカセット・ソフトウェアをインストールする必要があります。インストールの説明については、ご使用の決済カセット固有の補足資料を参照してください。
- 現在 Payment Manager バージョン 2.1.4 を Cassette for SET バージョン 2.1.4 と一緒に使用している場合、まず、バージョン 2.1.5 PTF (Payment Manager フレームワークと Cassette for SET の両方) をインストールしなければなりません。そして、少なくとも一度 Payment Engine を始動して停止してから、マイグレーションを実行します。そうしなければ、データ・マイグレーション中に Payment Manager 3.1.2 のインストールが失敗します。

データベース

以前のインストールからマイグレーションする場合、ご使用のデータベース製品、オペレーティング・システム、およびキャンペーン製品のレベルが、Payment Manager によってサポートされていることを確認してください。プラットフォーム

固有の情報については、*IBM WebSphere Payment Manager 3.1 for Multiplatforms インストール・ガイド* 内の、ご使用のプラットフォームに対応したインストールに関する章を参照してください。

DB2 7.2 FixPak 6 より前のバージョンの UDB を使用している場合、このバージョンの UDB までアップグレードする必要があります。CD-ROM からの UDB ソフトウェアのアップグレードについては、*IBM WebSphere Payment Manager 3.1 for Multiplatforms インストール・ガイド* の『ワークステーション・プラットフォームでのデータベース (UDB) のインストール』を参照してください。Payment Manager インストールで UDB 以外のデータベースを使用していて、それを UDB に変更する場合、UDB を手動でインストールして、Payment Manager のデータを UDB にマイグレーションすることもできます。

インストール前の考慮事項

マイグレーションの前に、以下に挙げる追加の考慮事項に注意してください。

レジストリー・キー内のパス

Payment Manager の前提条件 (Web サーバー、WebSphere Application Server、データベース) のいずれかをアップグレードまたは再インストールした場合、Payment Manager または Payment Server レジストリー・キーに保管されたパスが、現行のインストールに対して正しいか確認してください。これらのキーは、`/root/bin/jitk.db` というファイルにあります。

以下のキーにあるパスが、システム上で Payment Manager の前提条件がインストールされている場所を反映していない場合、それらを更新し、ご使用のシステムに適したパスになるようにしてください。以下の例で示されているパスは、Windows 上のレジストリー・キーを反映しています。値を更新する際に、ご使用のプラットフォームに適した構文を使用してください。

以下をチェックしてください。

- データベースをアップグレードまたは再インストールした場合は、以下をチェックします。

```
21db_jdbcClasspathAddition="/home/db2_instance_name/sqllib/bin/java12/db2java.zip"
21db_jdbcSharedLibPathAddition="/home/db2_instance_name/sqllib/bin"
21db_odbcSharedLibPathAddition="/home/db2_instance_name/sqllib/bin"
```

- IBM カセットがインストールされている場合、上記と同名のレジストリー・キーに各カセットに固有の接頭部が付いたものもチェックしてください。たとえば、Cassette for CyberCash がインストールされている場合は、以下をチェックします。

```
CC_21db_jdbcClasspathAddition="/home/db2_instance_name/sqllib/bin/java12/db2java.zip"
CC_21db_jdbcSharedLibPathAddition="/home/db2_instance_name/sqllib/bin"
CC_21db_odbcSharedLibPathAddition="/home/db2_instance_name/sqllib/bin"
```

21db_jdbcClasspathAddition キーとそれに関連するすべてのカセット・キーを更新する際は、必ずデータベース・ドライバーの JDBC 2.0 バージョンを使用してください。たとえば、以下のレジストリー・キーがあるとしたら、

```
21db_jdbcClasspathAddition="/home/db2_instance_name/sqllib/bin/java/db2java.zip"
```

これを、次のように変更します。

```
21db_jdbcClasspathAddition="/home/db2_instance_name/sqllib/bin/java12/db2java.zip"
```

- データベースとして UDB を使用している場合は、以下もチェックしてください。

```
UDB_DIR="/home/db2_instance_name/sql1ib"
```

- Web サーバーをアップグレードまたは再インストールした場合は、以下をチェックします。

```
WEB_PUBDIR="/opt/IBMHTTPServer/htdocs"
```

- WebSphere Application Server をアップグレードまたは再インストールした場合は、以下をチェックします。

```
WAS_dir="/opt/WebSphere/AppServer"  
WAS_ver="4.0.1"  
javaExe="/opt/WebSphere/AppServer/java/jre/bin/java"  
javaClasses="/opt/WebSphere/AppServer/java/jre/lib/rt.jar;  
/opt/WebSphere/AppServer/java/jre/lib/i18n.jar"
```

- また、WebSphere Application Server ディレクトリーが正しいものとなるように (たとえば、 /opt/WebSphere/AppServer/java/jre/bin/java ディレクトリーなど)、 `ibmpaysserver.sh` スクリプトを更新する必要もあります。

Test Cassette

Payment Manager バージョン 2.1.x で提供されていた Test Cassette は、バージョン 2.2 からは提供されなくなりました。インストール・プログラムで Test Cassette が検出されると、削除されます。現在 Test Cassette をご使用の場合は、OfflineCard Cassette の使用に切り替えるようにお勧めします。

OfflineCard Cassette の詳細については、*Payment Manager 管理者ガイド* を参照してください。

EventListener SocksHost の長さ制限

現在 SocksHost フィールドの長さは 254 文字に制限されています。254 文字よりも長い SocksHost フィールドを使用しているアプリケーションによって EventListener が作成されている場合、マイグレーション・プログラムによって 254 文字に切り捨てられます。イベント通知の詳細については、*Payment Manager プログラマーのガイドとリファレンス* を参照してください。

PSWAS2Realm のマイグレーション

Payment Manager 環境で PSWAS2Realm を使用していた場合 (PaymentServlet.properties ファイルで示されている)、Payment Manager 3.1.2 をインストールする際に、レルムはマイグレーションされません。代わりに、Payment Manager では PSDefaultRealm が使用されます。この変更の結果として、自分で何らかのアクションを行う必要はありません。

インストールおよびマイグレーション

Payment Manager 3.1.2 のインストールおよびマイグレーション・プログラムでは、Payment Manager バージョン 2.1.4.0 以降からのデータがマイグレーションされます。既存のデータベースが引き続き使用され、インストール中に必要なすべてのデータ・マイグレーションが行われます。

ご使用のプラットフォームで Payment Manager 3.1.2 をインストールする場合は、*IBM WebSphere Payment Manager 3.1 for Multiplatforms インストール・ガイド* の、該当するインストールのセクションを参照してください。

注: *IBM WebSphere Payment Manager 3.1 for Multiplatforms* インストール・ガイドでは、`wpm.RealmClass` パラメーターを `com.ibm.etill.framework.payserverapi.PSDefaultRealm` から `com.ibm.commerce.payment.realm.WCSRealm` に切り替えるように指示が出されます。これによって Payment Manager は、WebSphere Commerce を使用してログイン時にユーザーを認証するようになります。つまり、WebSphere Commerce は実行していなければならない、Payment Manager へのログインに使用されるユーザー ID は有効な WebSphere Commerce 管理者 ID でなければならないということです。

Payment Manager を使用する前に、少なくとも一度は WebSphere Commerce 管理コンソールにログインすることをお勧めします。WebSphere Commerce 管理コンソールにログインするには、以下にアクセスします。

```
https://host_name:8000/adminconsole
```

デフォルトの管理コンソールのユーザー ID (`wcsadmin`) と、デフォルトのパスワード (`wcsadmin`) を入力します。最初のログイン時にパスワードを変更するように求められます。

インストール後の考慮事項

カセット

Payment Manager フレームワークのインストールとマイグレーションが正常に行われた後、ご使用の IBM カセットとバージョン 2.x の非 IBM カセットを、バージョン 3.1.2 に更新し、正しく機能するようにしなければなりません。

除去されるファイル

Payment Manager および IBM Payment Manager カセットに関して、以下のような、名前に言語クォリファイア (** で表される) が含まれているすべての PSPL ファイルは削除されます。

```
pspl/admin.**.PSPL  
pspl/payment.**.PSPL  
pspl/reports.**.PSPL
```

サード・パーティーの、言語クォリファイアが付いている PSPL ファイルは除去されません。

Payment Manager の Tivoli® Ready サポートは提供されなくなったため、`PMInstallDir>/tivsupport` ディレクトリーは除去されます。

加えて、以下の点に注意してください。

- 正常にインストールされたら、`jre` サブディレクトリー全体が除去されます (Payment Manager では、WebSphere Application Server が使用する IBM SDK for Java を使用します)。
- Web 公開ディレクトリー `WebPubDir` の下に以前リストされていた Web ページは除去され、`WAS_Home/installedApps` ディレクトリーにある `PaymentManager.ear` ファイルに含まれている `Payment Manager.war` ファイルで使用可能になります。

- これまでトレース・ログ情報をフォーマット設定するために使用されていた `FormatServletTrace.cmd` ファイルは除去されます。代わりに、`FormatTrace.cmd` ファイルを使用してトレース情報をフォーマット設定できます。このコマンド・ファイルの詳細については、*Payment Manager 管理者ガイド* を参照してください。

バックアップされるファイル

マイグレーション中に、これらのキー・ファイルおよびサブディレクトリーが、新しいバックアップ・ディレクトリー `<pmInstallDir>/!pm22Backup!` または `<pmInstallDir>/!pm21Backup!` (以前に使用していたバージョンによる) に保管されます。

```
.payment
IBMPayServer [.cmd ]
IBMPaymentServerUI.properties
PaymentServlet.properties
PSDefaultRealm
StopIBMPayServer [.cmd ]
log/(all files within this directory)
samples/(all files within this directory)
```

UDB のアップグレード

データベース・マネージャー製品として UDB を使用している場合、今回のバージョンの *Payment Manager* で必要なレベルの UDB (DB2 7.2 FixPak 6 以降) を使用していることを確認しなければなりません。今回のレベルの UDB にアップグレードする必要がある場合、*IBM WebSphere Payment Manager 3.1 for Multiplatforms インストール・ガイド* の『UDB のアップグレード』のセクションを参照してください。

UDB を初めてインストールし、マイグレーションが必要ない場合は、インストール手順について、*IBM WebSphere Payment Manager 3.1 for Multiplatforms インストール・ガイド* の『ワークステーション・プラットフォームでのデータベース (UDB) のインストール』を参照してください。

product.xml の更新

`/opt/WebSphere/CommerceServer/xml` ディレクトリーにある `product.xml` ファイルの `<migrationFrom>` セクションを以下のように変更します。(`<migrationFrom>` セクションは、マイグレーション元の以前のマシンについての情報を示します。)

```
<migrationFrom>
  <edition>
    <name>name</name>
  </edition>
  <version>5</version>
  <release>rel</release>
  <modification>mod</modification>
  <fixpack>0</fixpack>
  <path>path</path>
</migrationFrom>
```

上記の内容は以下を表します。

- `name` は以下のいずれか 1 つです。

Start WebSphere Commerce Suite Start Edition — Windows または Linux のみ

Pro WebSphere Commerce Suite Pro Edition

Business WebSphere Commerce Business Edition 5.1 — Windows のみ

Business Limited Availability

(WebSphere Commerce 5.4 Limited Availability リリース)

- *rel* は以下のいずれか 1 つです。

1 これは以下のいずれかです。

- WebSphere Commerce Suite 5.1.1 — Windows のみ
- WebSphere Commerce Suite 5.1
- WebSphere Commerce Business Edition 5.1 — Windows のみ

4 WebSphere Commerce 5.4 Limited Availability リリース

- *mod* は以下のいずれか 1 つです。

0.1 WebSphere Commerce Suite 5.1.0.1

1 WebSphere Commerce Suite 5.1.1 — Windows のみ

- *path* は、たとえば /opt/WebSphere/CommerceSuite for WebSphere Commerce Suite 5.1 などの、以前の Commerce システムのインストール・パスです。

以下に示すのは、 WebSphere Commerce Suite 5.1.0.1 Pro Edition からマイグレーションする場合に、 `product.xml` ファイルに含まれている必要があるものの例です。

```
<migrationFrom>
  <edition>
    <name>Pro</name>
  </edition>
  <version>5</version>
  <release>1</release>
  <modification>0.1</modification>
  <fixpack>0</fixpack>
  <path>/opt/WebSphere/CommerceSuite</path>
</migrationFrom>
```

第 4 章 データベースのマイグレーション

この章では、Commerce Suite 5.1 データベース・スキーマを WebSphere Commerce 5.4 スキーマ・レベルにマイグレーションするためのステップを説明します。スキーマをマイグレーションするためのステップを実行する前に、31 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』および 35 ページの『第 3 章 ソフトウェアのアップグレード』のステップを完了することをお勧めします。

重要

この章および本書の残りの部分に示されているマイグレーションの手順を進める前に、以下を実行します。

- 以下のデータベース・マイグレーション・ユーティリティを実行するには、DB2 データベースがリモート・データベースの場合、これを適切にカタログ化する必要があります。これがローカル・データベースの場合は、デフォルトでカタログ化されます。
- (テーブルに列を追加するなどして) WebSphere Commerce Suite スキーマを拡張した場合、159 ページの『付録 B. データ・マイグレーション・スクリプトの拡張』で説明されているステップを実行する必要があります。
- WebSphere Commerce 要約テーブルのロードは、WebSphere Commerce 5.4 のインストールまたはマイグレーション・プロセスでは行われません。マイグレーションの後に要約テーブルをロードする場合、マイグレーションを完了した後で、WebSphere Commerce 構成マネージャーの「Search Configuration (構成の検索)」パネルを使ってロードできます。WebSphere Commerce インストール・ガイド バージョン 5.4 にある構成のセクションを参照してください。
- Commerce Suite 5.1 データベースのバックアップを収めるのに十分なフリー・ディスク・スペースがマシン上にあることを確かめてください。データベース・マイグレーション・スクリプトはデータベースのバックアップをマイグレーション時に 3 回実行するので、この追加のディスク・スペースは、少なくとも現在の WebSphere Commerce Suite 5.1 データベースの 3 倍のサイズが必要です。

STORE テーブルおよび populatedb スクリプトのセットアップ

各国語バージョンの WebSphere Commerce を使用する場合、データ・マイグレーションの前に、デフォルトのストア用の LANGUAGE_ID が、STORE テーブルと populatedb スクリプトとで、使用する言語に正しく設定されていることを確認します。

注: Linux 上の WebSphere Commerce 5.4 では、以下の各国語はサポートされていません。

- Redhat Linux 7.2 — 中国語 (簡体字)、中国語 (繁体字)

- SuSE Linux Enterprise Server 7.0 — 韓国語、日本語、中国語 (簡体字)、中国語 (繁体字)

各国語サポートの詳細については、*WebSphere Commerce* インストール・ガイドバージョン 5.4 を参照してください。

1. STORE テーブルで、LANGUAGE_ID が、デフォルトのストア (つまり STORE_ID=0 のストア) に対して、使用する言語の正しい値に設定されていることを以下のよう
にチェックします。

LANGUAGE_ID 値	言語
-1	米国英語
-2	フランス語
-3	ドイツ語
-4	イタリア語
-5	スペイン語
-6	ブラジル・ポルトガル語
-7	中国語 (簡体字)
-8	中国語 (繁体字)
-9	韓国語
-10	日本語

2. /opt/WebSphere/CommerceServer/bin ディレクトリーにある元の populatedb スクリプトをバックアップするには、以下のようになります。

- DB2 データベースの場合 — populatedb.db2.sh

3. wcs.comment_multi_en_US.xml を、/opt/WebSphere/CommerceServer/schema/xml ディレクトリー内にコピーして、ロケール固有のファイル名を付けます。たとえば、日本語ロケール ja_JP を使用する場合は、ファイル wcs.comment_multi_en_US.xml をコピーして wcs.comment_multi_ja_JP.xml にします。

4. バックアップした populatedb スクリプトを変更して、すべてのロケール情報が、使用する言語に対して正しいことを確認します (たとえば日本語の場合は ja_JP)。英語 (米国) ロケール en_US のすべての オカレンスをスクリプトから検索します。

```

.
.
.
%JAVA_HOME%bin/java -Dcom.ibm.wca.logging.configFile=%wcaloggerconfigfile%
-Dcom.ibm.wcm.ErrorReporterDir=%errorlogdir% -classpath %CP% %massLoader%
-infile %schema%/xml/wcs.bootstrap_en_US.xml -dbname %database%
-dbuser %user% -dbpwd %password% -method sqlimport -commitcount 100 -schemaname
%schemaowner% >> %log%
%JAVA_HOME%/bin/java -Dcom.ibm.wca.logging.configFile=%wcaloggerconfigfile%
-Dcom.ibm.wcm.ErrorReporterDir=%errorlogdir% -classpath %CP% %massLoader%
-infile %schema%/xml/wcs.comment_multi_en_US.xml -dbname %database%
-dbuser %user% -dbpwd %password% -method sqlimport -commitcount 100 -schemaname
%schemaowner% >> %log%
.
.
.

```

上記の例では、2 つのオカレンスだけを示していますが、実際には多数のオカレンスがあることに注意してください。

en_US のすべての オカレンスを、使用する言語のロケールに変更します。たとえば、使用する言語が日本語であり、STORE テーブルで STORE_ID=0 である場合に LANGUAGE_ID=-10 であれば、以下の populatedb スクリプトで太字で示しているとおりに、すべての en_US のオカレンスを ja_JP に変更します。

```
%JAVA_HOME%bin/java -Dcom.ibm.wca.logging.configFile=%wcaloggerconfigfile%
-Dcom.ibm.wcm.ErrorReporterDir=%errorlogdir% -classpath %CP% %massLoader%
-infile %schema%/xml/wcs.bootstrap_**_JP.xml -dbname %database%
-dbuser %user% -dbpwd %password% -method sqlimport -commitcount 100 -schemaname
%schemaowner% >> %log%
%JAVA_HOME%/bin/java -Dcom.ibm.wca.logging.configFile=%wcaloggerconfigfile%
-Dcom.ibm.wcm.ErrorReporterDir=%errorlogdir% -classpath %CP% %massLoader%
-infile %schema%/xml/wcs.comment_multi_**_JP.xml -dbname %database%
-dbuser %user% -dbpwd %password% -method sqlimport -commitcount 100 -schemaname
%schemaowner% >> %log%
```

上記の例で示しているよりも数多くのオカレンスが存在する可能性があることに注意してください。

以下の Java ロケールが WebSphere Commerce 5.4 ではサポートされています。

注: Linux 上の WebSphere Commerce 5.4 では、以下の各国語はサポートされていません。

- Redhat Linux 7.2 — 中国語 (簡体字)、中国語 (繁体字)
- SuSE Linux Enterprise Server 7.0 — 韓国語、日本語、中国語 (簡体字)、中国語 (繁体字)

各国語サポートの詳細については、*WebSphere Commerce* インストール・ガイド バージョン 5.4 を参照してください。

ロケール名	政治的、地理的、または文化的領域
en_US	アメリカ合衆国
fr_FR	フランス
de_DE	ドイツ
it_IT	イタリア
es_ES	スペイン
pt_BR	ブラジル
zh_CN	中国
zh_TW	台湾
ko_KR	韓国
ja_JP	日本

5. STORE テーブルで、デフォルトのストアの LANGUAGE_ID が、使用したい言語の正しいロケールになっていることを確認します。

```
DB2 CONNECT TO database
DB2 SELECT LANGUAGE_ID FROM STORE WHERE STORE_ID=0
```

たとえば、使用する言語が日本語で、LANGUAGE_ID が日本語用に正しく設定されていない場合、つまり -1 (en_US 用) などになっている場合、それを手動で -10 (ja_JP 用) に変更する必要があります。

```
DB2 UPDATE STORE SET LANGUAGE_ID=-10 WHERE STORE_ID=0
```

6. この章の残りの部分で説明しているとおりにデータベースのマイグレーションを実行します。データベースのマイグレーションを完了したら、LANGUAGE_ID を元の設定に戻します。この例では en_US 用の -1 です。

リモート DB2 データベースのカタログ化 (DB2 のみ)

リモートの DB2 データベースを使用している場合、以下のようにして、ローカル・ノードでこれをカタログ化します。

1. データベースの DB2 インスタンス・ユーザー ID (たとえば db2inst1) に切り替えます。

```
su - db2_instance_user_id
```

2. 次のように入力して、リモート・データベース・サーバー・ノードをカタログ化します。

```
db2 catalog tcpip node node_name
remote database_server_hostname server port_number
```

ここで

node_name

ユーザーが指定する、DB2 が TCP/IP ノードを識別するための固有
名。

database_server_hostname

データベース・サーバーの TCP/IP ホスト名。

port_number

DB2 の使用するポート。(インストール時には、DB2 はデフォルトで
50000 の使用を試みます。このポート番号を使用できない場合は、使用
可能な別の未使用ポートを使用します。)

3. リモート・ノードでリモート DB2 データベースをカタログ化します。

```
db2 catalog db db_name at node node_name
```

ここで

db_name

DB2 データベース (たとえば mall)。

node_name

上でカタログ化したリモート・ノード。

避けるべきシナリオ

マイグレーション時には、以下のようなデータベースの使用のシナリオは避けてください。

たとえば、hostA という 1 マシン上に 2 つのデータベース・インスタンスがあり、1 つが db2inst1 というデフォルトのインスタンスで、もう 1 つが ecom というインスタンスだとします。インスタンス db2inst1 の下には、dbtest というデータベースがあります。

別の hostB というマシン上にも 2 つのデータベース・インスタンスがあり、1 つが db2inst1 で、もう 1 つが ecom だとします。インスタンス ecom の下には、dbtest というデータベースがあります (データベース・ユーザーとパスワードは、hostA のデータベース dbtest と同じです)。

マシン hostA 上で以下を実行します。

```
su - ecom
db2 catalog tcpip node hostB remote hostB server 50000
db2 catalog db dbtest as dbtest at node hostB
```

したがって hostA 上には、hostB にリダイレクトされる dbtest というリモート・データベースが存在することになります。

この条件下では、hostA 上で WebSphere Commerce を実行している場合、実際の接続先のデータベースは hostB 上の dbtest ではなく、hostA 上の dbtest です。デフォルトのインスタンスが db2inst1 であるため、WebSphere Commerce はこのインスタンス名を使用してデータベースに接続します。

データベース準備スクリプトの実行

このスクリプトは、データベースのアクセス・コントロール・テーブルを変更して、それらが適切にマイグレーションできるようにします。したがって、このスクリプトを適切なデータベース・マイグレーションに対して実行する必要があります。

データベース準備スクリプトも、既存の Commerce Suite データベースを分析して、以下の特性を持つデータについてのレポートを生成します。

- 親商品がないすべてのアイテム
- 組織エンティティ内に親がないすべてのメンバー

このレポートには、72 ページの『必須のデータベース・プレマイグレーション項目』および 74 ページの『オプションのデータベース・プレマイグレーション項目』に説明されているアイテムがリストされます。

必須項目については、データベースのマイグレーションに進む前に、要求されているアクションを実行する必要があります。

オプション項目については、データベースのマイグレーションに進む前の推奨アクションの実行は、必須ではありません。ただし、マイグレーション・スクリプトによるデフォルトの割り当てを受けたくない場合、フラグが立てられたデータを訂正しておくことをお勧めします。データを訂正した後、データベース準備スクリプトを再実行して、すべての項目が修正されたことを確認する必要があります。

必要なすべての項目を修正した後、31 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』に説明されているように、再度 Commerce Suite データベースをバックアップして、データベースの最新のコピーを所有するようにします。

以下の手順で、WebSphere Commerce 5.4 データベース準備スクリプトを実行します。

DB2 データベース

1. コマンド・プロンプト・ウィンドウをオープンします。
2. root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、データベースの DB2 インスタンス・ユーザー ID (たとえば db2inst1) に切り替えます。

```
su - db2_instance_user_id
```

非 root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、WebSphere Application Server ユーザー ID に切り替えます (たとえば wasuser)。

```
su - WAS_user
```

__ 3. WebSphere Commerce 5.4 がインストールされている bin サブディレクトリに移動します。たとえば、/opt/WebSphere/CommerceServer/bin。

__ 4. データベース準備スクリプトを以下のように実行します。

```
./premigrate.db2.sh db_name db_userID db_userID_password db2insthome
```

ここで

- *db_name* は、マイグレーション先の Commerce Suite データベースです (たとえば mall)。
- *db_userID* は、マイグレーション先の Commerce Suite データベースに接続するためのユーザー ID です (たとえば mydbuser)。
- *db_userID_password* は、マイグレーション先の Commerce Suite データベースに接続するためのユーザー ID のパスワードです (たとえば mypasswd)。
- *db2insthome* は、DB2 インスタンス・ユーザー ID のホーム・ディレクトリです (たとえば /home/db2inst1/sqllib)。

たとえば、データベース準備スクリプトを実行するには、以下のコマンドを使用することができます。

```
./premigrate.db2.sh mall mydbuser mypasswd /home/db2inst1/sqllib
```

__ 5. /opt/WebSphere/CommerceServer/logs/migration ディレクトリに生成された premigrate.log ファイル内に、追加のメッセージまたはエラーがないかを調べます。続行する前に、フラグが立てられたエラーを解決する必要があります。

データベース準備スクリプトによって通知されたエラーまたは警告を修正した場合や、データベースに何らかの変更を加えた場合には、33 ページの『データベースのバックアップ』の説明に従ってデータベースをバックアップすることをお勧めします。

必須のデータベース・プレマイグレーション項目

データベース・プレマイグレーション・スクリプトが以下のリターン・コードを生成する場合、データベースのマイグレーションを続行する前に以下のアクションを実行しなければなりません。

リターン・

コード 説明 / アクション

122 スクリプトは、Commerce Suite 5.1 データベースが以下のメンバー ID (0 ~ -8) を、対応するメンバー・グループに使用しているかどうか (たとえば、サイト管理者メンバー・グループのメンバー ID が 1 であるかどうか) をチェックします。

0 Site Owner (サイト所有者)

- 1 Site Administrator (サイト管理者)
- 2 Customer (顧客)
- 3 Customer Service Representative (顧客サービス担当者)
- 4 Merchant (マーチャント)
- 5 Order Clerk (オーダー・クラーク)
- 6 Store Administrator (ストア管理者)
- 7 Store Developer (ストア・デベロッパー)
- 8 Merchandising Manager (取引管理マネージャー)

注:

1. すべてのブートストラップ値を保存することが必要です。そうしないと、データベースのマイグレーションは失敗します。
2. リターン・コード 122 は、英語のシステムだけに適用可能です。英語以外の各国語バージョンを使用している場合、ご使用のシステムに対するこのリターン・コードは無視することができます。各国語バージョンのユーザーは、ブートストラップ・データを変更していないことを確認する必要があります。
3. WebSphere Commerce 5.4 では、Merchant 役割は Seller に名前変更され、Merchandising Manager は Product Manager (プロダクト・マネージャー) に名前変更されています。

200

スクリプトは Commerce Suite 5.1 内の AUCTION テーブルの REFCODE フィールドをチェックします。固有であるはずの REFCODE フィールドがありますが、Commerce Suite 5.1 スキーマはこのことを強制していません。WebSphere Commerce 5.4 スキーマは、それが固有索引であることを指定しています。

AUCTION テーブル内の REFCODE フィールドが固有であることを確認する必要があります。

318

スクリプトは、契約名の長さが 200 文字を超えていないことをチェックします。

AUCTION テーブル内の NAME フィールドのデータが 200 文字を超えていないことを確認する必要があります。

319

スクリプトは、ORGENTITY テーブル記述の長さが 512 文字を超えていないことをチェックします。

ORGGRP テーブル内の DESCRIPTION フィールドのデータが 200 文字を超えていないことを確認する必要があります。

340

スクリプトは、MBRGRP テーブル記述の長さが 512 文字を超えていないことをチェックします。

MBRGRP テーブル内の DESCRIPTION フィールドのデータが 200 文字を超えていないことを確認する必要があります。

オプションのデータベース・プレマイグレーション項目

データベース・プレマイグレーション・スクリプトが以下のリターン・コードを生成する場合、データベースのマイグレーションを続行する前に以下のアクションを実行することが推奨されています。これらのアクションは必須ではありませんが、マイグレーション後のシステムに対する効果を注意深く考慮する必要があります。

リターン・
コード

説明 / アクション

- 103** スクリプトは、Commerce Suite 5.1 割引データをチェックします。手動で作成したカスタム割引データ、つまり Commerce Suite Accelerator の「マーチャンダイズ」メニューで作成したものではない割引データがある場合、スクリプトは警告を出します。データベース・マイグレーション・スクリプトは、割引データを WebSphere Commerce 5.4 要件にマイグレーションしません。しかし、このデータは現状のままデータベースに残ります。このデータを WebSphere Commerce 5.4 割引ツールによって表示することはできません。
- この割引データ (Commerce Suite 5.1 ツールの外側で作成されたと想定される) を表示したい場合、Commerce Suite 5.1 での割引データの処理で以前使っていたものと同じ手順に従う必要があります。
- 305** スクリプトは BUSPROF テーブル内にレコードがあるかどうかをチェックしますが、ORG_ID および ORGUNIT_ID エントリはヌルです。これらの行の ORG_ID および ORGUNIT_ID データに値を入れる必要があります。値を入れない場合、マイグレーション・スクリプトはデフォルト組織をビジネス・ユーザーの親として割り当てます。さらに、ユーザーの profileType を B (B2B ユーザー) から C (B2C ユーザー) に変更することも検討してください。
- 307** REGISTERTYPE の値が S であり、それがユーザーに対してより特定の役割を持つものとして ACCMBRGRP テーブルに表示されない場合、データベース・マイグレーション・スクリプトは、マイグレーション時にそのユーザーに Site Administrator 役割を自動的に割り当てます。特に、マイグレーション・スクリプトはこれらのユーザーのエントリを MBRROLE テーブル内に作成して、Site Administrator 役割をそれらの祖先の組織エントリに割り当てます。Site Administrator 役割は非常に強力な役割なので、スクリプトは警告を出してユーザーにこのことを通知します。十分に考慮せずに Site Administrator 役割を組織エンティティやユーザーに割り当てることはしないでください。
- 309** Commerce Suite 5.1 での Customer は、すべてのユーザーのグループを表していました。WebSphere Commerce 5.4 に同梱されている AllUsers メンバー・グループによって、Customer アクセス・グループが置き換えられます。ユーザーが Commerce Suite 5.1 で Customer アクセス・グループ (-2) に割り当てられていた場合、WebSphere Commerce 5.4 ではマイグレーション・スクリプトは、そのユーザーを AllUsers メンバー・グループに明示的に割り当て

ます。そのような明示的な割り当ては WebSphere Commerce 5.4 の設計上必須ではないので、プレマイグレーション・スクリプトは警告メッセージを出します。

そのような明示的な割り当てが必要かどうかを検討してください。

- 310** Commerce Suite 5.1 は、Order Clerk 役割をサポートしていましたが、 WebSphere Commerce 5.4 では不要になり使用されなくなりました。Order Clerk 役割で実行に使用されるタスクは、自動化されているか、または WebSphere Commerce 5.4 の顧客サービス・スーパーバイザーで実行できます。ユーザーに Commerce Suite 5.1 で Order Clerk 役割 (-5) があり、ACCCMDGRP テーブルにエントリーがある場合、そのユーザーは、アクセス・コントロール・マイグレーションの一部としてマイグレーションされ、その役割は、ユーザー定義の役割として扱われます。Order Clerk 役割を持つユーザーが存在しない場合、その役割はマイグレーションされません。

まだ Order Clerk 役割が必要かどうかを確認してください。必要でなければ、WebSphere Commerce 5.4 ではサポートされなくなったので、除去してください。

- 316** スクリプトは、ORGENTITY テーブル内に組織エンティティの親メンバー ID があるかどうかをチェックします。

ORGENTITY テーブル内にある、フラグが立てられたアイテムの親 MEMBER_ID に、値を入れることができます。値を入れない場合、データベース・マイグレーション・スクリプトが Default Organization を表す値 -2001 を割り当てます。

- 401** スクリプトは、親商品を持たないすべてのアイテムをチェックします。WebSphere Commerce 5.4 では、各アイテムに 1 つの親商品が存在する必要があります。

親を持たないアイテムについては、そのアイテムを CATGPENREL テーブルに追加して、CATALOG_ID および CATGROUP_ID を割り当ててください。

フラグの立てられたアイテムについて、親商品を作成することができます。フラグの立てられたアイテムについて親商品を作成しない場合、データベース・マイグレーション・スクリプトによってそれが作成されます。

- 404** スクリプトは、複数の親商品を持つすべてのアイテムをチェックします。WebSphere Commerce 5.4 では、各アイテムが持つことのできる親商品は 1 つだけです。

WebSphere Commerce 5.4 カタログ・ツールを使用してカタログ・データを表示したい場合、1 つの親商品を残して他のすべての親商品を除去する必要があります。

- 414** アクセス・コントロールに関する問題があります。アクセス・コントロール検査メッセージのあるログで、詳しいメッセージを見つけて調べてください。メッセージは以下のようなものです。

- The user xxxx does not have an entry in the BUSPROF table.
(BUSPROF テーブルには、ユーザー xxxx のエントリーがありません)

ん。) The user will be migrated to MBRROLE but will not actually be able to perform that role without being assigned to an organization or organization unit that can perform the role. (このユーザーは MBRROLE にマイグレーションされますが、この役割を実行可能な組織または組織単位に割り当てられなければ、実際には役割を実行できません。)

- The owner xxxx for the member group defined in ACCMBRGRP is not an organization and will not be migrated. (ACCMBRGRP で定義されたメンバー・グループの所有者 xxxx は組織ではないため、マイグレーションされません。)

- 415** スクリプトは、オーダーの状況コードが M (支払いが開始されました - 顧客は支払いを開始しました。与信は処理中です)であることをチェックします。
- ORDERS テーブルで、すべてのオーダー・アイテムの STATUS 列が M に設定されていることを確認してください。
- 416** スクリプトは、オーダー・アイテムの状況コードが M (支払いが開始されました - 顧客は支払いを開始しました。与信は処理中です)であることをチェックします。
- ORDERITEMS テーブルで、すべてのオーダー・アイテムの STATUS 列が M に設定されていることを確認してください。

Commerce Suite 5.1 キャッシュ・トリガーの除去

Commerce Suite 5.1 のキャッシュ・トリガーをインストールしている場合は、データベースをマイグレーションする前に、それらを除去する必要があります。以下のディレクトリーに備えられている `wcs.cache.delete.trigger1.sql` スクリプトを使用できます。

- DB2 データベースの場合: `/opt/WebSphere/CommerceServer/schema/db2`

DB2 データベース

DB2 データベースで `wcs.cache.delete.trigger1.sql` スクリプトを実行するには、以下のようにします。

1. コマンド・プロンプト・ウィンドウをオープンします。
2. ご使用のデータベースの DB2 インスタンス・ユーザー ID (たとえば、`db2inst1`) に切り替えます。

```
su - db2_instance_user_id
```
3. `/opt/WebSphere/CommerceServer/schema/db2` ディレクトリーに移動します。
4. Commerce Suite データベースに接続します。

```
db2 connect to db_name user db_user_name using db_password
```
5. `wcs.cache.delete.trigger1.sql` スクリプトを以下のように実行します。

```
db2 -tvf wcs.cache.delete.trigger1.sql
```

カスタム制約の除去

出荷された Commerce Suite 5.1 テーブルへの外部キー・リンクを含むテーブルをカスタマイズした場合、データ・マイグレーション時に、参照保全制約 (外部キー、基本キー、索引など) を除去しようとする、データ・マイグレーション・スクリプトは失敗する場合があります。以下のセクションに示す SQL ステートメントを使用して、これらの制約を除去する必要があります。 82 ページの『カスタム制約のリストア』で説明しているように、データを WebSphere Commerce 5.4 スキーマにマイグレーションした後で、これらの制約をリストアします。

1. 新しく追加したすべてのテーブル、およびすべての Commerce Suite 5.1 テーブルを確認します。
2. 新規テーブルから Commerce Suite 5.1 テーブルへ、またはその逆に向けられるすべての制約 (ビュー、要約テーブル、トリガー、SQL 関数、SQL メソッド、参照に関する制約) を確認します。
3. これらすべての制約を除去します。データベース・マイグレーション・スクリプトを実行した後で、制約による SQL エラーが `migratedb.log` に記録される場合は、データベースをマイグレーションする前に制約を除去する必要があります。

制約を除去するには、後続のセクションにある説明に従って SQL ステートメントを実行します。

DB2 データベース

DB2 データベースでは、以下の手順で SQL ステートメントを実行します。

1. コマンド・プロンプト・ウィンドウをオープンします。
2. ご使用のデータベースの DB2 インスタンス・ユーザー (たとえば、`db2inst1`) に切り替えます。

```
su - db2_instance_user_id
```

3. Commerce Suite データベースに、データベースに対する管理権限を持つユーザーとして接続します。

```
db2 connect to db_name user db_user_name using db_password
```

4. 以下の SQL ステートメントを入力します。

```
db2 ALTER TABLE table_name DROP constraint constraint_name
```

ここで

- `table_name` は、制約を含むカスタマイズされたテーブルの名前です。
 - `constraint_name` は、除去したい参照保全制約の名前です。
5. 除去した制約は、データ・マイグレーションの後にリストアする必要があるの、メモしておきます。

データベース・スキーマのマイグレーション

データベース・プレマイグレーション・スクリプトによってフラグが立てられたすべてのアイテムを除去した後、以下のデータベース・マイグレーション・スクリプトを実行して、スキーマを WebSphere Commerce 5.4 レベルに更新することができます。マイグレーション・スクリプトの働きの概要については、161 ページの『付録 C. マイグレーション・スクリプトの概要』を参照してください。Commerce Suite 5.1 と WebSphere Commerce 5.4 とのデータベース・スキーマの変更の要約については、171 ページの『付録 E. データベース・スキーマの変更点』を参照してください。

データベース上にマイグレーション・スクリプトを実行できるのは 1 回だけであることに注意してください。

注: UNIX[®] システム上で実行している DB2 データベースの場合、データベースは通常、デフォルトで *Rdata_base_name* としてカタログ化されます。たとえば、MALL は、デフォルトでは RMALL としてカタログ化されます。リモート・データベースでデータベース・マイグレーション・スクリプトを実行する際に起きる問題を回避するには、以下を行う必要があります。

1. `migratedb.sh` スクリプトで、以下の行を更新します。

```
db2 connect to $database
```

これを以下のように変更します。

```
db2 connect to $database user $user using $password
```

2. `migratedb.sh` を実行する際、データベース名を *Rdata_base_name* とする必要があります (たとえば RMALL)。

- 1. DB2 を使用している場合、コマンド・プロンプト・ウィンドウをオープンします。
- 2. DB2 を使用しており、かつ root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、データベースの DB2 インスタンス・ユーザー ID (たとえば `db2inst1`) に切り替えます。

```
su - db2_instance_user_id
```

非 root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、WebSphere Application Server ユーザー ID に切り替えます (たとえば `wasuser`)。

```
su - WAS_user
```

- 3. WebSphere Commerce 5.4 がインストールされている `bin` サブディレクトリに移動します。たとえば、`/opt/WebSphere/CommerceServer/bin`。
- 4. 以下のように、データベース・マイグレーション・スクリプトを実行します。

```
./migratedb.sh db_type db_name db_userID db_userID_password  
backup_dir atp | noatp [db2_install_dir] [db2_inst_home]  
[host_name] [port_number] [schema_owner]
```

ここで

- *db_type* は、使用しているデータベース管理システムです。値は `db2` でも構いません。

- *db_name* は、マイグレーション先の Commerce Suite 5.1 データベースです (たとえば mall)。
- *db_userID* は、マイグレーション先の Commerce Suite 5.1 データベースに接続するためのユーザー ID です (たとえば mydbuser)。
- *db_userID_password* は、マイグレーション先の Commerce Suite 5.1 データベースに接続するためのユーザー ID のパスワードです (たとえば mypasswd)。
- *backup_dir* は、マイグレーション・スクリプトがデータベースのバックアップ・コピーを保管するディレクトリーです。たとえば、/my_backup です。非 root ユーザー (wasuser) として WebSphere Application Server を実行している場合、このユーザーがバックアップ・ディレクトリーへの読み取りおよび書き込みアクセスを持っていることを確認します。
- *atp* は、既存の Commerce Suite 5.1 在庫データを、WebSphere Commerce 5.4 予定可能 (ATP) 在庫表記にマイグレーションすることを指定します。
noatp は、既存の Commerce Suite 5.1 在庫データを、WebSphere Commerce 5.4 予定可能 (ATP) 在庫表記にマイグレーションしないことを指定します。

