

IBM WebSphere Commerce Business
Edition
IBM WebSphere Commerce Professional
Edition



Linux マイグレーション・ガイド

(WebSphere Commerce Suite 5.1 用)

バージョン 5.5

IBM WebSphere Commerce Business
Edition
IBM WebSphere Commerce Professional
Edition



Linux マイグレーション・ガイド

(WebSphere Commerce Suite 5.1 用)

バージョン 5.5

ご注意!

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

本書は、IBM WebSphere Commerce Business Edition Version 5.5 for Linux、IBM WebSphere Commerce Professional Edition Version 5.5 for Linux、および新しい版で明記されていない限り、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)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： IBM WebSphere Commerce Business Edition
IBM WebSphere Commerce Professional Edition
Migration Guide for Linux
from WebSphere Commerce Suite 5.1
Version 5.5

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 2003.10

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

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

© Copyright International Business Machines Corporation 2001, 2003. All rights reserved.

© Copyright IBM Japan 2003

目次

本書について	vii
変更内容の要約	vii
WebSphere Commerce 5.5 の新機能	vii
マイグレーションを実行すべき担当者	xii
サポートされるマイグレーション・パス	xiv
前のバージョンからのマイグレーション	xiv
オペレーティング・システム別のマイグレーション・パス	xv
サポートされないマイグレーション・パス	xv
その他の考慮事項	xv
本書の表記規則	xv
パス変数	xvi

第 1 部 必要なマイグレーション・ステップ

第 1 章 Commerce Suite 5.1 からの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	5
MSGSTORE テーブルに新規メッセージが保管されていないことの確認	5
既存オークションのクローズ	6
マイグレーションのためのステージング・サーバーの準備	6
CONTRACT テーブル内の固有索引の検査	7
マイグレーション前の考慮事項	7
ATP 在庫へのマイグレーション	8
オーダーおよびオーダー・アイテム	10
マスター・カタログ	11
デフォルト契約	13
アクセス制御	13
メンバー・サブシステム	15
識別名	20
WebSphere Application Server 5.0 への移動	20

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	21
ディレクトリーおよびファイルのバックアップ	21
データベースのバックアップ	23

第 3 章 ソフトウェアのアップグレード

Commerce Suite 5.1 および WebSphere Commerce 5.5 IBM ソフトウェアのマッピング	25
ソフトウェアのアップグレードのアプローチ	26
単独のマシンへの WebSphere Commerce 5.5 のインストール (リモート・マイグレーション)	26
遷移に必要な追加のマシン要件	27
WebSphere Commerce テスト・インスタンスの作成	27

前の WebSphere Application Server 構成のマイグレーション	28
次のステップ	29

第 4 章 Commerce インスタンス構成のマイグレーション

LANG 環境変数の検査	31
Commerce Suite 5.1 product.xml ファイルのコピー	32
WebSphere Commerce 5.5 product.xml ファイルの検査 (実稼働マシン上でのマイグレーションのみ)	32
フィックスバック・レベルを調整するための予備手段	34
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	34
WCIM を使用したインスタンス構成のマイグレーション	35
WCIM 実行の前提要件	36
Commerce Suite 5.1 インスタンスのマイグレーション	37
WCIM スクリプトの実行後に	47
WebSphere Application Server での古いクラスパスの除去	47
WebSphere Application Server での古い JVM プロパティの除去	48
wcimWasConfig.jacl スクリプトの実行	48
JDBC プロバイダーの検査	50
マイグレーションした .ear ファイルのデプロイ	51
カスタム・コードの遷移およびデプロイ	53
デプロイされたインスタンス XML ファイルの更新	53
httpd.conf でのカスタマイズ	53
Web サーバーの再構成	54
IBM HTTP Server での SSL の使用可能化	54
WebSphere Application Server EJB セキュリティーの使用可能化	54
静的コンテンツを持つファイルの文書ルート・ディレクトリーへのコピー	54

第 5 章 データベースのマイグレーションの前に

DB2 インスタンスの検査	57
列の順序の考慮	58
リモート・データベースの考慮事項	58
リモート DB2 データベースのカタログ	59
MSGTYPES テーブル内の固有索引の検査	60
データベース準備スクリプトの実行	61
DB2 データベース	61
カスタム制約を除去にする	62
DB2 データベース	63

第 6 章 Commerce Suite 5.1 データベースのマイグレーション 65

データベースのマイグレーション	67
DB2 データベース	67
カスタム制約のリストア	69
DB2 データベース	69
識別名の更新	70
DB2 データベース	71
大文字小文字を区別する検索の除去	71
マスター・カタログの割り当て	72
DB2 データベース	73
オーダーおよびオーダー・アイテムの状況の変更	73
DB2 データベース	73
データベースの整合性チェッカーの実行	74
DB2 データベース	74
データベースの再マイグレーション	75

第 7 章 インスタンスおよびデータベースのマイグレーション後 77

WebSphere Application Server 始動時の考慮事項	77
セキュリティー構成のマイグレーション	77
Java 仮想マシンのヒープ・サイズに関する考慮事項	78
暗号化設定の確認	78
DB2 データベース	78
ストア資産のマイグレーション	81
Commerce Suite 5.1 JSP ファイルの更新	81
Commerce Suite 5.1 JSP ファイルの更新	84
ツール XML ファイルのマイグレーション	85
割引、配送、または課税用のカスタム・コードの更新	86
キャッシュ・ポリシーのマイグレーション	89
WebSphere Application Server 動的キャッシュの使用可能化	89
動的キャッシュ・サービスおよびサブレット・キャッシングの使用可能化	90
Web サーバー・プラグインに関する考慮事項	90
3 層インストールの場合	90
Web サーバーおよび WebSphere Application Server の再始動	91
ホスト名の変更 (リモート・マイグレーションのみ)	91
インスタンスおよびデータベースが正常にマイグレーションされたことの検証	92

第 8 章 Commerce Payments へのマイグレーション 93

Payments のマイグレーションの概要	93
マイグレーションするコンポーネント	93
遷移の方法論	94
サード・パーティー・カセットに関する考慮事項 (カスタム・コード)	94
マイグレーション前の考慮事項	95
Payments インスタンスおよびデータベースのマイグレーション	95

WCIM を使用した Payments インスタンスのマイグレーション	95
仮想ホスト・ポートの更新	100
Payments データベースのマイグレーション	100
JDBC プロバイダーの検査	101
Payments マイグレーションの使用シナリオ	102
以前の Payment Manager のマイグレーション	102
Payments の 1 インスタンスを指す複数の WebSphere Commerce インスタンス	103
Payments サブシステムのマイグレーション考慮事項	104
データ・マイグレーションの処理	105
Payments サブシステムのマイグレーションに関する追加の考慮事項	106
PAYMTHD テーブルから支払ポリシーへのマイグレーション	107
ビジネス・ポリシーおよびビジネス・ポリシー・コマンド	107
支払い用の WebSphere Commerce 5.5 ビジネス・ポリシー・コマンドへのマイグレーション	110
DoCancelCmd の CMDREG エントリーの変更	110
サンプル JavaServer ページ・ファイル - PayStatusPM.jsp	111

第 2 部 追加のマイグレーション・ステップ 113

第 9 章 メンバー・サブシステムのマイグレーション 115

マイグレーション手順の概説	115
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	118
WebSphere Commerce 5.5 での Commerce Suite 5.1 ディレクトリー・サーバーの継続使用	118

第 10 章 アクセス制御サブシステムの考慮事項 121

ACCCUSTEXC テーブルのマイグレーション	122
------------------------------------	-----

第 11 章 他の WebSphere Commerce コンポーネントのマイグレーション 123

ユーザー役割の構成	123
ステー징・サーバーのマイグレーション	124
データベース・クリーンアップ・ユーティリティーの再構成	125
ルール・サーバー構成のマイグレーション	125
オークション	127
ビジネス・アカウントおよび契約	128
配送計算コード	128
edit_registration ページにおけるログオン ID の形式 (LDAP が使用される場合)	128
商品アドバイザーのマイグレーション	129
ATP 在庫の丸め	130
デフォルトの通貨の動作	130
アクセス制御ポリシーのサブスクリプション	131

第 3 部 付録 133

付録 A. 詳細情報の入手場所 135

WebSphere Commerce についての情報 135
 WebSphere Commerce オンライン・ヘルプ 135
 WebSphere Commerce テクニカル・ライブラリ
 ー 135
WebSphere Commerce Payments についての情報 135
IBM HTTP Server についての情報 137
WebSphere Application Server についての情報 137
DB2 Universal Database についての情報 137
その他の IBM 資料 137

**付録 B. データ・マイグレーション・ス
クリプトの拡張 139**

**付録 C. WCIM ツールおよびデータ・マ
イグレーション・スクリプトの概要 . . . 141**

WCIM を使用したインスタンス・マイグレーション 141
データ・マイグレーション・スクリプト 142
 メンバーのマイグレーション 142
 カタログのマイグレーション 144
 ATP 在庫のマイグレーション 144
 オーダー・アイテムのマイグレーション 145
 配送計算コード 145
 割引データのマイグレーション 145
 契約のマイグレーション 146

 キャンペーンのマイグレーション 147
 アクセス制御のマイグレーション 148
データ・マイグレーションの補足情報 149
 データベース準備スクリプトの戻りコード 149
 列の長さの確認 154
 ATP マイグレーションに関する考慮事項 155
 データベース・マイグレーション・ログ・ファイ
 ルのチェック 155
 データベース整合性チェッカーの出力 160

**付録 D. 後で実行する ATP 在庫への変
換 163**

DB2 データベースの場合 164

付録 E. サンプル JSP の更新 167

register.jsp 168
account.jsp 173
infashiontext_en_US.properties 177

付録 F. トラブルシューティング 179

ロギングとトレースの使用可能化 181
 ロギング 182
 トレース 184

特記事項 187

商標 189

本書について

本書は、Linuxで IBM® WebSphere® Commerce Suite 5.1 から IBM WebSphere Commerce 5.5 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。前のバージョンの WebSphere Commerce Studio から WebSphere Commerce Studio 5.5 へのマイグレーションを行う場合は、「*WebSphere Commerce Studio* マイグレーション・ガイド」を参照してください。

本書では、前のリリースおよびそれに適用された関連修正パッケージを総称して「Commerce Suite 5.1」と呼びます。本書での「WebSphere Commerce 5.5」または「WebSphere Commerce」という表現は、現行リリースの WebSphere Commerce 5.5 を意味しています。

変更内容の要約

このマイグレーション・ガイドとその更新版は、WebSphere Commerce Technical Library Web ページ (<http://www.ibm.com/software/genservers/commerce/library/>) から入手可能です。WebSphere Commerce の各版についての追加情報は、以下の概要のページを参照してください。

- Business Edition
(<http://www.ibm.com/software/genservers/commerce/wcbe/>)
- Professional Edition
(<http://www.ibm.com/software/genservers/commerce/wcpe/>)

追加サポート情報については、WebSphere Commerce サポート・サイト (<http://www.ibm.com/software/genservers/commerce/support/>) を参照してください。

この製品に加えられた最新の変更について知るには、製品の README ファイルを参照してください。このファイルは上記の Web サイトからも入手できます。WebSphere Commerce 5.5 およびそれがサポートする製品のインストールの手順については、「*WebSphere Commerce* インストール・ガイド」を参照してください。

本書の更新内容を以下に要約します。

WebSphere Commerce 5.5 の新機能

Commerce Suite 5.1 の後、重要な機能強化と新機能が WebSphere Commerce 5.5 で取り入れられました。WebSphere Commerce 5.5 に備わっている新しい機能および機能強化についての詳細は、バージョン 5.5 向けの「*WebSphere Commerce* 新着情報」を参照してください。機能強化を要約すると、以下のようになります。

- アクセス制御ポリシー — 組織ベースの許可ポリシーおよびポリシー・グループ・サブスクリプションを含むアクセス制御ポリシーの更新。

- 高度な割引および販売促進機能 — 割引および販売促進の単純なモデルと複雑なモデルの両方に対応した、従来より柔軟で容易にカスタマイズできるフレームワークを組み込む新規機能。
- 分析性およびビジネス・インテリジェンス — WebSphere Commerce Analyzer Advanced Edition が WebSphere Commerce に同梱されるようになり、より高度なデータ抽出ができるようになりました。
- 添付ファイルのサポート — 多様な添付ファイル形式をサポートしています。
- ビジネス・モデル — 消費者向け (企業顧客間取引) モデル、企業向け (企業間取引) モデル、および複数企業での企業間取引を可能にする企業向け間接モデルが含まれます。
- Catalog Manager — Web Editor は組み込まれなくなりました。WebSphere Commerce 5.5 の商品管理ツールを使用してください。
- チャンネル管理 — 新しいチャンネル管理機能により、企業向けモデルと企業向け間接モデルがいくつか追加され、販売店、流通業者、製造業者、バイヤーなどのさまざまな取引者間の関係および互いの対話がサポートされます。
- コラボレーション — 複数キューのサポートや、顧客サービス担当者が援助を待っている顧客をルーティングしたり優先順位を付けたりできるようにする機能を含む、さまざまな機能強化。
- 構成マネージャー — リモート Web サーバーを構成するためのツールが含まれるようになりました。パスワード・マネージャーも、このリリースでの新規ツールです。
- クーポンによる販売促進 — ゲスト顧客がクーポンを入手して特典を受けられるようにするさまざまな機能強化や、マーケティング・マネージャーがクーポンを管理できるようにする拡張機能。
- 資料 — WebSphere Commerce 5.5 向けの 2 つの新しいガイドとして、「管理ガイド」および「サンプル・ストア・ガイド」が備えられました。API およびコマンドのリファレンス情報などの開発者向けのオンライン・ヘルプは、WebSphere Commerce Studio に完全に統合されました。WebSphere Commerce ヘルプ・システムは、サイト管理者およびビジネス・ユーザー向けに調整されました。API およびコマンドのリファレンス情報は、WebSphere Commerce Studio のヘルプに移されました。
- E メール・アクティビティ — サイト・レベルでなくストア・レベルの E メールからのオプトアウト、E メール・アクティビティに関する統計の表示、E メール・アクティビティの内容を登録するためのユーザー・インターフェースなどの、機能強化。
- アウトバウンド・メッセージ通知 — アウトバウンド・メッセージ (E メール・メッセージを含む) が送信済みか送信済みでないかをモニターすることができます。
- インストール — WebSphere Commerce Install Shield で、WebSphere Commerce 5.5 とそれに関連したすべてのソフトウェアをインストールできるようになりました。標準、カスタム、および高速の 3 つのインストール方法が備えられました。高速インストールを使用すると、ユーザーは最小限の操作で WebSphere Commerce を迅速にインストールし、WebSphere Commerce インスタンスを作成できます。

- InterChange Server アダプター (以前の CrossWorlds[®])、つまり IBM WebSphere InterChange Server (ICS) アダプターは、ICS を使用する WebSphere Commerce ビジネス・インテグレーションを拡張する新しいメカニズムを提供します。このアダプターを使用すると、ICS を介して同期メッセージを送信することにより、WebSphere Commerce を外部システムと統合できます。
- ログイン — WebSphere Application Server のログイン機能である JRas ログイン・ツールキットが使用されるようになりました。JRas はメッセージ・ログインおよび診断トレース用のプリミティブを提供する複数の Java[™] パッケージから成り、ロガー、ハンドラー、フォーマッターなどが含まれています。
- メッセージング — Java 2 Enterprise Edition Connector Architecture (J2EE/CA) 標準にマイグレーションしました (ただし既存のメッセージ・システムの機能は同じです)。
- WebSphere MQ — WebSphere MQ (これまでの MQ トランスポート・アダプター) の既存の listener で、JMS/MQ との統合に、新規 Java 2 Enterprise Edition Connector Architecture (J2EE/CA) インフラストラクチャーが使用されるようになりました。WebSphere Commerce 5.5 は WebSphere MQ バージョン 5.3 をサポートします。
- 組織管理コンソール — 組織の検索、検索基準に適合した組織のリスト表示、およびメンバーが暗黙的に所属するグループにそのメンバーを含めたり除外したりする機能、などの新規機能があります。
- PRM メッセージ — 企業向け間接モデル用に、販売店マーケットプレイスと流通業者間のビジネス・フローを表す PRM (Partner Relationship Management) メッセージが導入されました。
- 商品アドバイザー — 商品アドバイザーのタスクに関する多数の機能強化。
- 商品管理 — バンドルとパッケージ、および取引管理アソシエーション用のウィザードとノートブックが新たに作成されました。作成にはウィザードを、変更にはノートブックを使用することができます。
- サンプル・ストア — FashionFlow および Tooltech (Business Edition のみに付属) といった既存のサンプル・ストアが拡張され、新たに CommercePlaza ストアが導入されました (Business Edition のみ)。
- ストア開発 — Publish ユーティリティーがストア・サービスから管理コンソールに移りました。「ストア・プロフィール (Store Profile)」、「税 (Tax)」、および「配送 (Shipping)」ノートブックは、WebSphere Commerce アクセラレーターに移りました。これにより、ストア・アーカイブではなくデータベース内で生きたデータを編集できます。その結果、ストア・サービスはなくなりました。パッケージ化および発行機能がさらに柔軟になりました。
- サブシステム — サブシステムの変更内容についての詳細は、バージョン 5.5 向けの「WebSphere Commerce 新着情報」を参照してください。
- システム管理 — 管理、問題判別、ログイン、メッセージング、およびパフォーマンス・モニターに対して機能強化が行われました。
- Web サービス — Web サービスは自己完結型で自己記述型のモジュラー・アプリケーションで、Web を介して発行、位置指定、呼び出しを行うことができます。呼び出し可能な Web サービス機能は、単純な要求から複雑なビジネス・プロセスにわたります。Web サービスがデプロイされて登録されると、他のアプリケーションはデプロイされたそのサービスを発見して呼び出すことができま

す。Web サービスは、Simple Object Access Protocol (SOAP)、Web サービス記述言語 (WSDL)、 Universal Description Discovery and Integration (UDDI) レジストリーなどの標準に基づいています。WebSphere Commerce は、そのビジネス機能を外部システムがアクセスできる Web サービスとして使用可能にすることにより、サービス・プロバイダーにすることができます。また、外部システムが提供する Web サービスを呼び出せるようにすることにより、WebSphere Commerce をサービス・リクエストにすることもできます。

- WebSphere Commerce Payments — これまで Payment Manager と呼んでいた WebSphere Commerce Payments が、WebSphere Commerce に統合されました。したがって、Payments は WebSphere Commerce のインストールおよび構成の一部となります。新規機能には新しい Cassette for Paymentech のサポートが含まれます。Cassette for VisaNet は、SSL ゲートウェイによるインターネット経由のクレジット・カード・トランザクション処理をサポートしています。また、Vital Processing Services 決済ネットワークの他に、First Horizon Merchant Services 決済ネットワークを使ってトランザクションを処理することもできます。SET™ および CyberCash 支払メソッドは、WebSphere Commerce Payments ではサポートされなくなりました。
- WebSphere Commerce の役割 — このリリースの WebSphere Commerce には、Channel Manager と Registered Customer という 2 つの役割が新たに導入されました。

このマイグレーション・ガイドの説明に従って Commerce Suite 5.1 システムから WebSphere Commerce 5.5 へのマイグレーションを終えてから、WebSphere Commerce 5.5 のオンライン・ヘルプや他の製品資料で、マイグレーションしたシステムでのこれらの機能の使用方法を調べてください。

さらに、WebSphere Commerce 5.4 では、WebSphere Commerce Suite 5.1 になかった重要な機能強化と機能を取り入れられました。以下の分野におけるこれらの機能と機能強化は、WebSphere Commerce 5.5 にも引き継がれました。この機能強化の詳細については、WebSphere Commerce Technical Library Web ページ (<http://www.ibm.com/software/commerce/library/>) から入手可能である「*IBM WebSphere Commerce 5.4 新着情報*」の資料を参照してください。

- コラボレーション

WebSphere Commerce は 2 つのタイプのコラボレーション機能: コラボレイティブ・ワークスペースおよびカスタマー・ケアをサポートします。

コラボレイティブ・ワークスペースは、複数の関係者がビジネス情報を共有するためのディスカッション・フォーラムです。たとえば、バイヤーとセラー (またはセラーのアカウント担当者) 間、またセラーの組織内でのビジネス・ユーザーの間で取引条件の交渉をすることができます。コラボレイティブ・ワークスペースは、Lotus® QuickPlace® による非同期通信をサポートします。

カスタマー・ケアは、Lotus Sametime® サーバーを使用した同期テキスト・インターフェイスを介して、リアルタイムの顧客サービス・サポートを提供します。顧客はサイトに入り、ストア・ページにあるリンクをクリックして、顧客サービス担当者 (CSR) に接続することができます。こうして、顧客と担当者はインターネット経由で通信またはチャットすることができます。

- 契約ベースのコマース

WebSphere Commerce のアカウント・フィーチャーとは対照的に、契約では、セラーとバイヤー組織の 1 単位間のトランザクションの詳細情報が定義されます。契約には、契約が適用するすべての顧客からのオーダーに関するある情報が含まれます。WebSphere Commerce アクセラレーターを使用することにより、アカウント担当者およびセールス・マネージャーは、ストアに定義されたアカウントおよび契約を制御することができます。

- 在庫管理

在庫サブシステムは、リアルタイムの在庫管理を提供する WebSphere Commerce のコンポーネントです。在庫サブシステムは、取引先から受け取った在庫や顧客によって返品された在庫に関する情報を記録し、在庫の数量を調整し、返品された在庫の処分を決定し、在庫を配送して受け取るといった機能を提供します。WebSphere Commerce は以下の領域での在庫機能をサポートします。