マイグレーションを行うかどうかの判断に役立つ、新しい ATP 在庫表記システムの要約については、5 ページの『ATP 在庫へのマイグレーション』を参照してください。在庫データを ATP 表記にマイグレーションする場合、在庫データに関連したカスタマイズ済みコードを変更する必要があります。詳しくは、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。在庫データを後でマイグレーションすることにした場合、169 ページの『付録 D. 後からの ATP インベントリーへの変換』で説明しているように、*migrateATP* スクリプトを実行する必要があります。

atp を指定して WebSphere Commerce 5.4 にマイグレーションする際に、データベース・マイグレーション・スクリプトはマイグレーション時に RECEIPT テーブルにどのベンダーも含めないことに注意してください。ストアごとに複数のベンダーがあったり、またはベンダーがない可能性があるからです。ベンダーがストアに割り当てられていない場合、ATP 在庫表記にマイグレーションする前に WebSphere Commerce Accelerator を使用してベンダーを作成する必要があります。

- *db2_install_dir* は、DB2 のインストール・パスです。たとえば、/opt/IBMDB2/V7.1 のようになります。このパラメーターが必要なのは、DB2 データベースだけです。
- *db2_inst_home* は、DB2 インスタンスの DB2 インスタンス・パスです。たとえば、/home/db2inst1/sqllib のようになります。このパラメーターが必要なのは、DB2 データベースだけです。
- *host_name* は、マシンの完全修飾ホスト名です。このパラメーターが必要なのは、**Oracle** データベースだけです。**Linux** 上では必要ありません。
- *port_number* は、Oracle listener のポート番号です。一般に、デフォルトではこれはポート 1521 です。このパラメーターが必要なのは、**Oracle** データベースだけです。**Linux** 上では必要ありません。
- *schema_owner* は、データベース・スキーマ所有者のユーザー ID です。
schema_owner は、大文字で入力する必要があります。

たとえば、DB2 データベースのデータベース・マイグレーション・スクリプトを実行して、在庫表記を ATP に変換するためには、以下のコマンドを使用することができます。

```
./migratedb.sh db2 mall mydbuser mypasswd /backup atp /usr/IBMDB2/V7.1  
/home/db2inst1/sqllib DB2INST1
```

__ 5. migratedb.log ファイルをチェックします。

このログ・ファイルは、/opt/WebSphere/CommerceServer/logs/migration ディレクトリーに生成されます。

続行する前に、スクリプトの実行に際してエラーが起きていないことを確認してください。ログ・ファイル内で、error または fail ストリングを検索してください。さらに、ログ・ファイル内を検索する際に、SQLSTATE をチェックしてください。これらのストリングが存在する場合、メッセージを分析して、データを修正し、データ・マイグレーションを再試行する必要があります。

以下の SQLSTATE メッセージが migratedb.log 内にあっても、それらは無視することができます。

```
SQLSTATE=23505 (duplicate entries)
```

```
=====
```

```
LINE      MESSAGES FOR db2c1pnc.bnd  
-----  
SQL0595W  Isolation level "NC" has been escalated to "UR".  
SQLSTATE=01526
```

```
LINE      MESSAGES FOR db2arxnc.bnd  
-----  
SQL0595W  Isolation level "NC" has been escalated to "UR".  
SQLSTATE=01526
```

```
=====
```

```
SQL0100W  No row was found for FETCH, UPDATE or DELETE; or the result of a  
query is an empty table.  SQLSTATE=02000
```

```
-----Warning-----  
Message: [IBM][CLI Driver][DB2/NT] SQL0513W  The SQL statement will modify an  
entire table or view.  SQLSTATE=01504
```

```
SQLState: 1504  
Error Code: 513
```

```
=====
```

```
Executed: update store set ALLOCATIONGOODFOR = 0  
-----Warning-----  
Message: [IBM][CLI Driver][DB2/NT] SQL0513W  The SQL statement will modify an  
entire table or view.  SQLSTATE=01504
```

```
SQLState: 1504  
Error Code: 513
```

以下のエラー・メッセージがデータベース・マイグレーション・ログに現れても、それは無視できます。なぜなら、データベースにデータを取り込む際、データベースは要約テーブルをロードしていないからです。デフォルトでは、wcs.summary.sql スクリプトを使用して WebSphere Commerce 5.4 データベースにデータが取り込まれます。

```
DROP TABLE category
```

```
DB21034E  The command was processed as an SQL statement because it was not a  
valid Command Line Processor command.  During SQL processing it returned:
```

```
SQL0204N  "DB2ADMIN.CATEGORY" is an undefined name.  SQLSTATE=42704
```

```
DROP TABLE product
```

```
DB21034E The command was processed as an SQL statement because it was not a  
valid Command Line Processor command. During SQL processing it returned:
```

```
SQL0204N "DB2ADMIN.PRODUCT" is an undefined name. SQLSTATE=42704
```

```
DROP TABLE richattr
```

```
DB21034E The command was processed as an SQL statement because it was not a  
valid Command Line Processor command. During SQL processing it returned:
```

```
SQL0204N "DB2ADMIN.RICHATTR" is an undefined name. SQLSTATE=42704
```

```
DROP TABLE richattrcatgp
```

```
DB21034E The command was processed as an SQL statement because it was not a  
valid Command Line Processor command. During SQL processing it returned:
```

```
SQL0204N "DB2ADMIN.RICHATTRCATGP" is an undefined name. SQLSTATE=42704
```

```
DROP TABLE storeinv
```

```
DB21034E The command was processed as an SQL statement because it was not a  
valid Command Line Processor command. During SQL processing it returned:
```

```
SQL0204N "DB2ADMIN.STOREINV" is an undefined name. SQLSTATE=42704
```

データベース・マイグレーション・ログおよびトレース・ファイル

データベース・マイグレーション・スクリプトは、
/opt/WebSphere/CommerceServer/logs/migration ディレクトリーにさまざまなログ・ファイルを生成します。

生成されるログ・ファイルを以下に示します。

migratedbscripts.log

スキーマ・スクリプトを生成する際のログ・ファイル

migratedb.log

ブートストラップ・データに値を入れることを含む、全体的なデータ・マイグレーションの際のログ・ファイル

OrigSchema.log

データベース内にある、元となるリリースの WebSphere Commerce Suite 5.1 テーブルの詳細リスト。たとえば、WebSphere Commerce Suite データベース・レベル 5.1.0.1 からのマイグレーションの場合、このログには、5.1.0.1 WebSphere Commerce Suite スキーマ・テーブルのすべてのリストが含まれます。

TargetSchema.log

マイグレーション・スクリプトが正常に実行された後、データベース内にある WebSphere Commerce 5.4 スキーマ・テーブルの詳細リスト。これには、固有索引、列定義、および制約が含まれます。 TargetSchema.log ファ

イルと OrigSchema.log ファイルを比べることにより、出荷時のオリジナルの Commerce Suite テーブルに対してどのようなカスタマイズがなされたかが分かります (たとえば列を追加した、あるいはテーブルを追加したなど)。

データベース・マイグレーション・スクリプトは、
/opt/WebSphere/CommerceServer/schema ディレクトリー内に messages.txt ファイルも生成します。このファイルには、Mass Loader のメッセージが入っています。

トレース情報 (ecmsg_XXXXX ファイル) を使用可能にするには、224 ページの『トレース情報の使用可能化』を参照してください。

カスタム制約のリストア

Commerce Suite 5.1 データベースをマイグレーションした後、77 ページの『カスタム制約の除去』で除去した、参照制約をリストアする必要があります。

以下のセクションで説明されているように、SQL ステートメントを実行します。

DB2 データベース

DB2 データベースでは、以下の手順で SQL ステートメントを実行します。

1. コマンド・プロンプト・ウィンドウをオープンします。
2. データベースの DB2 インスタンス・ユーザー ID (たとえば db2inst1) に切り替えます。

```
su - db2_instance_user_id
```

3. Commerce Suite データベースに、データベースに対する管理権限を持つユーザーとして接続します。

```
db2 connect to db_name user db_user_name using db_password
```

4. 以下の SQL ステートメントを入力します。

```
db2 ALTER TABLE table_name  
ADD CONSTRAINT constraint_name FOREIGN KEY (column_name)  
REFERENCES foreign_table_name ON DELETE CASCADE
```

ここで

- *table_name* は、制約をリストアする必要のある、カスタマイズされたテーブルの名前です。
- *constraint_name* は、リストアしたい参照保全制約の名前です。
- *column_name* は、参照制約が適用されている列の名前です。
- *foreign_table_name* は、参照制約が適用されている外部テーブルの名前です。

参照制約を追加する SQL ステートメントの例については、
/opt/WebSphere/CommerceServer/schema/db2 ディレクトリー内のファイル
wcs.referential.sql をご覧ください。

識別名の更新

WebSphere Commerce 5.4 では、ORAGENTITY および USERS テーブルの識別名 (DN) 列に値を取り込む必要があります。migrateDN スクリプトを使用して、ゲスト・ユーザー (タイプ G) を除き、これらのテーブルに推奨値を入れることができます。このスクリプトは、ORAGENTITY テーブル内の DN 列に値を取り込む fillorgDN.sql スクリプトを呼び出して、USERS テーブル内のユーザーの DN 列に値を取り込みます。fillorgDN.sql スクリプトは、データベース・マイグレーション・スクリプトの実行時に生成されます。

migrateDN スクリプトを実行する前に、以下を行います。

- fillorgDN.sql スクリプトを見つけてその内容を表示し、DN 列の値と、それが更新される値を参照してください。fillorgDN.sql スクリプトは、schema サブディレクトリー (/opt/WebSphere/CommerceServer/schema など) にあります。
- 推奨されている更新を行いたくない場合、スクリプトを適切にカスタマイズする必要があります。

注: LDAP サーバーを使用している場合、生成された fillorgDN.sql を編集して、組織エンティティーの識別名 (DN) が希望どおりのものになるようにします。後で WebSphere Commerce によって組織エンティティーが LDAP サーバー上に作成されるとき、ORAGENTITY テーブル内の DN 値が使用されます。たとえば、Root Organization という名前の組織エンティティーを、DN 値が c=US である LDAP エントリーの下で作成したい場合、fillorgDN.sql 内で Root Organization の DN を、o=Root Organization から o=Root Organization,c=US に変更します。他の組織エンティティーの識別名も、これに応じて変更する必要があります。たとえば、DN エントリーの o=YourOrganization,o=Root Organization を、o=YourOrganization,o=Root Organization,c=US に変更します。

fillorgDN.sql スクリプトの内容に満足できれば、次のセクションで説明している手順で、migrateDN を実行します。

DB2 データベース

DB2 データベースで migrateDN スクリプトを実行するには、以下のようになります。

1. コマンド・プロンプト・ウィンドウをオープンします。
2. root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、データベースの DB2 インスタンス・ユーザー ID (たとえば db2inst1) に切り替えます。

```
su - db2_instance_user_id
```

非 root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、WebSphere Application Server ユーザー ID に切り替えます (たとえば wasuser)。

```
su - WAS_user
```

3. WebSphere Commerce 5.4 がインストールされている bin サブディレクトリーに移動します。たとえば、/opt/WebSphere/CommerceServer/bin。

- __ 4. 以下のようにして、整合性チェッカーを、マイグレーションされたデータベース上で実行します。

```
./migrateDN.db2.sh db_name db_userID db_userID_password db2insthome
```

ここで

- `db_name` は、 WebSphere Commerce 5.4 データベース・スキーマ・レベルにマイグレーションされた Commerce Suite 5.1 データベースです (たとえば `mall`)。
- `db_userID` は、マイグレーションされたデータベースに接続するためのユーザー ID です (たとえば `mydbuser`)。
- `db_userID_password` は、マイグレーションされたデータベースに接続するためのユーザー ID のパスワードです (たとえば `mypasswd`)。
- `db2insthome` は、 DB2 インスタンス・ユーザー ID のホーム・ディレクトリーです (たとえば `/home/db2inst1/sqllib`)。

たとえば、マイグレーションされた `mall` データベースに対してスクリプトを実行するには、以下のコマンドを使用できます。

```
./migrateDN.db2.sh mall mydbuser mypasswd /home/db2inst1/sqllib
```

- __ 5. `/opt/WebSphere/CommerceServer/logs/migration` ディレクトリーに生成された `migrateDN.log` ファイルをチェックします。続行する前に、スクリプトの使用に際してエラーが起きていないことを確認してください。

マスター・カタログの割り当て

WebSphere Commerce 5.4 では、それぞれのストアごとに指定されたマスター・カタログを持つことが必要です。ストアにマスター・カタログを割り当てるには、データベース・マイグレーション・スクリプトの実行時に生成された `choosemc.sql` スクリプトを実行できます。このマイグレーション・スクリプトは、データベース内に複数のカタログがあるかどうかを検出します。データベース内にカタログが 1 つしか存在しない場合、このマイグレーション・スクリプトは、そのカタログをマスター・カタログとして指定します。この場合は、以下のステップを実行する必要はありません。

`choosemc.sql` スクリプトを実行する前に、それを編集する必要があります。ストリング `MASTERCATALOG_ID` を見つけ、それをマスター・カタログとして指定したいカタログに対応する、参照番号 (基本キー) に置き換えます。たとえば、スクリプト内で以下のステートメントを見つめます。

```
--please replace MASTERCATALOG_ID with one of the catalog of the store you want to designate as MasterCatalog
update storecat set mastercatalog='1' where catalog_id=MASTERCATALOG_ID
and storeent_id=10001;
insert into catgrptpc (catgroup_id,catalog_id,tradeposcn_id)
values (0,MASTERCATALOG_ID,10006);
```

カタログ ID 6000 をマスター・カタログとして選択するには、以下のようにしてステートメントを更新します。

```
update storecat set mastercatalog='1' where catalog_id=6000
and storeent_id=10001;
insert into catgrptpc (catgroup_id,catalog_id,tradeposcn_id)
values (0,6000,10006);
```

スクリプトを実行した後の出力例を以下に示します。

```
--store :10001 has 20 catalogs.
--catalog:311000
--catalog:321000
--catalog:341000
--catalog:6000
--catalog:361000
--catalog:371000
--catalog:322000
--catalog:391000
--catalog:411000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:611000
--catalog:612000
--catalog:10001
--please replace MASTERCATALOG_ID with one of the catalog of the store you want to
designate as MasterCatalog
update storecat set mastercatalog='1' where catalog_id=6000
and storeent_id=10001;
insert into catgrptpc (catgroup_id,catalog_id,tradeposcn_id)
values (0,6000,10006);
```

このスクリプトを実行するには、以降のセクションを参照してください。

DB2 データベース

DB2 データベースで choosemc.sql スクリプトを実行するには、以下のようになります。

1. コマンド・プロンプト・ウィンドウをオープンします。
2. データベースの DB2 インスタンス・ユーザー ID (たとえば db2inst1) に切り替えます。

```
su - db2_instance_user_id
```

3. /opt/WebSphere/CommerceServer/schema ディレクトリーに移動します。
4. Commerce Suite データベースに接続します。

```
db2 connect to db_name user db_user_name using db_password
```

5. choosemc.sql スクリプトを以下のように実行します。

```
db2 -tvf choosemc.sql
```

オーダーおよびオーダー・アイテムの状況の変更

Commerce Accelerator ツールを使用してオーダーおよびオーダー・アイテムを処理するには、状況が C のすべてのオーダーおよびオーダー・アイテムの状況を S に変更することをお勧めします。ただし、これは必須ではありません。必要な考慮事項は、7 ページの『オーダーおよびオーダー・アイテム』で説明しています。状況を変更するには、データベース・マイグレーション・スクリプトの実行時に生成された ctos.sql スクリプトを使用できます。

このスクリプトを実行するには、以降のセクションを参照してください。

DB2 データベース

DB2 データベースで `ctos.sql` スクリプトを実行するには、以下のようにします。

1. コマンド・プロンプト・ウィンドウをオープンします。
2. データベースの DB2 インスタンス・ユーザー ID (たとえば `db2inst1`) に切り替えます。

```
su - db2_instance_user_id
```

3. `/opt/WebSphere/CommerceServer/schema` ディレクトリーに移動します。
4. Commerce Suite データベースに接続します。

```
db2 connect to db_name user db_user_name using db_password
```

5. `ctos.sql` スクリプトを以下のように実行します。

```
db2 -tvf ctos.sql
```

デフォルト・ストアのブートストラップ・データ

マイグレーションの前に、デフォルト・ストア (0 に設定された `STOREENT_ID` によって識別される) のブートストラップ・データに対して変更を加えた場合、WebSphere Commerce 5.4 にマイグレーションした後も、デフォルト・ストアのブートストラップ・データに対して同様の変更を加える必要があります。デフォルト・ストアの場合、データ・マイグレーション・プロセスは、デフォルトのブートストラップ・データをロードしますが、これによって、ブートストラップ・データに対して行ったカスタマイズは上書きされます。非デフォルト・ストアの場合、データ・マイグレーション・プロセスは、ストアのブートストラップ・データに対して変更を加えません。

たとえば、マイグレーションの前に、`CMDREG` テーブルで、`OrderProcessCmd` コマンドのインプリメンテーションを、`OrderProcessBonusImpl` に変更したとします。データベース・マイグレーション・スクリプトを変更した後、エントリーはデフォルト値の `OrderProcessImpl` にリセットされます。この場合、これを手動で `OrderProcessBonusImpl` に変更し直す必要があります。そうしないと、`OrderProcessCmd` コマンドにアクセスした際、ブランク・ページが戻されてしまいます。この変更を行えば、問題なくページにアクセスすることができます。

注: 通常は、ブートストラップ・ファイルは変更しないようお勧めします。変更すると、インスタンスは正常に作成されない場合があります。

Payment Manager に関する考慮事項

WebSphere Commerce Suite 5.1 におけるデフォルト・ストアのブートストラップ・データでは、`DoPaymentCmd` コマンドのデフォルト・インプリメンテーション・クラスは `DoPaymentCmdImpl` になります。このインプリメンテーションは、`Payment Manager` を使用しません。

一方、WebSphere Commerce Business Edition 5.1 または WebSphere Commerce 5.4 における `DoPaymentCmd` のデフォルト・インプリメンテーション・クラスは、`DoPaymentMPFCmdImpl` になります。このインプリメンテーションは、`Payment Manager` を使用します。

WebSphere Commerce Suite 5.1 がデフォルト・インプリメンテーション・クラス DoPaymentCmdImpl を使用しており、 WebSphere Commerce 5.4 にマイグレーションした後も DoPaymentCmdImpl を使用する場合は、以下のうちのいずれかを行うことができます。

1. このストアで引き続き DoPaymentCmdImpl を使用する場合は、CMDREG テーブルで STOREENT_ID を指定します。 WebSphere Commerce 5.4 のブートストラップ・データは変更されず、新しいバージョンにマイグレーションするにはより適しているため、この方法を優先的に採用してください。たとえば、CMDREG に新規エントリーを挿入し、ストアで、STOREENT_ID=0、interfacename=DoPaymentCmd、classname=DoPaymentCmdImpl を指定します。
2. CMDREG テーブルにあるブートストラップ・データを変更します。 STOREENT_ID=0 となっているデフォルト・ストアに関しては、DoPaymentCmd のデフォルト・インプリメンテーション・クラスを DoPaymentCmdImpl に変更してください。デフォルト・ストアのブートストラップ・データを変更することになるため、これは推奨されていません。

Payment Manager に関するその他の考慮事項については、143 ページの『第 9 章 Payment コンポーネントのマイグレーション考慮事項』を参照してください。

データベース・マイグレーションの検証

このセクションでは、データベースが正常にマイグレーションされていることを検証するためのいくつかのガイダンスを記載しています。

migratedb.log ファイルの表示

マイグレーション・スクリプトを実行した後、migratedb.log ファイルの内容を検討してください。ストリング Migrating を検索して、Total errors=0 が存在するかどうかを調べます。警告は無視することができます。マイグレーションが正常に実行された場合は、migratedb.log ファイルが以下のようなものになっているはずです。

```
Migration starts... Date:2001-10-02
Migrating Member table...
...
...
...
Summary
Total changed =
Total inserted =
Total queries =
Total warnings = 0
Total errors = 0
```

```
Migrating Catalog Entries..
...
...
...
Summary
Total changed =
Total inserted =
Total queries =
Total warnings =
Total errors = 0
```

Migrating Inventory..

...
...

Summary
Total changed =
Total inserted =
Total queries =
Total warnings = 0
Total errors = 0

Migrating Discount Data...

...
...

Summary
Total changed =
Total inserted =
Total queries =
Total warnings = 0
Total errors = 0

Migrating Contract Component..

...
...

Summary
Total changed =
Total inserted =
Total queries =
Total warnings =0
Total errors = 0

Migrating Calculation Framework..

...
...

Summary
Total changed =
Total inserted =
Total queries =
Total warnings =0
Total errors = 0.

追加のチェック

データベース・マイグレーション・ログ・ファイルをチェックした後、以下の SQL ステートメントをデータベースに対して実行してください。

- 次の Select ステートメントを実行して、FLOW テーブルをチェックします。

```
DB2 SELECT * FROM FLOW
```

FLOW テーブルが空である場合は、Mass Loader のビジネス・フロー・データのロードに問題があることを意味します。詳細については、migratedb.log および message.txt ファイルを参照してください。

- 以下の Select ステートメントを実行して、ORGENITY および USERS テーブル内の識別名列 DN をチェックします。

```
DB2 SELECT DN FROM ORGENTITY
```

```
DB2 SELECT DN FROM USERS
```

いずれかの DN エントリーが空であれば、83 ページの『識別名の更新』で説明しているように、データベースの migrateDN スクリプトが実行されなかった可能性があります。

データベースの整合性チェッカーの実行

生成されたマイグレーション・スクリプトをデータベースに対して正常に実行した後、整合性チェッカーを実行して、マイグレーションされたデータベースの状態をチェックします。

チェッカーはレポートを生成して、90 ページの『データベース整合性チェッカーの出力』で説明している項目をリストします。

注: 整合性チェッカー・ツールは、マイグレーションされたデータベースに既知の健全性問題があるかどうかをチェックします。整合性チェッカーを実行してエラー・メッセージを受け取らない場合でも、このことは必ずしも整合性問題がデータベースに存在しないことを保証するものではありません。このツールは考えられるすべてのケースを扱うことはできません。特にカスタマイズされたデータベースの場合はそうです。

DB2 データベース

以下の手順で、DB2 データベースに対して整合性チェッカーを実行します。

1. コマンド・プロンプト・ウィンドウをオープンします。
2. root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、データベースの DB2 インスタンス・ユーザー ID (たとえば db2inst1) に切り替えます。

```
su - db2_instance_user_id
```

非 root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、WebSphere Application Server ユーザー ID に切り替えます (たとえば wasuser)。

```
su - WAS_user
```

3. WebSphere Commerce 5.4 がインストールされている bin サブディレクトリに移動します。たとえば、/opt/WebSphere/CommerceServer/bin。
4. 以下のようにして、整合性チェッカーを、マイグレーションされたデータベース上で実行します。

```
./dbchecker.db2.sh db_name db_userID db_userID_password db2insthome
```

ここで

- *db_name* は、WebSphere Commerce 5.4 データベース・スキーマ・レベルにマイグレーションされた Commerce Suite 5.1 データベースです (たとえば mall)。
- *db_userID* は、マイグレーションされたデータベースに接続するためのユーザー ID です (たとえば mydbuser)。

- `db_userID_password` は、マイグレーションされたデータベースに接続するためのユーザー ID のパスワードです (たとえば `mypasswd`)。
- `db2insthome` は、DB2 インスタンス・ユーザー ID のホーム・ディレクトリー (たとえば `/home/db2inst1/sqllib`) です。

たとえば、マイグレーションされた `mall` データベース上で整合性チェッカーを実行するには、以下のコマンドを使用できます。

```
./dbchecker.db2.sh mall mydbuser mypasswd /home/db2inst1/sqllib
```

- 5. `/opt/WebSphere/CommerceServer/logs/migration` ディレクトリーに生成される `dbchecker.log` をチェックします。続行する前に、スクリプトの使用に際してエラーが起きていないことを確認してください。

データベース整合性チェッカーの出力

整合性チェッカーが以下のリターン・コードを生成する場合、システムのマイグレーション・プロセスを続行する前に、リストされている必須のアクションを実行しなければなりません。そうしないと、WebSphere Commerce 5.4 ランタイムは、マイグレーションされたデータに対して機能しなくなります。

リターン・
コード

説明 / アクション

- | | |
|------------|---|
| 401 | マイグレーションされたデータに、親商品のないアイテムが含まれています。これらの各アイテムに親商品を割り当てなければなりません。アイテムの親商品を作成するには、エントリーを <code>CATENTREL</code> テーブルに追加します。 |
| 402 | マイグレーションされたデータに、複数の親商品を持つアイテムが含まれています。各アイテムに 1 つだけの親商品が割り当てられるようにしなければなりません。複数の親商品を持つとしてフラグが立てられたアイテムから余分の親商品を削除するには、エントリーを <code>CATENTREL</code> テーブルから除去します。 |
| 405 | マイグレーションされた <code>USER</code> テーブルに、組織上の不整合があります。フラグが立てられたアイテムを訂正する必要があります。 <code>BUSPROF</code> テーブルにレコードを持つユーザーの場合、 <code>ORG_ID</code> 列と <code>ORGUNIT_ID</code> 列がヌルでなければ、 <code>ORGENITY</code> テーブルを使用し、組織階層を <code>ORGUNIT_ID</code> から上方向に調べます。これは、最終的に <code>MEMBER_ID</code> にヌルが見つかるか、 <code>ORGENITY_ID</code> と同じ値が見つかるまで続きます。 <code>ORGENITY</code> 内の <code>MEMBER_ID</code> 列の値が、 <code>BUSPROF</code> 内の <code>ORG_ID</code> 列の値と異なっています。 <code>ORGENITY</code> 内の <code>MEMBER_ID</code> 列の値が、 <code>BUSPROF</code> 内の <code>ORG_ID</code> 列の値と同じになるようにしてください。 |
| 500 | <p>マスター・カタログとして指定されているカタログがありません。ストアごとに複数のカタログがある場合、カタログの 1 つがマスター・カタログとして指定されている必要があります。</p> <p>84 ページの『マスター・カタログの割り当て』で説明しているように、マスター・カタログを設計して、<code>choosemc.sql</code> スクリプトを実行することによってマスター・カタログを選択します。</p> |
| 503 | マイグレーションされたデータには、最上位のカタログ・グループ |

が含まれていません。データには、それぞれのマスター・カタログに対して、少なくとも 1 つの最上位のカタログ・グループが含まれていなければなりません。エントリーを CATTOGRP テーブルに追加することによって、最上位のカタログ・グループの関係を追加してください。

整合性チェッカーが以下のリターン・コードを生成する場合、システムのマイグレーション・プロセスを続行する前に、リストされているオプションのアクションを実行することが推奨されていますが、これは必須ではありません。このアクションを実行しない場合、マイグレーションされたデータに対して Product Management ツールなどの WebSphere Commerce 5.4 ツールを使用できなくなります。

**リターン・
コード**

説明 / アクション

- 408** マイグレーションされたデータには、複数の取引位置コンテナの下にあるカタログ・グループが含まれています。カタログ・グループが、必ず複数の取引位置コンテナの下にないようにする必要があります。CATGRPTPC テーブルからエントリーを除去することによって、追加の取引位置コンテナの関係を削除してください。
- 501** マイグレーションされたデータには、複数の親カタログ・グループを持つカタログ・グループが含まれています。各カタログ・グループは、親として 1 つのカタログ・グループだけを指定する必要がある必要があります。CATGRPREL テーブルからエントリーを除去することによって、追加の親カタログ・グループを削除してください。
- 502** マイグレーションされたデータには、複数のカタログ・グループに属するカタログ・エントリーが含まれています。すべてのカタログ・エントリーが、1 つのカタログ・グループだけに属する必要がある必要があります。CATGPENREL テーブルからエントリーを除去することによって、追加のカタログ・グループを削除してください。
- 614** ATTRVALUE テーブルで、CatEntryId = 0 によって定義された属性値ごとに、1 つの行が必要です。たとえば、色が赤 (red) のアイテムが 2 つある場合、ATTRVALUE テーブルは以下のようになります。

CATENTRY_ID	ATTRIBUTE_ID	STRINGVALUE
0	color_id	red
item1_id	color_id	red
item2_id	color_id	red

これは完全なテーブルではないことに注意してください。この方法によって、定義されたアイテムが存在しない場合でも、属性に定義された有効な属性値を表示することができます。複数の言語が存在する場合、言語ごとに各行を再定義する必要があります。

このステップが必要なのは、Product Management ツールなどの WebSphere Commerce 5.4 ツールを使用してカタログを管理したい場合だけです。

複数のアイテムに同じ ATTRIBUTE_ID があるケースごとに、ATTRVALUE テーブルに CATENTRY_ID を 0 (ゼロ) に設定した 1 行を追加します。

Commerce Suite 5.1 のサンプル・ストアのカタログである InFashion や WebFashion には、WebSphere Commerce 5.4 マスター・カタログに適した構造のカタログ・ツリーがありません。9 ページの『重要な構造上の考慮事項』で説明しているマスター・カタログの要件に基づいて、カタログを再設計する必要があります。そのようにしないと、WebSphere Commerce 5.4 のカタログ・エディター・ツールである Product Management で、カタログのナビゲート時に問題が生じる可能性があります。適切な構造のカタログ・ツリーの例については、`/opt/WebSphere/CommerceServer/samples` ディレクトリーにある、WebSphere Commerce 5.4 に同梱されているサンプル・ストアを参照してください。

LANGUAGE_ID のリセット

67 ページの『STORE テーブルおよび populatedb スクリプトのセットアップ』で説明しているとおりに、デフォルトのストアについての STORE テーブルに LANGUAGE_ID を設定している場合、データベース・マイグレーションが完了した後にその元の値にリセットし直す必要があります。たとえば、LANGUAGE_ID が変更前は -1 (en_US) に設定されていた場合、他のマイグレーション・ステップを続行する前に、その設定を以下のようにして戻す必要があります。

```
DB2 UPDATE STORE SET LANGUAGE_ID=-1 WHERE STORE_ID=0
```

データベースの再マイグレーション

データベース・マイグレーションを完了し、何らかの理由でデータベースを再マイグレーションすることが必要になった場合は、元のデータベースへの再マイグレーションを試行する前に、データベース・マイグレーション・スクリプトによって以前に生成されたすべてのファイルが除去されていることを確認してください。

削除するファイルのリストを以下に示します。

- DB2 の場合 — `/opt/WebSphere/CommerceServer/schema/db2/migration` ディレクトリーの下にある以下のファイル。
 - `delwcs.constraint.sql`
 - `wcs.schema.create.sql`
 - `wcs.temp.drop.sql`
- `/opt/WebSphere/CommerceServer/schema` ディレクトリーの下にある以下のファイル。
 - `fillorgDN.sql`
 - `ctos.sql`
 - `choosemc.sql` (ある場合)
- `/opt/WebSphere/CommerceServer/logs/migration` ディレクトリーの下にある以下のファイル。
 - `migratedb.log`
 - `premigrate.log`

- migrateDN.log
- migratedscripts.log
- migrate.log
- dbchecker.log
- OrigSchema.log
- TargetSchema.log

データベースを再マイグレーションする場合、71 ページの『データベース準備スクリプトの実行』で説明しているとおりにデータベース準備スクリプトを再実行することを含め、この章のすべてのステップを実行する必要があります。

注: 95 ページの『第 5 章 データ・マイグレーション後のアクション』にある、WebSphere Commerce インスタンス・マイグレーションおよびマーチャント・キー・マイグレーションを完了した後に、データベースを再マイグレーションする場合、これらのステップを再度実行する必要はありません。ここで示すのは、インスタンスおよびマーチャント・キーを再マイグレーションせずに、データベースを再マイグレーションするためのステップです。

1. データベースを再マイグレーションする前に、上に挙げたすべてのファイルを削除します。
2. マイグレーション済みの WebSphere Commerce 5.4 データベースをバックアップします。
3. WebSphere Commerce 5.4 データベース内のマイグレーション済みの USERREG テーブルをテキスト・ファイルにエクスポートします。このテーブルには、ご使用の WebSphere Commerce システムのユーザーのマイグレーション済みのパスワードが入っています。
4. バックアップ済みの Commerce Suite 5.1 データベースをリストアします。
5. この章のすべてのステップを実行して、リストアした Commerce Suite 5.1 データベースを WebSphere Commerce 5.4 レベルにマイグレーションします。
6. ステップ 3 のテキスト・ファイルから USERREG テーブルをインポートします。このテーブルには、ご使用の WebSphere Commerce システムのユーザーのマイグレーション済みのパスワードが入っています。

第 5 章 データ・マイグレーション後のアクション

以下のセクションでは、データを WebSphere Commerce 5.4 レベルにマイグレーションした後に必要のあるマイグレーション・アクションについて説明します。これには以下が含まれます。

- 『Commerce Suite 5.1 インスタンス構成のマイグレーション』
- 100 ページの『セキュリティー構成のマイグレーション』
- 103 ページの『ストア・ファイル資産のマイグレーション』
- 110 ページの『割引、配送、または課税用のカスタム・コードの更新』

Commerce Suite 5.1 インスタンス構成のマイグレーション

このセクションでは、Commerce Suite 5.1 インスタンス構成を WebSphere Commerce 5.4 インスタンス構成にマイグレーションする方法について説明します。新規の WebSphere Commerce 5.1 インスタンスを作成する方法の詳細については、プラットフォームに対応する *WebSphere Commerce Suite 5.1 Installation Guide* の、構成マネージャーによるインスタンスの作成または変更に関するセクションを参照してください。

インスタンス構成をマイグレーションする前のステップ

注意:

インスタンスをマイグレーションする前に、データベース・スキーマを **WebSphere Commerce 5.4** にマイグレーションしなければなりません。

WebSphere Commerce Suite 5.1 インスタンスを WebSphere Commerce 5.4 にマイグレーションする前に、以下のようにします。

- システムの LANG 環境変数が、使用する言語に正しく設定されていることを確認します。LANG 環境変数の値を表示するには、コマンド・プロンプトから以下のコマンドを入力します。

```
echo $LANG
```

これが正しく設定されていない場合、以下のステートメントを `.profile` ファイルに追加する必要があります。

```
export LANG=locale
```

ここで *locale* は、使用する言語に対応するロケールです。たとえば `en_US` は英語 (米国) です。LANG 環境変数は、インスタンス・マイグレーションを適正に動作させるために正しく設定することが必要です。

- WebSphere Commerce Suite 5.1 は、`instance_name.xml` ファイルで UTF-8 をすべての言語のデフォルトのエンコードとして使用していました。WebSphere Commerce 5.4 の場合、`instance_name.xml` ファイルは言語固有のエンコードが必要です。ご使用の `instance_name.xml` ファイルのエンコードをチェックして、使用する言語に対して正しく設定されていることを確認してください。たとえば、

/opt/WebSphere/CommerceServer/instances/your_instance/xml にある instance_name.xml ファイルを編集して、以下を検索します。

```
<?xml version="1.0" encoding="xxxxxx"?>
```

上記のストリングで、encoding は以下のように設定する必要があります。

言語	エンコード方式
日本語	Shift_JIS
韓国語	euc-Kr
中国語 (簡体字)	GB2312
中国語 (繁体字)	Big5
その他の言語 (英語を含む)	ISO-8859-1

たとえば、使用する言語が日本語の場合は、エンコードを以下のように設定します。

```
<?xml version="1.0" encoding="Shift_JIS"?>
```

- 57 ページの『SSL の使用可能化』の説明に従って、SSL が使用可能になっていることを確認します。
- Commerce Suite 5.1 のカスタマイズ済みファイルおよびディレクトリーを手動でバックアップした場合、ここでこれらのファイルとディレクトリーを、バックアップ場所から適切な WebSphere Commerce 5.4 の場所にコピーしなければなりません。

17 ページの『マイグレーション時のダウン時間の最小化』の手順に従っている場合、上記のファイルに加えて、インスタンス構成ファイルもリストアしなければなりません。

バックアップ場所からファイルとロケーションをリストアするには、以下のようになります。

1. コマンド・ウィンドウをオープンして、一時バックアップ・ディレクトリーにナビゲートします。
 2. 適切なファイルまたはディレクトリーを選択して、それらを適切な WebSphere Commerce 5.4 ディレクトリーにコピーします。
- 以下のアイテムを手動でコピーします。(ご使用の WebSphere Commerce Suite 5.1 システムが、WebSphere Commerce 5.4 マシンからはリモートのマシン上にある場合、以下のアイテムを、WebSphere Commerce Suite 5.1 マシンから WebSphere Commerce 5.4 マシンにコピーする必要があります。)
 - /opt/WebSphere/CommerceSuite/instances にある WebSphere Commerce Suite 5.1 wcs_instances ファイルを、新規 /opt/WebSphere/CommerceServer/instances ディレクトリーにコピーする。
 - /opt/WebSphere/CommerceSuite/instances/your_instance ディレクトリー (そのサブディレクトリーを含む) を、新規 /opt/WebSphere/CommerceServer/instances/your_instance ディレクトリーにコピーする。
 - /opt/WebSphere/CommerceSuite/bin にあるファイル cfg.passwd を、新規 /opt/WebSphere/CommerceServer/bin ディレクトリーにコピーする。
 - /opt/WebSphere/CommerceServer/instances ディレクトリーにある wcs_instances ファイルを編集します。

/opt/WebSphere/CommerceSuite/instances/*your_instance* (WebSphere Commerce Suite 5.1 インスタンスのパス名) があれば、いずれも

/opt/WebSphere/CommerceServer/instances/*your_instance* (WebSphere Commerce 5.4 インスタンスのパス名) に変更する。

- インスタンス・マイグレーションの前に、
/opt/WebSphere/CommerceServer/instances/*your_instance* ディレクトリーにある *instance_name.xml* ファイル (たとえば *demo.xml*) に、以下の更新を加える必要もあります。
 - 非 root ユーザー (**wasuser**) として **WebSphere Application Server** を実行している場合、*instance_name.xml* ファイルで、WebSphere ノードの下の *port* 属性を 900 から 2222 に変更します。
 - WebSphere Commerce 5.4 では、WebSphere Commerce 管理ツールはサーバーとは異なるポート上で実行します。これらのツールが実行するデフォルトのポートは 8000 です。この設定をオーバーライドして管理ツール用に別のポートを使用したい場合は、*instance_name.xml* ファイルで WebSphere ノードを見つけて、*ToolsPort="port_number"* 属性をその属性リストに追加し、*port_number* の値を非デフォルトのポート番号に設定します。
- 注意:**
instance_name.xml ファイルの編集はエラーが発生しやすいので、デフォルト・ポートを指定変更しないことを強くお勧めします。
- /opt/WebSphere/CommerceServer/instances/*your_instance* ディレクトリー内に *instance_name.xml.bak* があればすべて除去します。

重要: WebSphere Application Server セキュリティーがオンになっている場合、インスタンスを作成する前に以下の方法でそれを使用不可にしなければなりません。

1. WebSphere Application Server 管理者コンソールをオープンします。
2. 「コンソール > **Security Center (セキュリティ・センター)**」をクリックして、「一般」タブの「**Enable Security (セキュリティを使用可能にする)**」チェック・ボックスからチェックを外します。
3. 「終了」をクリックします。
4. WebSphere Application Server 管理サーバーを再始動します。

インスタンス構成のマイグレーション

以下のステップを実行して、実行中の各 Commerce Suite インスタンスをマイグレーションする必要があることに注意してください。

Commerce Suite 5.1 インスタンスをマイグレーションするには、以下のようになります。

- 1. WebSphere Application Server を始動します。
- 2. root ユーザーとしてインスタンス・マイグレーションを実行している場合は、必ず root ユーザーとしてログオンしてください。
非 root ユーザーとしてインスタンス・マイグレーションを実行している場合、以下のようにして、WebSphere Application Server ユーザー ID に切り替えます (たとえば *wasuser*)。

```
su - WAS_user
```

- __ 3. コマンド・プロンプトで `/opt/WebSphere/CommerceServer/bin` ディレクトリーに移動して、`instMigration.sh` スクリプトを実行します。
- __ 4. インスタンスのマイグレーションが正常に完了したことを検証するには、`/opt/WebSphere/CommerceServer/instances` ディレクトリー内の `instMigrate.log` ファイルを調べます。

ログの末尾にマイグレーションが正常に完了したことを示す文があり、インスタンスが正常にマイグレーションされたことを示しているはずですが。

さらに、Enterprise JavaBeans (EJB) のデプロイメントが正常に終了したかどうかを調べるには、

`/opt/WebSphere/CommerceServer/instances/instance_name/logs` ディレクトリー内の `WASConfig.log` ファイル (EJB のインポートに関するログ・ファイル) を調べます。例外が発生していないことを確認してください。

Commerce Suite 5.1 で Web サーバー構成をカスタマイズした場合、それが現行の Web サーバー・ファイルにもあることを確認してください。存在しない場合、構成ファイルにカスタマイズ内容を再適用する必要があります。

次のセクションで説明しているように、セキュリティー構成をマイグレーションした後で、Web サーバーと WebSphere Commerce Server - `instance_name` アプリケーション・サーバーを再始動します。

ルール・サービスのオフへの切り替え

Commerce Suite 5.1 でルール・サーバー・コンポーネントを使用不可にしているも、インスタンスのマイグレーション・プロセスは、デフォルトで Commerce ルール・サーバーを使用可能にします。インスタンス・マイグレーションの後に WebSphere Commerce 5.4 内のルール・サーバーを使用不可にするには、`instance>name.xml` ファイル内の `enable` ディレクティブ・ファイルを `true` から `false` に変更します。このファイルは、`/opt/WebSphere/CommerceServer/instances/your_instance` ディレクトリーにあります。

以下の行を見つけて、`enable` ディレクティブを以下の行から更新します。

```
<component enable="true"
  name="Rule Services"
  compClassName="com.ibm.commerce.rules.RuleSystemComponentConfiguration">
```

以下のようにします。

```
<component enable="false"
  name="Rule Services"
  compClassName="com.ibm.commerce.rules.RuleSystemComponentConfiguration">
```

新規ロケーションへのストア資産のコピー

WebSphere Commerce 5.4 にマイグレーションする際、以下のテーブルにあるリストに従っていくつかの WebSphere Commerce Suite ディレクトリーにあるストア資産を WebSphere Application Server ディレクトリーにコピーする必要があります。ストア用の Web アプリケーションを構成する資産は、WebSphere Application Server 4.0.2 で必要となる新しい Web アプリケーションの構造の理由から、コピーする必要があります。

以下の表に示されているソース・ディレクトリーの内容を、対応するターゲット・ディレクトリーに手動でコピーする必要があります。

注: ソース・ディレクトリーおよびターゲット・ディレクトリーは、以下の表の *instance_name.xml* 内にある <devtools> エLEMENTの下のエントリーの値を使って構成されます。

表 5. WebSphere Commerce 5.4 の新規ロケーションにストア資産をコピーする

<i>instance_name.xml</i> ファイル内のエントリー	(WebSphere Commerce Suite 5.1 の) ソース・ディレクトリー (これらのディレクトリーは、デフォルトのインストール・パスにあると想定します。)	(WebSphere Commerce 5.4 の) ターゲット・ディレクトリー (これらのディレクトリーは、デフォルトのインストール・パスにあると想定します。)
StoresDocRoot + StoreWebPath	/opt/WebSphere/CommerceSuite/stores/web	/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear/wcstores.war
StoresDocRoot + StoresPropertiesPath	/opt/WebSphere/CommerceSuite/stores/properties	/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear/wcstores.war/WEB-INF/classes

ストア・アーカイブ・ファイルがストアの URL から無許可で検索され、ストアをロードするために使用されたすべてのアイテムが表示されることがないようにするために、ご使用のマシン上のすべての *store_name.sar* ファイルが /opt/WebSphere/CommerceServer/instances/*your_instance*/sar に移動されたことを確認してください。 sar サブディレクトリーがマイグレーション済みのシステムに存在しない場合は、それを手動で作成する必要があります。ストア・サービスを使用して新規ストアを WebSphere Commerce 5.4 に作成する場合は、ツールによって sar サブディレクトリーが自動的に作成されます。

WebSphere Commerce インスタンスの更新

Infashion または WebFashion などの、マイグレーション済みストアを公開するには、WebSphere Commerce インスタンス・ファイルの DevTools ノード内の TempPath 属性を以下のように変更します。

```
TempPath="/opt/WebSphere/CommerceServer/tools/devtools/temp"
```

以下のようにします。

```
TempPath="/opt/WebSphere/CommerceServer/temp/your_instance/tools/devtools"
```

your_instance.xml ファイルは

/opt/WebSphere/CommerceServer/instances/*your_instance*/xml ディレクトリーにあります。

Web サーバーおよび WebSphere Application Server の再始動

インスタンスのマイグレーションが完了した後に、以下の方法で Web サーバーおよび WebSphere Application Server を再始動します。

1. WebSphere Application Server を停止します。
2. Web サーバーを停止します。
3. Web サーバーを再始動します。
4. WebSphere Application Server を再始動します。
5. WebSphere Application Server コンソールで、WebSphere Commerce Server - *instance_name* アプリケーション・サーバーを再始動します。

WebSphere Application Server EJB セキュリティーの使用可能化

WebSphere Commerce 5.1 で WebSphere Application Server EJB セキュリティーを使用可能にした場合、それを WebSphere Commerce 5.4 で再度使用可能にする必要があります。EJB セキュリティーを再デプロイするためのステップについては、*WebSphere Commerce インストール・ガイド バージョン 5.4* のセクション

『WebSphere Application Server セキュリティーを使用不可にする』を参照してください。

ご使用のマシンが以下の要件を満たしていることが強く推奨されていることに注意してください。

- 最低でも 1 GB のマシン・メモリー
- WebSphere Commerce アプリケーションの場合、最低でも 384 MB のヒープ・サイズ

インスタンスが正常にマイグレーションされたことの検証

インスタンスが正常にマイグレーションされたことを検証するには、

`/opt/WebSphere/CommerceServer/instances` ディレクトリー内の `instMigrate.log` ファイルを調べます。さらに、WebSphere Commerce サーバーをエラーなしで開始できることを確認します。エラーが発生した場合、それは

`/opt/WebSphere/CommerceServer/instances/instance_name/logs` ディレクトリーの `wcs.log` ファイルに記録されます。

セキュリティー構成のマイグレーション

Commerce Suite 5.1 のインスタンス構成を WebSphere Commerce 5.4 レベルにマイグレーションした後、マーチャント・キー・マイグレーション・ユーティリティー `MigrateEncryptedInfo.sh` を実行する必要があります。このユーティリティーは、以下を行います。

- 指定したインスタンスのマーチャント・キーを新規の鍵に変更して、関連付けられた構成ファイルに応じてストア・データベース内の暗号化データを更新します。WebSphere Commerce 5.4 では、デフォルト以外のマーチャント・キーを使用する必要があります。Commerce Suite 5.1 でデフォルトのマーチャント・キーを使用する場合は、このデフォルト鍵をデフォルト以外の鍵に変更する必要があります。
- ログオン・パスワードがデータベースに保管される方法を変更して、WebSphere Commerce 5.4 の要件に適合するようにします。
- `instance.xml` ファイルの `PDIEncrypt` フラグの設定に基づいて、暗号化されていないすべてのクレジット・カード・データは暗号化され、暗号化されているすべてのクレジット・カード・データは暗号化解除されます。

更新される暗号化データには、暗号化されたパスワードおよびクレジット・カード・データが含まれます。ユーティリティーは以下の表にある暗号化データを更新します。

- USERREG
- PATTRVALUE
- ORDPAYINFO
- ORDPAYMTHD

暗号化設定の確認

マーチャント・キーのマイグレーション・ユーティリティーを実行する前に、WebSphere Commerce インスタンスに対する *your_instance.xml* ファイルでの PDI 暗号化の設定が正しく設定されていることを確認してください。

your_instance.xml ファイルは、
`/opt/WebSphere/CommerceServer/instances/your_instance/xml` ディレクトリーにあります。*your_instance.xml* ファイルを編集し (たとえば `demo.xml`)、ストリング `PDIEncrypt` を検索し、その値を "on" または "off" のいずれかに設定します。

PDIEncrypt="on"

クレジット・カード・データなどの機密情報は、マーチャント・キーを使用して暗号化された形式で保管されます。

PDIEncrypt="off"

クレジット・カード・データなどの機密情報は、プレーン・テキスト形式 (暗号化されていない) で保管されます。

DB2 データベース

`MigrateEncryptedInfo.sh` ユーティリティーを実行する前に、それを実行するユーザー ID (つまり `db2inst1` または `wasuser`) が、
`/opt/WebSphere/CommerceServer/instances/your_instance` ディレクトリー内のファイルおよびサブフォルダーに対する完全アクセス権を持っていることを確認します。

`MigrateEncryptedInfo` ユーティリティーを以下のように実行します。

- __ 1. WebSphere Commerce Server - *instance_name* アプリケーション・サーバーを、WebSphere Application Server コンソールから停止します。
- __ 2. コマンド・プロンプト・ウィンドウをオープンします。
- __ 3. root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、データベースの DB2 インスタンス・ユーザー ID (たとえば `db2inst1`) に切り替えます。

```
su - db2_instance_user_id
```

非 root ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、WebSphere Application Server ユーザー ID に切り替えます (たとえば `wasuser`)。

```
su - WAS_user
```

- __ 4. WebSphere Commerce 5.4 インストール・ディレクトリーの下の `bin` サブディレクトリーに移動します。たとえば、以下のとおりです。

```
/opt/WebSphere/CommerceServer/bin
```

- __ 5. DB2 に対して `MigrateEncryptedInfo` ユーティリティーを以下のように実行します。

```
./MigrateEncryptedInfo.sh db2 [instance_name] [current_key] [new_key]
```

ここで

- *instance_name* は、更新されるインスタンスの名前です。インストールされているインスタンスが 1 つしかない場合は、このパラメーターを省略することができます。

- *current_key* は、制御文字がないテキスト (ASCII) 形式による、現在のマーチャント・キーです。このパラメーターを指定する必要があるのは、現在デフォルト以外の Commerce Suite 5.1 マーチャント・キーを使用している場合だけです。デフォルトの Commerce Suite 5.1 マーチャント・キーを使用している場合、デフォルト鍵を使用していることをユーティリティーが検出するので、このパラメーターは指定しないでください。
- *new_key* は、制御文字がないテキスト (ASCII) 形式による、新規のマーチャント・キーです。これは、以下の規則に準拠していなければなりません。
 - 長さが 16 進文字で 16 文字であること。使用可能な文字は、0、1、2、3、4、5、6、7、8、9、a、b、c、d、e、または f です。
 - 最低 1 つの英字を含むこと。
 - 最低 1 つの数字を含むこと。
 - 小文字であること。
 - 同一の文字を連続して 5 回以上使用しないこと。

たとえば、aaaa1aaaa1aaaa12 や abcdeaaaa3aaaa12 は使用できますが、aaaaabaaaa1aaaa1 は使用できません。

異なるシナリオでのコマンドの指定方法については、以下のセクションを参照してください。

- ___ 6. Web サーバーおよび WebSphere Application Server を再始動します。
 - a. WebSphere Application Server を停止します。
 - b. Web サーバーを停止します。
 - c. Web サーバーを再始動します。
 - d. WebSphere Application Server を再始動します。
- ___ 7. WebSphere Application Server コンソールで、WebSphere Commerce Server *instance_name* アプリケーション・サーバーを開始します。

このツールによって、/opt/WebSphere/CommerceServer/logs ディレクトリーに以下の 2 つのログ・ファイルが生成されます。

- CCInfoMigration.log
- MKChangeUserAndCCInfoMigration.log

これらのログ・ファイル内の情報を見て、エラー・メッセージが含まれていないことを確認する必要があります。

このユーティリティーを使用してマーチャント・キーを更新できるのは、マイグレーション時だけであることに注意してください。後にマイグレーションが終了してからマーチャント・キーを変更したい場合、構成マネージャーを使用して鍵を更新します。構成マネージャーの使用の詳細については、WebSphere Commerce 5.4 オンライン・ヘルプ・セクションを参照してください。

DB2 データベースの例

以下の例で、インスタンス名 `myinstance` が必要となるのは、複数の Commerce Suite 5.1 インスタンスが存在する場合だけです。

- デフォルトの Commerce Suite 5.1 マーチャント・キーを使用している場合、および `myinstance` と呼ばれるインスタンスがある場合、新規の鍵だけを指定します。

```
./MigrateEncryptedInfo.sh db2 myinstance abcdef0123456789
```

- デフォルト以外の Commerce Suite 5.1 マーチャント・キーを使用している場合、つまりすでにマーチャント・キーを固有のものに変更した場合に、鍵を WebSphere Commerce 5.4 用に変更するためには、現行鍵と新規の鍵の両方を指定します。

```
./MigrateEncryptedInfo.sh db2 myinstance 0123456789abcdef  
abcdef0123456789
```

- デフォルト以外の Commerce Suite 5.1 マーチャント・キーを使用している場合、つまりすでにマーチャント・キーを固有のものに変更した場合に、鍵を WebSphere Commerce 5.4 用に更新しない ためには、どちらの鍵も指定しません。

```
./MigrateEncryptedInfo.sh db2 myinstance
```

(現行鍵と新しい鍵に同じ値を指定した場合、そのことを示すエラー・メッセージを受け取るので注意してください。)

ストア・ファイル資産のマイグレーション

このセクションでは、公開済みの Commerce Suite 5.1 ストアを WebSphere Commerce 5.4 にマイグレーションする方法について説明します。

JavaServer Pages を変更する前に、以下が実行済みであることを確認してください。

- Commerce Suite 5.1 ストア・アーカイブを公開したこと。ストア・アーカイブを公開するための詳細ステップについては、Commerce Suite 5.1 オンライン・ヘルプのセクション『ストア・アーカイブの発行』を参照してください。
- 67 ページの『第 4 章 データベースのマイグレーション』に従って、データベース・スキーマをマイグレーションします。
- 95 ページの『Commerce Suite 5.1 インスタンス構成のマイグレーション』に従って、インスタンスを WebSphere Commerce 5.4 レベルにマイグレーションします。

これによって、データベース内のストア・データが WebSphere Commerce 5.4 に自動的にマイグレーションします。

上記の操作を完了した後、示されたサンプル・ストアに基づいて新規の WebSphere Commerce 5.4 ストア・アーカイブを作成する必要があります。 WebSphere Commerce 5.4 オンライン・ヘルプのセクション『ストアの作成』および『ストア・アーカイブの作成』を参照してください。古い Commerce Suite 5.1 ストア・アーカイブは、WebSphere Commerce 5.4 のストア・アーカイブ・ツールでは処理されません。

重要

- `index.jsp` は、WebSphere Commerce 5.4 で新しく使用されるようになったファイルなので、マイグレーションされたストアにはこれがありません。そのため、WebSphere Commerce ストア・サービスの「launch store (ストアの立ち上げ)」ボタンは、マイグレーションされたストアに対しては機能しません。ストア・サービスの「launch store (ストアの立ち上げ)」ボタンを使用してストアを立ち上げるには、独自の `index.jsp` を作成する必要があります。 `index.jsp` ファイルの例については、WebSphere Commerce 5.4 で提供されているサンプル・ストアを参照するとともに、付属のサンプル・ストアについて説明している WebSphere Commerce オンライン・ヘルプを参照してください。
- マイグレーションの前に Commerce Suite 5.1 で使用した Web アドレスによって、ストアを立ち上げることができます。たとえば、以下のようにします。

```
http://hostname/webapp/wcs/stores/servlet/StoreCatalogDisplay?  
storeId=storeId&langID=-1&catalogId=catalogId
```

`storeId` はストアのストア ID 番号、`catalogId` はカタログ番号です。詳細については、WebSphere Commerce オンライン・ヘルプの『コマンド行を使用したストア・アーカイブの公開』を参照してください。

- WebSphere Commerce 5.1 または WebSphere Commerce Business Edition 5.1 の Java または Enterprise JavaBeans オブジェクトで作成されたコードまたはコマンドをカスタマイズした場合、それらを WebSphere Commerce 5.4 で求められるレベルに再デプロイする必要があります。 *WebSphere Commerce Studio* マイグレーション・ガイド バージョン 5.4 の『カスタマイズ・コードの遷移』のセクションを参照してください。この遷移は、IBM WebSphere Application Server 3.5 から WebSphere Application Server 4.0.2 への移動を行うために必要です。

いくつかの EJB をカスタマイズした場合、JNDI 名が、マイグレーション後にカスタマイズされたコマンドで呼び出される名前と同じであることを確認してください。

ご使用の JSP またはカスタマイズ済みコードが以前に IBM WebSphere Application Server 3.5.x からの `com.ibm.util` パッケージを使用していた場合、コードまたは JSP を、WebSphere Application Server 4.0.2 に同梱されている IBM SDK for Java からの同等のクラスを使用して再作成する必要があります。 `com.ibm.util` パッケージは、現行バージョンの WebSphere Application Server からは除去されています。

Commerce Suite 5.1 JavaServer Pages の更新

Commerce Suite 5.1 JavaServer Pages (JSP) が WebSphere Commerce 5.4 で機能するようになるには、いくつかの変更を加える必要があります。これを自動的に実行するツールとして、`migrateJSP.sh` が備えられています。このツールを実行するには、以下のようにします。

1. コマンド・ウィンドウをオープンします。
2. `root` ユーザーとして WebSphere Application Server を実行している場合は、`root` ユーザーとしてスクリプトを実行してください。

非 `root` ユーザーとして WebSphere Application Server を実行している場合、以下のようにして、WebSphere Application Server ユーザー ID に切り替えます (たとえば `wasuser`)。

```
su - WAS_user
```

3. `/opt/WebSphere/CommerceServer/bin` ディレクトリーに移動します。
4. `migrateJSP` スクリプトを以下のように実行します。

```
./migrateJSP.sh inputDir outputDir
```

ここで

- `inputDir` は、Commerce Suite 5.1 JSP が存在するディレクトリーの完全修飾名です。
- `outputDir` は、変換後の WebSphere Commerce 5.4 JSP が置かれるディレクトリーの完全修飾名です。

たとえば、以下のようにします。

```
./migrateJSP.sh /my_wcs_jsp_dir /my_wcbe_jsp_dir
```

注: JavaServer Pages に対するこのツールの実行は一度だけに行ってください。2 度以上実行すると、JSP で構文エラーが生じる場合があります。たとえば、初回にスクリプトを実行する際は、JSP は `<jsp:include page="<%=incfile%>"/>` から `<jsp:include page="<%=incfile%>" flush="true"/>` へ正しく更新されますが、もう一度このスクリプトを実行すると、`<jsp:include page="<%=incfile%>" flush="true" flush="true"/>` のように更新されてしまいます。

上記のツールが JSP に対して自動的に加える、必須の変更点を以下に示します。このリストは、InFashion ストアの Commerce Suite 5.1 から WebSphere Commerce 5.4 へのマイグレーション時のものです。

- WebSphere Commerce 5.1 は JavaServer Page 1.1 レベルを使用するので、以下のすべての出現を変更する必要があります。

```
<jsp:include page="<%=incfile%>"/>
```

これを次のように変更します。

```
<jsp:include page="<%=incfile%>" flush="true"/>
```

Commerce Suite 5.1 で実行されるストアで使用される JavaServer Page テンプレートは、JavaServer Page 1.0 の仕様をサポートすることが求められていました。ストアを WebSphere Commerce 5.4 にマイグレーションする場合、そのストアの JavaServer Page テンプレートが、Sun Microsystems によって作成された

JavaServer Page 1.1 の仕様に準拠していることを確認する必要があります。
JavaServer Page 1.1 の仕様については、<http://java.sun.com> にある
Sun Microsystem の Java Web サイトを参照してください。

- 以下のリンク (存在する場合) のすべての出現を変更します。

```
<Form NAME="BillAddressForm" METHOD="POST" action="<%= "OrderCopy"%>">
```

または

```
<Form NAME=BillAddressForm METHOD="POST" action="<%= "OrderCopy"%>">
```

これを次のように変更します。

```
<Form NAME="BillAddressForm" METHOD="POST" action="OrderCopy">
```

Commerce Suite 5.1 JSP に対して加えることができる、追加の変更点のリストを以下に示します。これらの変更はオプションです。

- いくつかの Commerce Suite 5.1 コマンドは、そのコマンドの現在の WebSphere Commerce 5.4 バージョンに置き換えることができます。これらの変更は必須ではありませんが、適当な時期に実行することをお勧めします。

- `getCalculatedPrice()` メソッドのすべての出現を、
`getCalculatedContractPrice()` メソッドに置き換えることができます。これらのメソッドは、`ItemDataBean` および `ProductDataBean` クラスで生じます。これらのメソッドおよびクラスの詳細については、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。

- `CatalogEntryMPE Bean` の使用法を `EMarketingSpot Bean` に更新できます。たとえば、`InFashion` ストアの Commerce Suite 5.1 `newarrivals.jsp` で、`CatalogEntryMPE Bean` が以下のように使用されていたとします。

```
<%  
// create the e-Marketing Spot  
CatalogEntryMPE productSpot = new CatalogEntryMPE();  
  
//LOOK: Set the right spot name.  
productSpot.setName("NewArrivalsPage");  
productSpot.setMaximumNumberOfItems(new Integer(20));  
  
//Set the default list of promoted products to the  
//contents of the HOMEPAGE_PROMO category.  
List defaultCatalogEntryIdList = new ArrayList();  
if (newArrivalCategoryId != null )  
{  
    CategoryDataBean subCategories[];  
%>
```

`WebFashion` ストアの WebSphere Commerce 5.4 `newarrivals.jsp` では、この同じ機能が以下ようになります。

```
<!-- START PROMO -->  
<%  
// create the e-Marketing Spot  
EMarketingSpot eMarketingSpot = new EMarketingSpot();  
  
// IMPORTANT - set the correct name here  
eMarketingSpot.setName("StoreHomePage");  
  
// instantiate the bean  
DataManager.activate(eMarketingSpot, request);  
  
EMarketingSpot.CatalogEntry[] productResults = eMarketingSpot.getCatalogEntries();  
if (productResults != null && productResults.length > 0)  
{
```



```

for (int i = 0; i < productResults.length; i++) {
EMarketingSpot.CatalogEntry catalogBean = productResults[i];
CatalogEntryDescriptionAccessBean catalogDescriptionBean =
catalogBean.getDescription();

```

*/>

- UsablePaymentTCListDataBean を ProfileCassetteAccountDataBean の代わりに使用し、 UserRegistrationDataBean.findUser() を UserRegistrationDataBean.getRegisterType() の代わりに使用して、ユーザー・タイプをチェックすることができます。

新規の WebSphere Commerce 5.4 ストアの作成方法を知るためには、 WebSphere Commerce 5.4 オンライン・ヘルプの『ストアの作成』を参照してください。

shipaddress.jsp に対する変更

マイグレーション済みストアを WebSphere Commerce 5.4 ストア・サービスを使用して立ち上げたい場合、そしてショッパーがストア内でショッピング・フローを完了するようにしたい場合は、 Commerce Suite 5.1 に同梱されている shipaddress.jsp に対して以下の変更を加える必要があります。

Commerce Suite 5.1 shipaddress.jsp から以下の行を見つけて変更します。

```
if ( !addr.getAddress1().equals("-"))
```

以下のようにします。

```
if (addr.getAddress1()!=null && !addr.getAddress1().equals("-"))
```

以下のディレクトリーにある shipaddress.jsp を更新します。

```
/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear/wcstores.war/store_dir
```

store_dir は、ストアのディレクトリーです (たとえば、webfashion1)。

ユーザー独自のアプリケーションで Commerce Suite 5.1 shipaddress.jsp からコードを使用している場合、それに対応した変更を加える必要があります。

ストア・サービスなどのツールを実行するために使用したすべてのブラウザーをシャットダウンした後、ショッパーは、クリーンな (つまり新規に立ち上げた) ブラウザーからストアを立ち上げる必要があります。

register.jsp および account.jsp に対する変更

サンプルの InFashion ストア用として Commerce Suite 5.1 に同梱されていた register.jsp および account.jsp JSP は、ストアへのログインが失敗した場合でもエラーを戻しませんでした。

顧客がログオンに失敗したときにストアがエラー・コードを戻すようにするには、 register.jsp および account.jsp を更新する必要があります。さらに、 infashiontext_en_US.properties ファイルも更新する必要があります。

更新済みの JSP およびプロパティー・ファイルのリストについては、以下を参照してください。

- 206 ページの『register.jsp』
- 211 ページの『account.jsp』
- 215 ページの『infashiontext_en_US.properties』

Commerce Suite 5.1 に同梱されていた元のファイルと比較して、 WebSphere Commerce 5.4 用のこれらの JSP に必要な変更は、太字フォントで示しています。

ユーザー独自のアプリケーションで Commerce Suite 5.1 の register.jsp および account.jsp からのコードを使用している場合、それに対応したコードの変更を加える必要があります。

サンプルの WebFashion ストアのユーザー登録

Commerce Suite 5.1 Web サイトからダウンロードして使用可能なサンプルの Commerce Suite 5.1 WebFashion ストアでは、提供される register.jsp に次の変更を加える必要があります。コマンド RegisterNAddToMemberGroup をコマンド UserRegistrationAdd に置き換えてください。(migrateJSP.sh スクリプトを実行すると、このコマンドは自動的に更新されることに注意してください。)
UserRegistrationAdd コマンドの使用法および構文についての情報は、 WebSphere Commerce オンライン・ヘルプを参照してください。

この変更が必要なのは、RegisterNAddToMemberGroup コマンドが owner_id を 0 (ゼロ) の値にハードコーディングするためです。 WebSphere Commerce 5.4 では、owner_id は実際には -2001 の値になります。この変更を行わないと、新規のユーザーを WebSphere Commerce 5.4 に登録することができません。

上記の変更を register.jsp ファイルに加えた後、登録ユーザーに割引が適用されるようにするには、顧客プロファイルを変更する必要があります。 Commerce Accelerator を使用して、以下のようにしてマイグレーションされた WebFashion ストアの register10 顧客プロファイルを変更します。

1. WebSphere Commerce Accelerator にログインし、WebFashion ストアを選択します。
2. 「マーケティング」>「顧客プロファイル」の順に選択します。
3. **register10** を選択して、右のナビゲーション・バーで「変更」をクリックします。
4. 「顧客プロファイルの変更」ページが表示されます。左のナビゲーション・バーで、「登録」>「登録状況」の順に選択します。
5. 登録状況を「登録済み」に設定します。
6. 画面の右下の角にある「OK」をクリックして、register10 顧客プロファイルを変更します。

Commerce Suite 5.1 では、新規のユーザーが Commerce Suite 5.1 WebFashion ストアに登録されると、RegisterNAddToMemberGroup コマンドがそのユーザーを、メンバー・グループ register10 の下に自動的に割り当てます。そのため、Commerce Suite 5.1 WebFashion ストアのすべての登録済みユーザーは、MBRGRPMBR テーブルの register10 メンバー・グループに属します。register.jsp ファイルを変更した後は、マイグレーションされた WebFashion ストアに新規に登録されたユーザーは、register10 メンバー・グループに属さなくなります。

ユーザー独自のアプリケーションで Commerce Suite 5.1 WebFashion register.jsp 内のコードを使用している場合、それに対応したコードの変更を加える必要があります。

ストア・プロパティ・ファイルの更新

マイグレーションされた Commerce Suite 5.1 ストアでは、ストア固有のバージョンの以下のプロパティ・ファイルが使用されていることがあります。

- 各国語の `Address.properties` または `Address_locale.properties` (英語用の `Address_en_US.properties` など)
- 各国語の `UserRegistration.properties` または `UserRegistration_locale.properties` (英語用の `UserRegistration_en_US.properties` など)

これらのストアでは、これらのファイルを除去して WebSphere Commerce 5.4 システムのデフォルトのプロパティが使用されるようにする必要があります。Address および UserRegistration プロパティの形式は WebSphere Commerce 5.4 で変更されており、WebSphere Commerce Accelerator ツールは新しいプロパティ・ファイルに対してのみ動作します。

以下のようにして、ストアのための 2 セットのプロパティ・ファイル、UserRegistration および Address を更新できます。

1. `/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear/wcstores.war/WEB-INF/classes/store_dir` ディレクトリーにあるすべての `Address_*.properties` ファイル (つまり、`Address_` で始まるプロパティ・ファイル) をバックアップします。
2. すべての `Address_*.properties` ファイルを `/opt/WebSphere/CommerceServer/samples/properties` ディレクトリーから `/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear/wcstores.war/WEB-INF/classes/store_dir` ディレクトリーにコピーします。

UserRegistration または Address プロパティ・ファイルをカスタマイズした場合、それらを新しい形式に変換する必要があります。これらの新規バージョンのプロパティを作成する方法については、WebSphere Commerce 5.4 オンライン・ヘルプで、クラス `PropertyResourceBundleReader` の説明を参照してください。

新規のサンプル・ストア・ディレクトリー

WebSphere Commerce Suite 5.1 では、`samples` サブディレクトリー (`/opt/WebSphere/CommerceServer/samples`) にサンプル・ストアのストア・アーカイブ・ファイルを保管しました。WebSphere Commerce 5.4 ストア・サービス・ツールが Commerce Suite 5.1 ストアを表示するようにするには、`.sar` ファイルを古い `samples` サブディレクトリーから、新規の `sample store` サブディレクトリーに移動する必要があります。

ストア・サービスで非公開のストア状況

ストアをマイグレーションした後、以前に Commerce Suite 5.1 で公開されたマイグレーション済みストアの状況は、ストア・サービス・ツールにおいて `not published` (公開されていない) と誤って表示されます。WebSphere Commerce 5.4 にマイグレーションする場合、データベース・マイグレーション・スクリプトは、SCHSTATUS テーブルで SCSSTATE が 'C' (complete (完了)) に設定されているすべてのエントリーを消去します。データベース・マイグレーション・スクリプトは、SCHSTATUS テーブルにある保留または未完了状態の他のすべてのエントリーを

SCHACTIVE テーブルに移動させます。以前に完了と設定されていたエントリーが SCHACTIVE テーブルに移動されないと、マイグレーション後にストア・サービスではストアの状態を not published (公開されていない) と表示します。

ストアは正しく機能し続けることに注意してください。

すべてのキャッシングをオフに切り替える

キャッシングを使用可能にした場合、ストアを公開する前にそれを使用不可にしてください。キャッシングをオフに切り替えるには、構成マネージャーを使用して、*instance.xml* 内の構成を更新します。

キャッシングを使用不可にしないと、*message.txt* ログで「Transaction log for the database is full (データベースのトランザクション・ログがフルです)」というメッセージを受け取ることがあります。この場合、217 ページの『付録 H. トラブルシューティング』にある、この問題に対する推奨の解決策を参照してください。

割引、配送、または課税用のカスタム・コードの更新

割引、配送、または課税用のカスタム・コードがある場合、以下の手順を使用して、計算コードを WebSphere Commerce 5.4 にマイグレーションできます。

1. 割引、配送、または課税用のカスタム・コードがあることを確認します。

以下の照会を使用して、CMDREG テーブルをチェックします。

```
SELECT * FROM CMDREG WHERE INTERFACENAME LIKE '%.ApplyOrder%Cmd'
```

割引、配送、または課税サブシステム用のカスタム・コードがある場合、結果セットはヌルになりません。CMDREG テーブルの列 CLASSNAME から、コマンドによって参照されるクラス名をチェックします。

2. カスタマイズ・コードをマイグレーションする必要があるかどうかを確認します。

ApplyOrderXXXCmd タスク・コマンドのカスタム・インプリメンテーションをチェックして、それらが Commerce Suite 5.1 のデフォルトのインプリメンテーションを拡張または参照しているかどうかを調べます。対象となるのは以下のものです。

- ApplyOrderAdjustmentCmdImpl
- ApplyOrderShippingChargesCmdImpl
- ApplyOrderTaxesCmdImpl
- ApplyCalculationUsagesCmdImpl — 基本インプリメンテーション

これらが上記のインプリメンテーションを拡張または参照していない場合、ApplyOrderXXXCmd タスク・コマンドは WebSphere Commerce 5.4 でも動作します。下記のタスク・コマンドのマイグレーションは、将来の互換性を維持するために引き続き推奨されていることに注意してください。

3. カスタム・コードをマイグレーションします。
 - a. カスタム・コードの計算使用法 ID を確認します。

配送	-2
消費税	-3
配送税	-4

- b. WebSphere Commerce 5.4 での対応するデフォルトのインターフェースおよびインプリメンテーションを確認します。

以下の照会を使用して、CALMETHOD テーブルをチェックします。

```
SELECT * FROM CALMETHOD
WHERE STOREENT_ID=-1 AND
CALUSAGE_ID=calculationUsageID AND
SUBCLASS=12
```

例として割引の場合の

com.ibm.commerce.order.calculation.ApplyCalculationUsageCmd などの、タスク・コマンド名に注目してください。対応するインプリメンテーションは、必ずタスク・コマンド名に Impl が付加されたものになります。たとえば割引の場合は

com.ibm.commerce.order.calculation.ApplyCalculationUsageCmdImpl となります。

- c. ステップ 3b のインターフェースを拡張する新規インターフェースを作成します。たとえば、以下のようになります。

```
package mypackage;
import com.ibm.commerce.order.calculation.*;
public interface MyApplyDiscountCmd extends ApplyCalculationUsageCmd {
    public static final String NAME = "mypackage.MyApplyDiscountCmd";
    public static final String_defaultCommandClassName = NAME + Impl;
}
```

String_defaultCommandClassName 変数は、カスタム・インプリメンテーションの名前 (この例では mypackage.MyApplyDiscountCmdImpl) に一致している必要があることに注意してください。

- d. ステップ 3b のインプリメンテーションを拡張し、ステップ 3c のインターフェースをインプリメントするために、カスタム・インプリメンテーションを変更します。たとえば、以下のようになります。

```
package mypackage;
import com.ibm.commerce.order.calculation.*;
public class MyApplyDiscountCmdImpl extends
    ApplyCalculationUsageCmdImpl implements MyApplyDiscountCmd {
}
```

- e. WebSphere Commerce 5.4 インターフェース・シグニチャーと一致するように、カスタム・インプリメンテーションを変更します。

WebSphere Commerce 5.4 の setItems() メソッドは、Commerce Suite 5.1 の setOrderItems() メソッドとおおむね同等であり、setItems() のデフォルトのインプリメンテーションをオーバーライドする必要はありません。Commerce Suite 5.1 とは異なり、performExecute() メソッドでは、保護メソッド getItem() を呼び出すことによって (オーダー) アイテムのリストを検索できるので、オーバーライドは必要なくなりました。

Item クラスは、OrderItemAccessBean クラスのラッパーです。

- OrderItemAccessBean インスタンス (またはオブジェクト) をラップするには、以下のようになります。

```
Item item = new Item(abOrderItem);
```

- OrderItemAccessBean インスタンス (またはオブジェクト) をアンラップするには、以下のようにします。

```
OrderItemAccessBean abOrderItem = item.getOrderItem();
```

- カスタマイズ・コードを、OrderItemAccessBean とではなく、Item と対話させることをお勧めします。
- 以下のようにして、Item に加えられた変更をコミットして、それらが下層の OrderItemAccessBean および EJB キャッシュによってピックアップできるようにします。

```
item.commit();
```

- OrderItemAccessBean を直接更新することを選択した場合、item.refresh() を呼び出して、Item が下層の OrderItemAccessBean インスタンス (またはオブジェクト) と確実に同期するようにします。
- WebSphere Commerce 5.4 の setCurrency() メソッドは、Commerce Suite 5.1 の setOrderCurrency() メソッドとおおむね同等です。(オーダー) 通貨は、保護メソッド getCurrency() を呼び出すことによって検索できません。

詳細については、上記の被参照クラスおよびインターフェースの WebSphere Commerce 5.4 オンライン・ヘルプを調べてください。

4. カスタム・インターフェースを CALMETHOD テーブルに登録します。以下の INSERT ステートメントは、この実行方法の一例です。

```
INSERT INTO CALMETHOD
(CALMETHOD_ID, STOREENT_ID, CALUSAGE_ID, TASKNAME, DESCRIPTION, SUBCLASS, NAME)
VALUES
(calculationMethodID, storeID, -1, 'interfaceName', 'my method to apply discount',
12, 'custom ApplyCalculationUsage (discount)')
```

ここで

- *calculationMethodID* は、たとえば 123456 などの、他の既存の *calculationMethodId* と競合しない任意の正数です。
 - *interfaceName* は、ステップ 3c (111 ページ) からのもので、たとえば *mypackage.MyApplyDiscountCmd* です。
5. STENCALUSG テーブルを更新して、この計算メソッドを参照するようにします。以下の INSERT ステートメントは、この実行方法の一例です。

```
INSERT INTO STENCALUSG
(STOREENT_ID, CALUSAGE_ID, ACTCC_CALMETHOD_ID, ACTRC_CALMETHOD_ID,
CALCODE_ID, CALMETHOD_ID_APP, CALMETHOD_ID_SUM, CALMETHOD_ID_FIN,
USAGEFLAGS, CALMETHOD_ID_INI, SEQUENCE)
VALUES
(storeID, -1, -1, -5, null, calculationMethodID, -203, null, 1, -201, 1.0)
```

ここで *calculationMethodID* は、ステップ 4 からのもので。

マイグレーションの検証

WebSphere Commerce 5.1 インスタンスを構成した後、*WebSphere Commerce Business Edition* インストール・ガイド バージョン 5.1 の『インストールの検証とトラブルシューティング』セクションで説明されている検証ステップを実行して、すべての WebSphere Commerce 5.4 コンポーネントが正常に作動していることを確認してください。

第 6 章 データ・マイグレーション後の追加のアクション

以降のいくつかのセクションで説明するマイグレーション・アクションは、データを WebSphere Commerce 5.4 レベルにマイグレーションした後に行うもので、要件に応じて行うかどうか決定できます。これには以下が含まれます。

- 『ルール・サーバー構成のマイグレーション』
- 121 ページの『オークション』
- 121 ページの『ビジネス・アカウント』
- 122 ページの『カスタマイズしたデータベース・テーブル用のステージング・サーバーの再構成』
- 123 ページの『使用されなくなった Commerce Suite 5.1 テーブルの除去』
- 123 ページの『新しいキャッシュ・トリガーのロード』
- 124 ページの『カスタマイズしたプロパティの改良』
- 125 ページの『配送計算コード』
- 125 ページの『商品および在庫の検索』
- 126 ページの『カスタマイズ・コマンド』
- 127 ページの『商品アドバイザーのマイグレーション考慮事項』

ルール・サーバー構成のマイグレーション

以下の場合、このセクションを省略することができます。

- Commerce Suite 5.1 でルール・サービスを構成していない。
- すべてのルール・サービスが「キャンペーン」ツールによって作成された。キャンペーンのマイグレーションは、本書の前のマイグレーション・ステップを完了した結果として、すでに完了しています。

現在 `/opt/WebSphere/CommerceServer/instances/your_instance/xml/rules` ディレクトリーにあるファイル `wcs.server` は、Advisor ルール・サーバーを構成するために Commerce Suite 5.1 で使用されていました。WebSphere Commerce 5.4 では、この構成情報はデータベースに保管されています。

Commerce Suite 5.1 で構成されたのと同じようにルール・サービスを構成するには、以下を実行してください。

1. 95 ページの『Commerce Suite 5.1 インスタンス構成のマイグレーション』で説明されているとおりにインスタンスをマイグレーションしたかを確認します。
2. `wcs.server` ファイルを表示するために、テキスト・エディターでそのファイルをオープンします。これは XML 形式のファイルで、以下のようなものです。

```
<?xml version="1.0" ?>
<DeployRulesServerConfig>
  <Name>Stateless Event Poster server</Name>
  <ServerFactory>
    <JavaName>com.blazesoft.server.deploy.NdStatelessServer</JavaName>
  </ServerFactory>
  <ServiceManagerFactory>
    <JavaName>com.blazesoft.server.local.NdLocalServiceManager</JavaName>
  </ServiceManagerFactory>
</DeployRulesServerConfig>
```

```

<DeployRulesServiceConfig>
  <Name>Loan Event Poster Argument Service</Name>
  <RulesServiceAgentFactoryFactory>
    <JavaName>com.blazesoft.server.rules.NdScriptRulesServiceAgentFactory</JavaName>
    <RulesProjectLoaderFactory>
      <JavaName>com.blazesoft.server.rules.NdRulesProjectFileLoader</JavaName>
      <Project>/opt/WebSphere/CommerceServer/instances/demo/rules/ConsumerCredit_POSTER.adv</Project>
    </RulesProjectLoaderFactory>
  </RulesServiceAgentFactoryFactory>
  <DeploymentType>Java</DeploymentType>
  <DeployRulesServiceClientContextFactory>
    <JavaName>com.blazesoft.server.deploy.rules.NdDeployPosterRulesServiceClientContext</JavaName>
    <Sr1MappingClass>ScoredLoanApplication</Sr1MappingClass>
  </DeployRulesServiceClientContextFactory>
  <NumAgents>2</NumAgents>
  <RecyclePolicy>0</RecyclePolicy>
</DeployRulesServiceConfig>

<DeployRulesServiceConfig>
  <Name>Loan Event Poster Argument Wrapping-Results Extractor Service</Name>
  <RulesServiceAgentFactoryFactory>
    <JavaName>com.blazesoft.server.rules.NdScriptRulesServiceAgentFactory</JavaName>
    <RulesProjectLoaderFactory>
      <JavaName>com.blazesoft.server.rules.NdRulesProjectFileLoader</JavaName>
      <Project>../../data/rules/ConsumerCredit_POSTER.adv</Project>
    </RulesProjectLoaderFactory>
  </RulesServiceAgentFactoryFactory>
  <DeploymentType>Java</DeploymentType>
  <DeployRulesServiceClientContextFactory>
    <JavaName>com.blazesoft.server.deploy.rules.NdDeployPosterRulesServiceClientContext</JavaName>
    <Sr1ArgumentsObjectFactory>
      <Sr1Class>ScoredLoanApplication</Sr1Class>
      <Sr1ObjectInitializerFunctional>
        <Sr1Name>initServiceData</Sr1Name>
        <Sr1ArgumentType>string</Sr1ArgumentType>
      </Sr1ObjectInitializerFunctional>
    </Sr1ArgumentsObjectFactory>
    <Sr1ResultExtractorFunctional>
      <Sr1Name>extractServiceResult</Sr1Name>
    </Sr1ResultExtractorFunctional>
  </DeployRulesServiceClientContextFactory>
  <NumAgents>2</NumAgents>
  <RecyclePolicy>0</RecyclePolicy>
</DeployRulesServiceConfig>
</DeployRulesServerConfig>