- 特別在庫受け取り
- 予定在庫
- 在庫の調整
- リリースおよび出荷
- 配送業務
- 運用の報告
- 在庫の返品および処分

- 組織管理コンソール

このバイヤー・サイドのインターフェースは、バイヤー管理者が組織内のショッパーとそれらのショッパーのオーダーを承認できるようにします。バイヤー管理者は、そのメンバーにアクセス・レベルを割り当てることもできます。

- 商品管理

WebSphere Commerce アクセラレーターの商品管理ツールにより、さまざまなウィザードやノートブックを使用してストアのマスター・カタログの商品を管理できます。商品管理ツールは、以前のバージョンの Catalog Editor に替わるものです。WebSphere Commerce アクセラレーターを使用して、以下を扱うことができます。

- 商品
- SKU
- カテゴリー
- 属性
- 価格設定

- 見積要求 (RFQ)

WebSphere Commerce Business Edition のフィーチャーには RFQ サポートがあります。これらの要求は、個々のアイテムまたは大量購入のための見積価格を得るために、セラー組織に送信されます。いったん受信されると、セラー組織は RFQ 応答を作成して応答します。オーダーは、RFQ 応答から直接生成できます。RFQ のフローを受領から解決まで定義および制御する XML ファイルを使用することにより、RFQ サポートがサイトでどのように処理されるかを完全に制御することが可能です。

- RMA 機能を含むリターンおよびリファンド

購入したものに満足しない顧客のために、マーチャントまたはセラーはリファンドのオファーをすることができます。リファンドは、交換商品取引（これは交換と同じではありません。交換は現在サポートされていません）の新規オーダーに使用できます。システムの設定は、すべて適切なレベルのセキュリティでオーバーライドすることができ、これが、アイテムがリファンド可能なものであるかどうか、元の商品が返品されなければならないかどうか、またリファンドの金額を制御します。マーチャントまたはセラーが選択すると、返品商品取引許可 (RMA) がセルフサービス・インターフェースを介し、構成可能な自動承認メカニズムを使用して、または顧客サービス担当者 (CSR) の援助を得て発行されます。返品された商品が受け取られると、売り物にならないとマークされて廃棄されるか、または別のオーダーを満たすために在庫に戻されます。

- 検索の機能強化

WebSphere Commerce には、ユーザーと顧客に検索機能を提供することを容易にするための、いくつかの検索方法が含まれています。以下の検索方法が提供されています。

- カタログ検索
- 商品アドバイザー
- WebSphere Commerce アクセラレーター・ツール

このマイグレーション・ガイドの説明に従って Commerce Suite 5.1 システムから WebSphere Commerce 5.5 へのマイグレーションを終えてから、WebSphere Commerce 5.5 のオンライン・ヘルプや他の製品資料で、マイグレーションしたシステムでのこれらの機能の使用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 を読み書きできること。

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
- 商品とカテゴリーの作成および管理
- 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：

Commerce Suite 5.1 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 5.5 Business Edition または Professional Edition へのマイグレーションを行うためのプロセスを説明します。

すべての言語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 5.5 Professional Edition または Business Edition 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、上記のバージョンより後のバージョンの Commerce Suite 5.1 (将来の修正パック、暫定修正、または他の機能強化を適用してアップグレードした WebSphere Commerce) から WebSphere Commerce 5.5 へのマイグレーションは扱っていません。

WebSphere Commerce 5.4 から WebSphere Commerce 5.5 への直接のマイグレーションもサポートされていることに注意してください。 WebSphere Commerce 5.4 からマイグレーションする場合は、「*WebSphere Commerce* マイグレーション・ガイド」で該当するパスについて参照してください。

将来の修正パック、暫定修正、または他の機能強化の適用によるマイグレーション上の問題については、 WebSphere Commerce サポート・ページ (<http://www.ibm.com/software/commerce/wscom/support/index.html>) を参照してください。

前のバージョンからのマイグレーション

このガイドは、前述のサポート対象マイグレーション・パスのマイグレーション・プロセスを説明します。

前のバージョンの Net.Commerce™ または Commerce Suite から WebSphere Commerce 5.5 への直接のマイグレーション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 からマイグレーションするには、まず、サポートされている上記のレベルのいずれかに既存のシステムをマイグレーションしてから、本書に従って WebSphere Commerce 5.5 にマイグレーションする必要があります。

- 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 5.5 は、以下の同一のオペレーティング・システム上での Commerce Suite 5.1 のマイグレーションをサポートしています。具体的には、以下のパスのマイグレーションがサポートされています。

- AIX[®] から AIX
- iSeries[™] から iSeries
- Linux から Linux
- Solaris オペレーティング環境から Solaris オペレーティング環境
- Windows[®] 2000 から Windows 2000
- Windows NT[®] から Windows 2000

上記のプラットフォームで WebSphere Commerce 5.5 にマイグレーションするステップについては、そのプラットフォームに該当する「*WebSphere Commerce* マイグレーション・ガイド (バージョン 5.5)」を参照してください。

WebSphere Commerce 5.5 は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。たとえば、Windows NT 上の Commerce Suite 5.1 から Linux 上の WebSphere Commerce 5.5 へのマイグレーションはサポートされていません。

サポートされないマイグレーション・パス

WebSphere Commerce 5.5 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

- WebSphere Commerce 5.5 システムを Commerce Suite 5.1 がインストールされているマシンにインストールする場合、Commerce Suite 5.1 のディレクトリーが保持されます。旧バージョンは WebSphere Commerce 5.5 にアップグレードされません。
- Commerce Suite 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 5.5 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」の『カスタマイズ・コードの変換』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.x から WebSphere Application Server 5.0 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、ご使用のシステムの該当する値に置換し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換してください。

host_name

WebSphere Commerce サーバーの完全修飾ホスト名 (たとえば、`mymachine.mydomain.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 を表していると理解してください。

パス変数

本書では、ディレクトリー・パスを表す以下の変数を使用します。

HTTPServer1312_installdir

この変数は、Commerce Suite 5.1 に付属する、IBM HTTP Server バージョン 1.3.12 の実際のインストール・ディレクトリーを表しています。このインストール・ディレクトリーのデフォルトは `/opt/IBMHTTPServer` です。

HTTPServer1326_installdir

この変数は、WebSphere Commerce 5.5 に付属する、IBM HTTP Server バージョン 1.3.26 の実際のインストール・ディレクトリーを表しています。このインストール・ディレクトリーのデフォルトは `/opt/IBMHttpServer` です。

WAS35_installdir

この変数は、Commerce Suite 5.1 に付属する、WebSphere Application Server バージョン 3.5.x の実際のインストール・ディレクトリーを表しています。このインストール・ディレクトリーのデフォルトは `/opt/IBMWebAS` です。

WAS50_installdir

この変数は、WebSphere Commerce 5.5 に付属する、WebSphere Application Server バージョン 5.0 の実際のインストール・ディレクトリーを表しています。このインストール・ディレクトリーのデフォルトは `/opt/WebSphere/AppServer` です。

WCS51_installdir

この変数は、Commerce Suite 5.1 の実際のインストール・ディレクトリーを表しています。このインストール・ディレクトリーのデフォルトは `/opt/WebSphere/CommerceSuite` です。

WC55_installdir

この変数は、WebSphere Commerce 5.5 の実際のインストール・ディレクトリーを表しています。このインストール・ディレクトリーのデフォルトは `/opt/WebSphere/CommerceServer55` です。

WPM221_path

この変数は、Commerce Suite 5.1 に付属する、以前のバージョンの

WebSphere Commerce Payments (WebSphere Payment Manager 2.2) の実際のインストール・ディレクトリーを表しています。このインストール・ディレクトリーのデフォルトは `/opt/PaymentManager` です。

Payments_installdir

この変数は、WebSphere Commerce 5.5 に付属する、WebSphere Commerce Payments バージョン 3.1.3 の実際のインストール・ディレクトリーを表しています。このインストール・ディレクトリーのデフォルトは `/opt/WebSphere/CommerceServer55/Payments` です。

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 5.5 へのマイグレーションに必要なタスクを説明します。これには以下が含まれます。

- 3 ページの『第 1 章 Commerce Suite 5.1 からのマイグレーションの前に』
- 21 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』
- 25 ページの『第 3 章 ソフトウェアのアップグレード』
- 31 ページの『第 4 章 Commerce インスタンス構成のマイグレーション』
- 65 ページの『第 6 章 Commerce Suite 5.1 データベースのマイグレーション』
- 77 ページの『第 7 章 インスタンスおよびデータベースのマイグレーション後』
- 93 ページの『第 8 章 Commerce Payments へのマイグレーション』

さらに、123 ページの『第 11 章 他の WebSphere Commerce コンポーネントのマイグレーション』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ます。

第 1 章 Commerce Suite 5.1 からの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しなければならない具体的なステップを説明します。さらにこのセクションでは、WebSphere Commerce 5.5 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

- 本書では、マイグレーション・プロセスの実施メソッドとして、以下の 2 つを説明します。

- 実稼働マシン上での マイグレーション - 必要なマシンは 1 台です。
- リモート・マイグレーション - マシンが 2 台必要です。

実稼働マシン上でのメソッドは 1 台のマシンしか使用しないため便利ではありますが、特に実動サイトでこの方法を使用すると、ほとんどのユーザーが、システムのダウン時間が長すぎると感じると思われます。完全なマイグレーション・プロセスの処理時間を占める主要な項目として、以下の 2 つがあります。

- データベース・マイグレーション - データベースのサイズ (ユーザー、商品、オーダー、ユーザー・トラフィックなどの数) にもよりますが、かなりの時間を要することがあります。
- コード・マイグレーション - コードのカスタマイズとテストに何日かかることがあります。

マイグレーションの作業を開始すると直ちに実動 Commerce Suite 5.1 サイトをシャットダウンする必要があるため、**ダウン時間が影響する実動サイトでは、実稼働マシン上でのメソッドよりもリモート・メソッドをお勧めします。**しかし、実動ではないサイトでマイグレーション・プロセスのカスタマイズとテストを行う場合は、実稼働マシン上でのメソッドが適しています。

- Commerce Suite 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 5.5 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」の『カスタマイズ・コードの変換』のセクションを参照してください。この遷移は、WebSphere Application Server 3.5.x から WebSphere Application Server 5.0 へのアップグレードを行うために必要です。
- WebSphere Commerce 5.5 をインストールした後に、IBM WebSphere Commerce 5.5.0.2 フィックスパックをインストールする必要があります。このフィックスパックには、マイグレーション・プロセスの修正と機能拡張が含まれています。このフィックスパックは、WebSphere Commerce サポート・サイト (<http://www.ibm.com/software/genservers/commerce/support/>) から入手可能です。ご使用の Edition へのリンクをクリックして、『**Download**』のセクションをご覧ください。このフィックスパックをダウンロードして、それをインストールするための指示に従ってください。このフィックスパックをインストールしなければ、本書でのマイグレーション・ステップ (インスタンスまたはデータベースのマイグレーションなど) を始めることはできません。

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。

MSGSTORE テーブルに新規メッセージが保管されていないことの確認

Commerce Suite 5.1 では、SendTransacted メッセージを送信すると、1 つのメッセージがデータベース内の MSGSTORE テーブルに保管されます。メッセージの送信後、そのエントリはデータベースから除去されます。つまり、このテーブル内には (設定値が誤っていない限り) メッセージが残っていないはずなので、このテーブル内のデータは一時的なものに見なされます。Commerce Suite 5.1 システムでは、このテーブルを定期的にクリーンアップする必要がありました。

WebSphere Commerce 5.5 へのマイグレーションを行うと、このテーブル内のデータを再利用することはできません。元の Commerce Suite 5.1 インスタンスをシャットダウンする前に、以下のステップを実行する必要があります。システムを安全に停止させるには、一定の期間、元のシステムが新たなオーダーや要求を受け取らないようにする必要があります。この期間に、以下のようにして、MSGSTORE テーブルに新規メッセージが保管されていないようにします。

1. サイト管理者として管理コンソールにログオンします。
2. サイトまたはストアの選択パネルで、「**サイト**」を選択します。
3. 「構成」 - 「トランスポート」メニューを開きます。
4. 各トランスポートの状況を「**非アクティブ**」に変更します。
5. 上記のステップを繰り返して、ストアごとにトランスポートを使用不可にします。

トランスポートを使用不可にしても、MSGSTORE テーブル内のメッセージの送信は妨げられません。新規メッセージが MSGSTORE テーブルに保管されないようになるだけです。

スケジューラーは MSGSTORE 内のすべてのメッセージを送付しようとします。(デフォルトでは、スケジューラーは SendTransactedMsg ジョブを 5 分間隔で実行します。この再試行回数は 3 です。)

15 分後には、MSGSTORE には再試行回数がゼロより大きいメッセージはないはずです。

6. 以下の SQL ステートメントを使って、MSGSTORE に再試行回数がゼロより大きいエントリが残っているかどうかを検査します。

```
select count(distinct msgid) from msgstore where retries > 0
```

この select ステートメントでは、Commerce Suite 5.1 がまだ送付する必要があるメッセージの数が示されます。結果は 1 行 1 列に 0 (ゼロ) 以上の数値結果で示されます。

結果が 0 (ゼロ) の場合、送付される保留メッセージがなく、テーブルはマイグレーションを行える状態にあることを意味しています。

結果が 0 (ゼロ) より大きい場合、Commerce Suite 5.1 がまだ残っているメッセージの送付を試行していることを意味します。なお、この数が 0 (ゼロ) になった後にこの SQL を実行しても、この数は増加しないはずで、結果が増え続けているようであれば、一部のトランスポートがシャットダウンされていないことを意味しています。上記のステップを参照して、該当するトランスポートを使用不可にしてください (サイトの他にすべてのストアに対しても行う必要があります)。

この SQL select ステートメントから 0 (ゼロ) が戻されたら、以下のステートメントを実行します。

```
select count(distinct msgid) from msgstore where retries = 0 or retries = -1
```

この select ステートメントは、まだ送付されていないメッセージがあるかどうかを判別します。

結果がゼロの場合、メッセージは残っていません。

結果がゼロより大きい場合は、MSGSTORE テーブルにメッセージが残っています。残っているメッセージは、削除することをお勧めします。(残っているメッセージを削除することにより、このテーブルを定期的にクリーンアップしておくことを常にお勧めします。)

既存オークションのクローズ

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 5.5 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。オークションをクローズする方法については、Commerce Suite 5.1 のオンライン・ヘルプのトピック『オークション入札のクローズ』を参照してください。

マイグレーションのためのステージング・サーバーの準備

マイグレーションの前に、ステージング・サーバー・データベースを実動 Commerce Suite 5.1 データベースと同期する必要があります。それには、以下のどちらかを行ってください。

- データ・マイグレーションの前に、ステージ伝搬ユーティリティー・コマンド (stagingprop) を実行します。このユーティリティーの実行の詳細については、Commerce Suite 5.1 オンライン・ヘルプの『ステージ伝搬ユーティリティー・コマンド』のトピックを参照してください。
- データベースへの変更を伝搬することを望まない場合には、ステージ・コピー・ユーティリティー・コマンド (stagingcopy) を使用してデータベースを同期化できます。(このオプションのほうが、通常かなりの時間を要します。) このユーティリティーの実行の詳細については、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』のトピックを参照してください。

ステージング・サーバー・データベースの同期化後に、以下を行ってください。

- マイグレーションをスピードアップするために、ステージング・データベースをクリーンアップします (オプション)。

ステージング・サーバーのデータベース・マイグレーションをスピードアップするために、ステージ・コピー・ユーティリティー (stagingcopy) を `-cleanup_stage_db_only_` オプションを使用して実行します。ステージ・コピー・ユーティリティーの実行方法の詳細については、Commerce Suite 5.1 オンライン・ヘルプの『ステージング・サーバー・コマンド』のセクションにある情報を参照してください。

注: カスタマイズされたトリガーすべてを除去する必要があります。除去しないと、ステージング・データベースのマイグレーション中にそれらのトリガーが実行されてしまい、予期しない問題を起こす可能性があります。必要な場合には、これらのトリガーを覚えておき、ステージング・データベース・マイグレーションの後で再度適用する必要があります。

- 65 ページの『第 6 章 Commerce Suite 5.1 データベースのマイグレーション』で説明しているとおり、ステージング・サーバー・データベースを、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します。このデータベースのマイグレーションのプロセスは、実動データベースの場合と似ています。

CONTRACT テーブル内の固有索引の検査

WebSphere Commerce 5.5 の CONTRACT テーブル内の固有索引は、以下の列に基づいて構成されています。

- NAME
- MEMBER_ID
- MAJORVERSION
- MINORVERSION
- ORIGIN

MEMBER_ID、MAJORVERSION、MINORVERSION、および ORIGIN の値が 2 つ以上の行で同じ場合は、CONTRACT テーブルの NAME 列がそれらの行で異なることを確認して、固有性を確かめる必要があります。これを行わないと、Commerce Suite 5.1 からデータをマイグレーションするときに問題が起こることがあります。

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 5.5 のいくつかのかぎとなる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・フレームワーク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 5.5 へのマイグレーション・プロセスのかぎとなる部分です。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、142 ページの『データ・マイグレーション・スクリプト』のデータベース・マイグレーション・スクリプトの説明をよく読んで、それでデータベースのマイグレーションを行うとき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	オーダー・アイテムの配送センターを判別します。	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 5.5 では、新しいタスク・コマンド・インターフェースによってこの機能が拡張され (もともとは WebSphere Commerce 5.4 リリースで拡張された)、使用可能な在庫または予定在庫のアイテムのチェック、割り振り、バック・オーダーを行えます。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、販売可能在庫数量による納期回答 (ATP) といいます。割り振りとバック・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. 販売可能在庫数量による納期回答 (ATP)

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	使用可能な在庫または予定在庫をチェック、割り振り、またはバック・オーダーします。配送センターと、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	配送センター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のチェック、割り振り、バック・オーダーを行ったり、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、割り振りやバック・オーダーがされていない OrderItems の在庫を、常に割り振りますが、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 5.5 にマイグレーションするときには、ATP 在庫への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 5.5 RECEIPT テーブルに移動されません。これ

は、WebSphere Commerce 5.5 在庫機能が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 5.5 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、163 ページの『付録 D. 後で実行する ATP 在庫への変換』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 5.5 は、マイグレーションされたインスタンスが再始動された後に PayResetPMCmd 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 5.5 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・プロセス中に C 状態のオーダーがあれば、データベース・マイグレーション・スクリプトによって、

`WC55_installdir/instances/instance_name` ディレクトリーに `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、以下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 5.5 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。

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 アクセラレーター・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 アクセラレーターを使用して、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。

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。WebSphere Commerce 5.4 でマスター・カタログが導入されたため、WebSphere Commerce 5.5 では、特定の方法でカタログ・データを構造化する必要があります。

WebSphere Commerce 5.5 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 5.5 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 5.5 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、その商品进行分类しているのと同じカテゴリで分類する必要があります。
- 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、72 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。

データベース・マイグレーション・スクリプト実行時の関連事項

カタログ管理用のツール (WebSphere Commerce 5.5 管理コンソールで使用できる商品管理ツール) では、商品はその開始点として使用します。すべての商品には一連の属性があり、共通の属性を持ったアイテムを、ある商品の下に追加することができます。

Commerce Suite 5.1 では、どのアイテムもある商品に属さなければならないという制約はありません。データベース・マイグレーションのマイグレーション前の段階で、Commerce Suite 5.1 データベース内に孤立アイテムがないかどうかを、プレマイグレーション・スクリプトが検査します。つまり、親商品を持たないアイテムが Commerce Suite 5.1 データベースに含まれているかどうかを検査します。

WebSphere Commerce 5.5 へのマイグレーションを行うときは、以下のことに留意してください。

- データベース・マイグレーション・スクリプトで孤立アイテムごとに 1 つの商品が作成されないようにするには、以下のようになります。
共通の属性を持つアイテムをグループ化し、商品を作成してから、WebSphere Commerce 5.5 へのマイグレーションを開始する必要があります。以下のテーブルにデータを移植する必要があります。

- CATENTRY (ProductBean を作成する)
- CATENTDESC (商品の説明)
- CATENTREL (商品とそのアイテムの関係)

この方法には、以下の利点があります。

- アイテムをそれぞれの商品の下に編成しておくので、WebSphere Commerce 5.5 へのマイグレーション後にアイテムを再編成する必要がない。
- 管理対象の商品が少なくなる。
- デフォルトのマイグレーション・スクリプトのアクションを使用することにした場合は、以下ようになります。

付属のデータ・マイグレーション・スクリプトで孤立アイテムごとにプレースホルダーとしての商品が作成されるようにしたい場合は、マイグレーション後に WebSphere Commerce 5.5 商品管理ツールを使って商品を管理し、類似する属性を持つアイテムを商品に追加することができます。類似する属性を持つ新規アイテムを、同じ商品の下に編成することができます。

この方法の利点は、WebSphere Commerce 5.5 商品管理ツールのすべての機能を活用できることです。この方法では、以下のことに留意してください。

- プレースホルダー商品とアイテムの両方を管理することが必要になります。
- データベース・スペースのディスク・スペース所要量が、以前よりも大きくなります。マイグレーション・スクリプト・ツールは、どのアイテムをどの商品の下にグループ化すればよいかを判別できません。そのため、孤立アイテムごとに 1 つのプレースホルダー 商品を作成します。プレースホルダー 商品とアイテムの関係を再編成することが必要になる場合があります。プレースホルダー 商品内の SKU (Partnumber) は、孤立アイテムの部品番号 (アイテムの P-Partnumber) に基づくものになります。

デフォルト契約

Business

WebSphere Commerce 5.4 で導入された、契約サポートを提供する条件は、WebSphere Commerce 5.5 にも引き継がれています。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 5.5 にマイグレーションした後に、WebSphere Commerce アクセラレーターを使用してそれを行います。WebSphere Commerce 5.5 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。

アクセス制御

アクセス制御は、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 5.5 のリソース・レベルでのアクセス制御は、ビジネス・オブジェクトを扱うコードの外部でアクセス制御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、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 5.5 では、アクセス制御は、GUI と、アクセス制御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 5.5 データベースに保管されています。そして、WebSphere Commerce 5.5 システムの始動時にメモリーにロードされます。

WebSphere Commerce 5.5 でアクセス制御に加えられた主な改善点を次に示します（これらはもともと WebSphere Commerce 5.4 で導入されたものです）。

- 柔軟性を向上させるために、アクセス制御ポリシーは、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。
- 階層的なアクセス制御が、アクセス制御モデルに組み込まれました。
- すべてのアクセス制御ポリシーは、ActionGroups、ResourceGroups、AccessGroups というグループに基づくようになりました。

さらに、組織は、ポリシーのグループにサブスクライブすることができるようになり、こうして組織が、それに適用するポリシーを完全にコントロールします。

Commerce Suite 5.1 に実装されているコマンド・レベルのアクセス制御で十分であり、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 アクセス制御テーブルを適切なポリシーとポリシー・グループ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、65 ページの『第 6 章 Commerce Suite 5.1 データベースのマイグレーション』の説明に従って、データベース・フレームワーク・マイグレーション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制御ポリシーはマイグレーションされません。
- `getResourceOwners()` メソッドを指定変更した場合、コマンド・レベルのアクセス制御では、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()` メソッドを指定変更していない場合は、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・レベルのアクセス制御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 5.5 のリソース・レベルのアクセス制御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のであれば、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・レベルのアクセス制御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制御は、リソース・レベルのアクセス制御によって実行できます。

アクセス制御の詳細については、121 ページの『第 10 章 アクセス制御サブシステムの考慮事項』を参照してください。

メンバー・サブシステム

WebSphere Commerce 5.5 と WebSphere Commerce Suite 5.1 の主な違いは、WebSphere Commerce 5.5 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、以下の事柄に基づいて、ユーザーおよび組織エンティティの親と祖先を決定します。

- 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祖先を判別したら、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 5.5 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、以下のようにして、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 (ORGENTITY) 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 5.5 の場合、以下の 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。
- ORGENTITY 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、以下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、ORGENTITY 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、ORGENTITY_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かどうかをチェックします。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 5.5 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 5.5 にデフォルトで付属している役割には、以下の例外を除いて、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 5.5 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 5.5 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、アクセス制御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 5.5 ではサポートされなくなったので、除去してください。

- Store Developer 役割は組み込まれていません。

Commerce Suite 5.1 は Store Developer 役割をサポートしていましたが、WebSphere Commerce 5.5 では不要になり使用されなくなりました。Store Developer 役割で実行されていたタスクは、自動化されているか、または WebSphere Commerce 5.5 の Site Administrator で実行できます。ユーザーに Commerce Suite 5.1 で Store Developer 役割 (-7) 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、アクセス制御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 Store Developer 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 5.5 ではサポートされなくなったので、除去してください。

Store Developer メンバー・グループに属しているユーザーがいる場合は、マイグレーション前に所属先を Site Administrator メンバー・グループに変更してください。ユーザーが WebSphere Commerce 5.5 でも Store Developer 役割を維持することを希望する場合、マイグレーション・スクリプトで Store Developer 役割を WebSphere Commerce 5.5 のユーザー定義の役割としてマイグレーションします。

- 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、以下の 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 5.5 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、アクセス制御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 5.5 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 5.5 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 5.5 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 5.5 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、以下のようになります。

- 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、データベース・マイグレーション・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 5.5 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、REGISTERTYPE=A を指定した XXX というアクセス・グループに Commerce Suite 5.1 のユーザーを割り当てましたが、WebSphere Commerce 5.5 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・グループに追加してください。
 - 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制御ポリシーに置き換えます。WebSphere Commerce 5.5 では、新しいアクセス制御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、アクセス制御ポリシーを使用することをお勧めします。13 ページの『アクセス制御』を参照してください。
 - ご使用の論理が REGISTERTYPE 列を検査するものの、それがアクセス制御のためでなければ、「メンバー・サブシステム」から使用できる以下のいずれかのプログラミング・インターフェースを使用してください。
 - 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 5.5 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 5.5 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。

識別名

WebSphere Commerce 5.5 では、ORGENTITY の DN 列にデータを移植する必要があります。これには、現在の組織エンティティの親階層がすべて含まれていなければなりません。データ・マイグレーション時に、マイグレーション・スクリプトによって、*WC55_installdir/instances/instance_name* ディレクトリーに *fillorgDN.sql* スクリプトが生成されます。ただし、マイグレーション・スクリプトは階層ツリーをすべて探索しなければならないため、組織エンティティが多数 (たとえば数百万) ある場合は生成に長時間を要することがあります。実際の組織エンティティがどのように編成されているかが分かっているならば、それに基づいて独自の *fillorgDN.sql* スクリプトを設計することを考慮できます。

カスタム *fillorgDN.sql* スクリプトは、必ず *WC55_installdir/instances/instance_name* ディレクトリーに置いてください。このスクリプトはデータ・マイグレーション後に使用されます。

fillorgDN.sql の内容は通常、以下のようになります。組織エンティティ (ORGENTITY) ごとに DN を更新して、祖父母および親の組織がすべて含まれるようにします。たとえば、以下のようになります。

```
update orgentity set dn='o=Root Organization' where orgentity_id=-2001;
update orgentity set dn='o=Default Organization,o=Root Organization'
  where orgentity_id=-2000;
update orgentity set dn='o=MyGrandParentOrg,o=Default Organization,o=Root Organization'
  where orgentity_id=3455;
update orgentity
  set dn='o=MyParentOrg,o=MyGrandParentOrg,o=Default Organization,o=Root Organization'
  where orgentity_id=3456;
```

独自の *fillorgDN.sql* スクリプトを作成することにした場合は、データ・マイグレーション・スクリプトによる *fillorgDN.sql* の生成をスキップすることができます。そのようにするには、*fillorgDN.sql* ファイル (空でも可) を、*WC55_installdir/instances/instance_name* ディレクトリーに置く必要があります。次に上記の例を指針として使用し、*fillorgDN.sql* スクリプトの内容を作成します。

WebSphere Application Server 5.0 への移動

IBM WebSphere Application Server バージョン 5 は、オープンな e-business プラットフォームの卓越した次世代インフラストラクチャーとなります。WebSphere ソフトウェア・プラットフォームの基礎として、WebSphere Application Server は、トランザクション管理、セキュリティ、クラスター化、パフォーマンス、可用性、接続性、スケーラビリティに関する機能を含むアプリケーション・サービスの完全セットを持つ、e-business アプリケーションの高度なデプロイメント環境を提供します。バージョン 5 は J2EE 仕様 (サーブレット 2.3、JSP 1.2、EJB 2.0 など) を完全にサポートし、さまざまな拡張機能もサポートしています。

WebSphere Application Server 5.0 で導入された新機能についての詳細は、WebSphere Application Server 5.0 Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>) の『What is new in Version 5』のトピックを参照してください。

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 5.5 へのマイグレーション中に問題が生じても、以前のシステムにリカバリーすることができます。

Commerce Suite 5.1 システムのバックアップ

Commerce Suite 5.1 システムをバックアップするには、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 5.1 システムの完全なシステム・バックアップを実行できます。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。システム・バックアップには、以下の重要なコンポーネントを必ず含めてください。

- 基礎になるすべてのサブディレクトリーおよびファイルを含む、Commerce Suite 5.1 メイン・インストール・ディレクトリー。これらのディレクトリーとファイルは、マイグレーション・プロセスの特定の時点で参照することが必要になる場合があるので、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する必要があります。特に、`WCS51_installdir` の下にある `instances` および `your_instance` サブディレクトリーのバックアップを取ってください。バックアップする必要がある具体的なファイルおよびディレクトリーのリストについては、『ディレクトリーおよびファイルのバックアップ』を参照してください。
- Commerce Suite 5.1 で使用されるすべてのカスタマイズ済みファイルとディレクトリー、およびデータベース、Web サーバー、WebSphere Commerce Payments (旧称 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。これらのファイルを、バックアップ・ディレクトリーにコピーします。
- Commerce Suite 5.1 データベース (23 ページの『データベースのバックアップ』の説明に従います)。

ディレクトリーおよびファイルのバックアップ

完全なシステム・バックアップを実行した場合、バックアップには以下のファイルおよびディレクトリーが含まれています。ただし、さらに、以下の重要なファイルおよびディレクトリーを、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することをお勧めします。次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、一時バックアップ・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 (`WCS51_installdir`) に切り替えます。
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。

以下に示すのは、追加のバックアップで考慮する必要がある重要なディレクトリーおよびファイルのリストです。

- *WCS51_installdir* ディレクトリーの下にある Commerce Suite 5.1 ファイルおよびディレクトリー。以下のファイルおよびディレクトリーは、35 ページの『WCIM を使用したインスタンス構成のマイグレーション』で WCIM インスタンス・マイグレーション・ツールを実行するときにバックアップされますが、それでも、これらのファイルをアクセス可能な場所にバックアップすることをお勧めします。

- web/*
- stores/*
- properties/*
- xml/product.*
- PaymentManager/profile.dtd
- instances/wcs_instances
- bin/cfg.passwd
- instances/instance_name/*

- *HTTPServer1312_installdir* ディレクトリーの下にある、以下の Web サーバー・ファイル:

- conf/httpd.conf
- ssl/keyfile.kdb
- ssl/keyfile.sth

- 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、Commerce Suite 5.1 で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 5.5 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 5.5 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。

- カスタム位置に保管したカスタマイズ・コード。カスタマイズ・コードは、完全なシステム・バックアップの一部として、または他の手動手段によって必ずバックアップする必要があります。カスタマイズ・コードは、35 ページの『WCIM を使用したインスタンス構成のマイグレーション』のインスタンス・マイグレーション・ステップ時に、WCIM ツールでバックアップされません。カスタム・

コードのマイグレーションの詳細については、「*WebSphere Commerce Studio* マイグレーション・ガイド」の『カスタマイズまたは拡張されたコードの変換』のセクションを参照してください。

- その他の以下の各種ファイル。
 - 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)。

データベースのバックアップ

以下のセクションでは、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、以下のマシンの 1 つから以下のアクションを実行します。

- DB2 サーバー・マシン — 2 層または 3 層環境の場合 (データベースは Commerce Suite 5.1 マシンからリモートのマシンにインストールされている)
- コマース・マシン — 単一層環境の場合 (データベースは Commerce Suite 5.1 マシンと同じマシンにインストールされている)

1. 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
```

2. Commerce Suite 5.1 システムをシャットダウンすることによって、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。
3. 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
```

データベースに接続しているアプリケーションがない場合は、以下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
```

データベースに接続中としてリストされるアプリケーションがある場合は、アプリケーションのソースを見つけてそれをシャットダウンする必要があります。接続中のアプリケーションをシャットダウンした後、db2 list applications コマンドを再発行して、状況モニターによって戻されるデータがないことを確認します。

4. 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
```

5. 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、

```
mkdir /db2_backup
```

6. 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、以下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 または WAS40 データベースのような Commerce Suite 5.1 以外のデータベース、または WebSphere Commerce Payments (以前の WebSphere Payment Manager) データベースである PAYMAN を、この時点でバックアップ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。

DB2 データベースのリストア: バックアップした DB2 データベースをリストアするには、以下のコマンドを使用します。

```
db2 restore database db_name from backup_dir
```

ここで、*db_name* はリストアするデータベースの名前 (たとえば、MALL) であり、*backup_dir* はデータベースのバックアップ・コピーを保持するディレクトリー (たとえば、/db2_backup) を表します。

DB2 データベースのリストアの詳細については、「DB2 管理の手引き」を参照してください。リストア・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 5.5 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、21 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、そのステップの詳細は、ご使用の製品とプラットフォームに該当するインストール・ガイドの付録を参照してください。

Commerce Suite 5.1 および WebSphere Commerce 5.5 IBM ソフトウェアのマッピング

以下の表は、Commerce Suite 5.1 と WebSphere Commerce 5.5 に同梱されているほとんどのソフトウェアについてのバージョン・レベルとデフォルト・インストール・パスのマッピングを示したものです。

表 4. WebSphere Commerce 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 5.5
WebSphere Commerce	バージョン 5.1 /opt/WebSphere/CommerceSuite	バージョン 5.5 /opt/WebSphere/CommerceServer55
DB2 Universal Database™ Enterprise Edition	バージョン 7.1.0.28 /usr/IBMdb2/V7.1	バージョン 8.1.0.16 (バージョン 8.1 に FixPak 2 を適用) /opt/IBM/db2/V8.1
IBM WebSphere Application Server — Advanced Edition	バージョン 3.5.4.1 /opt/IBMWebAS	バージョン 5.0.2 /opt/WebSphere/AppServer
IBM HTTP Server	バージョン 1.3.12 /opt/IBMHTTPServer	バージョン 1.3.26 /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
WebSphere Commerce Payments (以前の IBM WebSphere Payment Manager)	バージョン 2.2 /opt/PaymentManager	バージョン 3.1.3 /opt/WebSphere/CommerceServer55/payments (WebSphere Commerce 5.5 に統合)
IBM SecureWay® Directory Server	バージョン 3.2.1	バージョン 4.1
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 4.5.5 /opt/WebSphere/CommerceServer55/blaze

表 4. *WebSphere Commerce 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 5.5
Macromedia	バージョン	バージョン 5.5
LikeMinds		
Personalization Server		

ソフトウェアのアップグレードのアプローチ

クリーンな マシン (つまり、WebSphere Commerce 5.5 やその前のバージョンのソフトウェア・コンポーネントがインストールされていないマシン) に前提条件ソフトウェアを含む WebSphere Commerce 5.5 をインストールする必要があります。このクリーンな マシンは、ご使用の Commerce Suite 5.1 マシンのリモート・マシンであることが必要です。つまり、別個のマシンである必要があります。資産を容易に転送できるように 2 つのマシンをネットワーク上で接続しておく、マイグレーション・プロセスで役に立ちます。なお、WebSphere Commerce 5.5 システムは、サポートされている以下のいずれかのトポロジに従ってセットアップすることができます。

- 1 層 (WebSphere Commerce のすべてのコンポーネントが同一マシン上にある)
- 2 層 (データベースが WebSphere Commerce に対してリモート・マシン上にある)
- 3 層 (データベースと Web サーバーが WebSphere Commerce に対してリモート・マシン上にある)
- なお、WebSphere Commerce Payments は WebSphere Commerce マシンとは別のマシンにインストールすることができます。

詳しくは、『単独のマシンへの WebSphere Commerce 5.5 のインストール (リモート・マイグレーション)』を参照してください。

単独のマシンへの WebSphere Commerce 5.5 のインストール (リモート・マイグレーション)

WebSphere Commerce 5.5 システムを、Commerce Suite 5.1 とは別のマシンにインストールしてセットアップします。このターゲット・マシンは、前提条件を満たすメモリー量、CPU タイプ、およびディスク・スペースを備えた、WebSphere Commerce 5.5 対応プラットフォームのマシンでなければなりません。これらの前提要件については、ご使用のプラットフォーム用の「*WebSphere Commerce* インストール・ガイド」を参照してください。WebSphere Commerce 5.5 システムのインストールについては、ご使用のプラットフォーム用の「*WebSphere Commerce* インストール・ガイド」で、インストールに関する章を参照してください。

重要

WebSphere Commerce 5.5 をインストールした後に、IBM WebSphere Commerce 5.5.0.2 フィックスパックをインストールする必要があります。このフィックスパックには、マイグレーション・プロセスの修正と機能拡張が含まれています。このフィックスパックは、WebSphere Commerce サポート・サイト (<http://www.ibm.com/software/genservers/commerce/support/>) から入手可能です。ご使用の Edition へのリンクをクリックして、『**Download**』のセクションをご覧ください。このフィックスパックをダウンロードして、それをインストールするための指示に従ってください。このフィックスパックをインストールしなければ、本書でのマイグレーション・ステップ (インスタンスまたはデータベースのマイグレーションなど) を始めることはできません。

注: WebSphere Commerce 5.5 インストール・プロセス時にすべての WebSphere Application Server 暫定修正を正常に適用するには、必ず Web サーバーをシャットダウンする必要があります。さらに、WebSphere Commerce 5.5 インストールが完了した後に、`wcWASeFixer.log` ファイルを調べて、すべての WebSphere Application Server 暫定修正が正常にインストールされたことを確認します。

遷移に必要な追加のマシン要件

「*WebSphere Commerce インストール・ガイド*」に記載されている前提要件に加えて、マイグレーションのための以下の追加要件を満たす必要があります。

単一層構成の場合、ターゲット・マシンのハード・ディスク・スペース要件として、WebSphere Commerce 5.5 のインストールのハード・ディスク要件に加えて、現在の Commerce Suite 5.1 データベース・サイズの少なくとも 2 倍が必要になります。このガイドラインは、以下の要件に基づいています。

- Commerce Suite 5.1 データベースのバックアップ・コピーを保持するためのスペース。
- マイグレーションの各段階で WebSphere Commerce 5.5 データベースのバックアップ・コピーを保持するためのスペース。WebSphere Commerce 5.5 データベースの最終的なサイズは、Commerce Suite 5.1 データベースの現行サイズの 2 倍を超えることはないと推定されます。
- ログ用のスペース。

WebSphere Commerce テスト・インスタンスの作成

「*WebSphere Commerce インストール・ガイド*」の説明に従って WebSphere Commerce 5.5 ソフトウェアのインストールと検証を正常に終了したら、WebSphere Commerce 5.5 システムに WebSphere Commerce テスト・インスタンスを作成してください。

テスト・インスタンスを作成するには、WebSphere Commerce 構成マネージャー・インターフェースにログオンします。

構成マネージャーを使用して WebSphere Commerce インスタンスを作成するためのステップについては、ご使用のデータベースとプラットフォーム用の「*WebSphere Commerce インストール・ガイド*」で、構成についてのセクションを参照してください。

テスト・ストアを作成して、インストールした WebSphere Commerce 5.5 の機能性を検証することができます。テスト・インスタンスを残しておく場合は、Commerce Suite 5.1 からマイグレーションするものとは異なるインスタンス名にして、競合が起きないようにします。WebSphere Commerce 5.5 システムが機能していることを確認したら、テスト・インスタンスとテスト・データベースを削除できます。WebSphere Commerce 5.5 インスタンスの削除については、「*WebSphere Commerce インストール・ガイド*」を参照してください。

前の WebSphere Application Server 構成のマイグレーション

前の WebSphere Application Server 管理構成を新規 WebSphere Commerce 5.5 システムにマイグレーションするには、WASPreUpgrade および WASPostUpgrade ツールを使用します。

1. 以前の WebSphere Application Server 環境の管理サーバーが稼働していることを確認します。前の WebSphere Application Server マシン上で WASPreUpgrade ツールを実行して、前の管理構成をバックアップ・ディレクトリーに保存します。WASPreUpgrade ツールは WebSphere Application Server 5.0 製品 CD に入っています。このツールを実行するための構文の説明は、WebSphere Application Server 5.0 Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>) を参照してください。例:

```
WASPreUpgrade backupDirectory WAS35_installdir yourNodeName
```



EJB のインポートおよびデプロイに時間がかかるため、WebSphere Commerce 5.5 へのインスタンス・マイグレーションにおいてパフォーマンス上の問題が起こる可能性があります。インスタンス・マイグレーションのパフォーマンスを改善するためには、WASPreUpgrade ツールを以下のように実行した後、`websphere_backup.xml` ファイルを変更し、EJB に関連したセクションを除去する必要があります。

- a. WASPreupgrade の実行後、ファイル `websphere_backup.xml` を見つけてそのバックアップ・コピーを作成し、そのファイルをエディターで開きます。
- b. `<container name="WCS EJB Container" action="update">` エントリーを見つけて、以下のノード全体を除去します。

```
<container name="WCS EJB Container" action="update">  
...  
</container>
```

これにより、インスタンス・マイグレーション中に EJB がインポートまたはデプロイされるのを防ぐことができます。

2. 前の Commerce Suite 5.1 マシンにある前の管理構成のバックアップ・ディレクトリーの全体 (ファイルとサブディレクトリーをすべて含む) を、WebSphere Commerce 5.5 マシンに転送します。

3. バックアップ・ディレクトリーを転送したら、 `websphere_backup.xml` ファイルを編集します (このファイルは WebSphere Commerce 5.5 マシンのバックアップ・ディレクトリーにあります)。 Commerce Suite 5.1 マシンのノード名、IP アドレス、およびドメインが出現するすべての箇所を検索し、それらを WebSphere Commerce 5.5 マシンのノード名、IP アドレス、およびドメインにそれぞれ置き換えます。
4. WebSphere Application Server 5.0 マシンで WASPostUpgrade ツールを実行して、前の構成 (バックアップ・ディレクトリーに保管されている) を新規システムにマイグレーションします。 WASPostUpgrade ツールは `WAS50_installdir/bin` ディレクトリーにあります。このツールを実行するための構文については、WebSphere Application Server 5.0 Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>) を参照してください。
例:
`WASPostUpgrade backupDirectory`

`WAS50_installdir/logs` ディレクトリーにあるファイル `WASPostUpgrade.log` をチェックします。 EJB ファイルの欠落に関連したメッセージがあっても無視してください。このファイルは WebSphere Commerce 5.5 マシンでは使用できません。
5. WebSphere Application Server 管理コンソールにログオンし、 WebSphere Commerce Server `_instance_name_MigratedApp` アプリケーションがあればアンインストールします。このアプリケーションは WebSphere Application Server 5.0 の WASPostUpgrade によって作成されたものです。マイグレーション・プロセスでは、このアプリケーションは不要です。