```

最上位のタグは `DeployRulesServerConfig` です。このタグ全体がルール・サーバーを表します。これには、`DeployRulesServiceConfig` というタグがいくつか含まれています。これらのタグのそれぞれが、ルール・サービスを表します。ルール・サーバー・タグ (`DeployRulesServerConfig`) は無視しても構いません。各ルール・サービス・タグ (`DeployRulesServiceConfig`) から、4つの部分の情報を取り出す必要があります。

3. WebSphere Commerce 5.4 管理コンソールを立ち上げます。
4. ログオンして、**ストア**を選択します。ストア名を選択し、「**OK**」をクリックします。
5. 「Rule Service Administration (ルール・サービス管理)」ツールに進みます。(「ルール・サービス」>「管理」)。
6. それぞれの `DeployRulesServiceConfig` タグに対し以下を行います。
 - a. 「Rule Service Administration (ルール・サービス管理)」ツールから、「サービスの追加」を選択します。
 - b. システムは、以下の4つの入力フィールドを表示します。

ルール・サービスの名前

`<Name>...</Name>` タグの値を入力します。この例では、最初のルール・サービスの名前は `Loan Event Poster Argument Service` です。

プロジェクト・ファイル名

`<Project>...</Project>` タグの値を入力します。この例では、最初のルール・サービスのプロジェクト・パスは次のようになります。

/opt/WebSphere/CommerceServer/instances/demo/rules/ConsumerCredit_POSTER.adv</Project

別々のフォルダー名を区切るには、スラッシュ (/) を使用し、ルール・プロジェクトの .adv ファイルの完全修飾パス名を入力してください。

31 ページの『ディレクトリーおよびファイルのバックアップ』で説明されているように、Commerce Suite 5.1.adv ファイルがバックアップされている必要があります。

エージェントの数

<NumAgents>...</NumAgents> タグの値を入力します。この例では、最初のルール・サービスのエージェント値は 2 です。

セッション・タイムアウト

<ServiceSessionTimeout>...</ServiceSessionTimeout> タグがあれば、その値を入力します。この例では、最初のルール・サービスのセッション・タイムアウト値は指定されていません。この場合、デフォルト値である 30000 (30000 ミリ秒、つまり 30 秒) を使用できます。

c. 「OK」 をクリックします。

システムによって、ルール・サービスの新しいリストが表示されます。これには、追加したばかりのルール・サービスも含まれています。ファイル wcs.server 内で、ルール・サービスごとに上記のステップを繰り返してください。

7. 上記のステップを完了したら、WebSphere Commerce 5.4 サーバーを再始動して「Rule Service Administration (ルール・サービス管理)」ツールに戻り、ルール・サービスが正常にマイグレーションされたかをチェックします。

ルール・サーバー管理コマンド

WebSphere Commerce 5.4 では、ルール・サービス管理コマンドの振る舞いとインターフェースの両方が変更されています。パッケージ

com.ibm.commerce.rules.commands および

com.ibm.commerce.ruleservice.admin.commands にある Commerce Suite 5.1 バージョンのコマンドは、ルール・サービスを追加、変更、除去、または最新表示するために、スケジューラーを使用して、すべてのアプリケーション・クローンに要求をブロードキャストします。WebSphere Commerce 5.4 では、コマンドが同じパッケージ内でより適切な名前が付いたものに置き換えられています。また、コマンドはジャストインタイム方法で操作されるようになっていました。たとえば、ルール・サービスを最新表示する場合、各アプリケーション・クローンがルール・サービスを再度実行する必要が生じるとすぐに、そのルール・サービスの独自のインスタンスを最新表示します。この方法により、信頼性が向上し、不必要な更新が避けられます。ルール・サービス管理コマンドを拡張した場合、振る舞いに関するこの変更が、カスタマイズした拡張機能にどのように影響するかを知るため、新しいコマンドを実行する必要があります。

表 6. ルール・システムのプロジェクトおよびパッケージ・レベル・マッピング

プロジェクト名	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
WCS ルール・デプロイメント	com.ibm.commerce.rules.deployment.commands	
WCS ルール・プロジェクト・サポート	com.ibm.commerce.rules.project	com.ibm.commerce.rules.project com.ibm.commerce.rules.beans ¹
WCS ルール・システム	com.ibm.commerce.rules	com.ibm.commerce.rules
	com.ibm.commerce.rules.commands	com.ibm.commerce.rules.commands
	com.ibm.commerce.rules.beans ¹	
	com.ibm.commerce.rules.blaze ²	
	com.ibm.commerce.rules.messages	
	com.ibm.commerce.rules.selector	
	com.ibm.commerce.rules.selector.blaze	
	com.ibm.commerce.rules.util ³	
		com.ibm.commerce.rules.ecmessages
		com.ibm.commerce.rules.exception
WCS ルール・システム管理	com.ibm.commerce.ruleservice.admin.beans	com.ibm.commerce.ruleservice.admin.beans
	com.ibm.commerce.ruleservice.admin.commands	com.ibm.commerce.ruleservice.admin.commands
	com.ibm.commerce.ruleservice.admin.util	com.ibm.commerce.ruleservice.admin.util
		com.ibm.commerce.ruleservice.admin.beansrc
WCS ルール・システム Enterprise Bean	com.ibm.commerce.rules.objects	com.ibm.commerce.rules.objects
	com.ibm.commerce.rules.objimpl	com.ibm.commerce.rules.objimpl
	WCSMCRuleLayerEJBReserved	WCSMCRuleLayerEJBReserved
		WCSRuleServerEJBReserved
		com.ibm.commerce.rules.helpers
ルール・サーバー統合		com.ibm.commerce.rules.blaze ²
		com.ibm.commerce.rules.blaze.exception
ルール・インフラストラクチャー		com.ibm.commerce.rules.repository
		com.ibm.commerce.rules.util ³
ルール・インフラストラクチャー共通サービス		com.ibm.commerce.services
		com.ibm.commerce.services.logging

注: いくつかのパッケージは WebSphere Commerce Suite 5.1 と WebSphere Commerce 5.4 とでは類似していますが、それらはまったく異なるクラスを持っている場合があります。類似のクラスも、類似のメソッドを持つ場合とそうでない場合とがあります。

^{1 2 3} これらのパッケージは、WebSphere Commerce 5.4 の別のプロジェクトに移されました。

以降のいくつかのセクションの表では、ルール・サーバー管理のための Commerce Suite 5.1 コントローラー・コマンドおよびアプリケーション・プログラミング・インターフェースへの変更ついてリストしています。

コントローラー・コマンド

ルール・サーバー管理コントローラー・コマンドの振る舞いに変更されています。WebSphere Commerce 5.4 では一般に、すべてのアプリケーション・クローンへのブロードキャスト情報ではなく、データベース内のルール・サービス構成情報を更新します。唯一の例外は、ブロードキャストを必要とする以下のコマンド・セットの場合です。

`com.ibm.commerce.ruleservice.admin.commands.BroadcastUpdateRuleServiceStatusCommand`

これらのコマンドは対応する URL を介して使用されるもので、カスタマイズしたり拡張したりすることは意図されていません。

これらのコマンドのいずれかに関する詳細については、WebSphere Commerce 5.4 のオンライン・ヘルプを参照してください。

以下の表では、簡潔に表記するために、コマンドのベース名だけをリストしています。完全なコマンド名の `com.ibm.commerce.ruleservice.admin.commands.` 部分は含まれていません。たとえば、`AddRuleServiceCommand` コマンドの完全な名前は、`com.ibm.commerce.ruleservice.admin.commands.AddRuleServiceCommand` です。

表7. ルール・サーバー・コントローラー・コマンド

WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
AddRuleServiceCommand	AddRuleServiceCommand
なし	BroadcastUpdateRuleServiceStatusCommand
StopRuleServiceCommand	DisableRuleServiceCommand
EditRuleServiceCommand	EditRuleServiceCommand
StartRuleServiceCommand	EnableRuleServiceCommand
RefreshRuleServiceCommand	RefreshRuleServiceCommand
RemoveRuleServiceCommand	RemoveRuleServiceCommand
CheckRuleServiceStatusCommand	UpdateRuleServiceStatusCommand

アプリケーション・プログラミング・インターフェース呼び出し (タスク・コマンド)

Commerce Suite 5.1 で直接的なメソッド呼び出しだったものは、WebSphere Commerce 5.4 ではタスク・コマンドになっています。Commerce Suite 5.1 で最も一般的に使用されるアプリケーション・プログラミング・インターフェース (API) 呼び出しは、ルール・サービスを呼び出すためのものです。WebSphere Commerce 5.4 では、これは

`com.ibm.commerce.rules.commands.InvokePersonalizationRuleServiceCommand` コマンドを使用することによって行われます。

これらのタスク・コマンドのいずれかに関する詳細については、WebSphere Commerce 5.4 のオンライン・ヘルプを参照してください。

以下の表では、簡潔に表記するために、API およびコマンドのベース名だけをリストしています。Commerce Suite 5.1 コマンドで、完全な API 名の `com.ibm.commerce.rules.RulesSystem.` 部分はリストされていません。たとえば、`changeServiceConfiguration()` の完全な名前は、`com.ibm.commerce.rules.RulesSystem.changeServiceConfiguration()` です。同様

に、WebSphere Commerce 5.4 コマンドについても、コマンド名の `com.ibm.commerce.rules.commands` 部分はリストされていません。たとえば、`ChangePersonalizationRuleServiceCommand` の完全な名前は、`com.ibm.commerce.rules.commands.ChangePersonalizationRuleServiceCommand` です。

表 8. ルール・サーバー API 呼び出し (タスク・コマンド)

WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
<code>changeServiceConfiguration()</code>	<code>ChangePersonalizationRuleServiceCommand</code>
<code>addService()</code>	<code>CreatePersonalizationRuleServiceCommand</code>
<code>stopService()</code>	<code>DisablePersonalizationRuleServiceCommand</code>
<code>startService()</code>	<code>EnablePersonalizationRuleServiceCommand</code>
<code>invokeService()</code>	<code>InvokePersonalizationRuleServiceCommand</code>
なし	<code>MarkPersonalizationRuleServiceChangedCommand</code>
<code>removeService()</code>	<code>RemovePersonalizationRuleServiceCommand</code>
<code>getService().getStatus()</code>	<code>UpdatePersonalizationRuleServiceStatusCommand</code>

以下の例は、WebSphere Commerce Suite 5.1 と WebSphere Commerce 5.4 との間の API 呼び出しのインプリメンテーションの相違を示しています。

Commerce Suite 5.1 でのルール・サービスの呼び出し (コマンド・コンテキストが `context` に設定されており、ルール・サービス名が `ruleServiceName` であると想定)

```
RuleServiceKey key=null;
key = new RuleServiceKey(ruleServiceName, context.getStoreId());
RulesSystem rulesSystem = RulesSystemToolbox.getInstance().getRulesSystem();
if (rulesSystem.isAvailable()) {
    rulesSystem.invokeService (key,context);
}
```

WebSphere Commerce 5.4 でのルール・サービスの呼び出し (`commandContext` が `commandContext` に設定されており、ルール・サービス名が `ruleServiceName` であると想定)

```
InvokePersonalizationRuleServiceCommand command =
    (InvokePersonalizationRuleServiceCommand)
    CommandFactory.createCommand
        (InvokePersonalizationRuleServiceCommand.class.getName(), storeId);
command.setCommandContext(commandContext);
command.setServiceName(ruleServiceName);
command.execute();
```

上記の例では、WebSphere Commerce 5.4 のインターフェースが最も分かりやすく使用が簡単であることに注意してください。このインターフェースでは、多くのインプリメンテーションの詳細情報がユーザーからは隠されています。

例外処理

WebSphere Commerce 5.4 では、ルール・システムは例外処理を、さまざまな問題領域を明示的に識別するように拡張しています。WebSphere Commerce Suite 5.1 では、すべての汎用ルール・システム関連の例外を処理する 1 つの `RulesExceptionHandler` クラスがありました。WebSphere Commerce 5.4 では、`RulesExceptionHandler` クラスはいくつかのルール例外クラスによって置き換えら

れています。これらの異なるクラスによって、例外の実際の原因を識別することが一層容易になっています。これらのクラスは、WebSphere Commerce ルール・システム・プロジェクトの `com.ibm.commerce.rules.exception` パッケージに入っており、それらは一般の `WebSphere Commerce RuntimeException` クラスを拡張します。

ルール・システム例外を以下に示します。

- `InvalidRuleServiceKeyException`
- `PersonalizationRuleServerException`
- `PersonalizationRuleServerNotAvailableException`
- `PersonalizationRuleServiceExistsException`
- `PersonalizationRuleServiceNotEnabledException`
- `PersonalizationRuleServiceNotFoundException`
- `RuleServerConfigurationNotFoundException`
- `RuleServerNotFoundException`
- `RuleServiceNotFoundException`
- `RuleServiceConfigurationNotFoundException`
- `RulesSystemDataModelException`
- `RulesSystemRuntimeException`

オークション

Commerce Suite 5.1 でオークションを使用可能にしていた場合、以下を考慮する必要があります。

Commerce Suite 5.1 のすべてのオークション・オーダー・アイテムでは、デフォルト契約 (`CONTRACT` テーブルの `CONTRACT_ID`) が使用されます。WebSphere Commerce 5.4 へのマイグレーション後、オークション・オーダー・アイテムには WebSphere Commerce 5.4 の新しいデフォルト契約が入れられます。WebSphere Commerce 5.4 のデフォルト契約には、オークション・アイテムでは適切でない条件やリファンド・ポリシーなどが含まれている可能性があります。

デフォルト契約の条件が、オークション・オーダー・アイテムで自分が必要とするものかどうかを確認する必要があります。そうでない場合、`TRADING` テーブルの `TRADING_ID` を変更して適切な契約を指すことによって、オークションで適切なものに契約を変更する必要があります。デフォルトでは、データベース・マイグレーション・スクリプトによって、データ・マイグレーション中に作成されるデフォルト契約を指すように、`TRADING_ID` が設定されます。

ビジネス・アカウント

このマイグレーション・スクリプトでは、ビジネス・アカウントは作成されません。スクリプトは、マイグレーションされたすべてのオーダー・アイテムをデフォルト契約に関連付けます。WebSphere Commerce 5.4 へのマイグレーション後にアカウントを作成する場合、デフォルト契約を使用するように指定するか、またはオ

ーダー・アイテムを変更する必要があります。アカウントの作成に関する詳細については、 WebSphere Commerce 5.4 オンライン・ヘルプの『新規ビジネス・アカウントの作成』を参照してください。

カスタマイズしたデータベース・テーブル用のステージング・サーバーの再構成

Commerce Suite 5.1 でデータベース・テーブルをカスタマイズしており、WebSphere Commerce 5.4 でステージング・サーバーを使用する場合、商品データベースとステージング・データベースの整合性を保つために、データ・マイグレーションに先だって、ステージ・コピー・ユーティリティー・コマンド (stagingcopy) を実行する必要があります。データ・マイグレーションが完了した後で、ステージング・サーバーを再構成する必要があります。マイグレーション・プロセスでは、以前の Commerce Suite 5.1 構成はマイグレーションされません。

マイグレーション・プロセスでは、Commerce Suite 5.1 ステージング・サーバー・テーブルは、元の名前に `_WCS51` が付加されて名前変更されることに注意してください。したがって、Commerce Suite 5.1 ステージング・サーバー・テーブルは以下のように保存されます。

- STGSITETAB_WCS51
- STGMERTAB_WCS51
- STGMRSTTAB_WCS51
- STAGLOG_WCS51

参照用にこれらの名前変更されたテーブルの内容を表示できます。

データベースのログ・ファイルのサイズを増やすことが必要になる場合もあることに注意してください。ログ・ファイルの・サイズを増やすには、DB2 コマンド・ウィンドウから次のステートメントを入力します。

```
DB2 UPDATE DB CFG FOR db_name USING LOGFILSIZ nnn
```

ここで、*db_name* はデータベースの名前です (たとえば、MALL)。 *nnn* は LOGFILSIZ 値を表す数です。詳細については、DB2 管理情報を参照してください。

さらに、Commerce Suite 5.1 データベース・クリーンアップ・ユーティリティー・テーブルは、元の名前に `_WCS51` が付加されて名前変更されます。そのため、Commerce Suite 5.1 データベース・クリーンアップ・ユーティリティー・テーブルは、`CLEANCONF_WCS51` として保存されます。参照用にこの名前変更されたテーブルの内容を表示できます。

カスタマイズしたデータベース・テーブル用にステージング・サーバーを再構成する場合は、 WebSphere Commerce 5.4 オンライン・ヘルプの『カスタマイズ・テーブルのステージング・サーバーの構成』を参照してください。

カスタマイズしたデータベース・テーブル用にデータベース・クリーンアップ・ユーティリティーを再構成する場合は、 WebSphere Commerce 5.4 オンライン・ヘルプの『データベース・クリーンアップ・ユーティリティーに新規構成を追加する』を参照してください。

使用されなくなった Commerce Suite 5.1 テーブルの除去

Commerce Suite 5.1 からの以下のテーブルは、WebSphere Commerce 5.4 では使用されなくなっています。

- CMPGNINTV
- CMPGNRV
- INTVMPE
- INTVSGMT
- MAFAMILY
- MATYPE
- MPE
- MPETYPE
- ONQUEUE
- ONLOG
- ONSLOG
- ORDERMSG
- SEGMENT
- ZIPCODE
- ACCCMDGRP
- ACCMBRGRP
- ACCCUSTEXC

上記テーブルを現在使用していないことを確認してください。どのテーブルも必要なくなった場合、以下のようにしてテーブルを除去することができます。

DB2 コマンド・ウィンドウで次のように入力します。

```
db2 drop table table_name
```

table_name は、除去するテーブルの名前です。

注: 除去した古いテーブルに対する KEYS または SUBKEYS テーブルにエントリーがある場合、それらのエントリーを KEYS または SUBKEYS テーブルから除去する必要があります。

新しいキャッシュ・トリガーのロード

Commerce Suite 5.1 ではキャッシュ・トリガーが提供されており、オプションでそれらをロードしたり、パフォーマンス上の理由からアンロードしたりすることができました。トリガーはテーブルの行が更新、追加、または削除される際に発生するデータベースのイベントです。キャッシュ・コンポーネント (デフォルトで使用可能になっている) は、オブジェクトが無効にされる際に示す通知メカニズムとして、トリガーを使用します。

このマイグレーション・スクリプトでは、既存のキャッシュ・トリガーは更新されません。しかし、以下のファイルが以下のディレクトリーに備えられています。

- DB2 データベースの場合 — /opt/WebSphere/CommerceServer/schema/db2

wcs.cache.trigger.sql

Commerce Suite 5.1 と WebSphere Commerce 5.4 の両方のキャッシュ・トリガーをロードします。

wcs.cache.trigger1.sql

Commerce Suite 5.1 キャッシュ・トリガーだけをロードします。

wcs.cache.trigger2.sql

WebSphere Commerce 5.4 キャッシュ・トリガーだけをロードします。

wcs.cache.delete.trigger.sql

Commerce Suite 5.1 および WebSphere Commerce 5.4 のすべてのキャッシュ・トリガーをアンロードします。

wcs.cache.delete.trigger1.sql

Commerce Suite 5.1 キャッシュ・トリガーだけをアンロードします。

wcs.cache.delete.trigger2.sql

WebSphere Commerce 5.4 キャッシュ・トリガーだけをアンロードします。

すでに Commerce Suite 5.1 キャッシュ・トリガーをロードしていた場合、以下が必要です。

1. データベースのマイグレーション前に、 `wcs.cache.delete.trigger1.sql` を使用して、Commerce Suite 5.1 トリガーをアンロードします。これは、76 ページの『Commerce Suite 5.1 キャッシュ・トリガーの除去』に従ってすでに実行されているはずです。
2. `wcs.cache.trigger.sql` を使用して、Commerce Suite 5.1 と WebSphere Commerce 5.4 の両方のトリガーをロードします。

このステップは以下のとおりです。

DB2 データベースの場合

1. DB2 コマンド・ウィンドウをオープンします。
2. `/opt/WebSphere/CommerceServer/schema/db2` ディレクトリーに移動します。
3. 以下のようにして、Commerce Suite データベースに接続します。
`db2 connect to db_name user db_user_name using db_password`
4. 以下のようにして、Commerce Suite 5.1 と WebSphere Commerce 5.4 の両方のトリガーをロードします。
`db2 -tvd# -f wcs.cache.trigger.sql`

カスタマイズしたプロパティの改良

3 ページの『マイグレーション前のアクション』でバックアップした、カスタマイズ済みプロパティおよびコードに関して、カスタマイズ内容を改良して、対応する WebSphere Commerce 5.4 プロパティ・ファイルに入れてください。

配送計算コード

Commerce Suite 5.1 では、配送計算コードは、異なる配送先住所を持つオーダー・アイテムのグループごとに別個に計算されていました。つまり、配送計算コードでは、配送先住所別にオーダー・アイテムがグループ化されていました。現在では、配送先住所別のグループ化は、オプションの動作になっています。下位互換性動作を保証するため、マイグレーション・スクリプトは、マイグレーション時に、すべての配送計算コード (CALCODE.CALUSAGE_ID = -2) 用の CALCODE.GROUPBY 列に perAddress フラグを設定します。

商品および在庫の検索

データをマイグレーションした後で、WebSphere Commerce Accelerator を使用し、以下のステップを実行して商品および在庫を検索します。

商品を検索するには、以下のようにします。

1. WebSphere Commerce Accelerator にログオンして、すでに公開されているストアを選択します。
2. 「商品」をクリックし、「商品の検索」を選択します。
3. 商品コードを入力します。
たとえば、sku-105 が商品 sku-102 に属するアイテムの場合、商品 ID sku-102 を入力する必要があります。
4. 102 を入力します。これにより、この商品を検索できます。
5. 右のパネルで「SKU」を選択し、sku-105 を検索します。

在庫を検索するには、以下のようにします。

1. WebSphere Commerce Accelerator にログオンして、公開されているストアを選択します。
2. 「ロジスティック」をクリックし、「在庫の検索」を選択します。
3. そのフィールドでアイテムの SKU を入力します。

注:

- a. 商品の SKU ではなく、アイテムの SKU を入力します。「商品の検索」フィールドからアイテムの SKU を検索できます。
- b. 105 などの番号を入力してから、「検索」をクリックし、必要なアイテムを検索します。

マイグレーションされたアイテムのすべての名前および説明は、「default migrated baseitem description (デフォルトのマイグレーション済みベース・アイテムの説明)」に設定されます。これらのアイテムの名前および説明は、BASEITMDESC テーブルに入れられます。これは、WebSphere Commerce 5.4 での新しいテーブルです。Commerce Suite 5.1 では、アイテムに関する名前および説明がありませんでした。

カスタマイズ・コマンド

Commerce Suite 5.1 で任意のカスタマイズ・コマンドをデプロイした場合について考慮します (たとえば、EJB を使用してコントローラー・コマンドを作成した、など)。この場合、システムを WebSphere Commerce Business Edition 5.1 にマイグレーションした後、マイグレーション済みシステムにおいて実行する前に、*WebSphere Commerce Studio* *マイグレーション・ガイド* バージョン 5.4 の説明に従ってすべてのカスタマイズ・コマンドを再デプロイする必要があります。

edit_registration ページにおけるログオン ID の形式

LDAP を使用している場合、edit_registration ページのログオン ID は、RDN 形式ではなく DN 形式で表示されます。これを RDN 形式で表示するには、UserRegistrationDataBean で提供されているメソッドを使用して、ログオン ID を適切に取り出します。このメソッドを使用するには、各ストアの JSP で次のような小さな変更を加える必要があります。古いメソッドもまだサポートされているため、LDAP が使用されない場合でもマイグレーション済みのストアは正常に機能することに注意してください。

edit_registration.jsp から以下のようなコードを見つけ出します。

```
<%
    strLogonID = jhelper.htmlTextEncoder(bnRegister.getLogonId());
    strPassword = bnRegister.getLogonPassword();
    strFirstName = jhelper.htmlTextEncoder(bnRegister.getFirstName());
    strLastName = jhelper.htmlTextEncoder(bnRegister.getLastName());
}
%>
```

以下のように、太字で示されているコードを更新します。

```
<%
    // use getAttribute("RDN") here because getLogonId() will
    // return the DN value when LDAP is used
    strLogonID = jhelper.htmlTextEncoder(bnRegister.getAttribute("RDN"));
    strPassword = bnRegister.getLogonPassword();
    strFirstName = jhelper.htmlTextEncoder(bnRegister.getFirstName());
    strLastName = jhelper.htmlTextEncoder(bnRegister.getLastName());
}
%>
```

商品アドバイザーのマイグレーション考慮事項

商品アドバイザー構成を前のリリースからマイグレーションする場合、以下の各項を考慮してください。これは WebSphere Commerce Suite 5.1 商品アドバイザー検索スペースが作成済みであることと、前のリリースの WebSphere Commerce で商品アドバイザーが操作可能であることを想定しています。

1. WebSphere Commerce 5.4 のサンプル JavaServer Pages (JSP)
(/opt/WebSphere/CommerceServer/samples/web/pa ディレクトリーにある pe51.jsp、pc51.jsp、および sa51.jsp) は、Commerce Suite 5.1 において同じ名前を持つ、マイグレーション済みバージョンのファイルです。 WebSphere Commerce 5.4 の場合、データ・タイプ・パッケージ名は、以下の表に要約しているとおりに変更されています。これらのパッケージ名を参照するすべての JSP で、 com.ibm.commerce.datatype のオカレンスを com.ibm.commerce.pa.datatype に、以下の表で要約しているとおりに変更する必要があります。

WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
com.ibm.commerce.datatype.DsString	com.ibm.commerce.pa.datatype.DsString
com.ibm.commerce.datatype.DsInteger	com.ibm.commerce.pa.datatype.DsInteger
com.ibm.commerce.datatype.DsDouble	com.ibm.commerce.pa.datatype.DsDouble
com.ibm.commerce.datatype.DsCurrency	com.ibm.commerce.pa.datatype.DsCurrency
com.ibm.commerce.datatype.DsDecimal	com.ibm.commerce.pa.datatype.DsDecimal
com.ibm.commerce.datatype.DsURLLink	com.ibm.commerce.pa.datatype.DsURLLink
com.ibm.commerce.datatype.DsImage	com.ibm.commerce.pa.datatype.DsImage
com.ibm.commerce.datatype.DsDate	com.ibm.commerce.pa.datatype.DsDate

さらに、WebSphere Commerce 5.4 で導入された新規のデータ・タイプとして com.ibm.commerce.pa.datatype.DsLong があり、これは catentry_id または一般的な整数よりも大きな値を持つ他の属性に使用する必要があることに注意してください。

2. 商品アドバイザー検索スペースを作成するスクリプトで (たとえば createsearchspace.sh)、ステップ 1 の表で示しているとおりに、同じデータ・タイプパッケージ名の変更を加える必要もあります。

WebSphere Commerce 5.4 で提供されているサンプル createsearchspace.sh を参照することができます。これはこのパッケージ名の変更によって更新済みです。

WebSphere Commerce 5.4 では、PAConfig ユーティリティーを使用した検索スペース作成の新しい方法がサポートされていることに注目してください。このユーティリティーは、検索スペース作成のプロセスの自動化に役立ちますが、入力 XML ファイルを必要とします。XML ファイルは手動で作成する必要があります。このプロセスの詳細については、オンライン・ヘルプの『商品アドバイザー』のトピックを参照してください。

3. スクリプトを実行して、マイグレーション済みの WebSphere Commerce 5.4 システムで商品アドバイザー検索スペースを作成します。スクリプトの実行後に、ICEXPLFEAT テーブルの DATATYPE 列を検査して、すべてのパッケージ名が正常

にマイグレーションされたことを検査します。この列のすべてのクラス名は、新規のパッケージ名 `com.ibm.commerce.pa.datatype` を持っているはずで

4. メタフォーを作成するための PABatchXML ユーティリティーへの入力として使用される XML ファイルを、ステップ 1 (127 ページ) のテーブルでリストされている同じパッケージ名の変更で更新します。
5. PABatchXML ユーティリティーを実行します。

第 2 部 追加のマイグレーション考慮事項

マイグレーション・ガイドのこの部の章は、特定の Commerce Suite 5.1 ユーザーだけに適用されるマイグレーション考慮事項およびシナリオを記載しています。ほとんどの場合、これらのセクションはオプションのステップと見なすことができます。これには以下が含まれます。

- 131 ページの『第 7 章 メンバー・サブシステムのマイグレーション考慮事項』
- 137 ページの『第 8 章 アクセス・コントロール・サブシステムの考慮事項』
- 143 ページの『第 9 章 Payment コンポーネントのマイグレーション考慮事項』
- 149 ページの『第 10 章 その他のマイグレーション考慮事項』

第 7 章 メンバー・サブシステムのマイグレーション考慮事項

この章では、LDAP と WebSphere Commerce 5.4 データベースとの統合方法の決定によって異なる、メンバー・サブシステムを Commerce Suite 5.1 から WebSphere Commerce 5.4 にマイグレーションする際のマイグレーション・シナリオについて説明しています。

重要

マイグレーションの前、あるいはメンバー・リポジトリとしてデータベースの使用からディレクトリー・サーバーの使用に切り替える前には、常にデータベースをバックアップしなければなりません。データベースのバックアップを行う方法については、33 ページの『データベースのバックアップ』を参照してください。

DB2 を使用する場合、データベースをマイグレーションした後で、以下の SQL ステートメントおよびコマンドを実行して、`stmheap` パラメーターのサイズを増やす必要があります。

```
db2 update db cfg for database_name using stmheap 60000
db2stop
db2start
```

ここで、`database_name` はマイグレーションしたデータベースのデータベース名です。 `stmheap` パラメーターを適切な値に増やさない場合、新しく作成したアクセス・グループを削除できないことがあります。

マイグレーション手順の概説

このセクションでは、メンバー・サブシステムのマイグレーション手順について概説します。

下記の表の見方を以下に示します。

DB->DB

Commerce Suite 5.1 データベースから WebSphere Commerce 5.4 データベースへのマイグレーション

DS->DS

Commerce Suite 5.1 ディレクトリー・サーバーから WebSphere Commerce 5.4 ディレクトリー・サーバーへのマイグレーション

DB->DS

Commerce Suite 5.1 データベースから WebSphere Commerce 5.4 データベースへのマイグレーションを行ってから、ディレクトリー・サーバーの使用への切り替え。

データベースの使用からディレクトリー・サーバーの使用へのマイグレーションは、リリース間のマイグレーションの後であれば、いつでも実行できま

す。したがって、DB->DS シナリオの場合、まず DB->DB 列を見てから、順に DB->DS 列を見るのが正しい見方です。

Commerce Suite 5.1 では、データベースに対するブートストラップ・データで wcsadmin ユーザーが提供されています。しかし、Commerce Suite 5.1 でディレクトリー・サーバーを使用していた場合、ディレクトリー・サーバー内での wcsadmin に、対応するブートストラップは提供されていませんでした。そのため Commerce Suite 5.1 では、ディレクトリー・サーバー内に wcsadmin のエントリーがある場合に、そのディレクトリー・サーバーを使用しているのが誰であるかは分かりませんでした。以下のマイグレーション手順では、そのようなエントリーがディレクトリー・サーバー内に存在していることを想定しています。しかし、ディレクトリー・サーバー内で wcsadmin ユーザーが実際に置かれている場所に関係なく、メンバーシップ階層を取り込む MBRREL テーブルでは、WebSphere Commerce 5.4 内の wcsadmin の親メンバーは Root Organization に設定されます。

この後の自動化されたマイグレーションによるタスクは、メンバー・サブシステムのために行われるすべてのタスクのサブセットにすぎません。この後にリストされていないタスクが他にもあります (主に役割とメンバー・グループのマイグレーション)。**DB->DS** のマイグレーションの詳細については、WebSphere Commerce 5.4 オンライン・ヘルプで見つけることができます。

表9. メンバー・サブシステムのマイグレーション手順の概説

マイグレーション手順	DB->DB	DS->DS	DB->DS	注釈
ORG_ID および ORGUNIT_ID に適切に入力し、ビジネス・ユーザー用に BUSPROF テーブルにレコードを作成します。	手動	手動	注釈を参照	DB->DB マイグレーション中にすでに完了しています。
USERS テーブル内でユーザーの PROFILETYPE を修正します。12 ページの『メンバー・サブシステム』を参照してください。				
自動マイグレーション・スクリプトを実行します。				
STATE 列を MEMBER テーブルに追加し、データを取り込みます。	自動	自動	注釈を参照	(*) DB ->DS に関して自動化されたスクリプトによって行われることはすべて、DB->DB マイグレーション中に完了しています。
Root Organization をデータベースに追加します。	自動	自動	参照	(*)
BUSPROF テーブル内の wcsadmin ユーザーの ORG_ID を、-2000 から -2001 に変更します。	自動	自動	参照	(*)
USERS テーブル内の wcsadmin の PROFILETYPE を、C から B に変更します。				
それまでヌルであったものに関して、ORGENTY テーブル内の MEMBER_ID を、-2001 (ルート組織) に設定します。	自動	自動	参照	(*)

表9. メンバー・サブシステムのマイグレーション手順の概説 (続き)

マイグレーション手順	DB->DB	DS->DS	DB->DS	注釈
MBRREL テーブルを作成し、データを取り込みます。	自動	自動	参照 (*)	
自動データ・マイグレーションにより生成された組織エンティティ用の DN を調べます。必要なら、スクリプトの DN (識別名) を変更します。	N/A	手動	手動	
ORGENITY テーブルに組織エンティティの DN、および USERS テーブルにユーザーを取り込むためにスクリプトを実行します。	手動	手動	手動	
すべての必要なサフィックスをディレクトリー・サーバーに作成します。これらは、組織エンティティが Commerce Suite 5.1 によりディレクトリー・サーバーに自動的に作成される時に必要になるサフィックスです。	N/A	手動	手動	
1dapentry.xml ファイルを作成します。	N/A	手動	手動	下の注釈を参照してください。
DS->DS マイグレーションの場合、1dapentry.xml は Commerce Suite 5.1 の 1dapmap.xml ファイルの内容に基づいていなければなりません。				
DB->DS マイグレーションの場合、構成マネージャーを使用してディレクトリー・サーバーの使用に切り替えます。手動で instance_name.xml ファイルを編集し、MigrateUsersFromWCsdb オプションを「ON」に設定します。デフォルトでは、これは「OFF」に設定されています。	N/A	N/A	手動	

注: DS->DS マイグレーションの場合、1dapentry.xml を作成する時、ユーザーにどのようにログオンしてもらいたいかにより、ユーザー検索ベースの指定に関して若干異なる方法をとる必要があります。次の説明は 2 種類のユーザーの違いを述べています。

- **DS ユーザー** とは、ディレクトリー・サーバー内に存在するユーザーで、かつ WebSphere Commerce に認識してもらいたいユーザーです。しかしながら、これらのユーザーは Commerce Suite 5.1 にログオンしたことがなく、かつ Commerce Suite 5.1 に参照されたことのないユーザーです。そのため、これらのユーザーは WebSphere Commerce Suite データベースにまだエントリーがありません。
- **WCS DS ユーザー** とは、ディレクトリー・サーバー内に存在するユーザーで、かつ Commerce Suite がすでに認識しているユーザーです。なぜなら、これらのユーザーはすでに Commerce Suite 5.1 サイトにログオンしているからです。そうしたユーザーは Commerce Suite データベースにエントリーがあります。

DS ユーザーと WCS DS ユーザーの両方が RDN (相対識別名) を使用してログオンすることを望む場合には、両方のタイプのユーザーは、ディレクトリー・サーバー内ですべてのユーザーが固有のものに見なされるような RDN 値を持つ必要があります。それから、両方のタイプのユーザーを見つけられるよ

うな検索ベースを指定します。ディレクトリー・サーバーがユーザーを検索する時は、WebSphere Commerce はユーザーが 1 つだけ見つかることを期待します。複数のユーザーが見つかるなら、それはエラー状態です。

DS ユーザーおよび WCS DS ユーザーが同じ RDN を持つことを望む場合には (たとえば、ある DS ユーザーが 'uid=john, o=IBM, c=US' という DN を持っていて、別の WCS DS ユーザーが 'uid=john, o=CompanyA, o=Root Organization' という DN を持っている場合、どちらのユーザーも 'john' という RDN 値を持っているという点に注意してください)、以下のようにします。

- WCS DS ユーザーには、WebSphere Commerce Suite 5.1 で使用していたものと同じログオン ID を引き続き使用してログオンしてもらうことができます。DS ユーザーには、DN を使用してログオンしてもらいます。DS ユーザーが常駐する場所と検索ベースがオーバーラップしないように、WCS DS ユーザーの検索ベースを指定する必要があります。

DB->DS のマイグレーションに関する詳細については、WebSphere Commerce 5.4 のオンライン・ヘルプで、LDAP の統合のセクションを参照してください。

既存のディレクトリー・サーバーを使用する既存の Commerce Suite 5.1 ユーザー

このシナリオでは、いくつかのエントリーがある既存のディレクトリー・サーバーがすでにあります。Commerce Suite 5.1 を使用しているものの、既存のディレクトリー・サーバーは使用していません。WebSphere Commerce 5.4 にマイグレーションした後で、既存のディレクトリー・サーバーを WebSphere Commerce 5.4 と一緒に使用しようと思っています。Commerce Suite 5.1 ユーザーなので、アクセス Bean だけを使用して WebSphere Commerce 5.4 データベースから MEMBER データを取り出すコードを実行しています。

詳細については、WebSphere Commerce 5.4 オンライン・ヘルプの『LDAP シナリオ: メンバー・リポジトリーとしてのデータベース』を参照してください。

WebSphere Commerce 5.4 での 5.1 ディレクトリー・サーバーの継続使用

このシナリオでは、すでにディレクトリー・サーバーを Commerce Suite 5.1 と一緒に使用しています。今回 WebSphere Commerce 5.4 にマイグレーションし、ディレクトリー・サーバーを引き続き使用します。Commerce Suite 5.1 の顧客なので、アクセス Bean だけを使用してメンバー・データを取り出すコードを実行しています。5.1 でディレクトリー・サーバーを使用していたので、Commerce Suite 5.1 によって認識される登録済みユーザーがディレクトリー・サーバーに存在し、ユーザーのデータが Commerce Suite 5.1 データベースに複製されますが、組織エンティティーおよびメンバー・グループのデータは、Commerce Suite 5.1 データベースにしかありません。

このシナリオでは、以下を行う必要があります。

1. ビジネス・ユーザー (B2B ユーザー) を WebSphere Commerce 5.4 のメンバーシップ階層内で適切に配置するために、ORG_ID および ORGUNIT_ID を適切に設

定し、必要に応じて BUSPROF テーブル内にレコードを存在させる必要があります。さらに、必要に応じてユーザーの profileType を設定します。以下を手動で行う必要があります。