WebSphere Application Server JDBC プロバイダー・レベルに関する考慮事項

WebSphere Application Server 5.0 にマイグレーションしたときの WASPostUpgrade の振る舞いは、デフォルトとして、それまでの WebSphere Application Server 3.5.x の JDBC プロバイダー・レベルの有効範囲を維持します。このことは、JDBC プロバイダーの有効範囲は WebSphere Application Server 3.5.x でのデフォルトの有効範囲がグローバルとしてそのまま保持されることを意味します。 WebSphere Application Server 5.0 での JDBC プロバイダーの有効範囲のデフォルトは、グローバルではなくサーバー・レベルより下の範囲になります。

次のステップ

31 ページの『第 4 章 Commerce インスタンス構成のマイグレーション』に進んでください。

第 4 章 Commerce 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 5.5 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 5.5 インスタンスを作成する方法については、ご使用データベースおよびプラットフォーム用の「*WebSphere Commerce インストール・ガイド*」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』のセクションを参照してください。

Commerce Suite 5.1 インスタンス・マイグレーションは、まず基本 WebSphere Commerce 5.5 (EAR) インスタンスから始めて、Commerce Suite 5.1 インスタンスから必要なコンポーネントを追加します。Commerce Suite 5.1 と WebSphere Commerce 5.5 との間ではインスタンスのパッケージ化においていくつかの変更点があるので、インスタンス・マイグレーション・ツールは、インスタンス・ファイルを WebSphere Commerce 5.5 EAR 内の適切なモジュールに置き、インスタンス・ファイル、構成ファイル、およびデプロイメント記述子を必要に応じて更新します。

WCIM (WebSphere Commerce Instance Migrator) ツールは、WebSphere Commerce インスタンス・マイグレーションのステップを実行するために提供されています。

LANG 環境変数の検査

インスタンスまたはデータベース・マイグレーションに進む前に、ご使用のオペレーティング・システムの LANG 環境変数が、使用する言語に合わせて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
```

これが正しいロケールに設定されていない場合は、データベース・マイグレーションの前に、以下のコマンドを実行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、データベース・マイグレーションを適正に動作させるために、システムで正しく設定することが必要です。

これが設定されていない場合、以下のステートメントを *.profile* ファイルに追加できます。

```
export LANG=locale
```

WebSphere Commerce がサポートするロケールのリストについては、「*WebSphere Commerce インストール・ガイド*」の『WebSphere Commerce で使用されるロケール』のセクションを参照してください。

Commerce Suite 5.1 product.xml ファイルのコピー

WebSphere Commerce Suite 5.1 の Start および Pro Edition には product.xml ファイルはありませんでした。リモート・マイグレーションの場合、WebSphere Commerce 5.5 は *WC55_installdir/migration* ディレクトリーに以下の 2 つのプレビルド・ファイルを準備します。

- product.xml.51.start
- product.xml.51.pro

リモート・マイグレーションの場合

- マイグレーションするエディションに適したファイルを、リモートの Commerce Suite 5.1 マシン上の *WC51_installdir/xml* ディレクトリーにコピーする必要があります。
- コピーしたファイルを product.xml に名前変更します。(WCIM ツールは、バックアップ実行前に product.xml ファイルのリモート・マイグレーションを必要とします。)

- コピーした product.xml ファイルで、以下の行を

```
<path>$COMMERCE_INSTALL_PATH</path>
```

以下の *WC51_installdir* のインストール・パスに更新します。

```
<path>WC51_installdir</path>
```

- 最後に、WebSphere Commerce 5.5 マシン上の *WC55_installdir/xml* ディレクトリーから、リモートの Commerce Suite 5.1 マシン上の *WC51_installdir/xml* ディレクトリーに、product.dtd ファイルをコピーします。

WebSphere Commerce 5.5 product.xml ファイルの検査 (実稼働マシン上でのマイグレーションのみ)

WCIM を実行する前に、WebSphere Commerce 5.5 マシン上にある product.xml ファイルの <migrationFrom> セクションを検査して、すべての値がマイグレーション・パス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。product.xml ファイルは *WC55_installdir/xml* ディレクトリーにあります。

product.xml ファイルの <migrationFrom> セクションで値が正しく設定されていることを、以下のように検証します。(<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>
  <fixpak>fixpak</fixpak>
  <path>path</path>
  <altpath>alt_path</altpath>
</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 (Intel™) のみ
- Pro** WebSphere Commerce Suite Pro Edition
- Business** WebSphere Commerce Business Edition 5.1 — Windows のみ
- **5** は WebSphere Commerce のバージョンです。このマイグレーションで考えられる値は 5 だけです。
- *rel* は以下のとおりです。
 - 1** これは以下のいずれかです。
 - WebSphere Commerce Suite 5.1
 - WebSphere Commerce Suite 5.1.0.1
 - WebSphere Commerce Suite 5.1.1 — Windows のみ
 - WebSphere Commerce Suite 5.1.1.1
 - WebSphere Commerce Suite 5.1.1.2
 - WebSphere Commerce Suite 5.1.1.3
 - WebSphere Commerce Suite 5.1.1.4
- *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
 - 1** WebSphere Commerce Suite 5.1.1.1
 - 1** WebSphere Commerce Suite 5.1.1.2
 - 1** WebSphere Commerce Suite 5.1.1.3
 - 1** WebSphere Commerce Suite 5.1.1.4
- *fixpak* は以下のいずれか 1 つです。
 - 0** WebSphere Commerce Suite 5.1.0.1
 - 1** WebSphere Commerce Suite 5.1.1.1
 - 2** WebSphere Commerce Suite 5.1.1.2
 - 3** WebSphere Commerce Suite 5.1.1.3
 - 4** WebSphere Commerce Suite 5.1.1.4
- *path* は、*WCS51_installdir* などの、以前の Commerce Suite 5.1 システムのインストール・パスです。
- *alt_path* は、*WCS51_installdir* などの、以前の Commerce Suite 5.1 システムの完全または長いインストール・パスです。これは Windows 上でのみ使用されます。Windows 以外のプラットフォーム上では、これは *path* と同じ値にするかまたはブランクのままにしておくことができます。



現行リリースを調べるには、サイト・テーブルに対して以下の SQL 照会を使用します。

```
select * from SITE
```

PRODUCTVERSION 列の結果を参照して、ご使用のリリース・レベルを判別してください。

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>
  <fixpak>0</fixpak>
  <path>/opt/WebSphere/CommerceSuite</path>
  <altpath>/opt/WebSphere/CommerceSuite</altpath>
</migrationFrom>
```

フィックスパック・レベルを調整するための予備手段

Linux 上でのマイグレーションの場合、WebSphere Commerce 5.5 product.xml ファイルで、フィックスパック・レベルが値 0 (ゼロ) に設定されていることを確認する必要があります。たとえば、Professional Edition 用の product.xml ファイルでは、デフォルト値は以下のようになっています。

```
<commerceserver>
  <name>IBM WebSphere Commerce Server</name>
  <edition>
    <name>Pro</name>
  </edition>
  <version>5</version>
  <release>5</release>
  <modification>0</modification>
  <fixpak>1</fixpak>
  <build>
    .
    .
    .
  </build>
</commerceserver>
```

インスタンス・マイグレーションを実行する前に、1 を 0 (つまり、<fixpak>0</fixpak>) に変更し、 product.xml ファイルを保管してください。

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

Commerce Suite 5.1 インスタンスを WebSphere Commerce 5.5 にマイグレーションする前に、以下のようにします。

- WCIM ツールは、インスタンスのディレクトリー構造全体をバックアップします。インスタンス・ディレクトリー以外の場所に 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がある場合は、インスタンスのルート・ディレクトリー構造下にあるディレクトリーに、これらのファイルやディレクトリーをコピーしなければなりません。このように、WCIM はカスタマイズ済みファイルおよびディレクトリーをリストアします。カスタム・コードのマイグレーションの詳細については、「*WebSphere Commerce Studio* マイグレーション・ガイド」を参照してください。

他の場所からファイルとディレクトリーをリストアするには、以下のようにします。

1. コマンド・ウィンドウを開いて、カスタマイズ済みのファイルがあるディレクトリー、またはカスタム・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 Commerce Suite 5.1 ディレクトリーにコピーします。
- WebSphere Application Server セキュリティーがオンになっている場合、インスタンスをマイグレーション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開きます。
 2. 「コンソール」 → 「Security Center (セキュリティー・センター)」をクリックして、「一般」タブの「Enable Security (セキュリティーを使用可能にする)」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。
 - WebSphere Commerce 管理ツール・ポートの更新 (オプション)

WebSphere Commerce 5.5 では、管理ツールは WebSphere Commerce サーバーとは異なるポート上で実行します。インスタンス・マイグレーションでは、以下のデフォルトのポート番号がこれらの管理ツールに割り当てられます。

ポート番号

WebSphere Commerce ツール

- 8000** WebSphere Commerce アクセラレーター
- 8002** WebSphere Commerce 管理コンソール
- 8004** WebSphere Commerce 組織管理コンソール

上記のツールにデフォルト以外のポートを使用したい場合は、*instance_name.xml* ファイルの <InstanceProperties> ノードの下にある <Websphere> ノードに、以下の 3 行を追加します。このファイルは *WC55_installdir/instances/your_instance/xml* ディレクトリーにあります。以下の行はデフォルトの設定を指定変更し、これらの管理ツールにデフォルト以外のポートを使用します。

```
ToolsPort = "desired_tools_port_number"
AdminPort="desired_admin_port_number"
OrgAdminPort="desired_org_admin_port_number"
```

たとえば、以下のようにします。

```
<Websphere Name="WebSphere Commerce DB2 DataSource"
  JDBCDriverLocation=""
  StoresWebApp="true"
  ToolsWebApp="true"
  WCMWebApp="true"
  name="WebSphere"
  ToolsPort = "9000"
  AdminPort="9001"
  OrgAdminPort="9002"
  port="900"/>
```

WCIM を使用した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、WCIM を実行して WebSphere Commerce の以前のインスタンスをマイグレーションする方法を説明します。WCIM の動作の詳細については、141 ページの『WCIM を使用したインスタンス・マイグレーション』を参照してください。

WCIM パッケージでは、*WC55_installdir/bin* ディレクトリーで以下の 2 つの主なスクリプトが提供されています。

- *wcimenv.sh* — *wcim* の環境設定値をセットアップし、システムで適切に実行されるようにします。
- *wcim.sh* — 実際のインスタンス・マイグレーションを実行します。

wcim の実行の基本プロセスは以下のとおりです。

1. インスタンスのバックアップ用に *wcimenv.sh* ファイルを更新します。リモート・マイグレーションの場合は、Commerce Suite 5.1 システム上でこのことが行われます。
2. バックアップのために *wcim* を実行します。
3. インスタンスのマイグレーション用に *wcimenv.sh* ファイルを更新します。リモート・マイグレーションの場合は、WebSphere Commerce 5.5 システム上でこのことが行われます。
4. マイグレーションのために *wcim* を実行します。

スクリプトの実行の詳細は、続くいくつかのセクションで記載しています。

WCIM 実行の前提要件

WCIM ツールを実行してインスタンスをマイグレーションする前に、以下のことを確認してください。

- WCIM を実行するマシンに、Apache ANT バージョン 1.4.1 をインストールしておく必要があります。このバージョンの Apache ANT は、WebSphere Application Server 5.0 と共にインストールされます。詳細については、WebSphere Application Server 5.0 Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>) の『Ant tasks』のトピックを参照してください。Apache ANT の詳細については、Web アドレス <http://ant.apache.org/index.html> を参照してください。ANT は Java ベースの構築ツールです (Make ユーティリティーにいくらか似ています)。WCIM ツールは、ANT ベースのツールで、インスタンスをマイグレーションするプロセスの多くを自動化することによって、手動ステップを最小化します。これは必要なすべてのデータを Commerce Suite 5.1 インスタンスから収集し、必要な操作を実行します。WCIM ツールは、コマンド行スクリプトから呼び出します。
- WebSphere Commerce ソフトウェア・コンポーネントが、Web サーバーも含め、WebSphere Commerce 5.5 レベルにアップグレードされている。
- すべてのインストール済み WebSphere Commerce アプリケーションが、WebSphere Application Server 5.0 によって正常にマイグレーションされた。
- WCIM ユーティリティーを実行してインスタンスをマイグレーションする前に、ユーティリティーを実行するユーザー ID (つまり、root または wasuser) が、以下の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する。

– *WC55_installdir/instances*

WebSphere Commerce 5.5 インスタンスが以前に作成されていない場合、つまり WebSphere Commerce 5.5 のインストール後、推奨されたとおりにテスト・

インスタンスを作成しなかった場合には、このディレクトリーは存在しません。ディレクトリーが存在しない場合には、WCIM ツールがそれを作成します。

- `WC51_installdir/instances`
- マイグレーション作業ディレクトリー (たとえば、`WC55_installdir/temp`)。

注:

1. `wcim.sh` スクリプトを `wcimenv.sh` 内の `WCIM_BACKUP` 変数を `true` に設定して実行すると、WCIM は、必要なパラメーターをそのスクリプトに渡すことによって、前のインスタンスのバックアップ ZIP ファイルを生成します。ZIP ファイル `wcbbackup51.zip` が `WORK_DIR/zip` ディレクトリーに生成されます。ここで `WORK_DIR` は、`wcimenv.sh` ファイル内のユーザー定義の作業ディレクトリーです。たとえば、作業ディレクトリーを `WC51_installdir/temp/` と定義できます (この変数が定義されている表 5 を参照してください)。
2. 手動で ZIP ファイル `wcbbackup51.zip` を、WebSphere Commerce 5.5 マシンの `WORK_DIR/zip` ディレクトリー (`WC55_installdir/temp/zip`) にコピーする必要があります。

Commerce Suite 5.1 インスタンスのマイグレーション

wcimenv で指定する変数

インスタンスのマイグレーションで `wcim.sh` を実行する前に、ご使用の環境に合わせて `wcimenv.sh` で環境変数を正しくセットアップする必要があります。さらに、インスタンスおよびインスタンス関連のファイル資産のバックアップを実行するか、それとも実際にそれらをマイグレーションするかに応じて、異なる方法で変数を設定する必要があります。

以下の表は、変数とその値を要約しています。

表 5. `wcimenv.sh` で設定される変数

変数	値	説明
WCIM_BACKUP	true	WCIM ツールは、WebSphere Commerce インスタンス関連のファイル資産をバックアップします。
	false	WCIM ツールは、WebSphere Commerce インスタンスおよびインスタンス関連のファイル資産をマイグレーションします。
WCIM_INPLACE	true	常に <code>WCIM_INPLACE=true</code> と設定します (ただし、 <code>WCIM_BACKUP=true</code> 、かつマイグレーションされたインスタンスが別のマシン — リモート・マシンに移される場合は例外)
	false	<code>WCIM_BACKUP=true</code> 、かつマイグレーションされたインスタンスが別のマシン — リモート・マシンに移される場合のみ、 <code>WCIM_INPLACE=false</code> と設定します。

表 5. *wcimenv.sh* で設定される変数 (続き)


変数	値	説明
 <p>上記の WCIM_BACKUP および WCIM_INPLACE 変数は、以下の組み合わせで使用されます。</p> <ul style="list-style-type: none"> バックアップの場合： <ul style="list-style-type: none"> WCIM_BACKUP は常に true に設定されます。 WCIM_INPLACE = true (Commerce Suite 5.1 が WebSphere Commerce 5.5 と同じマシン上にある場合 (実稼働マシン)) WCIM_INPLACE = false (Commerce Suite 5.1 が WebSphere Commerce 5.5 と同じマシン上にない場合 (リモート)) マイグレーションの場合： <ul style="list-style-type: none"> WCIM_BACKUP は常に false に設定されます。 WCIM_INPLACE = true (Commerce Suite 5.1 が WebSphere Commerce 5.5 と同じマシン上にある場合 (実稼働マシン)) WCIM_INPLACE = false (Commerce Suite 5.1 が WebSphere Commerce 5.5 と同じマシン上にない場合 (リモート)) 		
WCIM_MIGRATE_FROM	51	バックアップまたは Commerce Suite 5.1 からのマイグレーション
	WPM221	バックアップまたは Payment Manager 2.2.1 からのマイグレーション。詳細については、95 ページの『Payments インスタンスおよびデータベースのマイグレーション』を参照してください。
	WPM312	バックアップまたは Payment Manager 3.1.2 からのマイグレーション。詳細については、95 ページの『Payments インスタンスおよびデータベースのマイグレーション』を参照してください。
	WCP313	バックアップまたは WebSphere Commerce Payments 3.1.3 からのマイグレーション。詳細については、95 ページの『Payments インスタンスおよびデータベースのマイグレーション』を参照してください。
DB_TYPE	db2	DB2 データベースのデータベース・タイプ。
WAS_PATH	WAS35_installdir または WAS50_installdir	WebSphere Application Server のインストール・パス： <ul style="list-style-type: none"> バックアップの場合 (WCIM_BACKUP が true)、前の WebSphere Application Server のインストール・パスを指定します。 マイグレーションの場合 (WCIM_BACKUP が false)、WebSphere Application Server 5.0 のインストール・パスを指定します。
JAVA_EXE	java	Java 実行可能ファイルの SDK。

表 5. *wcimenv.sh* で設定される変数 (続き)

変数	値	説明
WC_PATH	<i>WCS51_installdir</i> または <i>WC55_installdir</i>	WebSphere Commerce のインストール・パス: <ul style="list-style-type: none"> バックアップの場合 (<i>WCIM_BACKUP</i> が true)、Commerce Suite 5.1 のインストール・パスを指定します。 マイグレーションの場合 (<i>WCIM_BACKUP</i> が false)、WebSphere Commerce 5.5 のインストール・パスを指定します。
WEBSERVER_PATH	<ul style="list-style-type: none"> IBM HTTP Server の場合: <i>HTTPServer1312_installdir</i> または <i>HTTPServer1326_installdir</i> 	<p>Web サーバーのインストール・パス:</p> <ul style="list-style-type: none"> バックアップの場合 (<i>WCIM_BACKUP</i> が true)、前の Web サーバーのパスを指定します。 マイグレーションの場合 (<i>WCIM_BACKUP</i> が false)、現行の Web サーバーのインストール・パスを指定します。 <p>注: Payment Manager 2.1.2 からのマイグレーションで、<i>WCIM_MIGRATE_FROM</i> に PM221 を指定している場合は、<i>HTTPServer1312_installdir/htdocs</i> ディレクトリーを指定する必要があります。</p>
ANT_PATH	<i>Ant_installdir/lib</i>	マシン上の ANT ライブラリー・パス。バックアップの場合、これは、 <i>ant.jar</i> のコピー先である Commerce Suite 5.1 マシン上の <i>temp_work_dir/lib</i> に設定できます。マイグレーションの場合、ANT 1.4.1 は WebSphere Application Server 5.0 に付属しているため、 <i>WAS50_installdir/lib</i> に設定できます。
WORK_DIR	ユーザー定義	WCIM ツールの作業ディレクトリー。(たとえば、 <i>WC55_installdir/temp</i>)
WPM_PATH	<i>Payments_installdir</i>	この変数は、WCIM を使用して以前のバージョンの WebSphere Commerce Payments をマイグレーションする場合に使用します。Payments のマイグレーションのプロセスは、95 ページの『Payments インスタンスおよびデータベースのマイグレーション』で詳細に説明しています。
LOG_FILE	ユーザー定義	<p>WCIM のログ・ファイル。任意のディレクトリーの下にある任意のファイル名を指定できます。ディレクトリーを指定しない場合、ログ・ファイルは <i>WORK_DIR</i> によって指定された作業ディレクトリーに生成されます。たとえば、以下のようにします。</p> <ul style="list-style-type: none"> バックアップの場合、<i>wcimbackup.log</i> マイグレーションの場合、<i>wcimmigration.log</i>
INSTANCE	Commerce Suite 5.1 <i>instance_name</i>	WCIM がマイグレーションするか、またはそのファイル資産を WCIM がバックアップするインスタンスの名前。

インスタンス・マイグレーションでの WCIM の実行

以下のセクションでは、インスタンス・マイグレーションのステップについて説明します。

インスタンス・マイグレーションの前提条件:

バックアップでの WCIM の実行: WCIM を実行してマイグレーションする前に、以下のように WCIM を実行して、現行のインスタンスおよびインスタンス関連ファイルのバックアップを作成してください。

1. Commerce Suite 5.1 マシン上に一時作業ディレクトリーを作成します (たとえば、*WCS51_installdir/temp* など)。さらに、以下のサブディレクトリーを *temp* 作業ディレクトリーの下に作成する必要があります。
 - *lib*
 - *bin*
 - *xml/migration*
 - *zip*
2. 以下のファイルをコピーします。
 - WebSphere Commerce 5.5 マシンの *WAS50_installdir/lib* ディレクトリーから、リモートの Commerce Suite 5.1 マシンの作業ディレクトリーの下にある *lib* サブディレクトリー (*WCS51_installdir/temp/lib*) に。
 - *ant.jar*
 - *j2ee.jar*これらのファイルは、WCIM の適切な動作に必要な Apache ANT 1.4.1 ファイル (WebSphere Application Server 5.0 から入手可能) です。Apache ANT については、Web アドレス <http://ant.apache.org/index.html> を参照してください。
 - WebSphere Commerce 5.5 マシンの *WC55_installdir/lib* ディレクトリーから、リモートの Commerce Suite 5.1 マシンの作業ディレクトリーの下にある *lib* サブディレクトリー (*WCS51_installdir/temp/lib*) に。
 - *Utilities.jar*
 - *ConfigManager.jar*
 - *xerces.jar*
 - WebSphere Commerce 5.5 マシンの *WC55_installdir/xml/migration* ディレクトリーから、*instbackup51.xml* ファイルを、Commerce Suite 5.1 マシンの作業ディレクトリーの下にある *xml/migration* サブディレクトリー (*WCS51_installdir/temp/xml/migration*) にコピーします。

注: Payments をマイグレーションする場合に、*instbackup51.xml* をコピーする代わりに、*instbackupwpm221.xml*、*instbackupwpm312.xml*、または *instbackupwcp313.xml* のいずれかを、マイグレーションするインスタンスの Payments のバージョンに応じてコピーします。
 - WebSphere Commerce 5.5 マシンの *WC55_installdir/bin* ディレクトリーから、Commerce Suite 5.1 マシンの作業ディレクトリーの下にある *bin* サブディレクトリー (*WCS51_installdir/temp/bin*) に、以下のファイルをコピーします。

- wcim.sh
- wcimenv.sh

- Commerce Suite 5.1 マシンの *WCS51_installdir/xml* ディレクトリーから、*product.xml* および *product.dtd* ファイルを、マイグレーション作業ディレクトリー (*WCS51_installdir/temp/xml*) の下の *xml* サブディレクトリーにコピーします。

- Payments インスタンス・バックアップの場合 :

WebSphere Commerce 5.5 マシンの *WC55_installdir/payments/lib* ディレクトリーから、*eTillConfig.jar* ファイル (Payments インスタンスのバックアップ用) を、リモートの Commerce Suite 5.1 マシンの作業ディレクトリーの下の *lib* サブディレクトリー (*WCS51_installdir/temp/lib*) にコピーします。

3. Commerce Suite 5.1 マシンでは、バックアップのために *wcim.sh* スクリプトを実行する前に、以下のように (太字の部分)、SDK for Java 用のパス変数を指定するステートメントを、*wcimenv.sh* ファイルに追加してください。

```
.....
# only call setenv.sh if it exists
if [ -f $BINDIR/setenv.sh ]; then
  . $BINDIR/setenv.sh
elif [ -f ./setenv.sh ]; then
  . ./setenv.sh
else
  # set JAVA_EXE here since setenv.sh is not found
  if [ "$OS_NAME" = "OS400" ]; then
    JAVA_EXE="java"
  elif [ "$OS_NAME" = "Linux" ]; then
    JAVA_EXE="java"
  fi
  fi
.....
```

4. Linux では、バージョン WebSphere Application Server 5.0.2 の *ant.jar* ファイルは、WebSphere Application Server 3.5 と互換性がありません。WCIM を使用して次のステップに進むために、以下の事柄を行う必要があります。

a. *java* ディレクトリーを、WebSphere Commerce 5.5 マシン上の *WAS502_installdir/lib* ディレクトリーから、リモート Commerce Suite 5.1 マシン上の作業ディレクトリーにコピーします。

b. Commerce Suite 5.1 マシン上の *wcimenv.sh* スクリプトに、以下の変更を行います。

```
JAVA_EXE="WORK_DIR/java/jre/bin/java"
PATH=WORK_DIR/java/jre/bin:$PATH
```

5. Commerce Suite 5.1 マシン上で、*wcim.sh* スクリプトを実行する前に、(リモート・マシンの作業ディレクトリーにコピーした) *wcimenv.sh* ファイルの環境変数をバックアップ用に更新します。

wcimenv.sh ファイルには、以下のエントリーが含まれています。

```
...
WCIM_BACKUP="true"
WCIM_INPLACE="false"
WCIM_MIGRATE_FROM="51"
DB_TYPE="db2"
```

```

export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE

WAS_PATH="WAS35_installdir"
JAVA_EXE="WORK_DIR/java/jre/bin/java"
WC_PATH="WCS51_installdir"
WEBSERVER_PATH="HTTPServer1312_installdir"
ANT_PATH="WCS51_installdir/temp/lib"
WORK_DIR="WCS51_installdir/temp"
LOG_FILE="=wcimbackup.log"
INSTANCE="instance_name"
PATH=WORK_DIR/java/jre/bin:$PATH

```

```

export WAS_PATH
export JAVA_EXE
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH
...

```

LOG_FILE 変数で設定されたログ・ファイルは、WORK_DIR/logs/ ディレクトリに生成されることに注意してください。

6. Commerce Suite 5.1 マシン上で、/usr/bin/sh ディレクトリが存在することを確認します。存在しない場合には、/usr/bin/sh から /bin/sh へのシンボリック・リンクを作成する必要があります。または、wcim.sh および wcimenv.sh の最初の行を、#!/bin/sh に変更する必要があります。

Commerce Suite 5.1 のバックアップだけを行う場合には、Commerce Suite 5.1 マシンに root としてログオンし、wcim.sh コマンドを実行するために作業ディレクトリに切り替えます。

7. Commerce Suite 5.1 マシン上で、wcim.sh スクリプトを以下のように実行します。

```
./wcim.sh [wcimenv_path] [PM_database_password]
```

ここで

- *wcimenv_path* は、マシン上で *wcimenv.sh* が存在するパスです。このファイルが *wcim.sh* と同じディレクトリ内にある場合、このパラメーターは不要です。
- *PM_database_password* は、WebSphere Commerce Payments データベース・パスワードで、WebSphere Commerce Payments インスタンスのマイグレーション時にのみ使用されます。WebSphere Commerce Payments のマイグレーションにおける WCIM の使用方法については、95 ページの『WCIM を使用した Payments インスタンスのマイグレーション』で詳細に説明しています。

注: *PM_database_password* パラメーターを使用する場合、*wcimenv* パスが現行ディレクトリにある場合でも、そのパスを指定する必要があります。この場合、現行ディレクトリを指定するには「.」を使用します。たとえば、以下のようにします。

```
./wcim.sh . mypassword
```

8. `wcbakup51.zip` ファイルを、リモート Commerce Suite 5.1 マシンの作業ディレクトリー (たとえば、`WCS51_installdir/temp/zip`) から、WebSphere Commerce 5.5 マシンの作業ディレクトリーの `zip` サブディレクトリー (たとえば、`WC55_installdir/temp/zip`) へコピーします。

`WC55_installdir/temp/zip` ディレクトリーが WebSphere Commerce 5.5 マシン上に存在しない場合には、それを作成する必要があります。

マイグレーションでの WCIM の実行:

注:

1. ステップ 2 (40 ページ) で説明されているとおりに、必要なファイルをすべて、WebSphere Commerce 5.5 マシンからリモートの Commerce Suite 5.1 マシンにコピーしたことを確認してください。
2. `WCIM_INPLACE` 変数は、実際にはリモート・マイグレーションの実行時に `true` に設定されています。

WCIM を実行してバックアップした後に、以下のように WCIM を実行して、インスタンスおよびインスタンス関連ファイルをマイグレーションします。

1. WebSphere Commerce 5.5 マシン上で `wcim.sh` スクリプトを実行する前に、`wcbakup51.zip` パッケージ内のすべてのファイルで、Commerce Suite 5.1 マシンのノード名、IP アドレス、およびドメインのすべての出現箇所を検索してください。これらの出現箇所をそれぞれ WebSphere Commerce 5.5 マシンのノード名、IP アドレス、およびドメイン・ネームに置き換えてください。たとえば、更新する必要がある可能性のあるファイルには、以下のものがあります。

- `product.xml`
- `product.xml.migration`
- `instance_name.xml`
- `instance_name_was.xml`
- `instance_name_was_DB.xml`
- `instance_name_was_Start.xml`
- `instance_name_was._EJB.xml`
- `instance_name_was_VH.xml`
- `cache.properties`

`wcbakup51.zip` パッケージ内のすべてのファイルを調べる必要があります。ただし、Web サーバー構成ファイル (たとえば、IBM HTTP Server を使用している場合は `httpd.conf`) は除きます。Web サーバー構成ファイルは、WCIM によってマイグレーションされないため、更新する必要はありません。

2. マイグレーション用にマシンの `wcimenv.sh` ファイル内の環境変数を更新します。

`wcimenv.sh` ファイルには、一般に以下のエントリーが含まれています。

...

```
WCIM_BACKUP="false"
WCIM_INPLACE="true"
WCIM_MIGRATE_FROM="51"
DB_TYPE="db2"
```

```
export WCIM_BACKUP
```



```

export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE

WAS_PATH="WAS50_installdir"
JAVA_EXE=$WAS_PATH/java/jre/bin/java
WC_PATH="WC55_installdir"
WEBSERVER_PATH="HTTPServer1326_installdir"
ANT_PATH="WAS50_installdir/lib"
WORK_DIR="WC55_installdir/temp"
LOG_FILE="wcimmigration.log"
INSTANCE="instance_name"
PATH=$WAS_PATH/java/jre/bin:$PATH

```

```

export WAS_PATH
export JAVA_EXE
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH

```

...

LOG_FILE 変数で設定されたログ・ファイルは、WORK_DIR/logs/ ディレクトリに生成されることに注意してください。

3. WebSphere Commerce 5.5 マシン上で、wcim.sh スクリプトを以下のように実行します。

```
./wcim.sh [wcimenv_path] [PM_database_password]
```

ここで

- *wcimenv_path* は、マシン上で *wcimenv.sh* が存在するパスです。このファイルが *wcim.sh* と同じディレクトリ内にある場合、このパラメーターは不要です。
- *PM_database_password* は、WebSphere Commerce Payments データベース・パスワードで、WebSphere Commerce Payments インスタンスのマイグレーション時にのみ使用されます。WebSphere Commerce Payments のマイグレーションにおける WCIM の使用方法については、95 ページの『WCIM を使用した Payments インスタンスのマイグレーション』で詳細に説明しています。

注: *PM_database_password* パラメーターを使用する場合、*wcimenv* パスが現行ディレクトリにある場合でも、そのパスを指定する必要があります。この場合、現行ディレクトリを指定するには「.」を使用します。たとえば、以下のようにします。

```
./wcim.sh . mypassword
```

WCIM が正常に実行されたことの検証

WCIM は、*wcimenv.sh* ファイルで SET LOG_FILE ディレクティブによって指定されたログ・ファイルを生成します。たとえば、SET LOG_FILE=wcimmigration.log の場合、WORK_DIR が *WC55_installdir/temp* に設定されていれば、ログ・ファイル *wcimmigration.log* が *WC55_installdir/temp/logs/* ディレクトリに生成されます。このログ・ファイルで、以下のステートメントを検索します。

```
Info: Instance migration is completed successfully.
```

WCIM はさらに、instanceXmlMigration.log マイグレーション・ログ・ファイルを、WORK_DIR/logs ディレクトリーに生成します (たとえば WORK_DIR=WC55_installdir/temp であれば WC55_installdir/temp/logs)。instanceXmlMigration.log で、以下の行を検索します。

```
Info: WebSphere Commerce instance configuration migrated successfully.  
Info: WebServer configuration migrated successfully.
```

注: このメッセージが存在していても、この時点ではデータベースをマイグレーションしていなかったため、WebSphere Commerce インスタンスが正常にマイグレーションされたことの保証にはなりません。65 ページの『第 6 章 Commerce Suite 5.1 データベースのマイグレーション』でデータベースを WebSphere Commerce 5.5 スキーマにマイグレーションしてから、92 ページの『インスタンスおよびデータベースが正常にマイグレーションされたことの検証』でインスタンスがマイグレーションされたことを検証できます。

WCIM 障害からのリカバリー: 何らかの理由で WCIM に障害が発生した場合、ログを検査して原因となっている可能性のある問題を確認し、必要な修正を行います。すでに確認されている問題およびその修正方法について、179 ページの『付録 F. トラブルシューティング』を参照することもできます。そうでなければ、システムを以前のバックアップからリストアし、このセクションのステップを注意深く検討して、インスタンス・マイグレーションを再実行する必要があります。

インスタンス XML マイグレーションに関連する問題の場合、問題を修正し、WCIM マイグレーションを再実行すればよいことにご注意ください。Web サーバー・マイグレーションの場合、WCIM はローカルの IBM HTTP Server のみマイグレーションするため、http.conf を削除し、httpd.conf.old を httpd.conf に名前変更してから、WCIM マイグレーションを実行する必要があります (マイグレーションにより httpd.conf ファイルが変更されていた場合)。



WCIM を使用したインスタンス・マイグレーション中に以下のメッセージを受け取った場合には、障害が起きている ANT ステートメントを特定し、適切な処置をとる必要があります。

```
Event: MethodId: invokeAnt - executing ant task:postmigrationcopy  
Error: Ant task has failed. Please check the log file.
```

1. wcimbackup.log (バックアップで WCIM を実行している場合) または wcmigration.log (マイグレーションで WCIM を実行している場合) に、誤ったストリングやメッセージがないかを調べます。これらのログ・ファイルは、WCIM の実行時に、ストリング "Error:" について WORK_DIR で指定された作業ディレクトリーの下に logs サブディレクトリーに生成されます。以下は、エラー・メッセージの例です。

```
Event: MethodId: invokeAnt - executing ant task:postmigrationcopy  
Error: Ant task has failed. Please check the log file.
```

2. Event: または Error: ステートメントの直前の Debug: ステートメントを参照してください。たとえば、以下のようなステートメントです。

```
Debug: java -classpath classpath -buildfile ... migrate.jsp
```

3. Debug: ステートメントによって参照される Java コマンドを、ご使用のマシン上の Java ランタイム環境ロケーションにあるコマンド・プロンプト、
WAS50_installdir/java/jre/bin で実行します。エラー・メッセージが表示されます。
4. エラー・メッセージに基づいて、適切な処置を行ってください。
5. エラー・メッセージを解決したら、WCIM を再実行してください。

WebSphere Commerce Payments キャッシャー・プロファイルの検査

インスタンス・マイグレーションの実行後に、WebSphere Commerce Payments の以下の項目を Payments ディレクトリーで検査します。

WC55_installdir/instances/instance_name/xml/payment

1. Cassette for SET および Cassette for CyberCash のプロファイルが存在していないことを確認します。
2. 既存の Commerce Suite 5.1 キャッシャー・プロファイルが更新されており、「Payment Manager」へのすべての参照が除去されていることを確認します。キャッシャー・プロファイル (たとえば WC51_CustomOffline_BillMe.profile) を開いて、それらが更新されているかを調べてください。更新を確認する良い方法は、以下の例にあるとおり、「Payment Manager」への参照が、「Commerce Payments」に変更されていることを確認することです。

マイグレーション前 :

```
<!-- =====
Parameters required by Payment Manager for order creation
===== -->
```

マイグレーション後 :

```
<!-- =====
Parameters required by WebSphere Commerce Payments for order creation
===== -->
```

3. 新規プロファイル WC_Paymentech が作成済みであることを確認します。
4. WebSphere Commerce 5.5 Payments profile.dtd ファイルが存在していることを確認します。
5. 以下のプロファイル・ファイルが存在していることを確認します。
 - WC_Paymentech (新規)
 - WC51_BankServACH.profile
 - WC51_CustomOffline_BillMe.profile
 - WC51_CustomOffline_COD.profile
 - WC51_OfflineCard.profile
 - WC51_VisaNet.profile
 - WC51_VisaNet_PCard.profile
 - WCS51_CustomOffline.profile
 - WCS51_OfflineCard.profile

WebSphere Commerce Payments のマイグレーションの詳細については、93 ページの『第 8 章 Commerce Payments へのマイグレーション』を参照してください。

WCIM スクリプトの実行後に

以下のセクションでは、データベース・マイグレーション・スクリプトを実行する前に完了すべき残りの作業を説明します。

WebSphere Application Server での古いクラスパスの除去

マイグレーションした ear ファイルを WebSphere Application Server 5.0 にデプロイする前に、以下のステップを実行して、マイグレーション済みのアプリケーション・サーバー (WebSphere_Commerce_Suite_-_instance_name) の JVM 設定から、すべての古い Commerce Suite 5.1 クラス・パスを除去します。

1. WebSphere Application Server 管理コンソールにログオンして、WebSphere Application Server - server1 アプリケーションを開始します。
2. 「サーバー (Servers)」 を拡張表示し、左のフレームで 「アプリケーション・サーバー (Application Servers)」 をクリックします。
3. 「アプリケーション・サーバー (Application Servers)」 パネルから、「server_name」 → 「プロセス定義 (Process Definition)」 → 「Java 仮想マシン (Java Virtual Machine)」 を選択します。server_name は、マイグレーション済みのアプリケーション・サーバーです (たとえば、WebSphere_Commerce_Suite_-_instance_name)。
4. すべての Commerce Suite 5.1 パスを参照する 「クラスパス (Classpath)」 フィールドからすべての内容を除去します。ただし wcsjni.jar ファイルのパスは除外します (存在する場合)。たとえば、WCS51_installdir/lib パスが組み込まれている以下のファイルを除去することができます (存在する場合)。
 - WCS51_installdir/lib/wcsejbimpl.jar
 - WCS51_installdir/lib/wcsruntime.jar
 - WCS51_installdir/lib/wcsmcruntime.jar
 - WCS51_installdir/lib/wcsauction.jar
 - WCS51_installdir/lib/wcscatalog.jar
 - WCS51_installdir/lib/wcslogging.jar
 - WCS51_installdir/lib/wcsexception.jar
 - WCS51_installdir/lib/wcssfc.jar

構成に応じて、削除することが必要な古いファイルがさらに存在する場合もあります。これらのファイルには WCS51_installdir/lib パスが組み込まれています。wcsjni.jar ファイルへのパスが存在する場合、それは削除しないでください。それが存在しない場合は、そのパスをマシン上に追加してください。

カスタマイズ済みコード拡張および wcsjni.jar ファイルを参照するパスだけを残してください。

5. 更新内容を保管します (「適用 (Apply)」 および 「OK」 をクリックします)。

ここでの着想は、WebSphere Commerce 5.5 環境で必要とされない Commerce Suite 5.1 JAR ファイルへの参照を除去することです。ただし、新規の classpath エントリを Commerce Suite 5.1 システムに追加した場合には、それらは引き続き WebSphere Commerce 5.5 環境 (たとえば、カスタマイズされたコマンド) で必要であると想定されます。

WebSphere Application Server での古い JVM プロパティの除去

WCIM スクリプトを実行した後、競合を避けるために、以下のステップを実行して、マイグレーションされたアプリケーション・サーバー (WebSphere_Commerce_Suite_-_instance_name) から、古い JVM プロパティを除去します。

1. WebSphere Application Server 管理コンソールにログオンします。
2. 「サーバー (Servers)」 を拡張表示し、左のフレームで 「アプリケーション・サーバー (Application Servers)」 をクリックします。
3. 「アプリケーション・サーバー (Application Servers)」 パネルから、 「server_name」 → 「プロセス定義 (Process Definition)」 → 「Java 仮想マシン (Java Virtual Machine)」 → 「カスタム・プロパティ (Custom Properties)」 (「追加プロパティ (Additional Properties)」 の下) を選択します。 server_name は、マイグレーション済みのアプリケーション・サーバーです (たとえば、WebSphere_Commerce_Suite_-_instance_name)。
4. 以下の古いプロパティを除去します (システムに存在する場合)。
 - javax.rmi.CORBA.UtilClass
 - com.ibm.ivj.ejb.runtime.instanceName
5. 更新内容を保管します (「適用 (Apply)」 および 「OK」 をクリックします)。

wcimWasConfig.jacl スクリプトの実行

WCIM は、wcimenv.sh ファイルの WORK_DIR で指定されたディレクトリー内に、ファイル wcimWasConfig.jacl を生成します。WCIM を使用してインスタンス・マイグレーションを完了した後に、この JACL スクリプト・ファイルを実行する必要があります。これには、マイグレーション時における手動での WebSphere Application Server 構成を最小化する、WebSphere Application Server 5.0 管理コマンドが含まれています。

とりわけ、このスクリプトは必要に応じて仮想ホストを WebSphere Application Server に追加します。デフォルトでは、以下の仮想ホストを、マイグレーション済み WebSphere Commerce インスタンスに追加します。

- VH_instance_name、ポート 80 および 443 (WebSphere Commerce インスタンスの場合)
- VH_instance_name_Tools、ポート 8000 (WebSphere Commerce アクセラレーターなどのツールの場合)
- VH_instance_name_Admin、ポート 8002 (WebSphere Commerce 管理コンソールの場合)
- VH_instance_name_OrgAdmin、ポート 8004 (WebSphere Commerce 組織管理コンソールの場合)

スクリプトを実行する前に

- リモート・マイグレーションの場合のみ、WebSphere Commerce 5.5 マシン上の WCIM_work_dir に生成された wcimWasConfig.jacl ファイルを編集します。Commerce Suite 5.1 マシンのノード名、IP アドレス、およびドメインのすべての

出現箇所を検索して、それらをそれぞれ WebSphere Commerce 5.5 マシンのノード名、IP アドレス、およびドメインに置き換えてください。

- `wcimWasConfig.jacl` スクリプトで、`DB_IMPL_CLASSNAME` 変数の値がヌルでないことを確認してください。それがヌルの場合には、使用しているデータベース管理システムに応じた値を割り当てる必要があります。たとえば、DB2 の場合は `COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource`、

- マイグレーション済みの `.ear` ファイルを、WebSphere Application Server 単一サーバー構成または WebSphere Application Server ネットワーク・デプロイメント構成のどちらにデプロイしたいかに応じて、スクリプト・ファイルを実行する前にその中の値を検証または更新する必要があります。

確認や変更が必要とされる行が、スクリプト内で示されます。

- サーバー名を `wcimWasConfig.jacl` ファイルで変更し、`server1` (デフォルトのサーバー) 以外の、マイグレーションしている適切なサーバー (たとえば、`WebSphere_Commerce_Server_-_instance_name`) に更新を適用します。スクリプトの行を参照してください。

```
# Please verify the server name
set SERVER_NAME server1
```



WebSphere Application Server 5.0 管理コンソールの「サーバー (Servers)」 → 「アプリケーション・サーバー (Application Servers)」から、マイグレーションされた WebSphere Commerce サーバーの名前を調べることができます。

- 必要であれば、`WebSphere_Commerce_Server_-_instance_name` アプリケーション・サーバーを開始します。これは WebSphere Application Server ND を使用している場合には WebSphere Application Server 管理コンソールから、そうでない場合には WebSphere Application Server `startServer` コマンドを使用して、開始することができます。