- ビジネス・ユーザーの親および上位の組織エンティティがすでに Commerce Suite 5.1 データベースに存在する場合、以下を行います。
 - ビジネス・ユーザーに BUSPROF レコードがない場合、ビジネス・ユーザーの BUSPROF レコードを作成して、適切な組織エンティティを指すように ORG_ID および ORGUNIT_ID を設定します。
 - ビジネス・ユーザーに BUSPROF レコードがある場合、BUSPROF レコードで ORG_ID および ORGUNIT_ID が適切に設定されているかを確認します。
 - ビジネス・ユーザーの親および上位の組織エンティティが Commerce Suite 5.1 データベースに存在せず、それらの組織エンティティを作成できない場合、そのビジネス・ユーザーの profileType を C (B2C ユーザー) に設定することを考慮してください。
2. 78 ページの『データベース・スキーマのマイグレーション』でのデータ・マイグレーション・スクリプトを実行し、以下を行います。
 - STATE 列を MEMBER テーブルに追加します。
 - wcsadmin ユーザーの ProfileType を C から B に変更します。
 - wcsadmin の BUSPROF テーブル内の ORG_ID を、-2000 から -2001 に変更します。
 - Root Organization をデータベースに追加します。
 - Commerce Suite 5.1 ではヌルだったメンバー ID について、ORGENITY テーブル内の MEMBER_ID を入力します。
 - MBRREL テーブルを作成し、データを取り込みます。
 3. 自動化されたデータ・マイグレーションの一部として、83 ページの『識別名の更新』で説明しているように、ORGENITY テーブルに識別名 (DN) 値を取り込むスクリプトが提供されています。組織エンティティの DN 値を調べて、それらが適切かどうかを確認し、必要に応じて DN 値を変更する必要があります。それから fillorgDN.sql スクリプトを実行して、組織エンティティの DN 値を取り込みます。また、USERS テーブルに登録済みユーザーの DN 値を取り込みます。DN 値が取り込まれるのは登録済みユーザーについてだけであり、それらのユーザーの DN 値は、後で WebSphere Commerce 論理によって置き換えられることに注意してください。
 4. 必要なすべての接尾部がディレクトリー・サーバー内に適切に作成されているかを確認します。これらの接尾部は、WebSphere Commerce によってディレクトリー・サーバー内に組織エンティティが自動的に作成される際に必要です。
 5. ldapmap.xml に基づいて ldapentry.xml ファイルを作成し、組織エンティティ属性のマッピングを ldapentry.xml に追加します。
 6. instancename.xml ファイル内の MigrateUsersFromWCsdb オプションが OFF になっているかを確認します。

第 8 章 アクセス・コントロール・サブシステムの考慮事項

WebSphere Commerce 5.4 のアクセス・コントロール・モデルは、アクセス・コントロール・ポリシーの制約に基づいています。アクセス・コントロール・ポリシーは、アクセス・コントロール・ポリシー・マネージャーによって施行されます。一般に、ユーザーが保護可能リソースへのアクセスを試みる際、アクセス・コントロール・ポリシー・マネージャーは、ユーザーが、指定されたリソースで要求された操作を実行できるかどうかを判別します。

加えて、以下の点に注意してください。

- 以下の Commerce Suite 5.1 アクセス・コントロール・データベース・テーブルは、WebSphere Commerce 5.4 では使用すべきではありません。
 - ACCMBRGRP
 - ACCCMDGRP
 - ACCCUSTEXC
 - ACCCMDTYPE

これらのテーブルは、アクセス・コントロールを決定するためにサーバー・ランタイムによって使用されることはなくなりました。これらは、いくつかの新しいアクセス・コントロール・テーブルに置き換えられています。詳細については、WebSphere Commerce 5.4 オンライン・ヘルプの『アクセス・コントロール』を参照してください。

- デフォルトの Commerce Suite 5.1 ブートストラップ・アクセス・コントロール・ポリシー (ACCCMDGRP テーブル) に加えられる変更は失われます。ただし、このテーブルに対して行われた追加は保存され、データ・マイグレーション・プロセスによって適宜 WebSphere Commerce 5.4 にマイグレーションされます。
- Commerce Suite 5.1 では、ACCCUSTEXC が、指定されたストア内での、指定された顧客コマンドの実行を阻止する除外テーブルです。WebSphere Commerce 5.4 は、より寛容なアクセス・コントロール・モデルに従っているため、緩いポリシーがメンバー階層の上位に存在しないように適切な注意を払ってください。
- WebSphere Commerce 5.4 での、アクセス・コントロールの 2 つのレベルは以下のとおりです。
 - コマンド・レベル (粗い)
ユーザーがコントローラー・コマンドまたはビューへのアクセスを持つかどうかを決定します。
 - リソース・レベル (細かい) — 役割ベースのアクセス・コントロールとしても知られます。
ユーザーが特定のリソースのインスタンスに対してアクションを実行できるかどうかを決定します。

以下の表では、Commerce Suite 5.1 アクセス・コントロールと WebSphere Commerce 5.4 アクセス・コントロールとの違いについて説明しています。主な違いは、Commerce Suite 5.1 はプログラムに基づいたリソース・レベル・アクセス・コントロールを使用するのに対し、WebSphere Commerce 5.4 はポリシーに基づいた

リソース・レベル・アクセス・コントロールを使用するということです。カスタマイズ・コードの障害を最小に抑えるため、WebSphere Commerce ランタイムは、現行では Commerce Suite 5.1 と WebSphere Commerce 5.4 の両方のアクセス・コントロール関連コマンド・メソッドを処理しています。ただし、どのカスタマイズ・コードも WebSphere Commerce 5.4 メソッドを使用するようにマイグレーションして、ポリシー・ベースのアクセス・コントロール・モデルを使用することを強くお勧めします。提供されているどのコマンドも、以下のアクセス・コントロール・モデルのいずれかを必ず使用しているなら、正しく機能するはずですが、

- validateParameters および getResources メソッドを使用する推奨 WebSphere Commerce 5.4 モデル。
- checkParameters、checkPermission、および getResourceOwners メソッドを使用する WebSphere Commerce Suite 5.1 モデル。

表 10. アクセス・コントロール・サブシステムの相違

アイテム	Commerce Suite 5.1	WebSphere Commerce 5.4
アクセス・コントロール・モデル	<p>役割ベースのアクセス・コントロール</p> <p>Commerce Suite 5.1 では、コマンド・レベルのアクセス・コントロールは、ACCCMDGRP テーブルを使用してインプリメントされます。リソース・レベルのアクセス・コントロールは、ソース・コード内でプログラムに基づいて行われます。リソース・レベル・ポリシーを変更する場合は、ソース・コードを再コンパイルする必要があります。</p>	<p>ポリシー・ベースのアクセス・コントロール</p> <p>WebSphere Commerce 5.4 では、コマンド・レベルおよびリソース・レベルのアクセス・コントロールは、ACPOLICY テーブルを使用してインプリメントされます。ユーザーはソース・コードを再コンパイルせずにポリシーを変更できます。</p>
Databeans	方針に基づいて保護	<p>Delegator インターフェースを使用して直接および間接的に保護されます。 databean がこのインターフェースをインプリメントしない場合は、だれでもこれにデータを移植できます。さらに、 databean が Delegator インターフェースをインプリメントした場合でも、これが getDelgate メソッドで null を戻す場合は、だれでもこれにデータを移植できます。</p>
getResources()	N/A	<p>このコマンド・メソッドは、リソース・レベルのアクセス・コントロール・チェックを起動するために使用されます。</p> <p>これは、このコマンドによってアクセスされるすべての保護可能 1 次リソースを戻します。このコマンドによってアクセスされるリソースがない場合は、これはヌルを戻します。</p>

表 10. アクセス・コントロール・サブシステムの相違 (続き)

アイテム	Commerce Suite 5.1	WebSphere Commerce 5.4
getResourceOwners()	<p>デフォルト動作</p> <p>このコマンドに対し、requestProperties またはセッションで有効なストア ID パラメーターが定義されている場合に、ストアの所有者を戻します。</p> <p>使用可能なストア ID がない場合や、storeId が ECConstants.EC_NO_STOREID に設定されている場合は、EC_ACC_ALL_RESOURCES を戻します。</p>	<p>getResourceOwners() メソッドは、デフォルトで null を戻します。Commerce Suite 5.1 の動作をシミュレートするため、コマンド・レベルのアクセス・コントロール・チェックを実行するとき (つまり、コマンドが保護可能リソースとなるアクセス・チェックを実行するとき)、コマンド・フレームワークは getResourceOwners() から戻されたリソース所有者を使用します。さらにこれは、リソース所有者をコマンドの所有者として使用します。</p> <p>詳細については、142 ページの『getResourceOwners() の使用例』を参照してください。</p>
checkPermission()	<p>このメソッドは、細かいアクセス・コントロール・チェックを提供します。</p> <p>WebSphere Commerce Suite コマンド・フレームワークは、performExecute() メソッドの前にこのメソッドを呼び出します。</p> <p>管理コマンドが細かいアクセス・コントロールを持っていない場合は true を戻します。</p>	<p>checkPermission() メソッドは、下位互換性のために保持されていますが、アクセス・コントロールには使用するべきではありません。</p>
checkParameters()	<p>ここで、Commerce Suite 5.1 はパラメーターのチェックを実行します。デフォルトのインプリメンテーションでは、アクションは実行されません。</p> <p>ControllerCommandImpl および TaskCommandImpl の performExecute() は、checkParameters() を呼び出します。大半のコマンドは、checkParameters() を呼び出すために、その performExecute() の先頭行として super.performExecute() を呼び出します。</p>	<p>WebSphere Commerce 5.4 では、新しいアクセス・コントロール・モデルをサポートするために、このメソッドは validateParameters() に置き換えられています。デフォルトのインプリメンテーションでは、アクションは実行されません。下位互換性のため、ControllerCommandImpl および TaskCommandImpl の performExecute() は、checkParameters() を呼び出します。大半のコマンドは、正しいプログラム規則として super.performExecute() をその performExecute() の先頭行として呼び出します。このメソッド checkParameters() は、次のリリースでは使用できなくなります。</p>

表 10. アクセス・コントロール・サブシステムの相違 (続き)

アイテム	Commerce Suite 5.1	WebSphere Commerce 5.4
ターゲットを指定できる コマンド	<p>メソッド呼び出しの順序は以下のとおりです。</p> <pre>{ Command.checkPermission();... Command.performExecute();... Command.checkParameters(); }</pre>	<p>コマンドを WebSphere Commerce 5.4 アクセス・コントロール・モデルにマイグレーションする場合は、<code>validateParameters()</code> をインプリメントする必要があります。</p> <p>Commerce Suite 5.1 で <code>checkParameters()</code> を使用していた場合は、論理を <code>validateParameters()</code> に移し、コードから <code>checkParameters()</code> を除去します。</p> <p>メソッド呼び出しの順序は以下のとおりです。</p> <pre>{ Command.validateParameters(); Command.getResources(); Command.checkPermission(); // for backward compatibility only Command.performExecute(); Command.checkParameters(); // for backward compatibility only }</pre>
コントローラー・コマンドおよびビュー	<p>Commerce Suite 5.1 では、URLREG テーブル内で定義されていても、ACCCMDGRP テーブル内に対応するエントリーのないコントローラー・コマンドは、アクセス・コントロールの対象になりません。そのため、それらは、ゲスト・ショッパーを含めたすべてのユーザーによってアクセス可能です。同様に、VIEWREG テーブル内で定義されていても ACCCMDGRP テーブル内に対応するエントリーのないビューにも、すべてのユーザーがアクセスできます。</p> <p>注: ACCCMDGRP テーブル内で、<code>MbrGrp_Id = -2</code> (つまり Customer アクセス・グループへの割り当て) があるコントローラー・コマンドまたはビューにも、すべてのユーザーがアクセスできます。</p>	<p>WebSphere Commerce 5.4 では、アクセス・コントロール・モデルが変更されています。現在は、コントローラー・コマンドが、そのコマンドへのアクセスをすべてのユーザーに認可するアクセス・コントロール・ポリシーを明示的に持たない場合は、普通のユーザーがそのコマンドにアクセスすることはできず、サイト管理者だけがアクセスできます。同様に、ユーザーが URL からビューに直接アクセスする場合や、コマンドによってビューにリダイレクトする場合は、そのビューへのアクセスを認可する明示的なアクセス・コントロール・ポリシーが必要です。</p>

注:

1. Commerce Suite 5.1 から拡張されたコントローラー・コマンドを追加している場合、WebSphere Commerce 5.4 は、マイグレーション時にそれに対してコマンド・レベルのポリシーを追加するだけです。Commerce Suite コマンドが `getResources()` をインプリメントしている場合、それが戻すリソースを判別して、コマンドに適したリソース・レベルのポリシーを作成する必要があるか、または、リソース・レベルのアクセス・コントロールを必要としない場合は、コマンドがヌル値を戻すように `getResources()` でコマンドを指定変更する必要があるかのいずれかです。

WebSphere Commerce 5.4 コマンドがその `getResources()` に戻すものを判別するには、トレースを分析して `Action=WCBECommand` を探し、`getResources()` がチェックしているすべての保護可能リソースを見つけてください。上記のトレースでは、リソースは `Order` です。たとえば、`SERVER` トレースを使用可能にした場合のことを考慮してみましょう。ログ内には以下が示されています。

```
===== TimeStamp:    2001-11-16 02:42:30.937
Thread ID:    <Worker#3>
Component:    SERVER
Class:        AccManager
Method:       isAllowed
Trace:        isAllowed? User=10012; Action=com.fvt.ACCOrderItemAddCmd;
Protectable=com.ibm.commerce.order.objects._Order_Stub;
Owner=70000000000000002000resource is Groupable
```

```
=====
TimeStamp:    2001-11-16 02:42:30.984
Thread ID:    <Worker#3> Component:
SERVER Class: AccManager
Method:       isAllowed
Trace:        PASSED? =false
```

上記のトレースの意味は、リソース・レベルのポリシーが失敗したということです。この場合、`ACCOrderItemAddCmd` は、`getResources()` をインプリメントするサーバー `OrderItemAdd` コマンドから拡張されます。したがって、デフォルトでは、`ACCOrderItemAdd` も、それに対する `getResources()` がヌルを戻すように変更されていない限り、リソース・レベルのポリシーを必要とします。このリソース・レベルのポリシーは、マイグレーション時には、どの `WebSphere Commerce 5.4` コマンドを拡張するかが分からないので追加されません。

たいていの場合、コマンドはアクセス Bean を `getResources()` メソッドで戻します。たとえば、`com.ibm.commerce.xyz.objects.XYZAccessBean` を `getResources()` で戻すと、トレースには `com.ibm.commerce.xyz.objects._XYZ_Stub` として表されます。この違いは、`WebSphere Commerce 5.4` がアクセス Bean をそのリモート・インターフェースに狭めなければならないからです (これは実際に保護可能インターフェースに拡張する EJB のリモート・インターフェースであるため)。

2. `WebSphere Commerce Suite 5.1` では、リソース・レベルのアクセス・コントロールは、コマンド・ロジック内でプログラマチックに施行されていました。`WebSphere Commerce 5.4` では、リソース・レベルのアクセス・コントロール・ポリシーは外部的に指定されます。これはコマンド・レベルのアクセス・コントロール・ポリシーの指定方法と似ています。マイグレーション時に、コマンド・レベルのアクセス・コントロール・ポリシーは `Commerce Suite 5.1` から `WebSphere Commerce 5.4` にマイグレーションされます。`Commerce Suite 5.1` のデフォルトのアクセス・コントロール・ポリシーのカスタマイズによって必要とされるどのリソース・レベルのアクセス・コントロール・ポリシー (`ACCCMDGRP` テーブルに保管される) も、手動で追加する必要があります。そうしない場合は、予期しないアクセス・コントロール違反例外を受け取ります。詳細については、217 ページの『付録 H. トラブルシューティング』の関連項目を参照してください。

getResourceOwners() の使用例

WebSphere Commerce Suite 5.1 コマンド

- デフォルト動作に依存するコマンド

これらのコマンドは、`getResourceOwners()` をインプリメントせず、デフォルトでは `null` を返します。

コマンド・レベルのアクセス・コントロール・チェックの実行時に、コマンド・フレームワークは、コマンド所有者としてストア所有者を返します。使用可能なストア ID がない場合、これは `EC_SITE_ORGANIZATION` を返します。

- `getResourceOwners()` メソッドをインプリメントするコマンド

コマンド・レベルのアクセス・チェックの実行時に、コマンド・フレームワークは、`getResourceOwners()` によって戻されるそれぞれのリソース所有者に対して、コマンドのアクセス・チェックを実行します。たとえば、

`getResourceOwners()` メソッドが組織 1 と組織 2 という 2 つの所有者を返す場合、コマンド・フレームワークは、まず組織 1 を所有者としてコマンドのアクセス・チェックを実行します。このチェックがパスすると、今度は組織 2 をコマンド所有者として使って、同じコマンドのチェックを再度実行します。この両方のチェックをパスしなければなりません。

新しい WebSphere Commerce 5.4 コマンド

- これらのコマンドは、`getResourceOwners()` をインプリメントせず、デフォルトでは `null` を返します。
- コマンド・レベルのアクセス・コントロール・チェックの実行時に、コマンド・フレームワークは、コマンド所有者としてストア所有者を返します。使用可能なストア ID がない場合、これは `EC_SITE_ORGANIZATION` を返します。

第 9 章 Payment コンポーネントのマイグレーション考慮事項

このセクションでは、Payment コンポーネントのマイグレーション考慮事項およびアクションについて説明しています。

PAYMTHD テーブルから支払ポリシーへのマイグレーション

Commerce Suite 5.1 は、3 つのデータベース・テーブルを使用して、ストアまたはストア・グループによってサポートされる支払メソッドを定義します。それらのテーブルは以下のとおりです。

PAYMTHD

支払メソッド・テーブルは、サイト単位のテーブルで、モール内で使用されるすべてのキャッシャー・プロファイルをリストしています。個々のプロファイルには固有の整数 ID および名前があります。

PAYMTHDDSC

支払メソッド説明テーブルは、サイト単位のテーブルで、サポートされている言語での、各 Commerce Suite 5.1 キャッシャー・プロファイルの簡略説明が含まれています。

PAYMTHDSUP

サポート支払メソッド・テーブルには、ストアまたはストア・グループによってサポートされているすべてのプロファイルがリストされています。

Commerce Suite 5.1 には、PAYMTHD テーブル内に 5 つのエントリーと、5 つのキャッシャー・プロファイルが同梱されています。以下の表は、PAYMTHD テーブル内の 5 つのエントリーの要約です。

PAYMTHD_ID	PROFILENAME	注釈
100	WCS51_CustomOffline	CustomOffline Cassette 用の標準 Commerce Suite 5.1 プロファイル。
200	WCS51_OfflineCard	OfflineCard Cassette 用の標準 Commerce Suite 5.1 プロファイル。
300	WCS51_SET_MIA	MIA (Merchant Initiated Authorization) SET 拡張を使用する Cassette for SET (Secure Electronic Transactions) の標準 Commerce Suite 5.1 プロファイル。
400	WCS51_SET_Wallet	ウォレットを使用する Cassette for SET の標準 Commerce Suite 5.1 プロファイル。
500	WCS51_CyberCash	Cassette for CyberCash の標準 Commerce Suite 5.1 プロファイル。

これらの支払メソッドは、WebSphere Payment Manager がサポートする支払メソッドに限定されます。

ビジネス・ポリシーおよびビジネス・ポリシー・コマンド

WebSphere Commerce 5.4 は、ビジネス・ポリシー およびビジネス・ポリシー・コマンド の概念を導入しています。

ビジネス・ポリシーの 1 つのカテゴリは、支払ビジネス・ポリシー (略して支払ポリシー) です。支払ポリシーは、そのビジネス・ポリシーに関連したビジネス機能を実行するために、WebSphere Commerce 5.4 が呼び出すビジネス・ポリシー・コマンド・インターフェースのセットを定義します。それぞれの支払ポリシーは、それぞれのビジネス・ポリシー・コマンド・インプリメンテーションを持つことができます。

WebSphere Commerce 5.4 支払ポリシーは、Payment Manager がサポートする支払メソッドに限定されないため、Commerce Suite 5.1 に定義される支払メソッドよりも一般的です。

Commerce Suite 5.1 から WebSphere Commerce 5.4 にマイグレーションするとき、新しい WebSphere Commerce 5.4 の機能または支払動作を使用したい場合は、OrderProcess コマンドに payMethodId を指定する代わりに、policyId を指定する必要があります。以下の表を使用して適切な変更を行ってください。

Commerce Suite 5.1		WebSphere Commerce 5.4		
支払メソッド ID	プロファイル名	ポリシー ID	ポリシー名	プロファイル名
100	WCS51_CustomOffline	xxx ¹	CustomOffline_COD	WC51_CustomOffline_COD
		xxxy ¹	CustomOffline_BillMe	WC51_CustomOffline_BillMe
200	WCS51_OfflineCard	200	OfflineCard	WC51_OfflineCard
300	WCS51_SET_MIA	300	SET_MIA	WC51_SET_MIA
		301	SET_MIA_PCard	WC51_SET_MIA_PCard
400 [®]	WCS51_SET_Wallet	400	SET_Wallet	WC51_SET_Wallet
500	WCS51_CyberCash	500	CyberCash	WC51_CyberCash
		600	VisaNet	WC51_VisaNet
		601	VisaNet_PCard	WC51_VisaNet_PCard
		700	BankServACH	WC51_BankServACH

注: ¹ WCS51_CustomOffline プロファイル名のポリシーは提供されていません。ユーザーが独自に作成する必要があります。WebSphere Commerce 5.4 オンライン・ヘルプの『新規支払ビジネス・ポリシーの追加』のセクションを参照してください。

注: 事前定義支払ポリシーの policyId は、Commerce Suite 5.1 で使用される PayMethods の payMethodId と同じであるため、同じ値を使用できるということに注意してください。Commerce Suite 5.1 または WebSphere Commerce 5.4 のどちらの支払動作をインプリメントするかは、OrderProcess コマンドを呼び出すときに適切なパラメーターを使用することによって選択できます。

- payMethodId パラメーターを使用すると、PAYMTHD、PAYMTHDDSC、および PAYMTHDSUP テーブルを使用する Commerce Suite 5.1 の動作に合致する DoPaymentMPFCmdImpl クラスが呼び出されます。これは、ユーザーが WebSphere Commerce 5.4 で使用可能な、ATP 在庫、フルフィルメントのためのリリース、Balance[®] Payment などの、新しい機能またはコマンドを使用

しないことを前提としています。新しい機能またはコマンドを使用するには、payMethodId パラメーターの使用から policyId パラメーターの使用に切り替える必要があります。

- policyId パラメーターを使用すると、POLICY および POLICYCMD テーブルを使用する WebSphere Commerce 5.4 動作に合致する DoPaymentMPFCmdImpl クラスが呼び出されます。

たとえば、マイグレーションした InFashion ストアで、ATP を使用する場合、OrderDisplayPending.jsp の以下の行を置き換えます。

```
<input type=hidden name="<%= ECConstants.EC_PAYMTHDID %>" value="200">
```

これを以下の行に置き換えます。

```
<input type=hidden name="policyId" value="200">
```

そうしないと、PickPatches などの一部の機能が Commerce Accelerator で機能しないことがあります。

また、Commerce Suite 5.1 で ProfileCassetteAccountDataBean データ Bean を使用している場合は、WebSphere Commerce 5.4 では UsablePaymentTCListDataBean データ Bean を使用するように切り替える必要があります。ProfileCassetteAccountDataBean データ Bean は、Commerce Suite 5.1 テーブルの PAYMTHD、PAYMTHDDSC、および PAYMTHDSUP を使用します。UsablePaymentTCListDataBean データ Bean は、新規の WebSphere Commerce 5.4 テーブル POLICY および POLICYCMD を使用します。

OrderProcessCmd コントローラー・コマンドを使用してオーダーを処理する場合は、WebSphere Commerce 5.4 の DoPaymentCmd タスク・コマンド用の標準インプリメンテーション・クラスである、DoPaymentMPFCmdImpl クラスが呼び出されます。WebSphere Commerce 5.4 での支払いの処理方法の詳細については、オンライン・ヘルプの WebSphere Payment Manager についてのセクションを参照してください。上記のインターフェースの詳細については、オンライン・ヘルプを参照してください。

Business 指定の支払条件がある契約を使用する B2B ストアにマイグレーションする場合、アカウント、契約、および支払条件を作成するには、WebSphere Commerce 5.4 Commerce Accelerator を使用します。その場合は、使用する支払条件を識別するための tcId パラメーターも必要になります。支払条件に関連した tcId を戻すには、UsablePaymentTCListDataBean データ Bean を使用します。

支払ポリシーの追加情報については、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。WebSphere Commerce 5.4 には、ここでリストされているものに加えて、いくつかのその他の支払ポリシーがあります。

支払い用の WebSphere Commerce 5.4 ビジネス・ポリシー・コマンドへのマイグレーション

WebSphere Commerce 5.4 は、支払ポリシーに対して、以下の一連のビジネス・ポリシー・コマンド・インターフェースを指定します。

- DoPaymentPolicyCmd
- CheckPaymentAcceptPolicyCmd
- DoDepositPolicyCmd
- DoRefundPolicyCmd
- DoCancelPolicyCmd

それぞれの支払ポリシーは、これらのコマンドごとに異なるインプリメンテーションを持つことができます。

WebSphere Commerce 5.4 は、WebSphere Commerce 5.4 に含まれる支払ポリシー用の 2 セットのインプリメンテーション・クラスを定義しています。一方のセットは、WebSphere Payment Manager に基づく支払ポリシーをサポートし、他のセットは、WebSphere Payment Manager に基づかない信用限度支払ポリシーをサポートします。WebSphere Payment Manager に基づく支払ポリシー用のビジネス・ポリシー・コマンドのインプリメンテーション・クラスは、以下のとおりです。

- DoPaymentPMCImpl
- CheckPaymentAcceptPMCImpl
- DoDepositPMCImpl
- DoRefundPMCImpl
- DoCancelPMCImpl

信用限度支払ポリシー用のビジネス・ポリシー・コマンドのインプリメンテーション・クラスは、以下のとおりです。

- DoPaymentCLCmdImpl
- CheckPaymentAcceptCLCmdImpl
- DoDepositCLCmdImpl
- DoRefundCLCmdImpl
- DoCancelCLCmdImpl

使用される支払ポリシーに応じて、適切なビジネス・ポリシー・コマンドのインプリメンテーション・クラスが呼び出されます。

上記のインプリメンテーション・クラスの詳細については、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。

DoCancelCmd の CMDREG エントリーの変更

Commerce Suite 5.1 で InFashion ストアを使用する場合や、WebSphere Payment Manager を使用する独自のストアを作成した場合、これを WebSphere Commerce 5.4 で機能させるためには、DoCancelCmd の CMDREG エントリーを変更する必要があります。Commerce Suite 5.1 では、DoCancelCmd は、WebSphere Payment Manager が使用されている場合に、クラス

com.ibm.commerce.payment.commands.DoCancelPMCmdImpl に割り当てられます。
WebSphere Commerce 5.4 では、DoCancelCmd は、クラス
com.ibm.commerce.payment.commands.DoCancelCmdImpl に割り当てられる必要があり
ます。 DoCancelCmdImpl インターフェースは、呼び出しを、支払いに使用される
支払ポリシーに応じて DoCancelPMCmdImpl または DoCancelCLCmdImpl に経路指定
します。

78 ページの『データベース・スキーマのマイグレーション』で説明されているデー
タ・マイグレーション・スクリプトを実行している場合、この変更は自動的に行わ
れることに注意してください。

サンプル JavaServer Pages ファイル - PayStatusPM.jsp

/opt/WebSphere/CommerceServer/samples/web/payment ディレクトリー内のサンプ
ル JavaServer Pages ファイル PayStatusPM.jsp は、Payment Manager 3.1.2 用にい
くらか更新されています。この変更は、JSP がオーダーの支払いの状態に関連した
正しい状況メッセージを生成するために必要なものです。

前のリリースでは、PayStatusPM.jsp は、Payment Manager オーダーの状態
Refundable を、そのオーダーの支払いが、承認済み状態の先にまで進んでいること
を示すものとして扱います。したがって、これはショッパーに対してオーダーが
「承認された.....」ことを保証する状況メッセージを生成します。これは、
Commerce Suite 5.1 に同梱されている Payment Manager カセットでも同様でした。

しかし、Payment Manager 3.1 では、一部のカセットは Payment Manager オーダー
が作成されるとすぐに、そして支払いが承認される前に、Payment Manager オーダ
ーの状態を Refundable 状態に設定します。このため、正しい状況メッセージを生
成するには、PayStatusPM.jsp を変更してこれを使用可能化する必要があります。

この動作を表す Payment Manager カセットは以下のとおりです。

- Cassette for CyberCash
- Cassette for VisaNet
- CustomOffline Cassette
- OfflineCard Cassette

独自のバージョンの PayStatusPM.jsp を持っており、WebSphere Commerce 5.4
で上記のカセットを使用することを計画している場合は、対応する変更を JSP に加
えることによって、ショッパーがページを表示するときにショッパーに正しい状況
メッセージが表示されるようにする必要があります。(Payment Manager オーダー
の状態 Refundable は、支払いがすでに承認済み という意味ではない場合もあるこ
とに留意してください。)

詳細については、WebSphere Commerce 5.4 で提供されている PayStatusPM.jsp フ
ァイルを参照してください。これは、
/opt/WebSphere/CommerceServer/samples/web/payment ディレクトリーにありま
す。

第 10 章 その他のマイグレーション考慮事項

この章では、注意を向け、必要に応じてアクションを取る必要がある、いくつかの WebSphere Commerce 5.4 のその他の領域について説明しています。

デフォルトの通貨の動作

顧客がショッピングで希望する通貨を選択できるようにするために、サポートされる支払通貨のリストを、ストア・ページ上に組み込むことができます。

- 顧客の希望する通貨がストアでサポートされている場合、その通貨がショッピング通貨になります。
- 希望通貨がサポートされていない場合に、顧客の希望する通貨を有効なカウンター値として持つ別の通貨があれば、その通貨がショッピング通貨として使用されます。
- 希望する通貨をカウンター値として持つ別の通貨がなければ、ストアのデフォルト通貨は、STOREENT テーブルの新しい SETCURR 列から決定されます。この設定は、顧客の言語 ID には依存していません。この動作は、WebSphere Commerce 5.4 で新しくなった点であることに注意してください。
- STOREENT テーブルで指定されているデフォルト通貨が他になければ、ストアの顧客の言語 ID 用のデフォルト通貨が使用されます。この場合は、STORELANG データベース・テーブルが設定を決定します。

注:

1. 希望する通貨がないショッパーや、非サポートの希望する通貨 (サポートされている通貨用のカウンター値ではない) があるショッパーだけが、この変更の影響を受けます。ショッパーに、サポートされている希望する通貨がある場合は、どの言語を選択しているとしても、常にこの通貨が表示されます。
2. 以前の Commerce Suite 5.1 のデフォルト通貨を WebSphere Commerce 5.4 で保持する場合は、STOREENT テーブルにストアのデフォルト通貨を設定しないでください。新しい WebSphere Commerce 5.4 のデフォルト通貨をインプリメントするには、STOREENT テーブルにストアのデフォルトを設定します。Commerce Suite 5.1 の STORELANG テーブル・パラメーターを変更する必要はありません。したがって、マイグレーション済みのどのストアに新しいデフォルトの通貨の動作をインプリメントし、どのストアに以前の動作を保持するかを選択できます。
3. STORELANG テーブルの SETCURR 列は、将来使用されなくなる可能性があるため、ヌルに設定することをお勧めします。
4. STOREENT テーブル内のストアまたはストア・グループの SETCCURR 列を設定します。マイグレーションされたストアの場合は、最初はこれは当てはまりません。新しいストアはどれも、ストアまたはストア・グループのデフォルト通貨を設定する必要があります。

価格設定のための考慮事項

以下に示すのは、Commerce Suite 5.1 の価格設定用コマンドと置き換えられた、WebSphere Commerce 5.4 の新しいコマンドおよびメソッドです。

タスク・コマンド

- `GetBaseUnitPriceCmd` は置き換えられて、`GetContractUnitPriceCmd` になりました。
- `GetBaseSpecialPriceCmd` は置き換えられて、`GetContractSpecialPriceCmd` になりました。
- `GetProductContractUnitPriceCmd` は置き換えられて、`GetProductBaseUnitPriceCmd` になりました。

注: 下位互換性のため、Commerce Suite 5.1 コマンドは WebSphere Commerce 5.4 でも有効です。

データ Bean

以下のデータ Bean の場合、Commerce Suite 5.1 では、価格を検索するためにメソッド `getCalculatedPrice()` を使用することができました。WebSphere Commerce 5.4 では、これは新しいメソッド `getCalculatedContractPrice()` に置き換えられます。

- `ItemDataBean`
- `PackageDataBean`
- `ProductDataBean`
- `CatalogEntryDataBean`
- `InterestItemDataBean`
- `BundleDataBean`

注: 下位互換性のため、Commerce Suite 5.1 メソッドは WebSphere Commerce 5.4 でも有効です。

上記のコマンドまたはメソッドの詳細については、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。

第 3 部 付録

付録 A. WebSphere Commerce Suite 5.1 コンポーネントの開始と停止

この付録では、WebSphere Commerce Suite パッケージの一部として提供されている各製品や Domino™ Web Server (オプションの Web サーバー) を開始および停止する方法について説明します。この付録では、WebSphere Commerce Suite パッケージの一部として提供されている各製品を開始および停止する方法について説明します。この情報は、何らかのコンポーネントを再始動する必要がある場合に使用してください。

WebSphere Commerce Suite の開始と停止

WebSphere Commerce Suite インスタンスを開始または停止するには、以下のようにします。

1. ユーザー ID root でログインします。
2. データベース管理システムと WebSphere Application Server が開始されていることを確認してください。DB2 を使用している場合は、155 ページの『DB2 ユニバーサル・データベースの開始と停止』を参照してください。WebSphere Application Server については、『WebSphere Application Server の開始と停止』を参照してください。
3. 端末ウィンドウで以下のように入力し、WebSphere Application Server 管理コンソールを立ち上げます。

```
cd /opt/WebSphere/AppServer/bin
./adminclient.sh
```
4. 「WebSphere 管理可能ドメイン」を拡張表示します。
5. ホスト名を拡張表示します。
6. 「WebSphere Commerce Server」 — 「instance_name」 を選択してこれを右クリックします。適宜「開始」または「停止」を選択します。

WebSphere Application Server の開始と停止

WebSphere Application Server を開始するには、ユーザー ID root でログインして以下を行います。

1. データベース管理システムが開始されていることを確認します。
2. 端末ウィンドウで以下のコマンドを入力します。

```
cd /opt/WebSphere/AppServer/bin
./startupServer.sh &
```

/opt/WebSphere/AppServer/logs/tracefile を調べて、WebSphere Application Server が正常に開始されたかどうかを確認してください。

WebSphere Application Server を停止させるには、ユーザー ID root でログインして以下を行います。

1. 端末ウィンドウで以下のように入力して、WebSphere Application Server 管理クライアントを開始します。

```
cd /opt/WebSphere/AppServer/bin
./adminclient.sh
```

2. WebSphere Application Server 管理コンソールで、使用しているホストの非修飾ホスト名が付いたノードを選択します。

3. 「停止」 ボタンをクリックします。次のようなメッセージが戻されます。

You are trying to stop the node that the console is connected to. This will cause the console to exit after the node is stopped. Do you want to continue? (コンソールの接続先のノードを停止させようとしています。このまま処理を続けると、ノードが停止した後コンソールは終了します。処理を続けてもよろしいですか?)