重要

これらの仮想ホストのポート番号は既存のポート番号と競合する可能性があります。たとえば、以前の WebSphere Commerce システムのインストールのデフォルト設定を使用している場合、デフォルトのホストのポート 443 があります。このポートは WebSphere Commerce 5.5 と競合し、マイグレーション済みのストアを正しく立ち上げるときの障害となります。

これらのポート番号または仮想ホスト名が、システムに既に存在していたり使用されていたりしないことを検証する必要があります。それらがシステムによって使用されている場合、スクリプトを実行する前に、スクリプト内で新規の仮想ホストの名前またはその定義済みのポート番号のいずれかを、未使用の値に変更する必要があります。さらに、新規の仮想ホストがすでに定義済みの場合は、それを除去することもできます。既存の仮想ホストを表示するには、WebSphere Application Server 5.0 管理コンソールから、「**環境 (Environment)**」を展開して「**仮想ホスト (Virtual Hosts)**」をダブルクリックすると、それらが WebSphere Application Server パネルに表示されます。

たとえば、WCIM スクリプトにより、Commerce Suite 5.1 バックアップから WebSphere Commerce 5.5 システム上に生成された default_host について、以下のホスト別名定義があるとします。

```
<X.X.X.X> :80
<X.X.X.X> :443
localhost :80
localhost :443
host_name :80
host_name :443
fully_qualified_host_name :80
fully_qualified_host_name : 443
```

上のエントリーがあると、マイグレーションの完了後、ホーム・ページを表示する際に問題が生じる場合があります。これらのエントリーを除去するだけで、ストア・ホーム・ページを表示させることができます。実際に、この default_host がコマース要求をつかむので、この場合、それらの要求は正しい宛先 — VH_instance_name に到達しません。WebSphere Commerce ツールは他のポート (たとえば 8000、8002、およびデフォルトで 80004) 上で実行されるので、管理コンソールのようなツールは影響を受けないからです。

JACL スクリプトを呼び出すには、以下のように入力します。

```
WAS50_installdir/bin/wsadmin -f WCIM_work_dir/wcimWasConfig.jacl
```

JDBC プロバイダーの検査

マイグレーションした .ear ファイルをアプリケーション・サーバーにデプロイする前に、WebSphere Application Server 管理コンソールで JDBC プロバイダーをチェックして、データ・ソースが現行の WebSphere Commerce 5.5 環境に合わせて正しく構成されていることを検査します。データ・ソースがヌルの場合、マイグレーション済みの .ear ファイルをデプロイする前に、それを手動で追加する必要があります。

1. WebSphere Application Server 管理コンソールに再びログオンします。管理コンソールから、「リソース (Resources)」を拡張表示して、「JDBC プロバイダー (JDBC Providers)」をクリックします。WebSphere Commerce 5.5 データ・ソースが表示されない場合には、以下のステップを実行します。
2. 「JDBC プロバイダー (JDBC Providers)」パネルから、「新規 (New)」をクリックし、ドロップダウン・リストから適切な JDBC プロバイダー (たとえば、DB2 データベースの DB2 JDBC プロバイダー) を選択します。
3. 「JDBC プロバイダー名 (JDBC Provider name)」、「JDBC クラスパス (JDBC classpath)」フィールドを構成します。「適用 (Apply)」および「OK」をクリックして、変更を保管します。
4. 「構成 (Configuration)」パネルの「追加プロパティ (Additional Properties)」セクションで、「データ・ソース バージョン 4 (Data Sources Version 4)」→「新規 (New)」をクリックします。
5. 「名前 (Name)」、「JNDI 名 (JNDI Name)」(たとえば DB2 データベースの場合は `jdbc/WebSphere Commerce Suite DB2 Datasource WC_instance_name`)、「データベース名 (Database Name)」、およびデータベースの「デフォルトのユーザー ID (Default user ID)」および「デフォルトのパスワード (Default Password)」フィールドを構成します。
6. 「適用 (Apply)」および「OK」をクリックします。
7. 変更を保管します。

マイグレーションした .ear ファイルのデプロイ

WCIM を使用し、`wcimWasConfig.jacl` スクリプトを実行してインスタンスをマイグレーションした後に、マイグレーションした .ear ファイルを以下のようにデプロイする必要があります。

1. WebSphere Application Server 管理コンソールに再びログオンします。
2. `node_name` →「アプリケーション (Applications)」→「新規アプリケーションのインストール (Install new application)」をクリックします。ブラウザーがマシン上でローカルの場合には、「ローカル・パス (Local path)」を選択し、マイグレーションした .ear ファイルへのパスを入力して (たとえば、`WCIM_work_dir/migrate/WebSphere_Commerce_Suite_-_instance_name.ear`)、「次へ」をクリックします。
3. 「EJB 1.1 CMPS のデフォルトのバインディング (Default bindings for EJB 1.1 CMPS)」(先頭にある「一般デフォルト・バインディング (Generate Default Bindings)」チェック・ボックスではない) を選択して、JNDI 名、データベース・ユーザー名、データベース・パスワードを入力し、このパネルの「EJB1.1 CMP バインディング (EJB1.1 CMP bindings)」のセクションのパスワードを確認して、「次へ」をクリックします。(たとえば、「JNDI 名 (JNDI name)」には `jdbc/WebSphere Commerce Suite DB2 Datasource WC_instance_name`、`username` には `db2inst1`、`password` には `db2inst1` などを入力します。)
4. 「新規アプリケーションのインストール (Install New Application)」ページが表示されます。以下のステップを完了します。
 - a. 「ステップ 1 (Step 1)」パネルで、デフォルトの構成を受け入れ、「次へ」をクリックします。

- b. 「**ステップ 2 (Step 2)**」パネルで、デフォルトの構成を受け入れ、「**次へ**」をクリックします。
 - c. 「**ステップ 3 (Step 3)**」パネルで、デフォルトの構成を受け入れ、「**次へ**」をクリックします。「**JNDI 名 (JNDI name)**」フィールドがブランクの場合は、ご使用のアプリケーションの JNDI 名をそのフィールドに追加します。
 - d. 「**ステップ 4 (Step 4)**」パネルで、リストされているすべての EJB を選択し、`node_name:jdbc/WebSphere Commerce Suite DB2 Datasource`
`WC_instance_name` などの JNDI 名を、「既存のリソース JNDI 名を指定 (Specify existing resource jndi name)」ドロップダウン・リストから選択します。「**適用**」、次いで「**次へ**」をクリックします。
 - e. 「**ステップ 5 (Step 5)**」パネルで、デフォルトの構成を受け入れ、「**次へ**」をクリックします。
 - f. 「**ステップ 6 (Step 6)**」パネルで、以下のような Web モジュール用の仮想ホストを調べて、必要であればマップします。

```
commerceAccelerator
    VH_instance_name_tools
OrganizationAdministration
    VH_instance_name_Orgadmin
SiteAdministration
    VH_instance_name_Admin
Stores  VH_instance_name
```
 - g. `server1` デフォルト・サーバーの代わりに、マイグレーション済みアプリケーション・サーバー `WebSphere_Commerce_Suite_-_instance_name` にデプロイするには、以下のようにします。
「ステップ 7: モジュールとアプリケーション・サーバーのマッピング (Step 7: Map modules to application servers)」パネルで、「**モジュール (Module)**」の近くにあるチェック・ボックスをチェックしてすべてのモジュールを選択し、マイグレーション済みサーバー (たとえば、`WC_demo`) を選択します。「**適用 (Apply)**」をクリックします。
デフォルトの構成を受け入れる場合は、マイグレーション済みの `.ear` ファイルが `server1` アプリケーション・サーバーにデプロイされます。
 - h. 残りのウィンドウについては、「**次へ**」をクリックします。
 - i. 「**終了 (Finish)**」をクリックし、更新成功のメッセージを受け取ります。
5. WebSphere Application Server 5.0 管理コンソールで、`node_name` → 「**環境 (Environment)**」 → 「**Web サーバー・プラグインの更新 (Update Web server plugin)**」 → 「**OK**」をクリックします。更新成功のメッセージが表示されることを確認します。
 6. 変更を保管します。
 7. 以下のようにして、Classloader モードと WAR Classloader ポリシーを変更します。
 - a. 「**アプリケーション (Applications)**」を拡張表示します。
 - b. 「**エンタープライズ・アプリケーション (Enterprise Applications)**」をクリックします。
 - c. `WC_` が先頭に付くアプリケーションをクリックします。
 - d. ClassLoader モードを **PARENT_LAST** に変更します。

- e. WAR Classloader ポリシーをアプリケーションに変更します。
 - f. 「OK」 をクリックします。
 - g. 変更を保管します。
8. WebSphere Application Server を再始動します。

注: JACL スクリプトで定義している仮想ホストを削除してしまったため、それらが 48 ページの『wcmWasConfig.jacl スクリプトの実行』で説明しているように既存のものとは競合しない場合は、EAR デプロイメント時に、既存のものを WAR モジュールにマップする必要があります。

WebSphere Application Server 5.0 管理コンソールを使用したアプリケーションのデプロイメントの詳細については、WebSphere Application Server 5.0 Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>) の『Deploying and managing applications』のトピックを参照してください。

カスタム・コードの遷移およびデプロイ

WebSphere Commerce 5.5 への移動の一部として、カスタム・コードおよびビジネス・ロジックを WebSphere Commerce 5.5 レベルに遷移する必要があります。コードを WebSphere Commerce 5.5 レベルにアップグレードする方法については、「*WebSphere Commerce Studio* マイグレーション・ガイド」の『カスタマイズまたは拡張されたコードの変換』を参照してください。

カスタマイズ・コード資産のデプロイメントの詳細については、「*WebSphere Commerce* プログラミング・ガイドとチュートリアル」でビジネス・ロジックまたはコントローラー・コマンドのデプロイメント、あるいはタスク・コマンドに関するセクションを必要に応じて参照してください。

デプロイされたインスタンス XML ファイルの更新

マイグレーションされた WebSphere Commerce インスタンスについて、以下のステップを実行して、インスタンス XML ファイルを WebSphere Commerce 5.5.0.2 レベルに更新してください。複数の WebSphere Commerce インスタンスがある場合には、変数 *instance_name* に対してインスタンス名を 1 つだけ使用してください。スクリプトはすべての既存のインスタンスを更新します。

1. WebSphere Commerce 非ルート・ユーザー ID に切り替えます。この ID は、WebSphere Commerce のインストール前に作成されています。
このユーザーに `config_ant` コマンドを実行する権限があることを確認してください。
2. `WC55_installdir/bin` ディレクトリーにナビゲートします。
3. 以下のコマンドを実行します。

```
./config_ant.sh -buildfile WC55_installdir/xml/config/updateInstances.xml  
-DupdateCEP=no -Duninstall=no -DinstName=instance_name
```

httpd.conf でのカスタマイズ

インスタンス・マイグレーション時に、WCIM は Web サーバー構成情報を以前の `instance.xml` ファイルから読み取り、その情報を使用して WebSphere Application Server 5.0 システムから WebSphere Commerce 5.5 `httpd.conf` ファイルを再構成します。前のバージョンの `httpd.conf` ファイルの、WebSphere Commerce インスタ

ンスに固有ではないカスタマイズ構成は、その構成が WCIM の扱う範囲外なので、マイグレーションされません。構成をカスタマイズした場合は、インスタンスおよび Web サーバーのマイグレーション後に、カスタマイズ構成を手動でマイグレーションする必要があります。

Web サーバーの再構成

Web サーバーを再構成するには、「*WebSphere Commerce インストール・ガイド*」にある Web サーバーのインストールに関するセクションに従ってください。そのセクションの、必須の事前構成ステップを完了させます。IBM HTTP Server を含め、リモート Web サーバーを構成するには、WebSphere Commerce 5.5 でいくらかの追加のユーザー構成が必要です。

この追加の構成を実行するには、マイグレーション後に構成マネージャーにログオンして、「Web サーバー (Web server)」パネルで必要なフィールドを完成させて Web サーバーを構成します。パネルで「適用 (Apply)」をクリックして、Web サーバーを構成します。構成マネージャーでのパネルの構成については、「*WebSphere Commerce インストール・ガイド*」で WebSphere Commerce インスタンスの構成についての章を参照してください。

IBM HTTP Server での SSL の使用可能化

WebSphere Commerce 5.5 の新規のインストールを完成させるリモート・マイグレーションの場合、デフォルトで Web サーバー上での SSL が使用可能になります。

SSL をシステムで使用可能にしておらず、SSL 暗号化を使用したい場合は、それを再び使用可能にする必要があります。

実動目的で SSL を使用可能にするには、「*WebSphere Commerce セキュリティー・ガイド*」の『IBM HTTP Server での実行のための SSL の使用可能化』を参照してください。

WebSphere Application Server EJB セキュリティーの使用可能化

Commerce Suite 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 5.5 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 セキュリティー・ガイド*」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。

EJB セキュリティーに関する追加のマシン推奨事項については、「*WebSphere Commerce セキュリティー・ガイド*」を参照してください。

静的コンテンツを持つファイルの文書ルート・ディレクトリーへのコピー

WebSphere Commerce WAR ディレクトリーの下にない Web 資産をマイグレーションするには、静的コンテンツを持つファイル (イメージや HTML ファイルなど) を、文書ルート・ディレクトリーにコピーする必要があります。

`WC51_installdir/instances/instance_name/web` ディレクトリー内の、静的コンテンツを持つファイルはすべて、`WC55_installdir/web` ディレクトリーにコピーしてください。

第 5 章 データベースのマイグレーションの前に

この章では、65 ページの『第 6 章 Commerce Suite 5.1 データベースのマイグレーション』で説明するデータベースの実際のマイグレーションを行う前に必要な、データベース・マイグレーション前のアクションと考慮事項を説明します。



- Commerce Suite 5.1 システムが操作可能な状態で、5 ページの『マイグレーション前のアクション』の他のセクションにあるステップ、および 5 ページの『MSGSTORE テーブルに新規メッセージが保管されていないことの確認』にあるステップを含めて 3 ページの『第 1 章 Commerce Suite 5.1 からのマイグレーションの前に』にあるステップを実行したことを確認してください。
- Commerce Suite 5.1 と WebSphere Commerce 5.5 のデータベース・スキーマの相違について詳しくは、「WebSphere Commerce Production and Development オンライン・ヘルプ」を参照してください。オンライン・ヘルプを起動したら、「**WebSphere Commerce 開発情報 (WebSphere Commerce Development information)**」 > 「**参照 (Reference)**」 > 「**データ (Data)**」 > 「**データベース・スキーマ (Database schema)**」の順に選択してください。ここから、「**データベース・テーブル (Database tables)**」を選択して、すべてのデータベース・スキーマの情報のアルファベット順リストを表示します。
「このリリースにおけるデータベースの変更 (**Database changes in this release**)」を選択して、バージョン 5.5 におけるスキーマの変更を表示します。オンライン・ヘルプでもデータ・モデルの情報を提供します。WebSphere Commerce Technical Library (<http://www.ibm.com/software/commerce/library/>) を参照して、オンライン・ヘルプ内のデータベース・スキーマ情報に対する変更も含めた、WebSphere Commerce 資料の最新バージョンを確認してください。

注: データベースをマイグレーションする前に、データベース・マイグレーション・スクリプトを実行するユーザーが、`WC55_installdir/instances` ディレクトリとそのサブディレクトリに対して、書き込み権限を持つことを確認する必要があります。

DB2 インスタンスの検査

DB2 インスタンス・ユーザーまたは WebSphere Commerce 5.5 上の WAS ユーザーのいずれかに切り替える前に、そのユーザーが `db2fgrp1` グループに属していることを確認してください。そうしないと、マイグレーション・スクリプトの実行時に問題が生じます (ストアード・プロシージャを正常に作成できません)。ユーザーを `db2fgrp1` グループに追加するには、ファイル `/etc/group` を編集し、ユーザー (たとえば `db2inst1` または `wasuser`) を、行 `db2fgrp1` の最後に追加します。

DB2 インスタンス・ユーザーまたは WAS ユーザーを `db2fgrp1` グループに追加した後、DB2 バージョン 8.2 インスタンスが正確にアップグレードされていること

を確認してください。ルート・ユーザーとしてログオンし、
DB2_installdir/instance ディレクトリーに切り替えて、以下のコマンドを実行し
ます。

```
./db2iupdt db2_instance_name
```

データベース・マイグレーション・スクリプトを実行する前に、現行パス (".") がシ
ステム変数 \$PATH に追加されていることを確認してください。\$PATH をエコー出
力して、現行ディレクトリー (".") が含まれているかを調べることができます。

列の順序の考慮

マイグレーション済みの WebSphere Commerce 5.5 データベース・テーブルの列の
順序は、 WebSphere Commerce 5.5 インスタンスを新たに作成したときに作成され
るテーブルと同じになるという保証はありません。select * を使って表を照会す
るときは、列位置ではなく列名を明示的に指定して列の値を検索することをお勧め
します。 JDBC プログラムでの例を以下に示します。

```
resultSet rs= statement.executeQuery("select * from address");
    (while rs.next())
    {
        p= rs.getObject(1); <---- 推奨しない方法

        p=rs.getLong("address_id"); <---- 推奨する方法
    }
```

逆に、データを挿入するときは、列名を明示的に指定します。たとえば、以下の挿
入ステートメントを考慮してください。

```
insert into address values( 10001,10002,'myaddress'); <-- 推奨しない方法

insert into address (address_id,addrbook_id, displayname)
    values (10001,10002,'myaddress'); <-- 推奨する方法
```

リモート・データベースの考慮事項

リモート・マイグレーションを実行する場合、パフォーマンス上の理由で、
Commerce Suite 5.1 データベースを WebSphere Commerce 5.5 マシンにローカルに
マイグレーションすることをお勧めします。これを行うには、以下のようになります。

1. Commerce Suite 5.1 データベースをバックアップします。
2. データベースのバックアップ (完全なバックアップ・ディレクトリーおよびファ
イル) を WebSphere Commerce 5.5 マシンに転送します。
3. バックアップしたデータベースを WebSphere Commerce 5.5 マシン上にリス
トアします。

データベースのバックアップおよびリストアについての詳細と構文は、23 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。 WebSphere Commerce 5.5
マシン上でのデータベースのリストアを完了した後、60 ページの『MSGTYPES テ
ーブル内の固有索引の検査』に進んで、データベースの事前マイグレーション・ス
テップを続行することができます。

データベース事前マイグレーションおよびマイグレーション・スクリプトを、
WebSphere Commerce 5.5 マシンから Commerce Suite 5.1 データベースに対してリ

モートに実行する場合には、Commerce Suite 5.1 データベースを WebSphere Commerce 5.5 マシン上でカタログする必要があります。下記のセクションを参照してください。

リモート DB2 データベースのカタログ

リモート DB2 データベースを使用していて、マイグレーション・スクリプトをリモートで実行したい場合は、そのデータベースを以下のようにローカル・ノードでカタログします。

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

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

各パラメーターの意味は以下のとおりです。

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の名。

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します。

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

各パラメーターの意味は以下のとおりです。

Rdb_name

リモート DB2 データベース (たとえば Rmall)。

db_name

リモート DB2 データベースの、カタログされた後のローカル名 (たとえば mall)。

node_name

上記の、前にカタログしたりモート・ノード。

たとえば、以下のようにします。

```
db2 catalog db RMALL as MALL at WCS151
```

WCS151 は、リモート・ノードのノード名です。

MSGTYPES テーブル内の固有索引の検査

WebSphere Commerce 5.5 には、MSGTYPES テーブルで作成される、独創的な新しい行があります (これは、マイグレーションされないシステムのためです)。MSGTYPE_ID 列は基本キーで、NAME 列は固有索引です。Commerce Suite 5.1 で独自のメッセージ・タイプを作成した場合、それらが新しいものと競合しないことを確認してください。競合する場合、MSGTYPE_ID 値または NAME、あるいはその両方を変更することができます。重要な点は、MSGTYPE_ID と NAME のどちらもテーブル内で固有でなければならないということです。カスタム・メッセージ・タイプを参照するコマンドを再コンパイルする必要があります。これを行わないと、Commerce Suite 5.1 からデータをマイグレーションするときに問題が起こることがあります。

以下のテーブルは、Commerce Suite 5.1 製品と比較して新しく追加された WebSphere Commerce 5.5 のメッセージ・タイプのリストです。新しく追加されている場合、MSGTYPES テーブルに、リストされているものと同じ MSGTYPE_ID または NAME を共有するものがあるかどうかチェックすることができます。

表 6. WebSphere Commerce 5.5 での MSGTYPES テーブルへの MSGTYPE_ID および NAME の追加

MSGTYPE_ID	NAME
130	OrderCancelForMerchant
150	OrderSummaryReportNotification
151	StoreUsageReportNotification
152	StoreCommerceReportNotification
153	SiteCommerceReportNotification
221	PriceAndAvailabilityCheck
222	BatchAvailability
223	ShoppingCartTransfer
224	CheckInventoryAvailabilityBE
250	CustomerMessage
251	CouponsSavedNotification
310	ResellerRegistrationApprovedNotification
320	ResellerRegistrationRejectedNotification
400	CollabEmail
501	RFQSubmitMessage
502	RFQCloseMessage
503	RFQCompleteMessage
600	ResponseNotification

なお、データ・マイグレーション後には、上記のすべてのエントリが表示されるわけではありません。

CollabEmail、RFQSubmitMessage、RFQCloseMessage、RFQCompleteMessage、および ResponseNotification は、Business Edition でのみ使用可能です。

データベース準備スクリプトの実行

データベース準備スクリプトは、Commerce Suite 5.1 データベースのアクセス制御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。データベース・マイグレーションを適切に行うには、このスクリプトを実行する必要があります。

データベース準備スクリプトは既存の Commerce Suite 5.1 データベースの分析を実行し、以下の特性を持つデータをレポートします。

- 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、149 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 151 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。要求されているアクションを行った後、データベース準備スクリプトを再実行して、必須のアクションがすべて行われたかを検証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、21 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、更新済み Commerce Suite 5.1 データベースを再度バックアップして、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 5.5 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・ウィンドウを開きます。
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 5.5 がインストールされている bin サブディレクトリに切り替えます。たとえば、WC55_installdir/bin などです。
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。

```
./premigratedb51.sh db2 db_name db_userID
db_userID_password instance_name db2_inst_home
```

各パラメーターの意味は以下のとおりです。

- `db2` は使用するデータベース管理システムを表し、DB2 データベースの場合は `db2` にする必要があります。
- `db_name` は、マイグレーションする Commerce Suite 5.1 データベースです (たとえば、`mall`)。
- `db_userID` は、マイグレーションする 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`)。
- `password` は、マイグレーションする 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`)。
- `instance_name` は、ご使用の Commerce Suite 5.1 インスタンスの名前です (たとえば、`demo`)。
- `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。

たとえば、Commerce Suite 5.1 データベースに対して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。

```
./premigratedb51.sh db2 mall db2inst1 db2inst1 demo
/home/db2inst1/sqllib
```

- __ 5. `WC55_installdir/logs/migrate` のサブディレクトリー `instance_name` に生成される `premigratedb51.log` ファイルを調べて、追加のメッセージまたはエラーがないかどうかをチェックします。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、23 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。

さらに、データベース準備スクリプトが WebSphere Commerce 5.5 の制限を超過するデータを含む列にフラグを立てた場合、154 ページの『列の長さの確認』を参照してください。

カスタム制約を除去にする

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ている場合、データ・マイグレーション時にそれらの参照保全制約 (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。

DB2 データベースの場合は、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。69 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 5.5 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 テーブルへの、またはその逆方向のすべての制約およびトリガー (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)、または Commerce Suite 5.1 テーブルに追加したすべての新規制約を確認します。

カスタム・テーブルを指す Commerce Suite 5.1 テーブルに追加された制約を判別するには、以下のようにします。

DB2 データベースに対して以下の照会を実行します。

```
db2 select CONSTNAME from SYSCAT.TABCONST
      where TABNAME='your_table_name'
```

3. DB2 データベースの場合、確認した制約をすべて除去します。データベース・マイグレーション・スクリプトの実行後、制約による SQL エラーが `migratedb_database_name.log` に記録されている場合は、データベースのマイグレーションを再試行する前にデータベースをリストアしてその制約を除去または使用不可にする必要があります。

DB2 データベースの場合、この後のセクションで示す SQL ステートメントを実行して制約を除去します。

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・ウィンドウを開きます。
2. 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
```

3. 以下のようにして、Commerce Suite 5.1 データベースに、データベースの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
```

`db_user_name` ユーザー ID は、`db_name` データベースのスキーマ所有者でなければなりません。

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
```

各パラメーターの意味は以下のとおりです。

- `table_name` は、制約を含むカスタマイズされた表の名前です。
- `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・マイグレーション後に追加し直す必要があるため、メモしておいてください。

第 6 章 Commerce Suite 5.1 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 5.5 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、3 ページの『第 1 章 Commerce Suite 5.1 からのマイグレーションの前に』、21 ページの『第 2 章 Commerce Suite 5.1 のバックアップ』、25 ページの『第 3 章 ソフトウェアのアップグレード』、および 57 ページの『第 5 章 データベースのマイグレーションの前に』のステップを完了することをお勧めします。また、この章で説明されているデータベースのマイグレーションに実際に取り掛かる前に、61 ページの『データベース準備スクリプトの実行』で扱われているデータベース準備スクリプトを必ず実行してください。WebSphere Commerce 5.5 に適したマイグレーションを実行するために、データベース準備スクリプトがデータベースを変更する点にご注意ください。

データをマイグレーションするためのスクリプトを完了した後に、追加のいくつかのデータベース・マイグレーション後ステップを実行して、データ・マイグレーションが正常に実行されたことを検証します。

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・ユーティリティーを実行するには、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されます。
- (テーブルに列を追加するなどして) Commerce Suite 5.1 スキーマを拡張した場合、139 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 5.5 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 5.5 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 5.5 インストール・ガイド」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空きディスク・スペースに加え、ログを収めるためのスペース、およびマイグレーション・スクリプトを実行するための一時ディスク・スペースがマシン上にあることを確かめてください。この追加のディスク・スペースは、少なくとも現在の Commerce Suite 5.1 データベースの 2 倍のサイズにする必要があります。
- DB2 データベースの場合、データベース・マイグレーション・スクリプトを実行する前に、ご使用のデータベースに合わせて DB2 ログ・ファイルのサイズを変更する必要があります。そうしない場合、トランザクション・ログ・ファイルが、スクリプトの実行時に満杯になる可能性があります。以下の SQL ステートメントを使用して、DB2 ログ・ファイルのサイズを最低でも 1000 ブロックに設定する必要があります。

```
db2 update database configuration for data_base_name using logfilsiz 1000
```

データベースが大規模な場合、ログ・ファイルのサイズを 10000 ブロックまたはそれ以上に変更することが必要になる場合があります。(ログ・ファイルのサイズは DB2 Control Center 内で変更することもできます。)

- データベースをマイグレーションする前に、それをバックアップすることを考慮できます。データベース・マイグレーション・スクリプトは、どの場合もデータベースのバックアップを実行することにご注意ください。現行データベースをバックアップする場合は、23 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。

データベースのマイグレーション

データベース準備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、Commerce Suite 5.1 スキーマを WebSphere Commerce 5.5 レベルに更新することができます。Commerce Suite 5.1 データベースのマイグレーション時のマイグレーション・スクリプトの働きの概要については、141 ページの『付録 C. WCIM ツールおよびデータ・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 5.5 との間のデータベース・スキーマの変更の要約については、「WebSphere Commerce Production and Development オンライン・ヘルプ」の『このリリースでのデータベース・スキーマの変更』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。

データベースをマイグレーションする前に、データベース・マイグレーション・スクリプトを実行するユーザーに `WC55_installdir` ディレクトリー内のファイルにアクセスする権限があるか確認してください。

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 `Rdatabase_name` としてカタログされます。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)。

DB2 データベース

以下のステップは、DB2 データベースのマイグレーション・スクリプトを実行する方法を示しています。

- __ 1. コマンド・プロンプト・ウィンドウを開きます。

- __ 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 5.5 がインストールされている `bin` サブディレクトリーに切り替えます。たとえば、`WC55_installdir/bin` などです。

__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。

```
./migratedb51.sh db2 db_name db_userID password instance_name atp_option  
db2_install_dir db2_inst_home schema_owner
```

ここで

- *db2* は、DB2 データベースをマイグレーションすることを示します。
- *db_name* は、マイグレーションする Commerce Suite 5.1 データベースです (たとえば、mall)。
- *db_userID* は、マイグレーションする 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)。
- *password* は、マイグレーションする 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)。
- *instance_name* は、データベースが関連付けられている Commerce Suite 5.1 インスタンスの名前です (たとえば、demo)。
- *atp_option* は、以下のように *atp* または *noatp* のいずれかにすることができます。
 - *atp* は、既存の Commerce Suite 5.1 在庫データを、WebSphere Commerce 5.5 が求める販売可能在庫数量による納期回答 (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、販売可能在庫数量による納期回答 (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。

この重要なオプションに関する詳細および考慮事項については、155 ページの『ATP マイグレーションに関する考慮事項』を参照してください。

- *db2_install_dir* は、DB2 のインストール・パスです。たとえば、`/usr/IBMDB2/V8.1` のようになります。
- *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。
- *schema_owner* は、DB2 データベースのデータベース・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。 *schema_owner* は、*db_userID* と全く同じである必要があります。

たとえば、以下のコマンドを使用して、DB2 データベースのデータベース・マイグレーション・スクリプトを実行在庫表記を ATP に変換できます。

```
./migratedb51.sh db2 mall db2inst1 db2inst1 demo atp  
/usr/IBMDB2/V8.1 /home/db2inst1/sqllib DB2INST1
```

マイグレーション・スクリプトの実行時には、追加のスクリプトおよびファイルが生成されます。それらは `WC55_installdir/instances/instance_name/migration` ディレクトリーに保管され、さまざまな Commerce Suite 5.1 インスタンスと関連付けられている複数のデータベースをマイグレーションする場合に、ファイル名の衝突やファイルの上書きを避けることができます。

データベース・マイグレーション・スクリプトの実行後、155 ページの『データベース・マイグレーション・ログ・ファイルのチェック』で説明されているように、

ログ・ファイルをチェックしてください。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。

ID リゾルバーのパフォーマンスの考慮事項

DB2 バージョン 8.x で、データ・マイグレーションの ID リゾルバー・フェーズを完了する際に、ローパフォーマンスが起こる場合には、DB2 構成パラメーターをいくつか変更する必要があります。通常、それぞれの ID リゾルバー・フェーズは、数分以内で完了します。ID リゾルバー・フェーズが完了するのに 1 時間近くかかるようであれば、パフォーマンスを改善するために、以下のアクションを考慮する必要があります。

- DB2 構成パラメーター CPUSPEED を値 -1 に変更します。以下の DB2 構成更新コマンドを入力します。

```
db2 -v update dbm cfg using cpuspeed -1
```

CPU 速度 (命令あたりミリ秒単位) が DB2 SQL 最適化プログラムによって使用されて、特定の操作を実行するコストが見積もられます。このパラメーターの値は、CPU 速度を測定するように設計されたプログラムからの出力に基づいて、データベース・マネージャーのインストール時に自動的に設定されます。このパラメーターの詳細については、DB2 テクニカル・ライブラリーにある、DB2 の管理に関する情報を参照してください。

- 以下のようにして、データベースに使用するバッファ・プール・サイズを増やすか、または SYSCAT テーブル・スペースに専用バッファ・プールを割り当てます。

```
CREATE BUFFERPOOL NEWBUFPOOL SIZE 20000 PAGESIZE 4K  
ALTER TABLESPACE SYSCATSPACE BUFFERPOOL NEWBUFPOOL
```

ID リゾルバーはメタデータのフェッチを何回も集中的に実行し、これが SYSCAT テーブル・スペース用に絞られていきます。新規のバッファ・プールを割り振ることで、ボトルネックが緩和されます。上記の変更を行った後は、データベース・マイグレーション・スクリプトを実行中の ID リゾルバー・フェーズの完了時に、パフォーマンスの向上が見られるはずですが、

カスタム制約のリストア

DB2 データベースの場合、マイグレーション・スクリプトを使用して Commerce Suite 5.1 データベースをマイグレーションした後に、62 ページの『カスタム制約を除去にする』で除去した参照制約をリストアする必要があります。

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・ウィンドウを開きます。
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. マイグレーション済みの Commerce Suite 5.1 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。

```
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表の名前です。
- *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表の名前です。

参照制約を追加する SQL ステートメントの例については、*WC55_installdir/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 5.5 では、ORGENITY 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、ORGENITY 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、*WC55_installdir/instances/instance_name/migration/db2* ディレクトリーにあります。
- 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 5.5 によって組織エンティティが LDAP サーバー上に作成されるとき、ORGENITY 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* スクリプトを実行します。

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、以下のようにします。

1. コマンド・プロンプト・ウィンドウを開きます。
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. WC55_installdir/bin サブディレクトリーに移動します。
4. 以下のようにして、migrateDN スクリプトを、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 instance_name  
db2insthome host_name
```

ここで

- *db_name* は、 WebSphere Commerce 5.5 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)。
- *instance_name* は、ご使用の Commerce Suite 5.1 インスタンスの名前です (たとえば、demo)。
- *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)。
- *host_name* は、マシンの完全修飾ホスト名です (たとえば、myhost.montreal.ca)。

たとえば、マイグレーションされた Commerce Suite 5.1 mall データベース上で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 demo /home/db2inst1/sqllib  
myhost.montreal.ca
```

5. WC55_installdir/logs/migration/*instance_name* 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。

大文字小文字を区別する検索の除去

WebSphere Commerce 5.5.0.2 の組織エンティティーを一意的に識別する新規インプリメンテーションでは、データベース内の組織エンティティーの識別名が常に小文字で保管されるようにします。識別名が小文字でない場合には、識別名の検索時に

問題が生じます。また、新規インプリメンテーションは、この組織エンティティーに対するファインダーのパフォーマンスを向上させます。小文字に変換するマイグレーション・スクリプトは、現行データベース内に組織エンティティーを持つ既存のユーザーを対象としています。

そのスクリプトを実行して、識別名を小文字に変換する必要があります。ご使用のプラットフォームでのスクリプトの実行についての詳細は、**WebSphere Commerce 5.5.0.2 (フィックスパック 2)** の「*WebSphere Commerce インストール・ガイド*」にある『Performance improvement by removing case sensitive search』の項を参照してください。

マスター・カタログの割り当て

WebSphere Commerce 5.5 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、`choosemc.sql` スクリプトは生成されず、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けてみます。

```
--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: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、以降のいくつか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・ウィンドウを開きます。
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. `WC55_installdir/instances/instance_name/migration` ディレクトリーに切り替えます。
4. Commerce Suite 5.1 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 アクセラレーター・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、10 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、以降のいくつか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、以下のようになります。

1. コマンド・プロンプト・ウィンドウを開きます。
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. *WC55_installdir/instances/instance_name* ディレクトリーに切り替えます。

4. Commerce Suite 5.1 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、160 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保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・ウィンドウを開きます。

— 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 5.5 がインストールされている *bin* サブディレクトリーに切り替えます。たとえば、*WC55_installdir/bin* などです。

— 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.sh db2 db_name db_userID db_userID_password  
instance_name db2_inst_home
```

ここで

- *db2* は使用するデータベース管理システムを表し、DB2 データベースの場合は *db2* にする必要があります。

- *db_name* は、 WebSphere Commerce 5.5 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)。
- *instance_name* は、ご使用の Commerce Suite 5.1 インスタンスの名前です (たとえば、demo)。
- *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。

たとえば、マイグレーションされた Commerce Suite 5.1 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.sh db2 mall mydbuser mypasswd demo
/home/db2inst1/sqllib
```

- 5. *WC55_installdir/logs/migration/instance_name* ディレクトリーに生成された *dbchecker_dbname.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。

- *WC55_installdir/instances/instance_name/migration* ディレクトリー:
 - delwcs.constraint.sql
 - wcs.schema.create.sql
 - wcs.temp.drop.sql
 - fillorgDN.sql
 - ctos.sql
 - choosemc.sql (ある場合)
- *WC55_installdir/logs/migration/instance_name* ディレクトリー:
 - 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、 61 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: 77 ページの『第 7 章 インスタンスおよびデータベースのマイグレーション後』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鍵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鍵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 5.5 データベースをバックアップします。
3. WebSphere Commerce 5.5 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 5.5 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 5.5 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 5.5 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 7 章 インスタンスおよびデータベースのマイグレーション後

以下のセクションでは、データおよび WebSphere Commerce インスタンスを WebSphere Commerce 5.5 レベルにマイグレーションした後で Commerce Suite 5.1 インスタンスをマイグレーションするのに必要なマイグレーション・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。

- 『セキュリティー構成のマイグレーション』
- 81 ページの『ストア資産のマイグレーション』
- 86 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』

WebSphere Application Server 始動時の考慮事項

Commerce Suite 5.1 インスタンスまたはこれまでの Payments インスタンスをマイグレーションした後、WebSphere Application Server 5.0 を WAS ユーザーとして実行する場合は、`WC55_installdir/bin` ディレクトリに移動して、以下のコマンドを実行します。

```
./wc55nonroot.sh
```

このコマンドを使用するのは、WAS ユーザーが WebSphere Application Server を始動するのを許可するためです。WebSphere Application Server を root として始動する場合は、上記コマンドを実行する必要はありません。

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 5.5 レベルにマイグレーションした後、マーチャント鍵マイグレーション・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鍵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 5.5 では、デフォルト以外のマーチャント鍵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鍵を使用している場合は、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 5.5 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・データを暗号化するか、暗号化されているすべてのクレジット・カード・データを暗号化解除します。

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

Java 仮想マシンのヒープ・サイズに関する考慮事項

MigrateEncryptedInfo スクリプトは、Java 仮想マシンの最大ヒープ・サイズ (JVM がヒープとして使用できるメモリの最大サイズで MB 単位) を 512 MB に設定します。スクリプトが JVM ヒープ・メモリの範囲を超えることが分かっている場合は、スクリプトを編集して 2 箇所ある `-Xmx512m` の設定を大きくします。たとえば、最大ヒープ・サイズを 1024 MB に増やす場合は、この設定を両方とも `-Xmx1024m` に変更します。この最大許容値は、データベース・サーバー・マシンに搭載されている物理メモリー量と、他のアプリケーションが使用するメモリー量で決まります。JVM 最大ヒープ・サイズの無理のない設定の目安は、通常は物理メモリーの 3 分の 1 です。このパラメーターは、64m (デフォルト値) より小さい値に設定することはできません。

暗号化設定の確認

マーチャント鍵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 `instance_name.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。

`instance_name.xml` ファイルは、`WC55_installdir/instances/instance_name/xml` ディレクトリにあります。`instance_name.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、マーチャント鍵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、`WC55_installdir/instances/your_instance` ディレクトリ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.sh ユーティリティーを以下のように実行します。

1. `WC_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・ウィンドウを開きます。
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. コマンドを実行しているユーザーに、*instance_name.xml* file (たとえば、*demo.xml*) への書き込み権限があることを確認してください。
- __ 5. WebSphere Commerce 5.5 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、以下のようにします。

```
WC55_installdir/bin
```

- __ 6. DB2 の場合、*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のオプション・パラメーターです。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そのままにしておくことができます。
- *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鍵のオプション・パラメーターです。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 マーチャント鍵を使用している場合だけです。デフォルトの Commerce Suite 5.1 マーチャント鍵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鍵のオプション・パラメーターです。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、以降のセクションを参照してください。マーチャント鍵を変更しない場合は、*old_key* and *new_key* パラメーターを省略します。*instance.xml* ファイルに保管されているマーチャント鍵が使用されます。

- __ 7. 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。

- e. マイグレーションされた WebSphere Commerce アプリケーション・サーバーを始動する前に、root ユーザーまたは wasuser としてログインし、以下のスクリプトを実行します。

```
.instance_home/sql1lib/db2profile
```

ここで *instance_home* は、DB2 インスタンスのホーム・ディレクトリーです (たとえば /home/db2inst1)。このコマンドは、アプリケーション・サーバーを始動する前に実行する必要があります。

- __ 8. WebSphere Application Server コンソールで、*WC_instance_name* アプリケーション・サーバーを始動します。

このツールによって、*WC55_installdir/logs* ディレクトリーに以下の 3 つのログ・ファイルが生成されます。

- CCInfoMigration.log
- MKChangeUserAndCCInfoMigration.log
- MigrateEncryptedInfoError.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してください。既存データの中に指定した *current_key* で暗号化されていないものがある場合に限り、以下のようなエラーは無視できます。異なる鍵を使用してデータが暗号化されている場合は、*MigrateEncryptedInfo* スクリプトを複数回実行し、その都度異なる *current_key* を指定する必要があります。この場合は暗号化されたデータのサブセットのみが各パスでマイグレーションされるため、こうしたエラーがログにいくつかできますが、それらは無視できます。しかし、こうしたエラーの存在が別の問題を示していることもあり、それらは無視することができません。無意識に誤った鍵を指定するとこのようなエラー・メッセージになり、顧客はログオンできなくなります。

```
%3DES-F-CIPHERINIT; Exception caught while initializing the cipher object.  
; java.lang.ArrayIndexOutOfBoundsException: 16
```

```
%3DES-F-DCRYPT; Exception caught while decrypting  
; javax.crypto.IllegalBlockSizeException: Input length (with padding)  
not multiple of 8 bytes
```

このユーティリティーを使用してマーチャント鍵を更新できるのは、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鍵を変更したい場合、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 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 マーチャント鍵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 マーチャント鍵を使用している場合、つまりすでにマーチャント鍵を固有のものに変更した場合に、鍵を WebSphere Commerce 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 マーチャント鍵を使用している場合、つまりすでにマーチャント鍵を固有のものに変更した場合に、鍵を WebSphere Commerce 5.5 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 にマイグレーションする方法について説明します。

重要

- `index.jsp` は、WebSphere Commerce 5.5 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 管理コンソールの「ストアの立ち上げ (launch store)」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。管理コンソールの「ストアの立ち上げ (launch store)」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 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 Production and Development* 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。

Commerce Suite 5.1 JSP ファイルの更新

Commerce Suite 5.1 JavaServer Pages (JSP ファイル) が WebSphere Commerce で機能するようにするには、JSP ファイルにいくつかの変更を加える必要があります。WCIM ツールは、いくつかのアイテムについて JSP ファイルを自動的に更新します。141 ページの『WCIM を使用したインスタンス・マイグレーション』を参照してください。

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 5.5 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 5.5 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、WebFashion 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 5.5 `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
DataBeanManager.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 5.5 ストアの作成方法については、「*WebSphere Commerce* ストア開発ガイド」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 管理コンソール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("-"))
```

これを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` からの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。

管理コンソール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` ファイルは、ストアへのログインが失敗した場合には、新しいエラー・コードを処理しませんし、エラーも戻しません。

顧客がログオンに失敗したときにストアがエラー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 ファイルおよびプロパティ・ファイルのリストについては、以下を参照してください。

- 168 ページの『`register.jsp`』
- 173 ページの『`account.jsp`』
- 177 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 用のこれらの JSP ファイルに必要な変更は、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` に以下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 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 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 アクセラレーターを使用して、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 アクセラレーターにログインし、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。