「はい」 をクリックして先に進みます。

4. WebSphere Application Server 管理コンソールの処理が終わった後、端末ウィンドウから以下のコマンドを実行して、 WebSphere Application Server 関連のプロセスがすべて停止したことを確認してください。

```
ps -ef | grep IBMWebAS
ps -ef | grep startupServer
```

5. このコマンドが何らかの Java プロセスを戻した場合は、検出されたプロセスごとに kill コマンドを出して、それを停止させてください。

IBM HTTP Server の開始と停止

IBM HTTP Server を開始するには、ユーザー ID root でログオンして、端末ウィンドウから以下のコマンドを入力します。

```
cd /opt/IBMHTTPServer/bin
./apachectl start
```

IBM HTTP Server を停止させるには、ユーザー ID root でログオンして、以下を行います。

1. WebSphere Commerce Suite と WebSphere Application Server が停止していることを確認します。

2. 端末ウィンドウから以下のコマンドを入力します。

```
cd /opt/IBMHTTPServer/bin
./apachectl stop
```

IBM HTTP Administration Server を開始するには、ユーザー ID root でログオンして、端末ウィンドウから以下のコマンドを入力します。

```
cd /opt/IBMHTTPServer/bin
./adminctl start
```

IBM HTTP Administration Server を停止させるには、ユーザー ID root でログオンして、端末ウィンドウから以下のコマンドを入力します。

```
cd /opt/IBMHTTPServer/bin
./adminctl stop
```

DB2 ユニバーサル・データベースの開始と停止

DB2 を開始するには、以下のようにします。

1. ご使用の DB2 インスタンス ID でログオンします。
2. db2start と入力します。

DB2 を停止させるには、以下のようにします。

1. 153 ページの『WebSphere Commerce Suite の開始と停止』にある指示に従って Commerce Suite を停止させます。
2. ご使用の DB2 インスタンス ID でログオンされているときに、db2stop と入力します。何か DB2 に接続されているアプリケーションがある場合は、代わりに以下のコマンドを実行します。

```
db2stop force
```

WebSphere Payment Manager の開始と停止

WebSphere Payment Manager を開始するには、まずデータベースが開始されていることを確認してから、『WebSphere Payment Manager Engine の開始』で説明されているように Payment Engine を開始し、次いで Web サーバーを開始します。WebSphere Application Server 4.0.2 が稼働していることを確認してください。

WebSphere Payment Manager Engine の開始

IBMPayServer スクリプトを使用して Payment Engine を開始するには、データベース管理者のパスワードを指定する必要があります。データベース管理者のパスワードは、コマンド行で引き数として指定することもできますし、あるいはもっと安全に、Payment Manager のインストール・ディレクトリーにある .payment ファイルを使用することもできます。コマンド行でパスワードを指定する場合、Payment Engine を開始するためのパスワードは、Payment Manager がインストールされているマシンへのアクセス権を持つすべてのユーザーが見ることができます。コマンド行でデータベース管理者のパスワードが入力されないと、Payment Manager は .payment ファイルを読み取ってパスワードを取得しようとします。

注: Payment Manager データベースに DB2 ユニバーサル・データベースを使用するときは、DB2 ユニバーサル・データベースでは常にオペレーティング・システムのユーザー ID とパスワードがデータベース管理者のユーザー ID とパスワードとして使用されることを覚えておいてください。それで、Payment Engine の始動時にデータベース管理者のパスワードを求める (.payment ファイル内で指定することも、コマンド行で入力することも可能) プロンプトが出されたなら、オペレーティング・システムのデータベース管理者パスワードを入力してください。

.payment ファイルを使用した Payment Engine の開始

パスワードの取得に .payment ファイルを使用すると、Payment Manager マシンへのアクセス権を持つすべてのユーザーからパスワードを、ひいては支払データを保護することができます。 .payment ファイルは、Payment Manager のインストール・プログラムによって、Payment Manager がインストールされるディレクトリー

に作成されます (古いバージョンからマイグレーションする場合を除く)。 .payment ファイルは読み取り専用のファイルとして作成され、オペレーティング・システムにおいて可能な限り隠されます。

.payment ファイルを使用して Payment Engine を開始するには、以下のようにします。

1. .payment ファイルのデータベース・パスワードに誤りがないことを確認してください。
2. 端末ウィンドウで以下のコマンドを入力します。

```
cd /opt/PaymentManager
./IBMPayServer
```

注: これらのステップを使用した場合、Payment Engine は、ユーザーがログオフすると停止します。ユーザーがログオフした後もバックグラウンドで Payment Engine を稼働させておきたいときは、 **nohup** コマンドを使用して nohup ./IBMPayServer & のようにします。 **nohup** コマンドを使用する場合、Payment Engine によって STDOUT に書き込まれるすべてのメッセージは、nohup.out ファイルに置かれます。このファイルは、**nohup** コマンドが実行されたディレクトリーにあります。

WebSphere Payment Manager ユーザー・インターフェースの開始

WebSphere Payment Manager Engine と WebSphere Application Server を開始した後、 WebSphere Payment Manager ユーザー・インターフェースを開始するには、以下のようにします。

1. Web ブラウザーで次のアドレスを参照します。

```
http://<host_name>/PaymentManager/
```

<host_name> には、Payment Manager のインストール先のマシンが入ります。

2. Payment Manager ログオン・ウィンドウで、Payment Manager 管理者のユーザー ID およびパスワードを入力して、「OK」をクリックします。デフォルトのユーザー ID とパスワードはいずれも wcsadmin です。Payment Manager ユーザー ID の作成に関する詳細は、 *IBM WebSphere Payment Manager for MultiPlatforms 管理者ガイド バージョン 2.2.1* を参照してください。

WebSphere Payment Manager ユーザー・インターフェース機能は、一部 WebSphere Commerce Suite の管理コンソールからでも使用できます。

WebSphere Payment Manager の停止

WebSphere Payment Manager を停止するには、次のようにする必要があります。

- Payment Engine を停止させます。
- WebSphere Application Server の下の Payment Manager アプリケーション・サーバーを停止させます。
- 他のすべての WebSphere Payment Manager サーブレットを停止させます。

Payment Engine の停止

Payment Engine は、StopIBMPayServer コマンドを使用して停止させることができます。 StopIBMPayServer スクリプトには、引き数はありません。このコマンドを使用するためには、 Payment Manager のインストール・ディレクトリーに .payment ファイルが必要です。 StopIBMPayServer コマンドは /opt/PaymentManager にあります。

StopIBMPayServer コマンドを使用しないで Payment Engine を停止するには、以下のようにします。

1. **ps** コマンドを使って Payment Engine のプロセス ID を確認します。
2. **kill** コマンドを使ってそのプロセス ID を停止します。

WebSphere Application Server を使った Payment Manager サブレットの停止

WebSphere Application Server 3.5 を使用しているときは、WebSphere Payment Manager アプリケーション・サーバーを停止させることによって、すべてのサブレットを停止させることができます。WebSphere Payment Manager アプリケーション・サーバーを停止させるには、以下のようにします。

1. WebSphere Application Server Administration Client に移動します。
2. WebSphere Payment Manager アプリケーション・サーバーを選択します。
3. アプリケーション・サーバーを右クリックして、「**停止**」を選択します。
4. Administration Client を終了します。

付録 B. データ・マイグレーション・スクリプトの拡張

標準の WebSphere Commerce Suite 5.1 データベース・スキーマを拡張した場合は、提供されているデータ・マイグレーション・スクリプトに以下のような変更を加える必要があります。ここでは例として、INTEGER DEFAULT 0 と FRIEND が MBRGRP の外部キーになっていて、標準の USERREG テーブルに FRIEND という列を追加したケースを考えてみましょう。

提供されているマイグレーション・スクリプトを使用して自動データ・マイグレーションを開始する前に、以下のようにします。

1. **DB2 データベースの場合**は、/opt/WebSphere/CommerceServer/schema/db2 ディレクトリーに移動します。
2. wcs.schema.sql ファイルのバックアップ・コピーを作成します。
3. この wcs.schema.sql ファイルを編集します。CREATE TABLE USERREG ステートメントを見つけてください。

```
CREATE TABLE USERREG (  
    ...  
);
```

4. この CREATE TABLE USERREG ステートメントに、次のようにして FRIEND という新しい列を追加します。

```
CREATE TABLE USERREG (  
    ...  
    FRIEND INTEGER DEFAULT 0  
);
```

列を追加するときには、ファイルのフォーマットが失われないようにしてください。データベース・マイグレーション・スクリプトは、このファイルを構文解析するとき、テーブル内の各列定義が固有の行であること、および行と行の間に空白行が入っていないことを前提として処理を行います。(これには、括弧も含まれます。)

5. 同じディレクトリー内に wcs.referential.sql ファイルのバックアップ・コピーを作成します。
6. この wcs.referential.sql ファイルを編集して、ファイルに参照制約を追加します。

```
ALTER TABLE USERREG ADD  
    CONSTRAINT F_FRIEND FOREIGN KEY (FRIEND) REFERENCES MBRGRP ON DELETE CASCADE
```

7. テーブルを調べて、標準の WebSphere Commerce スキーマ・テーブルを指す、何らかの新規テーブルからの制約を他に作成していないかどうか確認してください。そのような制約がある場合は、データ・マイグレーション・スクリプトを実行する前にこれを除去する必要があります。

注: これらの制約は、マイグレーションが完了した後、自分で作成し直す必要があります。

8. /opt/WebSphere/CommerceServer/schema/migration/FromRelease ディレクトリーにある MapFile.txt を調べてください。ここで FromRelease は、マイグレーション元のバージョンが以下のように示されます。

- 5.1.0.1 (WebSphere Commerce Suite 5.1 の場合)
- 20011015 (WebSphere Commerce Business Edition 5.1 の場合)
- 20011215 (WebSphere Commerce 5.4 Limited Availability リリースの場合)

変更したテーブルが **MapFile.txt** にリストされている場合は、ソースとターゲットの両方の列のリストに付加的な列を追加する必要があります。たとえば、MapFile.txt の ADDRBOOK テーブルのエントリーについて考えてみましょう。(以下の例では、表示上の都合で 2 行に分かれています。)

```
addrbook=(addrbook_id, description, displayname, member_id, type);
          (ADDRBOOK_ID, DESCRIPTION, DISPLAYNAME, MEMBER_ID, TYPE)
```

括弧で囲まれた 1 つ目の文字ストリングの集合は、Commerce Suite 5.1 データベースの ADDRBOOK テーブルの列を表しており、括弧で囲まれた 2 つ目の文字ストリングの集合は、WebSphere Commerce 5.4 データベースの ADDRBOOK テーブルの列を表しています。ここで、Commerce Suite 5.1 の ADDRBOOK テーブルに FRIEND 列を追加するとします。そうした場合、MapFile.txt では、以下に太字で示されているように FRIEND 列をアドレッシングするよう、編集を加えることが必要になります。

```
addrbook=(addrbook_id, description, displayname, member_id, type, friend);
          (ADDRBOOK_ID, DESCRIPTION, DISPLAYNAME, MEMBER_ID, TYPE, FRIEND)
```

これを行わないと、マイグレーション・スクリプトは FRIEND 列を WebSphere Commerce 5.4 にマイグレーションしません。

9. 67 ページの『第 4 章 データベースのマイグレーション』の説明に従ってマイグレーション・スクリプトを実行してください。

付録 C. マイグレーション・スクリプトの概要

このセクションでは、マイグレーション・スクリプトが何を実行し、どのような情報が組み込まれるのかについて説明しています。ユーザーが取るべきアクションはありません。

データベース・マイグレーション・スクリプトは、ユーザーが指定するディレクトリーにデータベースのバックアップ・コピーを作成してから、以下のサブシステムまたはコンポーネントをマイグレーションします。

- メンバー
- カタログ
- ATP 在庫
- オーダー・アイテム
- 契約
- キャンペーン
- アクセス・コントロール

メンバーのマイグレーション

マイグレーション・スクリプトは、メンバー・サブシステムに対して以下の更新を実行します。

- スクリプトは、MEMBER テーブルの状態を以下のように設定します。
 - スクリプトは、以下に対しては状態をヌル (状態なし) に設定します。
 - ゲスト・ユーザー (ユーザー登録タイプが G に設定されている)
 - メンバー・グループ
 - スクリプトは、以下に対しては状態を承認済み (1) に設定します。
 - 登録済みユーザー (ユーザー登録タイプが R に設定されている)
 - サイト管理者 (ユーザー登録タイプが S に設定されている)
 - 管理者 (ユーザー登録タイプが A に設定されている)
 - 組織エンティティー

MEMBER テーブルの状況には、以下のようなものがあります。

- 0** 承認保留中
- 1** 承認済み
- 2** 拒否済み

- MBRGRP テーブルの OWNER_ID 列が設定されていない (つまり値 0 が含まれている) 場合は、これを -2001 (Root Organization) に設定します。
- ユーザーのプロファイル・タイプを設定します。
 - Commerce Suite 5.1 でユーザー・タイプ S または A (Site Administrator または Administrator 役割) として登録されているユーザーに対しては、スクリプトは PROFILETYPE を B に設定します。

- Commerce Suite 5.1 でビジネス・プロフィールを持つ (つまり、BUSPROF テーブルが設定されている) ユーザーと、ヌルの PROFILETYPE を持つユーザーに対しては、スクリプトはこれらを B2B ユーザーと見なすため、PROFILETYPE を B に設定します。

たとえば、ユーザーの PROFILETYPE が C (B2C ユーザー) に設定されている場合は、スクリプトはプロフィール・タイプをリセットしません。

- MBRGRP テーブルをチェックします。

MBRGRPUSG テーブルに AccessGroup の MBRGRPTYPE_ID がある場合、以下の場合を除いて、スクリプトは対応する役割を ROLE テーブル内に作成します。

- MBRGRPTYPE_ID が -2 (CustomerGroup) に設定されている場合 (WebSphere Commerce 5.4 では顧客グループは役割ではないため)。
- Commerce Suite 5.1 で Order Clerk 役割が使用されている場合、 WebSphere Commerce 5.4 では、マイグレーション・スクリプトはこれをユーザー定義の役割にマイグレーションします。

スクリプトは、WebSphere Commerce 5.4 の ROLE テーブル内のすべての役割を MBRROLE テーブルに移動し、 MEMBER_ID を値 -2001 (Root Organization) に設定します。ルート組織はこれらのすべての役割にアクセスできます。

- ORGENTITY テーブルをチェックし、MEMBER_ID がヌルの場合、親 MEMBER_ID を -2001 (Root Organization) に設定します。
- ユーザー登録タイプが S であるすべてのユーザーをチェックします。スクリプトは以下を行います。
 - MBRROLE テーブル内に、Site Administrator 役割が -1 に設定されたエントリーを作成します。
 - すべての親および祖先に同じ役割が割り当てられるようにします。
 - ユーザーの登録タイプが A である場合、スクリプトはそれらを明示的に管理者グループ (管理者グループに関連付けられたいくつかの事前定義役割がある) にマイグレーションします。
 - MBRGRPMBR テーブル内に、新規管理グループを指す MBRGRP_ID があるエントリーを作成します。
- ACCMBRGRP テーブル内の各レコードに対して、スクリプトは以下を行います。
 - MBRROLE テーブルにレコードを追加します。
 - 管理者が属する親の組織エンティティ用の MBRROLE テーブルに、追加レコードを追加します。 OWNER_ID が 0 の場合、スクリプトはこれを -2001 に設定します。
- MBRREL テーブルを作成します。ただしこれは、登録済みユーザーに対してのみ作成し、 ORGENTITY テーブル内の MEMBER_ID が 0 (ゼロ) に設定されているゲスト・ユーザーに対しては作成しません。
- SQL ステートメントを生成して、ORGENTITY テーブルの DN を充てんします。スクリプトは fillorgDN.sql というファイルを作成します。 ORGENTITY テーブルを更新するには、このテーブルの DN (識別名) 列を手動で更新するか、またはこのファイルを使用します。詳細については、83 ページの『識別名の更新』を参照してください。

カタログのマイグレーション

マイグレーション・スクリプトは、カタログ・サブシステムに対して以下の更新を実行します。

- 親を持たないアイテムに対して親商品を作成します。
 - これはその商品タイプの CATENTRY テーブル・エントリーを作成し、そのアイテムの CATENTRY からそのエントリーの値をコピーします。
 - CATENTDESC テーブル・エントリー (カタログ・エントリーの説明) を作成します。
 - CATENTREL テーブル・エントリー (商品とアイテムとの関係) を作成します。
- 商品とアイテムのフルフィルメント・エントリーを作成します。
 - 各商品に対して BASEITEM、BASEITEMDSC、ITEMVERSN、および STOREITEM テーブル・エントリーを作成します。
 - 各アイテムに対して ITEMSPC および VERSIONSPC テーブル・エントリーを作成します。
- パッケージのフルフィルメント・エントリーを作成します。
 - 各パッケージに対して BASEITEM、BASEITEMDSC、ITEMVERSN、および STOREITEM テーブル・エントリーを作成します。
 - 各パッケージに対して ITEMSPC および VERSIONSPC テーブル・エントリーを作成します。

ATP 在庫のマイグレーション

マイグレーション・スクリプトは、在庫サブシステムに対して以下の更新を実行します。

- 新しい ATP 在庫表記にマイグレーションすることを選択した場合、スクリプトは ALLOCATIONGOODFOR を 43200 の値に更新して、ATP サポートを使用可能にします。

新しい ATP 在庫表記にマイグレーションしないことを選択した場合、スクリプトは、ALLOCATIONGOODFOR を値 0 に更新して ATP 在庫のマイグレーションを延期し、INVENTORY テーブルを使用して、Commerce Suite 5.1 と同様の方法での在庫のトラッキングを継続します。

この値の意味を理解するには、WebSphere Commerce 5.4 オンライン・ヘルプにある STORE テーブル用のデータベース・スキーマの資料を参照してください。非ゼロ値があれば、ATP 在庫は使用可能になります。

- 各商品ごとに DISTARRANG テーブル用のエントリーを作成します。これは、ENDDATE に大きな値 (59 年) を、および STARTDATE に現在日付を指定します。
- 各アイテムごとに、ITEMFFMCTR、RADETAIL、RECEIPT、および RCTAVAIL テーブル用のエントリーを作成します。
- 各パッケージごとに、ITEMFFMCTR、DISTARRANG、RADETAIL、RECEIPT、RCTAVAIL テーブル用のエントリーを作成します。

atp を指定して WebSphere Commerce 5.4 にマイグレーションする際に、データベース・マイグレーション・スクリプトはマイグレーション時に RECEIPT テーブルにどのベンダーも含めないことに注意してください。ストアごとに複数のベンダーがあ

ったり、またはベンダーがない可能性があるからです。ベンダーがストアに割り当てられていない場合、ATP 在庫表記にマイグレーションする前に WebSphere Commerce Accelerator を使用してベンダーを作成する必要があります。

オーダー・アイテムのマイグレーション

マイグレーション・スクリプトは、状況が P、I、または M であるすべてのオーダーをチェックします。

- これらのオーダーの下にあるすべてのオーダー・アイテムを検索します。
- ORDERITEMS テーブルの CATENTRY_ID に応じて ITEMSPC フィールドを埋めます。
- ストアのデフォルト契約を使用するすべてのオーダー・アイテムの TRADING_ID フィールドを埋めます。
- すべてのオーダーとオーダー・アイテムで、状況が C のものを状況 S に変換するスクリプトを生成します。

配送計算コード

Commerce Suite 5.1 では、配送計算コードは、異なる配送先住所を持つオーダー・アイテムのグループごとに別個に計算されていました。つまり、配送計算コードでは、配送先住所別にオーダー・アイテムがグループ化されていました。現在では、配送先住所別のグループ化は、オプションの動作になっています。下位互換性動作を保証するため、マイグレーション・スクリプトは、すべての配送計算コード (CALCODE.CALUSAGE_ID = -2) 用の CALCODE.GROUPBY 列に perAddress フラグを設定します。

支払いのマイグレーション

各 CMDREG エントリーごとに、マイグレーション・スクリプトは DoCancelPMCmdImpl を DoCancelCmdImpl に変更します。追加情報については、146 ページの『DoCancelCmd の CMDREG エントリーの変更』を参照してください。

割引データのマイグレーション

マイグレーション・スクリプトは、Commerce Suite 5.1 の Commerce Accelerator ツールによって作成された割引データが存在しているかどうかを判別します。割引サブシステムに対して以下を行います。

- スクリプトは、Commerce Suite 5.1 Commerce Accelerator 以外のツールで割引データが作成されたと検出した場合は、その割引データを現状のまま残しておきます。これは、その割引データを WebSphere Commerce 5.4 レベルにマイグレーションすることはありません。この場合、その割引データにアクセスして表示するには、Commerce Suite 5.1 で使用したのと同じツールとメソッドを使用する必要があります。
- スクリプトは、Commerce Accelerator ツールで割引データが作成されたと検出した場合は、CALCODE および CALCODEMGP テーブル内のその割引データを、WebSphere Commerce 5.4 で必要とされる割引データにマイグレーションします。

- 以前の割引データがマイグレーションされていたら、それを削除します。マイグレーションされない割引データは、現状のまま残ります。

契約のマイグレーション

マイグレーション・スクリプトは、契約サブシステムに対して以下の更新を実行します。

- Commerce Suite 5.1 でストアのデフォルト契約を使用していなかった場合は、WebSphere Commerce 5.4 では、スクリプトは必要に応じてストアのデフォルト契約を作成します。

Commerce Suite 5.1 で、ストアのデフォルト契約を使用していた場合は、スクリプトは、ご使用のストアのデフォルト契約を、WebSphere Commerce 5.4 ストアのデフォルト契約にマイグレーションします。これは、メンバー・グループ価格設定が使用できない場合は、セラー参加者と契約レベル参加者を作成します。各 TRADEPOSCN エントリーごとに、FLAGS 列がゼロに設定されている場合には、標準価格契約条件を作成します。FLAGS 列が非ゼロの場合、スクリプトはカスタム価格契約条件を作成します。

- 各 MGPTRDPSCN エントリーごとに、取引位置コンテナ・レベルのバイヤー参加者を作成します。MBRGRP_ID がゼロの場合は、バイヤー参加者 MEMBER_ID がヌルに設定され、すべてのバイヤーに資格があることを示します。
- 各カスタム価格契約条件ごとに、スクリプトは以下を行います。
 - カスタム価格表に対して、対応するカスタム商品セットを作成します。
 - 価格表からのデータを使用して、商品セット内にデータを取り込みます。
- 複数の契約がある場合、スクリプトは、STOREDEF.CONTRACT_ID に適切な値を設定することによって、1 つだけがデフォルト契約として活動化されるようにします。
- 各契約ごとに TRADING テーブルにエントリーを作成します。
- 新しい STORECNTR テーブルにエントリーを追加します。

デフォルト契約

WebSphere Commerce 5.4 では、契約サポートを提供する条件が導入されています。マイグレーション・プロセスでは、WebSphere Commerce Suite 5.1 ビジネス・フロー（たとえば配送料用）と同様の動作および特性を持つ、ご使用のシステムに対するデフォルトの契約が作成されます。

デフォルト契約は自動的に作成されるので、通常は、マイグレーション・プロセス中にユーザーがアクションを取る必要はありません。ビジネス・プロセスのために追加契約を作成する必要がある場合は、WebSphere Commerce 5.4 にマイグレーションした後に、WebSphere Commerce Accelerator を使用してそれを行います。WebSphere Commerce 5.4 オンライン・ヘルプの『新規契約の作成』のセクションを参照してください。

データベース・マイグレーション・スクリプトは、WebSphere Commerce 5.4 デフォルト契約用の以下の項目を作成します。

- マイグレーションされたストア用のポリシー (POLICY テーブル) およびポリシーの説明 (POLICYDESC テーブル)

- マスター・カタログ用の標準価格ポリシー (ストアごと) — Mastercatalog TC。
- 「セラーごとに送料を課金」ポリシー (POLICY_ID=-7001 ブートストラップ・データ)
- 「運送会社ごとに送料を課金」ポリシー (POLICY_ID=-7002 ブートストラップ・データ)
- 返品課金ポリシー — 日ごとの少額の課金 (ストアにつき 1 つずつ作成)
- 返品承認ポリシー — 日ごとの承認 (ストアにつき 1 つずつ作成)

さらに、スクリプトは、返品課金と返品承認用の 4 つのポリシー・コマンドを作成します (ストアごと)。

マイグレーション・スクリプトは、ユーザーがオリジナルの支払ポリシー (policy_id=-2001 ブートストラップ・データ) を使用することを想定しているため、それを作成しません。

- ストアのデフォルト契約用に作成された条件 — 1 つの配送 TC (契約ごと)
新しい JavaServer Pages を作成する必要なくストアを稼働できるようにするために、返品およびリファンド条件は作成されません。
返品およびリファンドの詳細情報は、各ストアに固有で、WebSphere Commerce 5.4 での新規事項です。このフィーチャーをデプロイする必要がある場合は、ご使用のストア用の新しい契約条件を作成する必要があります。WebSphere Commerce 5.4 オンライン・ヘルプの『新規契約の作成』のセクションを参照してください。
- 契約参加者
 - セラー参加者
 - 1 人以上のバイヤー参加者 (MGPTRDPSCN に応じる)
 - 1 人の契約レベル・バイヤー参加者

キャンペーンのマイグレーション

マイグレーション・スクリプトは、キャンペーン・イニシアチブ、e- マーケティング・スポット、および顧客プロファイルを WebSphere Commerce 5.4 の形式にマイグレーションします。SEGMENT テーブルに保管されていた顧客プロファイルは、MBRGRP テーブルに移動されます。キャンペーン・イニシアチブ規則は、BZRPESTSTG テーブルから抽出され、INITIATIVE テーブルの RULE 列に保管されます。各条件は別個のイニシアチブとして保管されます。e- マーケティング・スポットは、MPE テーブルから EMSPOT テーブルに移動されます。キャンペーン・イニシアチブのスケジュールリングは、INTVSCHED テーブルに移動されます。

アクセス・コントロールのマイグレーション

アクセス・コントロールのマイグレーションには、以下のコンポーネントのマイグレーションが含まれます。

- Commerce Suite 5.1 ACCMBRGP テーブルから WebSphere Commerce 5.4 MBRROLE テーブルへのマイグレーション。

- Commerce Suite 5.1 ACCCMDGRP テーブルから WebSphere Commerce 5.4 ACPOLICY テーブルへのマイグレーション。
- Commerce Suite 5.1 ACCCUSTEXC テーブルから WebSphere Commerce 5.4 ACPOLICY テーブルへのマイグレーション。
- Commerce Suite 5.1 の Order Clerk アクセス・グループのマイグレーション (必要に応じて)。

この詳細は以下のとおりです。

1. Commerce Suite 5.1 では、ユーザーは、ACCMBRGRP テーブル内でさまざまなアクセス・グループ (メンバー・グループ・タイプが AccessGoup に設定されたメンバー・グループ) に割り当てられることによってアクセス特権を与えられます。WebSphere Commerce 5.4 でも、アクセス・コントロール・ポリシーは部分的にアクセス・グループに基づいています。ただし、ユーザーは、アクセス・グループに直接割り当てられる代わりに、MBRROLE テーブル内で役割に割り当てられます。たいていのブートストラップ・アクセス・グループは、暗黙的に役割割り当てを参照します。たとえば、Sellers アクセス・グループには、MBRROLE テーブルで Seller 役割が割り当てられているすべての人が含まれます。

以下の表では、Commerce Suite 5.1 のアクセス・グループを、WebSphere Commerce 5.4 役割および WebSphere Commerce 5.4 アクセス・グループにマップしています。

表 11.

Commerce Suite 5.1 アクセス・グループ	WebSphere Commerce 5.4 役割	WebSphere Commerce 5.4 アクセス・グループ
Site Administrator (-1)	Site Administrator (-1)	SiteAdministrators (-1)
Customer (-2)	AllUsers アクセス・グループには暗黙的にすべての人が含まれるので、役割としては不要。	AllUsers (-2)
Customer Service Representative (-3)	Customer Service Representative (-3)	CustomerServiceRepresentatives (-3)
Merchant (-4)	Seller (-4)	Sellers (-4)
Order Clerk (-5)	ブートストラップでは使用されない	ブートストラップでは使用されない
Store Administrator (-6)	Store Administrator (-6)	StoreAdministrators (-6)
Store Developer (-7)	Store Developer (-7)	StoreDevelopers (-7)
Merchandising Manager (-8)	Product Manager (-8)	ProductManagers (-8)
Marketing Manager (-9)	Marketing Manager (-9)	MarketingManagers (-9)

注: 通常、役割の名前は単数形で、アクセス・グループの名前は複数形です。

ACCMBRGRP テーブル内のエントリーは、Commerce Suite 5.1 アクセス・グループ ID から WebSphere Commerce 5.4 役割 ID への上記のマッピングを使用して、MBRROLE テーブルにマイグレーションされます。Commerce Suite 5.1 システムでアクセス・グループを作成した場合、データ・マイグレーション・スクリプトは、そのアクセス・グループと同じ名前を持つ、対応する役割を作成しません。

2. データ・マイグレーション・スクリプトは、Commerce Suite 5.1 ACCCMDGRP テーブルに追加したすべてのエントリーを、WebSphere Commerce 5.4 ACPOLICY

テーブルに適切にマイグレーションします。エントリーがブートストラップ・アクセス・グループを参照していた場合、スクリプトは既存の WebSphere Commerce 5.4 ブートストラップ・アクセス・コントロール・ポリシーを更新して、コマンドまたはビューを組み込みます。エントリーが、Commerce Suite 5.1 で作成したアクセス・グループを参照している場合、マイグレーション・スクリプトは新しいポリシーを、適切なコンポーネント (Action、 ActionDescription、 ActionGroup、 ResourceCategory、 ResourceGroup など) と共に作成します。

3. Commerce Suite 5.1 ACCCUSTEXC テーブルのマイグレーションには、以下の適切な AllUsers テンプレート・アクセス・コントロール・ポリシーのオーバーライドが含まれます。
 - 特定の組織エンティティのコントローラー・コマンドの除外用の AllUsersExecuteAllUserCmdResourceGroup_TemplatePolicy。
 - 特定の組織エンティティのビューの除外用の AllUsersExecuteAllUsersViews_TemplatePolicy。

それから、スクリプトは、どのコマンドおよびビューが、この組織エンティティが所有するストアにアクセスできないかに応じて、この組織エンティティ用の適切な AllUsers ポリシーを作成します。

4. WebSphere Commerce 5.4 では、Order Clerk 役割は使用されません。そのため、WebSphere Commerce 5.4 ブートストラップ・アクセス・コントロール・ポリシーは、この役割を参照しません。ただし、顧客がこの役割によってアクセスされるいくつかのカスタマイズされたコントローラー・コマンドまたはビューを追加している場合は、スクリプトは ACPOLICY テーブル内に適切なアクセス・コントロール・ポリシーを作成します。

付録 D. 後からの ATP インベントリーへの変換

WebSphere Commerce 5.4 にデータベース・マイグレーション・スクリプトを使用してマイグレーションする場合、インベントリー・データは、デフォルトでは、WebSphere Commerce 5.4 によってサポートされる新規の ATP インベントリー表記に変換されます。インベントリー・データを、マイグレーション・スクリプトの `noatp` オプションを指定して変換しないことにした場合、以降のセクションで説明しているとおりに、後で `migrateATP` スクリプトを実行することによって、ATP に変換することができます。

ユーザーが ATP オプションを使用してマイグレーションすることを選択するかどうかにかかわらず、マイグレーション・スクリプトは、`PRODUCT` および `ITEM` エントリーの場合、以下のテーブルをユーザーに代わってセットアップします。

- `BASEITEM`
- `ITEMSPC`
- `ITEMVERSN`
- `STOREITEM`
- `VERSIONSPC`

このセットアップは、マイグレーション時に一度だけ実行されることに注意してください。新規の `PRODUCT` および `ITEM` エントリーをデータベース・マイグレーション・スクリプト (`migratedb`) を実行した後に追加する場合は、`migrateATP` スクリプトを実行する前に、上記のテーブルが、マイグレーション・スクリプトの実行以降に追加されたすべての新規の `PRODUCT` および `ITEM` エントリーに対して適切にセットアップされていることを確認する必要があります。WebSphere Commerce 5.4 ツールを使用して新しい商品およびアイテムを追加する場合、エントリーはユーザーに対して適切にセットアップされます。

Mass Loader (`massload.xml`) を使用して `ITEM` および `PRODUCT` エントリーを取り込む場合は、それらのテーブル (`BASEITEM`、`ITEMSPC`、`ITEMVERSN`、`VERSIONSPC`、および `STOREITEM`) も、スクリプトの実行前にセットアップする必要があります。Mass Loader の使用方法については、WebSphere Commerce 5.4 オンライン・ヘルプから、ローダーの使用についての情報と、それらのテーブルの説明を参照してください。

`migrateATP` スクリプトは、以下のテーブルを作成します。

- これは、商品ごとに `DISTARRANG` を作成します。これは、`ENDDATE` に大きな値 (59 年) を、また `STARTDATE` に現行日を指定します。
- これは、アイテムごとに `RADETAIL`、`RECEIPT`、`RCTAVAIL`、および `ITEMFFMCTR` を作成します。
- これは、パッケージごとに `DISTARRANG`、`RADETAIL`、`RECEIPT`、`RCTAVAIL`、および `ITEMFFMCTR` を作成します。(これを各親が親商品で、各アイテムが子アイテムであるかのように扱います。)

さらに、以下を行います。

- ストアのデフォルト契約を使用するすべてのオーダー・アイテムの TRADING_ID フィールドを埋めます。
- すべてのオーダーとオーダー・アイテムで、状況が C のものを状況 S に変換するスクリプトを生成します。

DB2 データベースの場合

DB2 データベースを使用する場合は、以下の手順で migrateATP スクリプトを実行します。

1. コマンド・プロンプト・ウィンドウをオープンします。
2. root ユーザーとして WebSphere Application Server を実行している場合は、ご使用のデータベースの DB2 インスタンス・ユーザー ID (たとえば db2inst1) に切り替えます。

```
su - db2_instance_user_id
```

非 root ユーザーとして WebSphere Application Server を実行している場合は、 WebSphere Application Server ユーザー ID (たとえば wasuser) に切り替えます。

```
su - WAS_user
```

3. WebSphere Commerce 5.4 がインストールされている bin サブディレクトリに移動します。たとえば /opt/WebSphere/CommerceServer/bin などです。
4. migrateATP マイグレーション・スクリプトを DB2 に対して以下のように実行します。

```
./migrateATP.db2.sh db_name db_userID db_userID_password db2insthome
```

ここで

- *db_name* は、マイグレーション先の Commerce Suite 5.1 データベースです (たとえば mall)。
- *db_userID* は、マイグレーション先の Commerce Suite 5.1 データベースに接続するためのユーザー ID です (たとえば、mydbuser)。
- *db_userID_password* は、マイグレーション先の Commerce Suite 5.1 データベースに接続するためのユーザー ID のパスワードです (たとえば、mypasswd)。
- *db2insthome* は、 DB2 インスタンス・ユーザー ID のホーム・ディレクトリ (たとえば /home/db2inst1/sqllib) です。

たとえば、インベントリー・データを新規の ATP 表記に変換する場合は、以下のコマンドを使用することができます。

```
migrateATP.db2.sh mall mydbuser mypasswd /home/db2inst1/sqllib
```

5. /opt/WebSphere/CommerceServer/logs/migration ディレクトリに生成される ATPmigrate.log の要約セクションをチェックして、エラーや警告が発生していないことを確認します。エラーが起きていない場合は続行できますが、起きていた場合は、すべてのエラー条件に対処して、続行前にスクリプトを再実行する必要があります。

付録 E. データベース・スキーマの変更点

この付録は、Commerce Suite 5.1 と WebSphere Commerce 5.4 との間でのデータベース・スキーマの変更点を要約しています。このリリースで導入されている新しいテーブルについては、*IBM WebSphere Commerce 新着情報 バージョン 5.4* を参照してください。

データベース・テーブルの用法と説明の詳細については、*WebSphere Commerce 5.4 オンライン・ヘルプのデータベース・テーブルの説明*を参照してください。

注: すべての基本キー索引名、ユニーク鍵索引名、および外部キー索引制約名は、Commerce Suite 5.1 に関連して変更されており、この表にはリストされていません。WebSphere Commerce 5.4 では、基本キー索引名とユニーク鍵索引名は、システムによって生成されます。また、外部キー制約は名前変更されています。例として、CALMETHOD テーブルについて、その変更点を以下の表に要約します。

表 12. CALMETHOD データベース・テーブル

アイテム	Commerce Suite 5.1 名	WebSphere Commerce 5.4 システム生成名
基本キー索引	P_CALMETHOD	SQL010926024834880
ユニーク鍵索引	UI_CALMETHOD	I0000055
外部キー制約	F_CALMETHOD1	F_164
	F_CALMETHOD2	F_163

データベース・スキーマの変更点

以下の表は、Commerce Suite 5.1 スキーマに加えられた主な変更点のリストです。

表 13. WebSphere Commerce 5.4 で変更されたテーブル

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
A				
		ACACGPDESC		新規テーブル
		ACACTACTGP		新規テーブル
		ACACTDESC		新規テーブル
		ACACTGRP		新規テーブル
		ACACTION		新規テーブル
		ACATTR		新規テーブル
		ACATTRDESC		新規テーブル
ACCCMDGRP				使用すべきでないテーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
ACCCMDTYPE				使用すべきでないテーブル
ACCCUSTEXC				使用すべきでないテーブル
		ACCLOGMAIN		新規テーブル
		ACCLOGSUB		新規テーブル
ACCMBRGRP				使用すべきでないテーブル
		ACCOUNT		新規テーブル
		ACORGPOL		新規テーブル
		ACPOLDESC		新規テーブル
		ACPOLICY		新規テーブル
		ACRELATION		新規テーブル
		ACRELDISC		新規テーブル
		ACRELGRP		新規テーブル
		ACRESACT		新規テーブル
		ACRESATREL		新規テーブル
		ACRESCGRY		新規テーブル
		ACRESGPDES		新規テーブル
		ACRESGPRES		新規テーブル
		ACRESGRP		新規テーブル
		ACRESMEMRL		新規テーブル
		ACRESPRIM		新規テーブル
		ACRESREL		新規テーブル
		ACRLGPDESC		新規テーブル
		APRVSTATUS		新規テーブル
		ATTACHMENT		新規テーブル
		ATTACHUSG		新規テーブル
ATTRIBUTE		ATTRIBUTE	USAGE	新規列
ATTRVALUE		ATTRVALUE	OPERATOR_ID	新規列
		AUCPAYINFO		新規テーブル
AUCTION		AUCTION	AUCMODE	新規列
			AUCPAYINFO_ID	
			DURLENGTH	
			INVRSRVID	
			PRICELIMIT	
			QUANTLIMIT	

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント	
テーブル	列名	テーブル	列名		
AUCTIONLOG		AUCTIONLOG	AUCMODE	新規列	
			AUCPAYINFO_ID		
			DURLENGTH		
			INVRSRVID		
			PRICELIMIT		
			QUANTLIMIT		
AUTOBID		AUTOBID	AUCPAYINFO_ID	新規列	
AUTOBIDLOG		AUTOBIDLOG	AUCPAYINFO_ID	新規列	
B					
		BASEITEM		新規テーブル	
		BASEITMDSC		新規テーブル	
BID		BID	AUCPAYINFO_ID	新規列	
			BIDMODE		
			INITPRICE		
			LIMITPRICE		
BIDLOG		BIDLOG	AUCPAYINFO_ID	新規列	
			BIDMODE		
			INITPRICE		
			LIMITPRICE		
BIDPAYMENT		BIDPAYMENT	AUCPAYINFO_ID	新規列	
			BKORDALLOC		新規テーブル
			BKORDITEM		新規テーブル
BUSPROF		BUSPROF	REQUISITIONERID	新規列	
			BUYERPO		新規テーブル
			BUYERPOTYP		新規テーブル
			BUYSUPMAP		新規テーブル
			BUYSUPSEC		新規テーブル
			BZSRVCFG		新規テーブル
			BZSVCCFG		新規テーブル
	BZSVCSTA	新規テーブル			
C					
CACHLOG	CACHASHVALUE	CACHLOG	CACHASHVALUE	変更された列	
	データ・タイプ: VARCHAR(64)		データ・タイプ: VARCHAR(254)		

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
CALCODE	CODE	CALCODE	CODE	変更された列
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
			DESCRIPTION	新規列
			DISPLAYLEVEL	
			ENDDATE	
			FLAGS	
			PRECEDENCE	
	STARTDATE			
		CALCODEMGP		新規テーブル
CALRANGE		CALRANGE	MARKFORDELETE	新規列
CAMPAIGN	ENDDATE	CAMPAIGN		除去された列
	LASTDEPLOY			
	LASTDEPLOYEDBY			
	MATYPE_ID			
	STARTDATE			
	USERSTATUS			
			STATUS	新規列
			TYPE	
		CATALGDESC		新規ビュー
CATALO		CATALO	TPCLEVEL	新規列
		CATCLSFCOD		新規テーブル
		CATCONFINF		新規テーブル
CATEGORY				除去された テーブル
		CATEGRYREL		新規ビュー
CATENCALCD		CATENCALCD	TRADING_ID	新規列
	CONTRACT_ID			除去された列
CATENTDESC		CATENTDESC	KEYWORD	新規列
CATENTRY		CATENTRY	BASEITEM_ID	新規列
			ITEMSPC_ID	
			STATE	
CATGPCALCD		CATGPCALCD	TRADING_ID	新規列
	CONTRACT_ID			除去された列
CATGPENREL		CATGPENREL	LASTUPDATE	新規列
CATGRPATTR		CATGRPATTR	SEQUENCE	新規列
CATGRPDESC		CATGRPDESC	KEYWORD	新規列
			CATGRPPS	
CATGRPREL		CATGRPREL	LASTUPDATE	新規列

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
		CATGRPTPC		新規テーブル
		CATPRDREL		新規ビュー
CATTOGRP		CATTOGRP	LASTUPDATE	新規列
			SEQUENCE	
		CHARGETYPE		新規テーブル
		CHKARRANG		新規テーブル
		CHKCMD		新規テーブル
		CHRGTYPDSC		新規テーブル
		CLASIFCODE		新規テーブル
CLEANCONF	CONDITION	CLEANCONF		除去された列
	TABNAME		OBJECTNAME	新規列
			SEQUENCE	
			STATEMENT	
	TYPE		TYPE	変更された列
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
	PK: No		PK: Yes	
	NAMEARGg		NAMEARG	
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
	DAYSARG		DAYSARG	
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
		CLSFCODEDS		新規テーブル
CMPGNINTV				除去された テーブル
CMPGNRV				除去された テーブル
		CNTRDISPLY		新規テーブル
		CNTRNAME		新規テーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
CONTRACT		CONTRACT	COMMENTS	新規列
			MAJORVERSION	
			MARKFORDELETE	
			MINORVERSION	
			ORIGIN	
			TIMEACTIVATED	
			TIMEAPPROVED	
			TIMECREATED	
			TIMEDEPLOYED	
			TIMEUPDATED	
	USAGE			
	STORE_ID			除去された列
	NAME		NAME	変更された列
	データ・タイプ: VARCHAR(254)		データ・タイプ: VARCHAR(200)	
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
	MEMBER_ID		MEMBER_ID	
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
CPENDORDER	MEMBER_ID	CPENDORDER	MEMBER_ID	変更された列
	PK: No		PK: Yes	
CPGNLOG	INITIATIVE_ID	CPGNLOG	INITIATIVE_ID	変更された列
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
	PK: No		PK: Yes	
		CPITMAP		新規テーブル
		CPOFFER		新規テーブル
		CPPMN		新規テーブル
		CPPMNDESC		新規テーブル
		CPPMNDISC		新規テーブル
		CPPMNORD		新規テーブル
		CPPMNPROD		新規テーブル
		CPPMNVAL		新規テーブル
		CPWALLET		新規テーブル
		CREDITLINE		新規テーブル
CURFMTDESC		CURFMTDESC	NUMBRUSG_ID	新規列

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
CURFORMAT		CURFORMAT	NUMBRUSG_ID MINAPPROVEAMOUNT	新規列
D				
DISPCGPREL		DISPCGPREL	LANGUAGE_ID	新規列
DISPENTREL		DISPENTREL	LANGUAGE_ID	新規列
		DISTARRANG		新規テーブル
E				
		EMSPOT		新規テーブル
		ENUMDESC		新規テーブル
F				
FFMCENTER		FFMCENTER	DEFAULTSHIPOFFSET MARKFORDELET	新規列
		FLCOMPOSE		新規テーブル
		FLDOMNDESC		新規テーブル
		FLINSTANCE		新規テーブル
		FLOW		新規テーブル
		FLOWADMIN		新規テーブル
		FLOWDESC		新規テーブル
		FLOWDOMAIN		新規テーブル
		FLOWTYPE		新規テーブル
		FLSTATEDCT		新規テーブル
		FLSTATEGP		新規テーブル
		FLSTATEREL		新規テーブル
		FLSTDCTDSC		新規テーブル
		FLSTGPDSC		新規テーブル
		FLTRANSDSC		新規テーブル
		FLTRANSITN		新規テーブル
		FLTYPEDESC		新規テーブル
H				
		HISTOATTR		新規テーブル
		HISTOFREQ		新規テーブル
		HISTONVP		新規テーブル
I				
ICEXPLDESC	NAME	ICEXPLDESC	NAME	変更された列
	データ・タイプ: VARCHAR(64)		データ・タイプ: VARCHAR(254)	

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
ICEXPLFEAT	COLUMNNAME データ・タイプ: CHAR	ICEXPLFEAT	COLUMNNAME データ・タイプ: VARCHAR(254)	変更された列
INITIATIVE		INITIATIVE	CAMPAIGN_ID RESULTTYPE RULE SELLTYPE STATUS TYPE	新規列
	DEPLOYABLE ENDDATE LASTDEPLOY LASTDEPLOYEDBY MATYPE_ID PROFILE RULEVALUE_ID STARTDATE USERSTATUS			除去された列
INTVMPE				除去された テーブル
		INTVSCHED		新規テーブル
INTVSGMT				除去された テーブル
		INVADJCODE		新規テーブル
		INVADJDESC		新規テーブル
		INVADJUST		新規テーブル
		INVOICE		新規テーブル
		INVRESERVE		新規テーブル
		INVRSRVDSC		新規テーブル
		INVRSRVTYP		新規テーブル
		ITEMFFMCTR		新規テーブル
		ITEMSPC		新規テーブル
		ITEMTYPE		新規テーブル
		ITEMVERSN		新規テーブル
J				
JURST		JURST	MARKFORDELETE	新規列
JURSTGROUP		JURSTGROUP	MARKFORDELETE	新規列
L				
LANGUAGE		LANGUAGE	MIMECHARSET	新規列

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
LISTPRICE				除去された テーブル
		LMEVENTMAP		新規テーブル
		LMSERVER		新規テーブル
		LPPOPURAMT		新規テーブル
M				
MAFAMILY				除去された テーブル
		MANIFEST		新規テーブル
MATYPE				除去された テーブル
		MBRATTR		新規テーブル
		MBRATTRVAL		新規テーブル
MBRGRP		MBRGRP	LASTUPDATE LASTUPDATEDBY	新規列
		MBRGRPCOND		新規テーブル
MBRGRPMBR		MBRGRPMBR	EXCLUDE	新規列
MBRGRPTYPE		MBRGRPTYPE	PROPERTIES	新規列
		MBRREL		新規テーブル
		MBRROLE		新規テーブル
MEMBER		MEMBER	STATE	新規列
		MLTIME		新規テーブル
MPE				除去された テーブル
MPETYPE				除去された テーブル
N				
		NUMBRUSG		新規テーブル
		NUMBRUSGDS		新規テーブル
O				
OFFER	IDENTIFIER ヌル・オプション: NOT NULL	OFFER	IDENTIFIER ヌル・オプション: NULL	変更された列
		OICOMPLIST		新規テーブル
		OPERATOR		新規テーブル
		OPERATRDSC		新規テーブル
		ORCPMAP		新規テーブル
ORDADJUST		ORDADJUST	DISPLAYLEVEL	新規列
		ORDCALCD		新規テーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
		ORDCHNLTY		新規テーブル
ORDERITEMS		ORDERITEMS	ALLOCADDRESS_ID	新規列
			ALLOCATIONGROUP	
			ALLOCFPMC_ID	
			ALLOCQUANTITY	
			CONFIGURATIONID	
			CORRELATIONGROUP	
			ESTAVAILTIME	
			FULFILLMENTSTATUS	
			INVENTORYSTATUS	
			ITEMSPC_ID	
			LASTALLOCUPDATE	
			LINEITEMTYPE	
			NEEDEDQUANTITY	
			ORDRELEASENUM	
			OUTPUTQ_ID	
			PROMISEDVAILTIME	
			SHIPPINGOFFSET	
			TERMCOND_ID	
			TIMERELEASED	
			TIMESHIPPED	
			TRADING_ID	
	CONTRACT_ID			除去された列
		ORDERMSG		新規テーブル
ORDERS		ORDERS	ORDCHNLTY_ID	新規列
			ORGENTITY_ID	
			PROVIDERORDERNUM	
			SHIPASCOMPLETE	
ORDICALCD		ORDICALCD	CALFLAGS	新規列
			CALPARMAMT	
			CALPARMTYPE	
			ORDICALCD_ID	
	CALCODE_ID		CALCODE_ID	変更された列
	PK: Yes		PK: No	
	ORDERITEMS_ID		ORDERITEMS_ID	
	PK: Yes		PK: No	
		ORDIMEEXTN		新規テーブル
		ORDIOFFER		新規テーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
			ORDIPROF	新規テーブル
			ORDITRD	新規テーブル
			ORDMEEXTN	新規テーブル
ORDOPTIONS	NOTIFYMERCHANT	ORDOPTIONS	NOTIFYMERCHANT	変更された列
	データ・タイプ: SMALLINT		データ・タイプ: INTEGER	
	NOTIFYSHOPPER		NOTIFYSHOPPER	
	データ・タイプ: SMALLINT		データ・タイプ: INTEGER	
ORDPAYMTHD		ORDPAYMTHD	ACCOUNT_ID	新規列
			ACTUALAMOUNT	
			BIGINTFIELD1	
			BIGINTFIELD2	
			BIGINTFIELD3	
			BUYERPO_ID	
			CHARGEAMOUNT	
			CHARGEAMTCURR	
			CHARGETIME	
			CREDITLINE_ID	
			DECIMALFIELD1	
			DECIMALFIELD2	
			DECIMALFIELD3	
			PAYSUMMARY_ID	
			POLICY_ID	
			REFUNDNUMBER	
			RMA_ID	
			STATUS	
			STRINGFIELD1	
			STRINGFIELD2	
			STRINGFIELD3	
			STRINGFIELD4	
			TRADING_ID	
			XMLDATA	
			ORDPICKHST	新規テーブル
			ORDRELEASE	新規テーブル
			ORDSHIPHST	新規テーブル
ORDTAX		ORDTAX	LASTUPDATE	新規列
			ORGCODE	新規テーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
ORGENITY		ORGENITY	MEMBER_ID	使用すべきでない列
		OUTPUTQ		新規テーブル
		OUTPUTQDSC		新規テーブル
P				
		PARTICIPNT		新規テーブル
		PARTROLE		新規テーブル
		PARTROLEDS		新規テーブル
		PATTRIBUTE		新規テーブル
		PATTRPROD		新規テーブル
		PATTRVALUE		新規テーブル
PAYMTHD		PAYMTHD		使用すべきでないテーブル
PAYMTHDDSC		PAYMTHDDSC		使用すべきでないテーブル
PATMTHDSUP		PAYMTHDSUP		使用すべきでないテーブル
PAYSTATUS	PENDING	PAYSTATUS	PENDING	変更された列
	データ・タイプ: SMALLINT		データ・タイプ: INTEGER	
		PAYSUMMARY		新規テーブル
		PICKBATCH		新規テーブル
PKGATTR	LANGUAGE_ID	PKGATTR	LANGUAGE_ID	変更された テーブル
	PK: No		PK: Yes	
PKGATTRVAL	LANGUAGE_ID	PKGATTRVAL	LANGUAGE_ID	変更された テーブル
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
	PK: No		PK: Yes	
PKGITEMREL		PKGITEMREL		使用すべきでない テーブル
		PLCYACCDSC		新規テーブル
		PLCYACCLCK		新規テーブル
		PLCYACCT		新規テーブル
		PLCYLCKDSC		新規テーブル
		PLCYPASSWD		新規テーブル
		PLCYPWDDSC		新規テーブル
		PLCYTYCMIF		新規テーブル
		PLCYTYPDSC		新規テーブル
		POLICY		新規テーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
		POLICYCMD		新規テーブル
		POLICYDESC		新規テーブル
		POLICYTC		新規テーブル
		POLICYTYPE		新規テーブル
		PRATRSTATR		新規ビュー
		PRCOFFRPRC		新規ビュー
		PRCEOFFPRD		新規ビュー
		PRDATRAVAL		新規ビュー
		PRICE		新規ビュー
		PRICEDESC		新規ビュー
		PROCBUYPRF		新規テーブル
		PROCMSGVW		新規テーブル
		PROCPROTCL		新規テーブル
		PROCSYS		新規テーブル
		PRODATR		新規ビュー
		PRODDSTATR		新規ビュー
		PRODSETDSC		新規テーブル
PRODUCT				除去された要約 テーブル
		PRODUCTSET		新規テーブル
		PRSETCEREL		新規テーブル
		PURCHASELT		新規テーブル
PVCBINDING		PVCBINDING	PVCBINDING_ID	新規列
	PVCSESSION_ID		PVCSESSION_ID	変更された列
	PK: Yes		PK: No	
	USERS_ID		USERS_ID	
	PK: Yes		PK: No	
PVCBUFFER		PVCBUFFER	PARAMETERS	使用すべきでない列
			PARAMETERS2K	新規列
PVCDEVMDL	MODELNAME	PVCDEVMDL	MODELNAME	変更された列
	ヌル・オプション: NOT NULL		ヌル・オプション: NULL	
			DEVMDLNAME	新規列
			MODELNAME	使用すべきでない列

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
PVCDEVSPEC	SESSIONTYPE ヌル・オプション: NOT NULL	PVCDEVSPEC	SESSIONTYPE ヌル・オプション: NULL	変更された列 使用すべきでない列
Q				
QTYFMTDESC		QTYFMTDESC	NUMBRUSG_ID	新規列
QTYFORMAT		QTYFORMAT	NUMBRUSG_ID	新規列
		QTYUNITMAP		新規テーブル
R				
		RA		新規テーブル
		RABACKALLO		新規テーブル
		RADETAIL		新規テーブル
		RCPTAVAIL		新規テーブル
		RECEIPT		新規テーブル
		REFUNDMTHD		新規テーブル
RICHATTR				再定義されたテーブル
RICHATTRCG				新規テーブル
RICHATTRCATGP				除去された要約 テーブル
		RFQ		新規テーブル
		RFQPROD		新規テーブル
		RFQRSP		新規テーブル
		RFQRSPPROD		新規テーブル
		RFQRSPTCRL		新規テーブル
		RLDISCOUNT		新規テーブル
		RMA		新規テーブル
		RMAAUTHLOG		新規テーブル
		RMACHARGE		新規テーブル
		RMAIADJCRD		新規テーブル
		RMAIDNYRSN		新規テーブル
		RMAITEM		新規テーブル
		RMAITEMCMP		新規テーブル
		RMATAX		新規テーブル
		ROLE		新規テーブル
		RTNDNYDESC		新規テーブル
		RTNDNYRSN		新規テーブル
		RTNDSPCODE		新規テーブル
		RTNDSPDESC		新規テーブル
		RTNRCPTDSP		新規テーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
			RTNREASON	新規テーブル
			RTNRECEIPT	新規テーブル
			RTNRSNDESC	新規テーブル
S				
			SCHCMD	新規テーブル
SCHCONFIG		SCHCONFIG	INTERFACENAME	新規列
			STOREENT_ID	
	SCCAPPTYPE		SCCAPPTYPE	変更された列
	データ・タイプ: VARCHAR		データ・タイプ: CHAR	
	SCCPATHINFO		SCCPATHINFO	
	データ・タイプ: VARCHAR(128)		データ・タイプ: VARCHAR(254)	
	SCCQUERY		SCCQUERY	
	データ・タイプ: VARCHAR		データ・タイプ: LONG	
SEGMENT				除去された テーブル
			SCHACTIVE	新規テーブル
SHIPMODE		SHIPMODE	MARKFORDELETE	新規列
			SHPARJURGP	新規テーブル
SHPARRANGE		SHPARRANGE	FLAGS	新規列
	SHIPMODE_ID		SHIPMODE_ID	変更された列
	ヌル・オプション: NOT NULL		ヌル・オプション: NULL	
SHPJCRULE	FFMCENTER_ID	SHPJCRULE	FFMCENTER_ID	変更された列
	ヌル・オプション: NOT NULL		ヌル・オプション: NULL	
	JURSTGROUP_ID		JURSTGROUP_ID	
	ヌル・オプション: NOT NULL		ヌル・オプション: NOT NULL	
	SHIPMODE_ID		SHIPMODE_ID	
	ヌル・オプション: NOT NULL		ヌル・オプション: NOT NULL	
STDPRICE				新規テーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
STENCALUSG		STENCALUSG	CALMETHOD_ID_APP	新規列
			CALMETHOD_ID_FIN	
			CALMETHOD_ID_INI	
			CALMETHOD_ID_SUM	
			SEQUENCE	
			USAGEFLAGS	
		STGUINDTAB		新規テーブル
STORE		STORE	ALLOCATIONGOODFOR	新規列
			AVSACCEPTCODES	
			BOPMPADFACTOR	
			DEFAULTBOOFFSET	
			FFMCSELECTIONFLAGS	
			MAXBOOFFSET	
			PRICEREFFLAGS	
			REJECTEDORDEXPIRY	
			RMAGOODFOR	
			RTNFFMCTR_ID	
			STORETYPE	
			STORECAT	
MASTERCATALOG				
		STORECNTR		新規テーブル
STOREENT		STOREENT	SETCCURR	新規列
STOREINV		STOREINV		要約テーブルが ビューに変更さ れたもの
			STOREITEM	
			STORITMFFC	
T				
TAXCGRY		TAXCGRY	MARKFORDELETE	新規列

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
TAXJCRULE		TAXJCRULE	TAXJCRULE_ID	新規列
	CALRULE_ID		CALRULE_ID	変更された列
	PK: Yes		PK: No	
	FFMCENTER_ID		FFMCENTER_ID	
	ヌル・オプション: NOT NULL		ヌル・オプション: NULL	
	PK: Yes		PK: No	
	JURSTGROUP_ID		JURSTGROUP_ID	
	ヌル・オプション: NOT NULL		ヌル・オプション: NULL	
	PK: Yes		PK: No	
TAXTYPE		TAXTYPE	SEQUENCE	使用すべきでない列
		TCDESC		新規テーブル
		TCPITMAP		新規テーブル
		TCSUBTYPDS		新規テーブル
		TCSUBTYPE		新規テーブル
		TCTYPE		新規テーブル
		TERMCOND		新規テーブル
		TFALGOPOL		新規テーブル
		TFALGOREG		新規テーブル
		TFALGOTYPE		新規テーブル
		TFALGPOLDS		新規テーブル
		TFALGTYPDS		新規テーブル
		TFALREGDSC		新規テーブル
		TFDOMAIN		新規テーブル
		TFDOMDSC		新規テーブル
		TFEXTENG		新規テーブル
		TFSBDOMAIN		新規テーブル
		TFSBDOMDSC		新規テーブル
		TFTRADENG		新規テーブル
		TFTRENGDSC		新規テーブル
		TMPBOLIST		新規テーブル
		TMPCMPLIST		新規テーブル
		TMPPROCID		新規テーブル
		TMPFFCLIST		新規テーブル
		TMPRADTL		新規テーブル

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
			TMPRCTLIST	新規テーブル
			TORCPMAP	新規テーブル
TRADEPOSCN		TRADEPOSCN	PRODUCTSET_ID	新規列
			PRECEDENCE	
			MARKFORDELETE	
			TYPE	
	NAME		NAME	変更された列
	ヌル・オプション: NULL		ヌル・オプション: NOT NULL	
			TRADING	新規テーブル
			TRDATTACH	新規テーブル
			TRDDEPAMT	新規テーブル
			TRDDESC	新規テーブル
			TRDPURAMT	新規テーブル
			TRDREFAMT	新規テーブル
			TRDTYPE	新規テーブル
			TRDTYPEDSC	新規テーブル
U				
USERPVCDEV		USERPVCDEV	DEVICEIDENTIFIER	使用すべきでない列
			PVCSESSION_ID	新規列
USERREG		USERREG	PLCYACCT_ID	新規列
			TIMEOUT	
			PASSWORDRETRIES	
			SALT	
			PASSWORDCREATION	
			PASSWORDINVALID	
USRTRAFFIC		USRTRAFFIC	SESSIONID	新規列
V				
			VENDOR	新規テーブル
			VENDORDESC	新規テーブル
			VERSIONSPC	新規テーブル
W				
			WCCATDESC	新規ビュー
			WCCATEGORY	新規ビュー
			WCCATRTCAT	新規ビュー
			WCCATSTORS	新規ビュー
			WCMLANG	新規テーブル
			WCPRDCTPID	新規ビュー

表 13. WebSphere Commerce 5.4 で変更されたテーブル (続き)

WebSphere Commerce Suite 5.1		WebSphere Commerce 5.4		コメント
テーブル	列名	テーブル	列名	
			WCPRDPRICE	新規ビュー
			WCPRODCATP	新規ビュー
			WCPRODDDESC	新規ビュー
			WCPRODSHIP	新規ビュー
			WCPRODUCT	新規ビュー
			WCSDTNRYS	新規テーブル
Z				
ZIPCODE				除去された テーブル

付録 F. 変更されたプログラミング・インターフェース

この付録では、WebSphere Commerce 5.4 で変更または廃止された、Commerce Suite 5.1 のコマンドやクラス名などのプログラミング・インターフェースに加えられた変更点をリストしています。このリリースで導入された新しいインターフェースについては、*IBM WebSphere Commerce 新着情報 バージョン 5.4* を参照してください。

変更されたインターフェースの使用法および構文の詳細については、WebSphere Commerce 5.4 オンライン・ヘルプを参照してください。

使用すべきでないコマンド

以下のいくつかの Commerce Suite 5.1 コマンドは、WebSphere Commerce 5.4 では使用すべきではなく、WebSphere Commerce の将来のリリースではサポートされなくなります。

- EntityAdmin

このコマンドは Commerce Suite 5.1 のオンライン・ヘルプには記載されていますが、WebSphere Commerce 5.4 のオンライン・ヘルプには記載されていません。さらに、このコマンドは以下のエンティティをサポートしていました。

- entity=MemberGroupType (MBRGRPTYPE テーブルにマップ)
- entity=MemberGroup (MBRGRP テーブルにマップ)
- entity=MemberGroupMember (MBRGRPMBR テーブルにマップ)
- entity=MemberGroupUsage (MBRGRPUSG テーブルにマップ)
- entity=**AccessControlMemberGroup** (ACCMBRGRP テーブルにマップ)
- entity=**AccessControlCommandGroup** (ACCCMDGRP テーブルにマップ)
- entity=**AccessControlCustomerCommandExclusion** (ACCCUSTEXC テーブルにマップ)

注: 上記の下から 3 つの (太字の) エンティティは、Commerce Suite 5.1 アクセス・コントロール・テーブルを管理するために使用されていましたが、WebSphere Commerce 5.4 では使用されなくなりました。したがって、このコマンドは WebSphere Commerce 5.4 のそれらのテーブルを引き続き操作できますが、アクセス・コントロール・ランタイムはそれらのテーブル内のデータには影響を受けません。

アクセス・コントロールの詳細については、10 ページの『アクセス・コントロール』および 137 ページの『第 8 章 アクセス・コントロール・サブシステムの考慮事項』を参照してください。

- HTTPCommandContext

HTTPCommandContext インターフェースは、同じ機能が CommandContext コマンドによって提供されているので、WebSphere Commerce 5.4 では使用すべきではあ

りません。コードでは HTTPCommandContext の代わりに CommandContext を使用する必要があります。そうしない場合、ランタイム ClassCastException 例外を受け取ります。

現在 HTTPCommandContext を使用している場合、コードを以下の方法で変更してください。

1. HttpServletRequest オブジェクトを取得するには、以下のようになります。

```
com.ibm.commerce.webcontroller.HttpControllerRequestObject req =
(com.ibm.commerce.webcontroller.HttpControllerRequestObject)commandContext.getRequest();
HttpServletRequest httpRequest = req.getHttpRequest();
```

2. HttpServletResponse オブジェクトを取得するには、以下のようになります。

```
HttpServletResponse httpResponse = (HttpServletResponse)commandContext.getResponse();
```

3. 要求がブラウザから来るかどうかをチェックするには、以下のようになります。

```
if (commandContext.getRequest() instanceof
com.ibm.commerce.webcontroller.HttpControllerRequestObject)
```

または

```
if (commandContext.getDeviceFormatTypeId().equals(new Integer(-1)))
```

変更されたコマンド

商品アドバイザー

商品アドバイザーのパッケージ名は、WebSphere Commerce Suite 5.1 のものとは異なるものになりました。商品アドバイザーを使用する場合は、これらのパッケージ名について 127 ページの『商品アドバイザーのマイグレーション考慮事項』を参照してください。

UserRegistrationAddCmd および UserRegistrationUpdateCmd

WebSphere Commerce Suite 5.1 では、com.ibm.commerce.usermanagement.command パッケージにおいて、UserRegistrationAddCmd および UserRegistrationUpdateCmd コマンドは、UserRegistrationCmd コマンドを拡張し、このコマンドは AddressBaseCmd コマンドを拡張します。

WebSphere Commerce 5.4 では、UserRegistrationCmd コマンドは AddressBaseCmd コマンドを拡張しなくなりました。それで、AddressBaseCmd から継承したものを使用するメソッドがある場合は、以下に太字で示しているとおりにコードを再作成する必要があります。

例

```
if ( isGuest ){
UserRegistrationAddCmd userRegAdd = (UserRegistrationAddCmd)
CommandFactory.createCommand(UserRegistrationAddCmd.NAME,getStoreId());
userRegAdd.setCommandContext(getCommandContext());
userRegAdd.setRequestProperties(register_Prop);
userRegAdd.execute();
newBillingAddress=userRegAdd.getAddressId(); <===in 51
//should change to the following in 54
String userId = userRegAdd.getWorkingUserId();
try
{
AddressAccessBean abBillingAddress =
```

```

        new AddressAccessBean().findSelfAddressByMember(new Long(userId));
        newBillingAddress=abBillingAddress.getAddressId() ;
    }
    catch(Exception e)
    {
        //User does not have a self address
    }
    ECTrace.trace(ECTraceIdentifiers.COMPONENT_USER, CLASSNAME, METHODNAME,
        "Successfully created a new user..");

}
else {
    UserRegistrationUpdateCmd userRegUpdate = (UserRegistrationUpdateCmd)
        CommandFactory.createCommand(UserRegistrationUpdateCmd.NAME,getStoreId());
    userRegUpdate.setCommandContext(getCommandContext());
    userRegUpdate.setRequestProperties(register_Prop);
    userRegUpdate.execute();
    newBillingAddress = userRegUpdate.getAddressId();<==in 51
    //should change to the following in 54
    String userId = userRegAdd.getWorkingUserId();
    try
    {
        AddressAccessBean abBillingAddress =
            new AddressAccessBean().findSelfAddressByMember(new Long(userId));
        newBillingAddress=abBillingAddress.getAddressId() ;
    }
    catch(Exception e)
    {
        //Use does not have a self address
    }

    ECTrace.trace(ECTraceIdentifiers.COMPONENT_USER, CLASSNAME, METHODNAME,
        "Successfully updated a new user..");
}
}

```

AddressAdd コマンド

WebSphere Commerce 5.4 では、AddressAdd コマンドには、指定したメンバー（組織を含む）に対するアドレスを作成する追加機能があります。したがってこれは、memberId をパラメーターとして取ります。memberId が指定された場合、これはそのメンバーのアドレスを作成します。指定されない場合、現行ユーザーのアドレスを作成します。

Commerce Suite 5.1 の場合、AddressAdd コマンドは memberId をパラメーターとして持たなかったことに注意してください。それは現行ユーザーに対するアドレスだけを作成します。それで Commerce Suite 5.1 の場合、memberId パラメーターは、指定しても無視されます。

変更されたプログラミング・インターフェース

以降のいくつかのセクションの表では、以下のパッケージに含まれているプログラミング・インターフェースに加えられた変更点をリストしています。

- WCS_Order
- WCS_Catalog
- WCS_User
- WCS_Databean
- WCS_EJB

これらの表では、すべてのクラスの完全名は `com.ibm.commerce.` で始まります。たとえば、`taxation.commands.GetDisplayTaxesCmdImpl` の完全名は、`com.ibm.commerce.taxation.commands.GetDisplayTaxesCmdImpl` です。以下の表では、簡潔に表現するために、`com.ibm.commerce.` は名前から取り除かれています。

WCS_Order

以下の表は、オーダー・サブシステムに関連したクラスに加えられた変更点を要約しています。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
<code>taxation.commands.GetDisplayTaxesCmdImpl</code>	<code>BigDecimal BIG_DECIMAL_ZERO;</code>	<code>public void setDisplayOnly(boolean);</code>
	<code>public void checkParameters()</code> で、 <code>ECEException;</code> がスローされます。	<code>public void validateParameters()</code> で、 <code>ECEException;</code> がスローされます。
		<code>public void setUOM(String);</code>
<code>taxation.commands.ApplyOrderTaxesCmd</code>	<code>public static final String defaultCommandClassName;</code>	
<code>taxation.commands.ResolveTaxJurisdictionCmdImpl</code>	<code>public void checkParameters()</code> で、 <code>ECEException;</code> がスローされます。	<code>public void validateParameters()</code> で、 <code>ECEException;</code> がスローされます。
<code>fulfillment.commands.InventoryBaseCmdImpl</code>	<code>public void checkParameters()</code> で、 <code>ECEException;</code> がスローされます。	<code>public void validateParameters()</code> で、 <code>ECEException;</code> がスローされます。
<code>fulfillment.commands.ApplyOrderShippingChargesCmd</code>	<code>public static final String defaultCommandClassName;</code>	
<code>fulfillment.commands.UpdateInventoryForOrderItemsCmdImpl</code>	<code>public void checkParameters()</code> で、 <code>ECEException;</code> がスローされます。	<code>public void validateParameters()</code> で、 <code>ECEException;</code> がスローされます。
<code>price.commands.GetBaseUnitPriceCmdImpl</code>	<code>public void checkParameters()</code> で、 <code>ECEException;</code> がスローされます。	<code>public void validateParameters()</code> で、 <code>ECEException;</code> がスローされます。
		<code>static Class class\$com\$ibm\$commerce\$price\$commands\$GetBaseUnitPrice CmdImpl;</code>
		<code>static Class class\$(String);</code>
<code>price.commands.GetBaseUnitPriceCmdImpl \$QualifyingOfferInfo</code>	<code>GetBaseUnitPriceCmdImpl this\$0;</code>	
<code>price.commands.ApplyOrderAdjustmentsCmd</code>	<code>public static final String defaultCommandClassName;</code>	
<code>price.commands.SetCurrencyPreferenceCmdImpl</code>	<code>public void checkParameters()</code> で、 <code>ECEException;</code> がスローされます。	<code>public void validateParameters()</code> で、 <code>ECEException;</code> がスローされます。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
		static Class class\$com\$ibm\$commerce \$price\$commands \$SetCurrencyPreference CmdImpl;
order.utils.GetCalculationUsagesCmdImpl \$PseudoOrderItemAccessBean	GetCalculationUsagesCmdImpl this\$0;	
order.utils.GetCalculationUsagesCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローさ れます。
order.utils.OrderConstants	public static final String ORDER_BID;	public static final String ORDER_BACKORDERED; public static final String ORDER_PAYMENT_ AUTH_REVIEW;
order.utils.GetShippingChargesCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローさ れます。 public void setCurrency(String);
order.utils.ResolveOrdersCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローさ れます。
order.utils.ResolveJurisdictionCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	static Class class\$java\$lang\$Long; public void validateParameters() で、ECEException; がスローさ れます。
order.utils.ApplyCalculationUsagesCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローさ れます。
order.utils.CalculationCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローさ れます。
order.utils.ResolveOrderItemsCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローさ れます。 static Class class\$java\$lang\$Long;
order.commands.CheckOrderCopyCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	
order.commands.OrderDisplayCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローさ れます。
order.commands.OrderProfileUpdateCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
order.commands.OrderListCmdImpl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。
order.commands.CheckOrderTemplateCmdImpl	public void checkParameters() で、ECEException; がスローされます。	
order.commands.ExtOrderCopyCmdImpl	public void checkParameters();	
order.commands.SetOrderPaymentInfoCmdImpl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。
order.commands.OrderProcessCmdImpl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。
order.commands.OrderProcessCmdImpl \$OrderExpiredException	OrderProcessCmdImpl this\$0; \$OrderExpiredException	
order.commands.OrderProcessingHelper \$DoubleNVP	OrderProcessingHelper this\$0; \$DoubleNVP	
order.commands.AdminOrderCancelCmd Impl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。
order.commands.ScheduledOrderCancelCmd Impl	public void checkParameters() で、ECEException; がスローされます。	public AccessVector getResources() で、ECEException; がスローされます。
order.commands.OrderCopyCmdImpl	public void checkParameters() で、ECEException; がスローされます。	public AccessVector getResources() で、ECEException; がスローされます。 public void validateParameters() で、ECEException; がスローされます。
order.commands.GetOrderPayInfoCmdImpl	public void checkParameters() で、ECEException; がスローされます。	
order.commands.SetCurrentPendingOrdersCmd Impl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。
order.commands.GetCurrentPendingOrdersCmd Impl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。
order.commands.OrderPrepareCmdImpl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。 public void setATPParms (ATPPParameters);
order.commands.ScheduledOrderProcessCmd Impl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
		public AccessVector getResources() で、 ECEException; がスローされま す。
order.commands.OrderProfile	public void updatePaymentInfo(Hashtable) で、 FinderException、CreateException、 RemoteException、NamingException、 EApplicationException、 ECEException; がスローされます。	
		public void updatePaymentInfo (Hashtable, CommandContext) で、 FinderException、 CreateException、 RemoteException、 NamingException、 EApplicationException、 ECEException; がスローされま す。
order.commands.SetOrderTemplateCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、 ECEException; がスローさ れます。
		static Class class\$java\$lang\$string;
order.commands.OrderScheduleCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、 ECEException; がスローさ れます。
order.commands.TestNotificationCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、 ECEException; がスローさ れます。
order.commands.OrderUnlockCmdImpl	public void checkParameters() で、 ECEException; がスローされます。	public void validateParameters() で、 ECEException; がスローさ れます。
		public AccessVector getResources() で、 ECEException; がスローされま す。

WCS_Catalog

以下の表は、カタログ・サブシステムに関連したクラスに加えられた変更点を要約しています。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
tools.catalog.commands.OfferNotebookUpdateImpl		除去されました
tools.catalog.commands.ItemNotebookUpdateImpl		除去されました
tools.catalog.commands.OfferNotebookUpdate		除去されました
tools.catalog.commands.ItemNotebookUpdate		除去されました
tools.catalog.beans.ItemListBean		除去されました
tools.catalog.beans.OfferListBean		除去されました
tools.catalog.beans.LanguageDataBean		除去されました
tools.catalog.beans.CalculationCodeSBBean		除去されました
tools.catalog.beans.OfferDataBean		除去されました
tools.catalog.beans.ListPriceDataBean		除去されました
tools.catalog.beans.OfferPriceDataBean		除去されました
tools.catalog.beans.CatalogListDataBean		除去されました
tools.catalog.beans.TaxCodeClassificationDataBean		除去されました
tools.catalog.beans.QtyUnitListData		除去されました
tools.catalog.beans.ProductSearchData		除去されました
tools.catalog.beans.AttributeValueDataBean	public void setCommandContext (CommandContext);	
	public TypedProperty getRequestProperties();	public Long getAttributeValueId();
	public CommandContext getCommandContext();	public String getDisplaySequence();
	public void setRequestProperties (TypedProperty);	public void setAttributeValueId(Long);
tools.catalog.helpers._CatalogSearch_BaseStub	public Vector findCatalogEntry (ProductSearchData) で、NamingException、SQLException、RemoteException、ObjectNotFoundException; がスローされます。	
		public Vector findCatalogEntry (ProductSearchParameters) で、NamingException、SQLException、RemoteException、ObjectNotFoundException; がスローされます。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
tools.catalog.helpers._CatalogSearch_Stub	public Vector findCatalogEntry (ProductSearchData) で、NamingException、SQLException、RemoteExceptionObjectNotFound Exception; がスローされます。	public Vector findCatalogEntry (ProductSearchParameters) で、NamingExceptionSQLException、RemoteExceptionObjectNotFound Exception; がスローされます。
tools.catalog.helpers.CatalogSearchBean	public static final String findCatalogEntryTypeID_WHERE;	public static final String findItem_WHERE; public static final String findProduct_WHERE; public void setCatalogEntryType(String);

WCS_User

以下の表は、ユーザー・サブシステムに関連したクラスに加えられた変更点を要約しています。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
usermanagement.commands.AddressDelete CmdImpl	public void checkParameters() で、ECEException; がスローされます。 public boolean isReadyToCallExecute();	
usermanagement.commands.UserRegistration CmdImpl	public void checkParameters() で、ECEException; がスローされます。	public boolean isLogonIdUnique(String) で、ECEException; がスローされます。 public void setCity(String);
usermanagement.commands.OrgEntity RegistrationCmdImpl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。
usermanagement.commands.AddressCheck CmdImpl	public void checkParameters() で、ECEException; がスローされます。 public boolean isReadyToCallExecute();	public void validateParameters() で、ECEException; がスローされます。
usermanagement.commands.AddressBase CmdImpl	public void checkParameters() で、ECEException; がスローされます。	public String getAddress3(); public AccessVector getResources() で、ECEException; がスローされます。
security.commands.LogonCmdImpl	public void checkParameters() で、ECEException; がスローされます。	public void validateParameters() で、ECEException; がスローされます。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
		public void updateLockoutInformation(boolean) で、EException; がスローされます。
security.commands.LDAPAuthenticationCmd Impl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
		public String getAuthenticateUserId();
security.commands.DBAuthenticationCmd Impl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
		public void markPasswordAsTemporary();
security.commands.LogoffCmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
security.commands.ResetPasswordRegister Cmd	public abstract String getErrorCode();	
security.commands.UpdateCredentials CmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
		public void markPasswordTemporary();
security.commands.ResetPasswordCmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
security.commands.VerifyCredentialsCmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
		public String getAuthenticateUserId();
security.commands.ResetPasswordGuest CmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
security.commands.MigrateUserEntries CmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
security.commands.SendPasswordNotification CmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
security.commands.ResetPassword AdministratorCmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。
security.commands.ResetPasswordRegister CmdImpl	public void checkParameters() で、 EException; がスローされます。	public void validateParameters() で、EException; がスローされ ます。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
	public String getErrorCode();	

WCS_Databean

以下の表は、データ Bean サブシステムに関連したクラスに加えられた変更点を要約しています。

クラス	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.4
user.beans.UserRegistrationInputDataBean	public abstract void setUserId(Long);	

Enterprise JavaBeans

以下の表は、Enterprise JavaBeans クラスに加えられた変更点を要約しています。

表 14.

コンポーネント	クラス名	WebSphere Commerce Suite 5.1 でのみのメソッド
カタログ	AttributeBean	public void ejbCreate(Long, Integer) で、 CreateExceptionRemoteException; がスローされます。 このメソッドは WebSphere Commerce 5.4 からは除去されています。
	AttributeFloatValueHome	public abstract AttributeFloatValue create(Long, Integer) で、 CreateExceptionRemoteException; がスローされます。 このメソッドは WebSphere Commerce 5.4 からは除去されています。
	AttributeHome	public abstract Attribute create(Long, Integer) で、 CreateExceptionRemoteException; がスローされます。 このメソッドは WebSphere Commerce 5.4 からは除去されています。
	AttributeIntegerValueHome	public abstract AttributeIntegerValue create(Long, Integer) で、 CreateExceptionRemoteException; がスローされます。 このメソッドは WebSphere Commerce 5.4 からは除去されています。
	AttributeStringValueHome	public abstract AttributeStringValue create(Long, Integer) で、 CreateExceptionRemoteException; がスローされます。 このメソッドは WebSphere Commerce 5.4 からは除去されています。
	AttributeValueBean	public void ejbCreate(Long, Integer) で、 CreateExceptionRemoteException; がスローされます。 このメソッドは WebSphere Commerce 5.4 からは除去されています。

表 14. (続き)

コンポーネント	クラス名	WebSphere Commerce Suite 5.1 でのみのメソッド
	AttributeValueHome	<p>public abstract AttributeValue create(Long, Integer) で、 CreateExceptionRemoteException; がスローされます。</p> <p>このメソッドは WebSphere Commerce 5.4 からは除去されています。</p>
	CatalogEntryHome	<p>public abstract CatalogEntry create(Long) で、 CreateExceptionRemoteException; がスローされます。</p> <p>このメソッドは WebSphere Commerce 5.4 からは除去されています。</p>
	CatalogGroupPageRelationBean	<p>public void ejbCreate(Long, Long, Integer) で、 CreateExceptionRemoteException; がスローされます。</p> <p>このメソッドは WebSphere Commerce 5.4 からは除去されています。</p>
	ItemHome	<p>public abstract Item create(Long) で、 CreateExceptionRemoteException; がスローされます。</p> <p>このメソッドは WebSphere Commerce 5.4 からは除去されています。</p>
	ProductHome	<p>public abstract Product create(Long) で、 CreateExceptionRemoteException; がスローされます。</p> <p>このメソッドは WebSphere Commerce 5.4 からは除去されています。</p>
	ProductUserGroupRelationBean	<p>このクラス・ファイルは WebSphere Commerce 5.4 からは除去されています。</p> <p>このメソッドは WebSphere Commerce 5.4 からは除去されています。</p>
オーダー	OfferBean	<p>public Enumeration getContracts() で、 FinderExceptionRemoteExceptionNaming Exception; がスローされます。</p> <p>public Enumeration getContracts(Integer) で、 FinderExceptionRemoteExceptionNaming Exception; がスローされます。</p> <p>これは、WebSphere Commerce 5.4 での設計変更です。現在では、契約は、オファーから検索することはできません。</p>
	OrderOptionBean	<p>すべての Short Java オブジェクト・タイプは、データベースと整合するように整数オブジェクト・タイプにマイグレーションされています。</p>

表 14. (続き)

コンポーネント	クラス名	WebSphere Commerce Suite 5.1 でのみのメソッド
オーダーの状況	OrderFulfillmentItemStatusBean	<p>public void ejbCreate(TypedProperty) で、 CreateExceptionNamingExceptionRemote ExceptionFinderException; がスローされます。</p> <p>TypedProperty は、EJB オブジェクトのランタイム従属関係を除去するため、現在では Hashtable に変更されています。</p>
	OrderFulfillmentStatusBean	<p>public void ejbCreate(TypedProperty) で、 CreateExceptionNamingExceptionRemote ExceptionFinderException; がスローされます。</p> <p>TypedProperty は、EJB オブジェクトのランタイム従属関係を除去するため、現在では Hashtable に変更されています。</p>
	OrderFulfillmentStatusHome	<p>public abstract OrderFulfillmentStatus create(TypedProperty) で、 NamingExceptionCreate ExceptionFinder ExceptionRemoteException; がスローされます。</p> <p>TypedProperty は、EJB オブジェクトのランタイム従属関係を除去するため、現在では Hashtable に変更されています。</p>
ユーザー	MemberBean	<p>public String getDisplayName();</p> <p>表示名列が、MEMBER テーブルから除去されています。</p>
	MemberBeanFinderHelper	<p>public static final String findDistinctStoreOwnersWhereClause;</p> <p>このメソッドは WebSphere Commerce Suite 5.1 では機能しなかったため、WebSphere Commerce 5.4 からは除去されました。</p>

付録 G. サンプル JSP の更新

マイグレーションの後に WebSphere Commerce 5.4 でご使用のストアのさまざまな面が正しく機能するために、JSP の一部を変更する必要があります。たとえば、ストア・サービスを使用してショッピング・フローを完了できるようにするには、Commerce Suite 5.1 に同梱されていた `shipaddress.jsp` を更新する必要があります。さらに、ログオン・エラー・メッセージを改善するため、Commerce Suite 5.1 に同梱されていた JSP である、`register.jsp` および `account.jsp` を更新する必要があります。

更新済みの JSP を参照用に以下にリストします。

register.jsp

```
<%
//*****
//*
//* Licensed Materials - Property of IBM
//*
//* 5697-D24
//*
//* (c) Copyright IBM Corp. 2000, 2002
//*
//* US Government Users Restricted Rights - Use, duplication or
//* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
//*
//*
//*
%>

<%@ page language="java" %>
<% // All JSPs requires the first 4 packages for getResource.jsp which is used for multi language support %>
<%@ page import="java.io.*" %>
<%@ page import="java.util.*" %>
<%@ page import="com.ibm.commerce.server.*" %>
<%@ page import="com.ibm.commerce.command.*" %>

<%@ page import="javax.servlet.*" %>
<%@ page import="com.ibm.commerce.catalog.beans.*" %>
<%@ page import="com.ibm.commerce.catalog.objects.*" %>
<%@ page import="com.ibm.commerce.beans.*" %>

<%@ page import="com.ibm.commerce.user.beans.*" %>
<%@ page import="com.ibm.commerce.user.objects.*" %>
<%@ page import="com.ibm.commerce.datatype.*" %>
<%@ page import="com.ibm.commerce.usermanagement.commands.ECUserConstants" %>
<%@ page import="com.ibm.commerce.common.beans.*" %>
<%@ page import="com.ibm.commerce.security.commands.ECSecurityConstants" %>
<%@ include file="getResource.jsp"%>
<%
// JSPHelper provides you with a easy way to retrieve
// URL parameters when they are encrypted
JSPHelper jhelper = new JSPHelper(request);

String storeId = jhelper.getParameter("storeId");
String catalogId = jhelper.getParameter("catalogId");
String languageId = jhelper.getParameter("langId");
%>

<jsp:useBean id="bnError" class="com.ibm.commerce.beans.ErrorDataBean" scope="page">
<% com.ibm.commerce.beans.DataBeanManager.activate(bnError, request); %>
</jsp:useBean>

<%
String strErrorMessage = null;
String strErrorCode = "";

String strLogonID = null;
String strPassword = null;
String strPasswordVerify = null;
String strLastName = null;
String strFirstName = null;
```

```

TypedProperty hshErrorProperties = bnError.getExceptionData();

if (hshErrorProperties != null)
{
    //We have a registration error.

    strErrorCode = hshErrorProperties.getString(EConstants.EC_ERROR_CODE, "");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_BAD_LOGONID))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE40");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_LOGONID_EXISTS))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE41");

    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_BAD_LOGONPASSWORD))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE42");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_BAD_LOGONPASSWORDVERIFY))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE43");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_PASSWORDS_NOT_SAME))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE44");

    if (strErrorCode.equals(ECUserConstants.EC_ADDR_ERR_BAD_LASTNAME))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE45");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_MISSING_LOGONPASSWORDVERIFY))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE46");

    if (strErrorCode.equals(ECSecurityConstants.ERR_MINIMUMLENGTH_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE21");
    if (strErrorCode.equals(ECSecurityConstants.ERR_MINIMUMDIGITS_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE22");
    if (strErrorCode.equals(ECSecurityConstants.ERR_MINIMUMLETTERS_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE23");
    if (strErrorCode.equals(ECSecurityConstants.ERR_USERIDMATCH_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE24");
    if (strErrorCode.equals(ECSecurityConstants.ERR_REUSEOLD_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE25");
    if (strErrorCode.equals(ECSecurityConstants.ERR_MAXCONSECUTIVECHAR_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE26");
    if (strErrorCode.equals(ECSecurityConstants.ERR_MAXINTANCECHAR_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE27");

    //Redisplay what was entered when the
    //invalid entry was submitted.
    strLogonID = jhelper.htmlTextEncoder(jhelper.getParameter(ECUserConstants.EC_UREG_LOGONID));
    strPassword = jhelper.getParameter(ECUserConstants.EC_UREG_LOGONPASSWORD);
    strPasswordVerify = jhelper.getParameter(ECUserConstants.EC_UREG_LOGONPASSWORDVERIFY);
    strLastName = jhelper.htmlTextEncoder(jhelper.getParameter(ECUserConstants.EC_ADDR_LASTNAME));
    strFirstName = jhelper.htmlTextEncoder(jhelper.getParameter(ECUserConstants.EC_ADDR_FIRSTNAME));
}
else
{
    //Form is loading under regular condition.
    //Initialize all fields to empty.

    strLogonID = "";
    strPassword = "";
    strPasswordVerify = "";
    strLastName = "";
    strFirstName = "";
}
%>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml11-transitional.dtd">
<html>
<head>
<title><%=infashiontext.getString("REGISTER_TITLE")%></title>

<link rel=stylesheet href="<%=storeDir%>/fashionfair.css" type="text/css">
</head>

<body marginheight="0" marginwidth="0">

<!-- Set the user id and e-mail to the same value -->
<SCRIPT language="javascript">
function prepareSubmit(form)
{
    form.<%=ECUserConstants.EC_ADDR_EMAIL1%>.value =
        form.<%= ECUserConstants.EC_UREG_LOGONID%>.value.toLowerCase()
    form.<%=ECUserConstants.EC_UREG_LOGONID%>.value =
        form.<%= ECUserConstants.EC_UREG_LOGONID%>.value.toLowerCase()

    form.submit()
}
</SCRIPT>

<%
String incfile;

incfile = "/" + storeDir + "/header.jsp";
%>
<jsp:include page="<%=incfile%>" flush="true"/>

<%
incfile = "/" + storeDir + "/sidebar.jsp";
%>

<jsp:include page="<%=incfile%>" flush="true"/>

<!--END SEARCH-->

<td bgcolor="#FFFFFF" width="600" rowspan="6" valign="top">

<!--MAIN CONTENT STARTS HERE-->

<table cellpadding="2" cellspacing="0" width="580" border="0" align="left">
<tr>

<td width="10" rowspan="10">&nbsp;  </td>

<td align="left" valign="top" colspan="3" class="categoryspace">
<font class="category"><%=infashiontext.getString("REGISTRATION")%></font>
<hr width="580" noshade align="left">
<font class="required">*</font><font class="text"><%=infashiontext.getString("REQUIRED_FIELDS3")%></font></td>
</tr><tr>

```

```

<td align="left" valign="top" width="400" class="topspace">
<%
if (strErrorMessage != null)
{
    //We have error message.
    %>
<p><font color="red"><%=strErrorMessage%></font><br><br></p>
<%
}
%>
<FORM name="Register" method=POST action="<%=UserRegistrationAdd"%>">
<INPUT TYPE="hidden" NAME="langId" Value="<%=languageId%>">
<INPUT TYPE="hidden" NAME="new" Value="Y">
<INPUT TYPE="hidden" NAME="storeId" Value="<%=storeId%>">
<INPUT TYPE="hidden" NAME="catalogId" Value="<%=catalogId%>">
<INPUT TYPE="hidden" NAME="URL" Value="LogonForm">
<INPUT TYPE="hidden" NAME="page" Value="account">
<INPUT TYPE="hidden" NAME="registerType" Value="G">
<INPUT TYPE="hidden" NAME="profileType" Value="C">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_EMAIL1%>" Value="">
<!--
Lots of mandetary fields are not displayed in this form.
We set them to "-".
-->
<INPUT TYPE="hidden" NAME="personTitle" Value="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_ADDRESS1%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_ADDRESS2%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_CITY%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_STATE%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_ZIPCODE%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_COUNTRY%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_PHONE1%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%= ECUserConstants.EC_UREG_CHALLENGEQUESTION %>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%= ECUserConstants.EC_UREG_CHALLENGEANSWER %>" VALUE="-">

<table cellpadding="3" cellspacing="0" border="0" align="left">
<tr>
<td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("EMAIL2")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%= ECUserConstants.EC_UREG_LOGONID%>" value="<%=strLogonID%>"></td>
</tr><tr><td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("PASSWORD3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%=ECUserConstants.EC_UREG_LOGONPASSWORD%>"
type="password" value="<%=strPassword%>"></td>
</tr><tr><td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("VERIFY_PASSWORD3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%=ECUserConstants.EC_UREG_LOGONPASSWORDVERIFY%>"
type="password" value="<%=strPasswordVerify%>"></td>
</tr>
<% if (locale.toString().equals("ja_JP")||locale.toString().equals("ko_KR")||
locale.toString().equals("zh_CN")||locale.toString().equals("zh_TW")) { %>

```

```

<tr>
<td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("LAST_NAME3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%=ECUserConstants.EC_ADDR_LASTNAME%>" value="<%=strLastName%>" type="text"></td>
</tr><tr><td align="right" valign="middle">
<font class="strongtext"><%=infashiontext.getString("FIRST_NAME3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%= ECUserConstants.EC_ADDR_FIRSTNAME %>" value="<%=strFirstName%>" type="text"></td>
</tr>
<% } else { %>
<font class="product">
<tr>
<td align="right" valign="middle">
<font class="strongtext"><%=infashiontext.getString("FIRST_NAME3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%= ECUserConstants.EC_ADDR_FIRSTNAME %>" value="<%=strFirstName%>" type="text"></td>
</tr><tr><td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("LAST_NAME3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%=ECUserConstants.EC_ADDR_LASTNAME%>" value="<%=strLastName%>" type="text"></td>
</tr>
<% } %>
<tr>
<td align="left" valign="top">
<font class="text">&nbsp;</font></td>
<td align="left" valign="top" class="categoryspace">
<table cellpadding="4" cellspacing="0" border="0">
<tr>
<td align="left" valign="middle" bgcolor="#FFCC99">
<A href="javascript:prepareSubmit(document.Register)"><font class="strongtext"><%=infashiontext.getString("SUBMIT")%></font></a>
</td>
</tr></table>
</td></tr></table>
</form>
</td>

<td width="180" valign="top" class="topspace">
<table cellpadding="3" cellspacing="0" border="1" width="180" bgcolor="#FFFFCC">
<tr>
<td align="left" valign="top">
<font class="strongtext"><%=infashiontext.getString("YOUR_PRIVACY")%></font><br>
<font class="text"><%=infashiontext.getString("PRIVACY_STATEMENT")%><br>
<a href="PrivacyView?langId=<%=languageId%>&storeId=<%=storeId%>&catalogId=<%=catalogId%>">
<%=infashiontext.getString("LEARN_MORE")%></a></font>
</td></tr></table>
</td>

</tr></table>
</td>

</tr></table>
<%
incfile = "/" + storeDir + "/footer.jsp";
%>

<jsp:include page="<%=incfile%>" flush="true"/>

</body>

</html>

```

account.jsp

```
<%
/**
*****
**
** Licensed Materials - Property of IBM
**
** 5697-D24
**
** (c) Copyright IBM Corp. 2000, 2002
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
**
**
*****
**
%>

<%@ page language="java" %>
<% // All JSPs requires the first 4 packages for getResource.jsp which is used for multi language support %>
<%@ page import="java.io.*" %>
<%@ page import="java.util.*" %>
<%@ page import="com.ibm.commerce.server.*" %>
<%@ page import="com.ibm.commerce.command.*" %>

<%@ page import="javax.servlet.*" %>
<%@ page import="com.ibm.commerce.catalog.beans.*" %>
<%@ page import="com.ibm.commerce.catalog.objects.*" %>
<%@ page import="com.ibm.commerce.beans.*" %>
<%@ page import="com.ibm.commerce.user.beans.*" %>
<%@ page import="com.ibm.commerce.user.objects.*" %>
<%@ page import="com.ibm.commerce.datatype.*" %>
<%@ page import="com.ibm.commerce.usermanagement.commands.ECUserConstants" %>
<%@ page import="com.ibm.commerce.security.commands.ECSecurityConstants" %>
<%@ page import="com.ibm.commerce.common.beans.*" %>
<%@ include file="getResource.jsp"%>

<%
CommandContext commandContext = (CommandContext)
    request.getAttribute(ECConstants.EC_COMMANDCONTEXT);

String catalogId = request.getParameter("catalogId");
String storeId = request.getParameter("storeId");
String languageId = request.getParameter("langId");

//Parameters may be encrypted.
if (catalogId == null)
    catalogId = ((String[]) request.getAttribute("catalogId"))[0];
if (storeId == null)
    storeId = ((String[]) request.getAttribute("storeId"))[0];
if (languageId == null)
    languageId = ((String[]) request.getAttribute("langId"))[0];
%>
```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">

<head><title><%=infashiontext.getString("ACCOUNT_TITLE")%></title>
<link rel=stylesheet href="<%=storeDir%>/fashionfair.css" type="text/css">
</head>

<body marginheight="0" marginwidth="0">

<%
String incfile;

incfile = "/" + storeDir + "/header.jsp";
%>
<jsp:include page="<%=incfile%>" flush="true"/>

<%
incfile = "/" + storeDir + "/sidebar.jsp";
%>

<jsp:include page="<%=incfile%>" flush="true"/>

<!--END SEARCH-->

<td bgcolor="#FFFFFF" width="600" rowspan="6" valign="top">

<%
//Deal with possible errors when logging in
String strPageTitle    = "Logon";
String strErrorMessage = null;
String strErrorCode     = null;

String[] strArrayAuth = (String [])request.getAttribute(ECConstants.EC_ERROR_CODE);

if (strArrayAuth != null){
    if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_DISABLED_ACCOUNT) == true){
        strErrorMessage = infashiontext.getString("ACCOUNT_LOCKED");
    }else if( strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_MISSING_LOGONID) == true){
        strErrorMessage = infashiontext.getString("LOGIN_ID_MISSING");
    }else if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_INVALID_LOGONID) == true){
        strErrorMessage = infashiontext.getString("LOGON_ID_INVALID");;
    }else if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_MISSING_PASSWORD) == true){
        strErrorMessage = infashiontext.getString("PASSWD_MISSING");
    }else if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_INVALID_PASSWORD) == true){
        strErrorMessage = infashiontext.getString("PASSWORD_INCORRECT");
    }else if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_LOGON_NOT_ALLOWED) == true){
        strErrorMessage = infashiontext.getString("WAIT_TO_LOGIN");
    }
}
}

```

```

/*
 * Register link behaves differently depending on if the user is logged
 * in. If the user is logged in (hence a registered user) clicking on
 * register will log the user off then display the registration form.
 * This will allow multiple registration using the same browser.
 * For guest shoppers clicking on Register link will simply display the
 * registration form.
 */
UserRegistrationDataBean regBean = new UserRegistrationDataBean();
com.ibm.commerce.beans.DataBeanManager.activate(regBean, request);
String regURL = null;

if (! regBean.findUser())
// findUser() return false if the customer is not registered
{
    //This is a guest user. Simply display the registration form.
    regURL = "UserRegistrationForm";
}
else
{
    //This is a registered/logged in user. Log him off first.
    //Through LogoffView registration page will be displayed.
    regURL = "Logoff";
}
%>
<!--MAIN CONTENT STARTS HERE-->

<table cellpadding="2" cellspacing="0" width="580" border="0">
<tr>

<td width="10" rowspan="10">&nbsp;&nbsp;&nbsp;</td>

<td align="left" valign="top" colspan="3" class="categoryspace">
<font class="category"><%=infashiontext.getString("MY_ACCOUNT3")%></font>
<hr width="580" color="#336666" noshade align="left">
</td>

</tr><tr>

<td align="left" valign="top" width="280" bgcolor="#CC6600">
<font class="subheader"><%=infashiontext.getString("PERSONAL_INFO")%></font></td>

<td width="20" rowspan="5">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td>

<td align="left" valign="top" width="280" bgcolor="#CC6600">
<font class="subheader"><%=infashiontext.getString("ADDRESS_BOOK")%></font></td>

</tr><tr>

<td align="left" valign="top" width="280" class="topspace">
<font class="text"><%=infashiontext.getString("UPDATE_NAME")%></font><p>

<table cellpadding="0" cellspacing="0" border="0" align="left">
<tr>
<td align="left" valign="top">

```

```

<table cellpadding="4" cellspacing="0" border="0">
<tr>
<td align="left" valign="middle" bgcolor="#FFCC99">
<A href="UserRegistrationForm?storeId=<%=storeId%>&langId=<%=languageId%>&catalogId=<%=catalogId%>">
<font class="strongtext"><%=infashiontext.getString("CHANGE_INFO")%></font></a></td>
</tr></table>

</td>
</tr></table>

</td>

<td align="left" valign="top" width="280" class="topspace">

<%
if (strErrorMessage != null)
{
    //We have an error message.
}%>
<p><font color="red" ><%=strErrorMessage%></font><br>
<%
}
%>
<table cellpadding="0" cellspacing="0" border="0">
<tr>
<td align="left" valign="top">
<font class="text"><%=infashiontext.getString("UPDATE_ADDRESS1")%></font><br><br></td>
</tr><tr>
<td align="left">
<table cellpadding="4" cellspacing="0" border="0">
<tr>
<td align="left" valign="middle" bgcolor="#FFCC99">
<A href="AddressBookForm?storeId=<%=storeId%>&langId=<%=languageId%>&catalogId=<%=catalogId%>">
<font class="strongtext"><%=infashiontext.getString("EDIT_ADD")%></font></a></td>
</tr></table>
</td></tr></table>

</tr></table>
</td>
</tr></table>

</td>
</tr></table>
<%
incfile = "/" + storeDir + "/footer.jsp";
%>

<jsp:include page="<%=incfile%>" flush="true"/>

</body>

</html>

```

infashiontext_en_US.properties

LOGON_ID_INVALID = The e-mail address entered is invalid. Type another e-mail address in the E-mail address field and try again.
PASSWORD_INCORRECT = The password entered is incorrect. Type your password in the Password field and try again.
LOGIN_ID_MISSING = Type an e-mail address in the E-mail address field.
PASSWD_MISSING = Type a password in the Password field.
ACCOUNT_LOCKED = Due to 3 unsuccessful password attempts, you will be unable to logon. Please contact a store representative to unlock your account.
WAIT_TO_LOGIN = Please wait a few seconds before attempting to log in again.

PASS_ERROR_MESSAGE21 = You entered a password with less than 6 characters. Passwords must be at least 6 characters in length, and include one digit and one letter. Please re-enter your password.
PASS_ERROR_MESSAGE22 = Your password does not contain a digit. Passwords must be at least 6 characters in length, and include one digit and one letter. Please re-enter your password.
PASS_ERROR_MESSAGE23 = Your password does not contain a letter. Passwords must be at least 6 characters in length, and include one digit and one letter. Please re-enter your password.
PASS_ERROR_MESSAGE24 = Your password is the same as your user-id. Please assure that your user-id and password are different.
PASS_ERROR_MESSAGE25 = Your new password is the same as the previous one. Please enter a new password, or choose 'My Account' on the menu bar to return to your account page.
PASS_ERROR_MESSAGE26 = A character in your password occurs more consecutively than the allowed limit of 3. Please re-enter your password.
PASS_ERROR_MESSAGE27 = A character in your password occurs more than the allowed limit of 4. Please re-enter your password.

infashion store translation text

ENCODESTATEMENT = text/html; charset=ISO_8859-1

header.jsp

SHOPPING_CART = SHOPPING CART
MY_ACCOUNT = MY ACCOUNT
CONTACT_US = CONTACT US
HELP = HELP
SEARCH = SEARCH
HOME = Home

#footer.jsp

SHOPPING_CART2 = Shopping cart
MY_ACCOUNT2 = My account
CONTACT_US2 = Contact us
HELP2 = Help
PRIVACY_POLICY = Privacy policy

#sidebar.jsp

CHOOSE_COUNTRY = CHOOSE A LANGUAGE
CANADA = Canada
UNITED_STATES = United States
CHINA = China
FRANCE = France
GERMANY = Germany
ITALY = Italy
JAPAN = Japan
SPAIN = Spain
TURKEY = Turkey
SERVICES = SERVICES
REGISTER = Register
NOW_BUY = now and get 10% off your first purchase!
NEED_HELP = Need help?
JUST_ASK = Just ask!
GO = GO

.
. .
.

付録 H. トラブルシューティング

このセクションでは、マイグレーション時に生じる可能性がある潜在的な問題と、そのような問題を解決するためのアクションをリストします。

- **問題:** ストア・サービス・ユーティリティの「ストア・アーカイブ」ページに、Commerce Suite 5.1 で公開し、WebSphere Commerce 5.4 にマイグレーションしたストアの状況が not published (公開されていない) と示されます。

解決法: WebSphere Commerce 5.4 にマイグレーションする場合、データベース・マイグレーション・スクリプトは、SCHSTATUS テーブルで SCSSTATE が 'C' (complete (完了)) に設定されているすべてのエントリーを消去します。データベース・マイグレーション・スクリプトは、SCHSTATUS テーブルにある保留または未完了状態の他のすべてのエントリーを SCHACTIVE テーブルに移動させます。以前に完了と設定されていたエントリーが SCHACTIVE テーブルに移動されないと、マイグレーション後にストア・サービスではストアの状態を not published (公開されていない) と表示します。

ストアは正しく機能し続けることに注意してください。

- **問題:** DB2 用のデータベース・マイグレーション・スクリプトを実行すると、以下のようなエラー・メッセージが表示されます (通常は、生成された message.txt ファイル中に生成されます)。

```
001.10.01 14:20:19.027 java.lang.Class executeUpdate  MassLoader
An SQL Exception was received [IBM][CLI Driver][DB2/NT]
SQL0964C The transaction log for the database is full.
SQLSTATE=57011 on record [attribute: [ATTRIBUTE_ID=10176][LANGUAGE_ID=-3]
[ATTRTYPE_ID=STRING][NAME=Farbe][SEQUENCE=1.0][DESCRIPTION=Farbe]
[CATENTRY_ID=11040][DESCRIPTION2=Farbe]].
[sqlcode: -964, sqlstate: 57011]
```

解決法: 基本的に上記のメッセージは、DB2 データベースのログ・サイズが小さすぎて、操作を実行できないことを示しています。以下のデータベース構成パラメーターがログ・サイズを制御しています。

- LOGFILSIZ
- LOGPRIMARY
- LOGSECOND

一例として、LOGFILSIZ が 1000 (4KB) で、LOGPRIMARY が 40 の場合、合計ログ・サイズは $1000 * 4K * 40 = 160 \text{ MB}$ になります。コミットされていないトランザクションの合計数が、これ以上のスペースを必要とするものであれば、LOGSECOND が使用されます。それでも十分なスペースでない場合、上記のエラー・メッセージを受け取ります。アプリケーションを開発する場合には、アプリケーションでさらに頻繁にコミットを実行する必要があります。

使用しているデータベースの現在のパラメーターを表示するには、DB2 コマンド・ウィンドウから次のステートメントを入力します。

```
DB2 GET DB CFG FOR db_name
```

ここで、*db_name* はデータベースの名前です。たとえば、MALL。

ログ・サイズを増やすには、DB2 コマンド・ウィンドウから次のステートメントを入力します。

```
DB2 UPDATE DB CFG FOR db_name USING LOGFILSIZ nnn
```

ここで、*db_name* はデータベースの名前です。たとえば、MALL。 *nnn* は LOGFILSIZ 値を表す数です。

これらの構成パラメーターの設定の詳細については、DB2 管理の資料を参照してください。さらに、次のような要約テーブルがシステムに存在していて、トランザクション・ログのオーバーフローの原因と思われる場合、この要約テーブルを除去することができます。次のような SQL ステートメントを使用して、テーブルを除去します。

```
DB2 DROP TABLE table_name
```

table_name は、次のいずれかの要約テーブルです。

- PRODUCT
- STOREINV
- CATEGORY
- RICHATTR
- RICHATTRCATGP

注: トランザクション・ログがオーバーフローするのは、たいていは PRODUCT テーブルが原因なので、他のテーブルを削除する前に、まず PRODUCT テーブルを削除してからストアの公開を試行してください。

DB2 コントロール・センターを使用して、2 次トランザクション・ログも作成できます。詳細については、DB2 管理の手引き を参照してください。

- **問題:** マイグレーションした WebFashion ストアに、新規ユーザーを登録しようとすると、失敗して次のようなエラーになります。

```
...
TimeStamp:    2001-10-13 18:17:46.456
Thread ID:    <Worker#2>
Class:        com.ibm.commerce.infashion.commands.RegisterNAddToMemberGroupImpl
Method:       performExecute
Severity:     1
Message Text: CMN0411E The following Finder Exception occurred during processing:
               "javax.ejb.ObjectNotFoundException".
Exception:    javax.ejb.ObjectNotFoundException
               at javax.ejb.FinderException.<init>(FinderException.java:36)
               at javax.ejb.ObjectNotFoundException.<init>(ObjectNotFoundException.java:38)
               ...
```

解決法: register.jsp を、コマンド RegisterNAddToMemberGroup をコマンド UserRegisterAdd で置き換えて更新し、108 ページの『サンプルの WebFashion ストアのユーザー登録』の説明に従って顧客プロフィールを変更します。

- **問題:** 出荷された Commerce Suite 5.1 テーブルへの外部キー・リンクを含むテーブルをカスタマイズした場合、データ・マイグレーション時に、参照保全制約 (外部キー、基本キー、索引など) を除去しようとする、データ・マイグレーション・スクリプトは失敗する場合があります。一般的なエラー・メッセージは、次のようなものです。

ERROR at line 1:
ORA-02273: this unique/primary key is referenced by some foreign keys

解決法: マイグレーション前に、Commerce Suite 5.1 テーブルに関する制約があれば除去し、WebSphere Commerce 5.4 へのマイグレーション後に、その制約をリストアする必要があります。マイグレーション・プロセスを開始している場合、次のようにして、Commerce Suite 5.1 データのバックアップ・コピーをロードすることが必要になる場合があります。

何らかの Commerce Suite 5.1 参照保全制約を変更した場合、すなわち、すでに存在する索引または外部キーの関係に別の列を追加した場合、その列はデータ・マイグレーション・プロセスの一部として削除されます。

- **問題:** WebSphere Commerce 5.4 へのマイグレーション後に、WebSphere Commerce 5.4 管理コンソールにログオンして、ユーザーを変更または作成する場合 (つまりリストからユーザーを選択して「変更」をクリックする)、コンソールはブランク・ページを表示します。ecmsg.log ファイルに以下のエラーが示されます。

```
=====
TimeStamp: 2002-02-04 19:04:40.908
Thread ID: <Servlet.Engine.Transports:10>
Class: UIPropertiesBean
Method: setRequestProperties
Severity: 1
Message Text: CMN7023E The system could not pass the XMLFile parameter into
the UI Property Element. The element could not be loaded.
```

```
=====
TimeStamp: 2002-02-04 19:04:41.036
Thread ID: <Servlet.Engine.Transports:10>
Class: HttpForwardViewCommandImpl
Method: performExecute
Severity: 1
Message Text: CMN1244E An error occurred when forwarding document
tools/common/NotebookNavigation.jsp.javax.servlet.ServletException
Exception: javax.servlet.ServletException at
org.apache.jasper.runtime.PageContextImpl.
handlePageException(PageContextImpl.java:604)
at tools.common._NotebookNavigation_jsp_0._jspService
(_NotebookNavigation_jsp_0.java:314)
at org.apache.jasper.runtime.HttpJspBase.
service(HttpJspBase.java:142)
.
.
.
```

解決法: この例外は、ツールが以前のバージョンの UserRegistration_locale.properties ファイルを検索しているときに発生します。これは、顧客プロファイルを表示する CSA ツールでの問題と、新規ユーザー登録時のストアでの問題の原因にもなります。WebSphere Commerce 5.4 の場合、これらのファイルは /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear/properties ディレクトリーだけに置くべきです。この問題を修正するには、システムで上記のディレクトリーにはない WebSphere Commerce Suite 5.1 UserRegistration_locale.properties ファイルのすべてのインスタンスを検出して、それが入っているフォルダーを名前変更するか、またはそれらのファイルを削除する必要があります。フォルダーを名前変

更するかまたはファイルを削除した後に、WebSphere Application Server を再始動します。これで WebSphere Commerce 5.4 管理者のユーザーを変更できるはずで
す。

- **問題:** WebSphere Commerce 5.4へのマイグレーション後に、予期しないアクセス・コントロール違反を検出する場合があります。これはメッセージ・ログ・ファイル (デフォルトでは `ecmsg_xxxx.log` に設定) に、以下のようなエラー・メッセージを書き込みます。

```
TimeStamp:    2002-02-15 09:03:24.14
Thread ID:    <Servlet.Engine.Transports:10>
Class:        AccManager
Method:        isAllowed
Severity:      1
Message Text: CMN1501E User 859 does not have the authority to perform action
"com.ibm.commerce.usermanagement.commands.OrgEntityAddCmd" on resource
"com.ibm.commerce.user.objects._Organization_Stub" for command "OrgEntityAdd".
```

解決法: この例外は、リソース・レベルのアクセス・コントロール・チェックが失敗したことを意味します。アクションは WebSphere Commerce コマンドなので、これはコマンド・レベルのアクセス・コントロール障害とは対照を成す、リソース・レベルのアクセス・コントロール障害です。これがコマンド・レベルのアクセス・コントロール障害であったなら、アクションは Execute であったはずで
す。この場合、アクションは OrgEntityAdd コマンドです。

WebSphere Commerce Suite 5.1 では、リソース・レベルのアクセス・コントロールは、コマンド・ロジック内でプログラマチックに施行されていました。

WebSphere Commerce 5.4 では、リソース・レベルのアクセス・コントロール・ポリシーは外部的に指定されます。これはコマンド・レベルのアクセス・コントロール・ポリシーの指定方法と似ています。マイグレーション時に、コマンド・レベルのアクセス・コントロール・ポリシーは Commerce Suite 5.1 から WebSphere Commerce 5.4 にマイグレーションされます。Commerce Suite 5.1 のデフォルトのアクセス・コントロール・ポリシーのカスタマイズによって必要とされるどのリソース・レベルのアクセス・コントロール・ポリシー (ACCCMDGRP テーブルに保管される) も、以下に例示するとおり手動で追加する必要があります。

WebSphere Commerce 5.4 のデフォルトのアクセス・コントロール・ポリシーに基づいて、メンバーシップ管理者およびサイト管理者だけが OrgEntityAdd コマンドへのアクセス権を持ちます。この例で、ユーザー 859 (上記のログに示されており、特別な役割を持たない) がこのコマンドにアクセスできるようにしたい場合は、既存のポリシーを変更するか、または新規のポリシーを追加する必要があります。このポリシーは、ユーザー 859 を含む、あまり限定的ではないアクセス・グループ (つまり、アクセス・コントロールを目的としたユーザーのグループ) を参照する必要があります。例として AllUsers アクセス・グループがありますが、これにはすべてのユーザーが組み込まれています。新規のアクセス・コントロール・ポリシーは、そのアクション・グループに、ログから判別したとおり、`com.ibm.commerce.usermanagement.commands.OrgEntityAddCmd` アクションを組み込むことが必要な場合もあります。リソース・グループは、ログから判別したとおり、`com.ibm.commerce.user.objects.Organization` リソースを組み込むことが必要な場合もあります。これは Organization EJB のリモート・インターフェースであることに注意してください。さらに、これはログに示されるリソースの名前とはやや異なることにも注意してください。この違いの理由は、コマ

ンドはそのリソースを保護するように指定するときに Organization アクセス Bean を戻しますが、このアクセス Bean は保護可能インターフェースを拡張するリモート・インターフェースなので、WebSphere Commerce ランタイムはこれをそのリモート・インターフェースに狭める必要があるからです。EJB リソースの保護の詳細については、*WebSphere Commerce プログラマーズ・ガイド* バージョン 5.4 を参照してください。

このサンプル・ポリシーに必要なコンポーネント (AccessGroup、ActionGroup、ResourceGroup) のほとんどは、既にデフォルトのアクセス・コントロール・ポリシーで定義済みです。これによって新規のアクセス・コントロール・ポリシーを追加するプロセスは簡略になります。ここに示すのは、デフォルトのアクセス・コントロール・ポリシー・ファイル defaultAccessControlPolicies.xml で指定されている、既存のリソース・レベル・ポリシーです。

```
<Policy Name="MembershipAdministratorsForOrgExecuteOrgEntityRegistrationCommandsOnOrganizationResource"
  OwnerID="RootOrganization"
  UserGroup="MembershipAdministratorsForOrg"
  ActionGroupName="OrgEntityRegistration"
  ResourceGroupName="OrganizationDataResourceGroup"
  PolicyType="template"/>
</Policy>
```

UserGroup は MembershipAdministratorsForOrg を指定していることに注意してください。この例では、目標はすべてのユーザーを組み込んだ新規のポリシーを追加することです。

新規の、あまり限定的ではないリソース・レベルのアクセス・コントロール・ポリシーをファイルに作成します。たとえば orgentityaddpolicy.xml などとします。このファイルは /opt/WebSphere/CommerceServer/xml/policies/xml ディレクトリに置く必要があります。これは、定義したら /opt/WebSphere/CommerceServer/bin ディレクトリにある以下のスクリプトを実行して、ロードすることができます。以下のようにスクリプトを実行します。(このスクリプトの実行の詳細については、*WebSphere Commerce アクセス・コントロール・ガイド* バージョン 5.4 を参照してください。)

```
apload database userid password orgentityaddpolicy.xml
```

以下に示すのは orgentityaddpolicy.xml の内容です。

```
<?xml version="1.0"encoding="ISO-8859-1"standalone="no"?>

<!DOCTYPE Policies SYSTEM "../dtd/accesscontrolpolicies.dtd">
<Policies>

<Policy Name="AllUsersExecuteOrgEntityRegistrationOnOrganizationResourceGroup"
  OwnerID="RootOrganization"
  UserGroup="AllUsers"
  ActionGroupName="OrgEntityRegistration"
  ResourceGroupName="OrganizationDataResourceGroup">

</Policy>

<Policies>
```

- **問題:** WebSphere Commerce 5.4 にマイグレーションした後に、JavaServer Pages を立ち上げたり、マイグレーション後の JSP の選択項目をクリックすると、以下のエラーが表示されます。

```
TimeStamp: 2002-02-27 11:24:17.04
Thread ID: <Servlet.Engine.Transports:10>
Class: HttpForwardViewCommandImpl
```

```

Method:      performExecute
Severity:    1
Message Text: CMN1244E An error occurred when forwarding document
              llbean/productvertical.jsp.org.apache.jasper.JasperException:
              /usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_demo.ear/
              wcstores.war/xxxx/productvertical.jsp(2676,0)
              JSPG0059E: Unable to compile class for JSP
              null/usr/WebSphere/AppServer/temp/prizm/WebSphere_Commerce_Server_-_demo/
              WebSphere_Commerce_Enterprise_Application_-_demo/wcstores.war/
              xxxx/_productvertical_jsp_3.java:57:
              Class com.ibm.util.Sorter not found in import.
              import com.ibm.util.Sorter;

```

解決法: カスタマイズしたコードおよび JSP が、IBM WebSphere Application Server 3.5.x からの `com.ibm.util.Sorter` クラスを使用しています。`com.ibm.util` パッケージ全体は現在の WebSphere Application Server バージョンからは除去されています。WebSphere Application Server 4.0.2 にパッケージされている IBM SDK for Java からの同等のクラスを使用して、コードおよび JSP を再作成する必要があります。

- **問題:** 新規ユーザー ID の登録後に、そのユーザー ID でログオンできません。

解決法: ユーザーのカスタム・ログオン・コマンドが、単に不適切なログオン・コマンドをラッピングするのではなく、個々のタスク・コマンドを直接呼び出すので、ログオンがエラーになる場合があります。たとえば、ユーザーのカスタム・ログオン・コマンドが、WebSphere Commerce 5.4 での登録だけに呼び出すべき `com.ibm.commerce.security.commands.AuthenticationPolicyCmdImpl` を呼び出すことが原因の場合があります。

この問題を回避するには、`your_instance.xml` を変更します。

```

<Security
...
passwordpolicy="true"/>

```

これを次のように変更します。

```

<Security
...
passwordpolicy="false"/>

```

これによって、`AuthenticationPolicyCmdImpl` クラス内のログオンが回避されません。

- **問題:** `/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear` ファイルを WebSphere Application Server にデプロイして WebSphere Commerce サーバーを始動すると、JNDI の問題を検出します。

解決法: ご使用の JNDI 名をチェックします。通常は、`ear` ファイルのインポート後に、JNDI 名が変更された可能性があります。WebSphere Application Server 管理コンソールから、JNDI 名を手動で変更する必要があります。ソース・ファイルの JNDI 名に応じて、すべての EJB グループの JNDI 名を変更する必要があります。たとえば、`demoOrderItemExt` を `democom/ibm/wcsx/some_company/order/objects/OrderItemExt` に変更します。

- **問題:** `/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear` ファイルを WebSphere Application Server に、XMLConfig を使用してインポートしてから、サーバー WebSphere Commerce Server - `instance_name` を始動した後に、WebSphere Application Server は、以下のようなサーブレット初期設定エラーを管理コンソールからレポートします。

```
Did not realize init() exception thrown by servlet WCS Stores Request Servlet:
javax.servlet.ServletException:
*** Error: Initialization passing property parameters to EJB failure.
```

解決法: これは、.ear ファイルのインポート時に XMLConfig がエラーなしとレポートしても、環境の問題が原因で、/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instance_name.ear ファイルが XMLConfig を使用して正しくインストールされていないことを示している場合があります。Enterprise Application (WebSphere Commerce Enterprise Application - *instance_name*) を、管理コンソールから除去し、.ear ファイルを再デプロイすることができます。通常はこれを実行することによって、WebSphere Commerce Server - *instance_name* サーバーは正しく始動できます。

- **問題:** インスタンス・マイグレーション・スクリプトを wasuser として実行すると、以下のエラーを受け取ります。

```
java.io.FileNotFoundException:
/opt/WebSphere/CommerceServer/instances/instance_name/logs/demo_export.xml
(A file or directory in the path name does not exist.)
```

解決法: これは、/opt/WebSphere/CommerceServer/instances/*your_instance*/xml ディレクトリーに *instance.xml* ファイルの別のインスタンス (たとえば *instance.xml.bak*) があることが原因の可能性がります。この問題を修正するには、インスタンス・マイグレーション・スクリプトを実行する前に、このディレクトリーからファイルの追加のインスタンスを除去するだけで構いません。

- **問題:** 新しく公開した WebSphere Commerce 5.4 ストアと、WebSphere Commerce Suite 5.1 からマイグレーションしたそのストアのバージョンとの共存に問題があります。たとえば、WebSphere Commerce Suite 5.1 で WebSphere Commerce Suite 5.1 InFashion ストアを公開している場合、そのストアを WebSphere Commerce 5.4 にマイグレーションして再発行するとします。WebSphere Commerce 5.4 にマイグレーションした後、WebSphere Commerce 5.4 InFashion ストアも発行します。マイグレーションした WebSphere Commerce Suite 5.1 InFashion ストアか、または新しく発行した WebSphere Commerce 5.4 InFashion ストアを立ち上げると、CatalogDisplay ページにはさらに多くの製品が表示されます。新しく表示された製品をクリックすると、以下のような例外が出されます。

```
=====
TimeStamp:    2002-04-12 18:06:42.912
Thread ID:    <Servlet.Engine.Transports:9>
Class:       com.ibm.commerce.catalog.commands.ProductDisplayCmdImpl
Method:      performExecute
Severity:    1
Message Text: CMN1222E Either the Catalog Entry "10,849" does not exist in the catalog,
              or it cannot be processed in store "10,001".
Exception:   javax.ejb.ObjectNotFoundException
              at com.ibm.commerce.catalog.objects.EJSJDBCPersisterCMPCatalogEntryBean.
              findByKeyAndStore(EJSJDBCPersisterCMPCatalogEntryBean.java:692)
```

```
.
.
.
```

解決法: 以下のいずれかの方法で、この問題を解決できます。

- WebSphere Commerce 5.4 InFashion .sar ファイルにパッケージされている catalog.xml ファイルを、公開前に手動で変更します。CATALOG テーブルの IDENTIFIER 列を InFashions から InFashion54 に変更します。ID リゾルバーは新しく公開された WebSphere Commerce 5.4 InFashion ストア用に、マイグレーションした InFashion ストアのカatalog ID と競合しない新規のカatalog ID を生成します。
- WebSphere Commerce 5.4 InFashion .sar ファイルを公開する際に、カatalog ID がマイグレーションした WebSphere Commerce Suite 5.1 InFashion ストアと競合しないように新しい組織を選択する必要があります。これは、新規のカatalog ID は ID リゾルバーによって、マイグレーションした InFashion ストアのカatalog ID と競合しないように生成されるからです。

トレース情報の使用可能化

トレース情報を使用可能にすると、
/opt/WebSphere/CommerceServer/instances/*instance_name*/logs ディレクトリーに、トレース情報ファイル *ecmsg_xxxx.log* が生成されます。このファイルを生
成して WebSphere Commerce 5.4 サーバー・コンポーネントのさまざまなトレースの
ログを記録できるようにする場合、構成マネージャーでログ・システム・ノードを
構成して、トレースを使用可能にすることをお勧めします。別の方法として、
instance_name.xml ファイルのトレース・タグに、以下を追加することができます。
ただし、このファイルを手動で編集するときには十分な注意が必要です。

```
<LogSystem name="Log System">
  <trace fileSize="40"
    display="false"
    traceFile="ecmsg.log" >
    <component name="SERVER" />
    <component name="CATALOG" />
    <component name="ORDER" />
    <component name="USER" />
    <component name="COMMAND" />
    <component name="PVC" />
    <component name="UBF" />
    <component name="INVENTORY" />
    <component name="RFQ" />
    <component name="REPORTING" />
    <component name="TOOLSFRAMEWORK" />
    <component name="DEVTOOLS" />
  </trace>

  <messageLog fileSize="40"
    messageFile="/opt/WebSphere/CommerceServer/instances/instance_name/logs/ecmsg.log"
    notification="false"
    display="false">
    <logSeverity type="ERROR" />
  </messageLog>
  <activityLog display="false">
    <userTraffic cacheSize="20" />
    <accessLogging logAllRequests="false"
      cacheSize="32" />
  </activityLog>
</LogSystem>
```

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものであり、米国以外の国においては本書で述べる製品、サービス、またはプログラムを提供しない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品、プログラムまたはサービスの操作性の評価および検証は、お客様の責任で行っていただきます。

本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。IBM 製品、プログラムまたはサービスに代えて、IBM の知的所有権を侵害することのない機能的に同等のプログラムまたは製品を使用することができます。ただし、IBM によって明示的に指定されたものを除き、他社の製品と組み合わせた場合の動作の評価と検証はお客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む。) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権の許諾については、下記の宛先に書面にてご照会ください。

〒106-0032 東京都港区六本木 3 丁目 2-31

IBM World Trade Asia Corporation

Intellectual Property Law & Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

本書は定期的に見直され、必要な変更 (たとえば、技術的に不適切な表現や誤植など) は、本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Canada Ltd.
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
Canada

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのもと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願います。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

この製品で使用されているクレジット・カードのイメージ、商標、商号は、そのクレジット・カードを利用して支払うことを、それら商標等の所有者によって許可された人のみが、使用することができます。

商標

以下は、IBM Corporation の商標です。

DB2	DB2 Universal Database	IBM
Net.Commerce	SecureWay	Tivoli
WebSphere		

Domino、および Lotus は、Lotus Development Corporation の商標です。

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

SET および SET ロゴは、SET Secure Electronic Transaction LLC の商標です。

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

UNIX は、The Open Group がライセンスしている米国およびその他の国における登録商標です。

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



Printed in Japan