Commerce Suite 5.1 JSP ファイルの更新

以下のセクションでは、Infashion などの Commerce Suite 5.1 サンプル・ストアのさまざまな JSP ファイルに行く必要がある更新について説明します。発行済み Commerce Suite 5.1 ストアで JSP ファイルを再利用していた場合は、説明に従って更新を行い、更新した JSP をマイグレーション済みストアに再デプロイする必要があります。

JSP ファイルの <jsp:root> セクションの除去

JSP ファイルの <jsp:root> および </jsp:root> のセクションが有効なのは、使用する JSP ファイルが XML 文書である場合だけです。JSP ファイルが XML 文書でない場合 (つまり JavaServer Pages 形式の場合) は、その JSP ファイルから <jsp:root>...</jsp:root> セクションを除去する必要があります。

XML 文書の詳細については、Sun Microsystems の「*JavaServer Pages Specification*」(Version 1.2) の『JSP.5.2 JSP Documents』のセクションを参照してください。

java.util.* パッケージのインポート

WebSphere Application Server 5.0 では、Vector デイレクティブを使用する JSP ファイルに、java.util.Vector パッケージを明示的に組み込む必要があります。

JSP に以下の行がある場合は、WebSphere Commerce 5.5 で機能させるためにその JSP を変更する必要はありません。

```
<%@ page import="java.util.*" %>
```

java.util パッケージをインポートせずにそのパッケージ内のクラスを使用する場合は、JSP ファイルに対して以下のような変更を行う必要があります。WebSphere Commerce で使用される共通クラスは、以下のとおりです。

Enumeration

この特定クラスをインポートするには、以下の行を使用します。

```
<%@ page import="java.util.Enumeration" %>
```

Vector

この特定クラスをインポートするには、以下の行を使用します。

```
<%@ page import="java.util.Vector" %>
```

ResourceBundle

この特定クラスをインポートするには、以下の行を使用します。

```
<%@ page import="java.util.ResourceBundle" %>
```

WebSphere Commerce 5.5 に付属するクラスの詳細については、「WebSphere Commerce Production and Development オンライン・ヘルプ」を参照してください。

JSP のその他の変更

他の JSP 変更には以下が含まれます。

- JSP 1.2 仕様では、サポートされている言語は "java" だけであると宣言されています。したがって、JSP での以下のページ言語宣言はもはや無効です。

```
<%@ page language="JAVA" %>
```

なお、WCIM ツールは、すべての `<%@ page language="JAVA" %>` を、ユーザーに代わって `<%@ page language="java" %>` に変換します。

- `AbstractAccessBean.getInitContext()` を使って JSP 内の初期コンテキストを検索している場合は、それを `AbstractAccessBean.getInitContext(null,null)` に変更することをお勧めします。

ツール XML ファイルのマイグレーション

WebSphere Commerce 5.5 の XML パーサーは、前のバージョンのパーサーよりも XML 1.0 仕様を厳密に適用します。そのため、XML ファイルに以下のエラーが含まれる場合は、そのファイルを更新する必要があります。

- `resources.xml` 内の `<XML>` エレメント

XML 1.0 仕様では、大文字小文字に関係なく、エレメント名を「XML」にすることはできません。たとえば、XML ファイル・マッピングに `<XML name="xxx" file="xxx">` の形式のエレメントを使用している場合は、それを `<resourceXML name="xxx" file="xxx">` に変更してください。

- コメントに 2 つまたはそれ以上のダッシュ (「--」) を含めることはできません。

XML 1.0 仕様では、コメントに「--」を使用できません。たとえば、以下のようになります。

```
<!-- comment ---> <!-- 誤
<!-- comment --> <!-- 正
<!-- comment ----- comment2 --> <!-- 誤
<!-- comment comment2 --> <!-- 正
```

- XML ヘッダー

XML ファイルの XML ヘッダーが正しくない場合 (たとえば `<?xml`) やこのヘッダーが欠落している場合は、以下のように修正します。

```
<?xml version="1.0" encoding="UTF-8"?>
```

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マイズ・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 5.5 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マイズ・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 5.5 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。

割引	-1
配送	-2
消費税	-3
配送税	-4

- b. WebSphere Commerce 5.5 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 5.5 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 5.5 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、`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 `getItems()` 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、以下のようにします。

```
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 5.5 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 5.5 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 (87 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,
USAGEFLÄGS, 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 からのものです。

キャッシュ・ポリシーのマイグレーション

WebSphere Commerce 5.5 は、WebSphere Application Server 5.0 の動的キャッシュ機能によるキャッシングをサポートしています。結果として、以前のキャッシュ・ポリシーをマイグレーションする必要があります。 *WC55_installdir/bin* ディレクトリから、*cacheMigration* スクリプトを以下のように実行します。

```
./cacheMigration.sh instance_name oldWCPath EARPath
```

各パラメーターの意味は以下のとおりです。

- *instance_name* は、マイグレーションされるインスタンス (たとえば *demo*) です。
- *oldWCPath* は、以前の Commerce Suite 5.1 インストール・パス (*WCS51_installdir*) です。
- *EARPath* は、WebSphere Commerce 5.5 の EAR パス (*WAS50_installdir/installedApps/cell_name/WC_instance_name.ear*) です。

たとえば、以下のようにします。

```
./cacheMigration.sh demo WCS51_installdir  
WAS50_installdir/installedApps/myhost/WC_demo.ear
```

キャッシュ・ポリシーのマイグレーション・スクリプトは

cachePolicyMigration.log ファイルを *WC55_installdir/instance_name/logs* ディレクトリに生成します。

WebSphere Application Server 動的キャッシュの使用可能化

Commerce Suite 5.1 では、WebSphere Commerce のキャッシュ・ページは、キャッシュ・クリーンアップ・ワーカー・プロセスによって無効にされていました。WebSphere Commerce の動的ページ・キャッシュでは、キャッシュされたオブジェクトがいつ無効になったかを示すための通知メカニズムとして、データベース・トリガーが使用されていました。これらのデータベース・トリガーは、*CACHLOG* テーブルにデータを移植することによって、キャッシュから除去する必要があるキャッシュ・エントリを示していました。行が *CACHLOG* テーブルにコミットされると、自動ページ無効化 (Automatic Page Invalidation) が使用可能になっていれば、キャッシュ・クリーンアップ・ワーカー・プロセスは、示されたページ (複数の場合あり) を次の反復時に除去していました。

WebSphere Commerce 5.5 は、新しい *CACHEIVL* テーブルを無効化に使用します。*CACHEIVL* テーブルを更新するために以前のキャッシング・メカニズムを継続して使用する必要がある場合、つまりデータベース・トリガーを継続して使用する場合は、*WC55_installdir/samples/dynacache/triggers/database_type* サブディレクトリで提供されている *cacheTriggers.sql* サンプル・ファイルを参照してください。

WebSphere Commerce 5.5 は、WebSphere Application Server 5.0 の動的キャッシュ機能をキャッシング・メカニズムとして使用します。WebSphere Application Server の動的キャッシングでは、キャッシュ・エントリは以下の状況で除去されます。

- キャッシュ・エントリがタイムアウトになったとき。
- キャッシュが満杯になっているため、指定の優先順位に基づいて、古いエントリを新規エントリに置き換えなければならないとき。

- いずれかのキャッシュ無効化メソッドが呼び出されたとき。
 - コマンド・ベースおよびサーブレット・パス情報ベースの無効化のとき。
 - キャッシュ・モニターを使用するとき。
 - CACHEIVL ベースの無効化のとき。

キャッシュ無効化の詳細については、「*WebSphere Commerce 管理ガイド*」のキャッシングについての章を参照してください。

WebSphere Application Server の動的キャッシュの詳細については、WebSphere Application Server 5.0 InfoCenter の『Improving performance through the dynamic cache』のトピックを参照してください。

注: WebSphere Commerce 5.5 キャッシング・メカニズムは、WebSphere Application Server の動的キャッシング機能を使用するので、CacheCommand はもはやサポートされていません。したがって、CacheCommand を使用する以前の JSP ファイルはすべて、WebSphere Commerce 5.5 では CacheCommand の機能を実行するために更新する必要があります。JSP ファイルの作成方法に関する詳細情報は、「*WebSphere Commerce ストア開発ガイド*」を参照してください。

動的キャッシュ・サービスおよびサーブレット・キャッシングの使用可能化

キャッシングを使用可能にするには、動的キャッシュ・サービスを使用可能にして、サーブレット・キャッシングを構成する必要があります。これらのステップの実行については、WebSphere Application Server Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>) から、『Enabling globally the dynamic cache service』および『Configuring servlet caching』のトピックを参照してください。

Web サーバー・プラグインに関する考慮事項

`WAS50_installdir/properties` ディレクトリーにある `bootstrap.properties` ファイルに以下の行がある場合は、除去してください。

```
cache.lib=WCS51_installdir/bin/lib51cache.so
```

WebSphere Commerce 5.5 は WebSphere Application Server の動的キャッシング機能を使用するため、上記の行はもはや不要です。

3 層インストールの場合

システムが 3 層ネットワーク・トポロジーで構成されている場合 (つまり、リモート・データベースと Web サーバーがある場合) は、以下のファイルおよびディレクトリーを除去してください。

- `WCS51_installdir/bin/lib51cache.so`
- `WCS51_installdir/instances/instance_name/xml`
- `WCS51_installdir/instances/wcs_instances`

Web サーバーおよび WebSphere Application Server の再始動

インスタンスおよびデータベー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. Web サーバーを開始済みの場合は停止します。
2. WebSphere Application Server を開始済みの場合は停止します。
3. Web サーバーを始動します。
4. WebSphere Application Server を始動します。
5. DB2 データベースの場合、マイグレーションされた WebSphere Commerce アプリケーション・サーバーを始動する前に、DB2 インスタンス・ユーザー ID (たとえば、db2inst1) でログインし、以下のスクリプトを実行します。

```
. db2_instance_home/sql1lib/db2profile
```

ここで *instance_home* は、DB2 インスタンスのホーム・ディレクトリーです (たとえば /home/db2inst1)。このコマンドは、アプリケーション・サーバーを始動する前に実行する必要があります。

6. WebSphere Application Server コンソールで、マイグレーションされた *WC_instance_name* アプリケーション・サーバーを再始動します。

ホスト名の変更 (リモート・マイグレーションのみ)

リモート・マイグレーションを行う場合は、マイグレーション済みの WebSphere Commerce 5.5 サイトを起動する前に、マイグレーション済みシステムのホスト名を、マイグレーション前の Commerce Suite 5.1 システムのときと同じ名前になるように変更する必要があります。名前変更した WebSphere Commerce 5.5 ホスト・マシンを起動するときは、競合が起こらないように、あらかじめ Commerce Suite 5.1 システムを停止するかネットワークから分離しておく必要があります。

以下の場合を考えてみます。

- マイグレーション済みの WebSphere Commerce 5.5 システムのホスト名は myhost2 である。
- このホスト名を元の Commerce Suite 5.1 のホスト名 myhost1 に変更して、WebSphere Commerce もそのホスト名を使用するようにする。

これを行うステップの概要は、以下のとおりです。必要なツールの使用については、ご使用のオペレーティング・システムおよびネットワークの資料を参照してください。

1. ネットワーク用ツールを使って、ネットワーク構成に myhost1 ホスト名と IP アドレスを追加します。
2. WebSphere Commerce 5.5 マシンで、Web サーバー構成ファイル httpd.conf を手動で更新して、myhost2 の代わりに myhost1 を使用するようにします。
3. WebSphere Application Server の管理コンソールで、以下のようになります。
 - a. 仮想ホストを変更して myhost1 を使用するようにします。
 - b. エンドポイントと HTTP トランスポートを変更して myhost1 を使用するようにします。

4. WebSphere Commerce インスタンス構成ファイル *instance_name.xml* を手動で更新して、*myhost2* の代わりに *myhost1* を使用するようになります。
5. ドメイン・ネーム・サーバーを更新して、*myhost2* の代わりに *myhost1* を参照するようになります。

インスタンスおよびデータベースが正常にマイグレーションされたことの検証

31 ページの『第 4 章 Commerce インスタンス構成のマイグレーション』での説明どおりに WCIM を使用してインスタンスをマイグレーションし、この章にあるデータベース・マイグレーションを完了したら、インスタンスのマイグレーション状況を検証してください。データベースが正常にマイグレーションされたら、マイグレーションされた WebSphere Commerce 5.5 インスタンスを始動できるはずです。マイグレーションされたインスタンスは、マイグレーションされたデータベースの新規の WebSphere Commerce 5.5 データベース・スキーマを使用します。

前のセクションで説明したとおり、WebSphere Application Server 5.0 を再始動して、ネットワーク状況が使用可能になっていることを確認します。インスタンスおよびデータベースが適切にマイグレーションされたことを検証するには、まず前述のとおりすべてのログを検証してから、Web ブラウザーで以下の位置を指定して、WebSphere Commerce アクセラレーターを起動します。

`https://your_hostname:8000/accelerator`

WebSphere Commerce アクセラレーター・ログオン・ページが表示され、そのページにログオンできたら、前のインスタンスおよびデータベースは WebSphere Commerce 5.5 に正常にマイグレーションされたこととなります。

第 8 章 Commerce Payments へのマイグレーション

この章では、以下に示す旧バージョンの Payment Manager を、 WebSphere Commerce 5.5 に付属する現行バージョンの Commerce Payments へマイグレーションするための手順とシナリオを説明します。

- Payment Manager 2.2.x

このマイグレーション・プロセスでは、31 ページの『第 4 章 Commerce インスタンス構成のマイグレーション』で説明した WebSphere Commerce 5.5 WCIM マイグレーション・ツールと、この章で説明するデータ・マイグレーション・スクリプトを使用します。



WebSphere Payment Manager の前のバージョン 2.2.1 がインストール済みで作動可能である場合、このセクションで説明する Payments マイグレーション・コマンドのみを実行する必要があります。

Payments のマイグレーションの概要

WebSphere Commerce の Payments コンポーネントのマイグレーションは、WebSphere Commerce 5.5 マイグレーション・ソリューション全体の一部です。このソリューションで Commerce をマイグレーションする手順は、すべてのプラットフォームで共通です。Payments コンポーネントでは、マイグレーション・ソリューションの一部として、WCIM (WebSphere Commerce Instance Migrator) ツールを使用します。このツールは WebSphere Commerce 内の各コンポーネントのマイグレーションに必要なインフラストラクチャーを提供し、インスタンスを WebSphere Commerce 5.5 レベルにマイグレーションします。Payments の場合、このツールを使って、以前の Payment Manager インスタンスを新規の Commerce Payments インスタンスにマイグレーションすることができます。マイグレーションで使用する別のツールは、データベース・マイグレーション・ツールです。これは、以前の Payments データベースから新規の WebSphere Commerce Payments データベースにデータを移動する手段とインフラストラクチャーを提供します。

マイグレーションするコンポーネント

マイグレーションする Payments のコンポーネントは、以下のとおりです。

- Payments フレームワーク
- IBM カセット
- ベンダー・カセット

ベンダー・カセットのマイグレーションには、後方互換性とマイグレーションの 2 種類があります。後方互換性により、2.2.x のカセットを WebSphere Commerce 5.5 Payments で扱うことができます。マイグレーション・ツールは、決済カセットの必要な資産を新規 EAR ファイルの適切な場所に移動します。カセットのマイグレーションは、IBM カセットのマイグレーションの場合と同じ原理で行います。ベンダーは新規バージョンのカセットを提供し、そのカセット用に新しいパッケージ化およびデプロイメント機能を使用します。デプロイメントのパッケージ化機能の一

部として、内部マイグレーションをインプリメントしてカセットを新規バージョンに更新するためのマイグレーション・インターフェースが、カセット作成者に提供されています。

遷移の方法論

かぎとなる原則は、WebSphere Commerce のすべての部分の遷移 (旧インスタンスと旧データを新規インスタンスと新規データへマイグレーションすること) において、1 つの方法論を使用することです。この原則に従って、Payments は WebSphere Commerce 5.5 が提供するツールを使用し、以前のリリースのように独自のマイグレーション・ツールやエンジンは提供されなくなりました。こうすることにより、以下の利点が得られます。

- ツールとフレームワークの共通セットの使用。
- Payments と Commerce の統合の促進。
- Payments コンポーネントは、フレームワーク全体を支援する追加のユーティリティーやヘルパーの他に、Payments 固有のマイグレーションを実行するためのツールを提供する。

さらに、古い Payments ファイルおよびプロパティを WebSphere Commerce 5.5 にマイグレーションする際の方法論が使用されます。前のバージョンの Payments は、以下のようにサポートされています。

- EAR ファイルなし (Payment Manager 2.2.x)

サード・パーティー・カセットに関する考慮事項 (カスタム・コード)

サード・パーティー・カセットのデータのマイグレーションは、Payments インスタンスのデータベース全体のマイグレーションで行われます。Payments は、マイグレーションが不完全なカセット (Payments が認識していない必須ファイルが欠落しているカセット) を扱うことができます。カセットの欠落ファイルのためにそのカセットがランタイムに適切に機能しない場合、Payments はそのカセットをロード不能としてマークします。この場合は、該当するカセット・ベンダーに連絡して、そのカセットを稼働できるように支援してもらう必要があります。このマイグレーションの意図は、Payments コンポーネントがマイグレーションが必要であると認識しているカセット内のアイテムをマイグレーションすることです。

これには以下のものが含まれます。

- Java アーカイブ・ファイル
マイグレーションされる Java アーカイブは通常、カセット作成者用の推奨命名規則に従っているファイルです。Payments コンポーネントは、前のリリースでの eTillClasses.zip のような Java アーカイブ・ファイルを想定しています。
- PSPL ファイル

サード・パーティーのカセットの大半は IBM サービス資産を使って作成されているため、サード・パーティーのカセットのマイグレーションは LDBCard カセットのインストール後にモデル化されました。このインストールは、サード・パーティー・ベンダーが独自のディレクトリー構造とカセットのセットアップを作成するときに使用するよう文書化されたモデルです。

Payments はマイグレーションごとに、既存のマイグレーション済み Payments インスタンスを使ってできる範囲で、サード・パーティーのカセットを取得しようと試みます。ただしカセットには Payments コンポーネントが認識できないファイルが含まれていることがあるので、これは保証されません。

注:

1. この資料はマイグレーションを扱ったものですが、既存のインストール済みカセットをマイグレーションに使用できる場合もあることを理解しておいてください。ただし、バージョン 2.2.x のカセットを WebSphere Commerce 5.5 Payments に新しくインストールしても、機能しません。これは、それらのインストールが WebSphere Commerce 5.5 のインプリメンテーションに基づいていないためです。
2. 以前のバージョン 2.2.x のカセットは、マイグレーション済み Payments インスタンスで認識されます。しかし、Payments の新規インスタンスを作成した場合、Payments は以前の 2.2.x のカセットのテーブル・レイアウトについての情報を持っていないので、それらのカセットは認識されません。

マイグレーション前の考慮事項

WebSphere Commerce Payments にマイグレーションする前に、以下のことを確認してください。

- すべての前提条件ソフトウェアのマイグレーションと更新を完了している。
- WebSphere Application Server 5.0 が、WebSphere Application Server 3.5.x とは別個のマシンにインストール済みである。
- 以前の Payment Manager および Commerce データベースにアクセスできる。

Payments インスタンスおよびデータベースのマイグレーション

Payments のマイグレーション・プロセスは 3 ステップのプロセスになり、以下のものを使用します。

- WCIM ツール (『WCIM を使用した Payments インスタンスのマイグレーション』で説明)。
- 必要な場合、仮想ホストの除去 (100 ページの『仮想ホスト・ポートの更新』で説明)。
- Payments データ・マイグレーション・スクリプト (100 ページの『Payments データベースのマイグレーション』で説明)。

このプロセスは、WebSphere Commerce の他のコンポーネントの一般的なマイグレーションの方法論を使用します。

WCIM を使用した Payments インスタンスのマイグレーション

35 ページの『WCIM を使用したインスタンス構成のマイグレーション』の説明に従って、WCIM を呼び出して Payments をマイグレーションする場合、まずバックアップのために WCIM を呼び出し、次いでマイグレーションのために WCIM を再度呼び出します。基本的なプロセスは 35 ページの『WCIM を使用したインスタンス構成のマイグレーション』での説明に似ていますが、Payments に必要ないくつかの付加的な変数があります。主なステップは以下のとおりです。

バックアップでの WCIM の実行

WCIM を実行してマイグレーションする前に、以下のように WCIM を実行して、現行のインスタンスおよびインスタンス関連ファイルのバックアップを作成してください。

1. `wcimenv.sh` をカスタマイズして、以前の Payments インスタンスのバージョン、インストール・パス、作業ディレクトリー名などの必要なマイグレーション情報を提供します。

以下に示すのは、Payments 用の WCIM スクリプトに加える必要がある変更内容です。

- `wcimenv.sh` スクリプトに、以下の行を追加します。

```
WPM_PATH="Payments_installdir"
export WPM_PATH
```

- このプラットフォーム用の前のバージョンの Payments ではインスタンスがサポートされていなかったため、`SET INSTANCE` 行のインスタンス名を「default」に設定する必要があります。
- 行 `WCIM_MIGRATE_FROM="PM_version"` を設定します。ここで `PM_version` は、以前の Payment Manager のバージョンに応じて以下のいずれかにします。
 - Payment Manager 2.2.1 — WPM221

注: `WCIM_MIGRATE_FROM` を `PM221` に指定する場合、`WEBSERVER_PATH` には `HTTPServer1312_installdir/htdocs` を指定する必要があります。

たとえば、以下のようにします。

...

```
WCIM_BACKUP="true"
WCIM_INPLACE="false"
WCIM_MIGRATE_FROM="WPM221"
```

```
DB_TYPE="db2"
```

```
export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE
```

```
WAS_PATH="WAS35_installdir"
WC_PATH="WCS51_installdir"
WEBSERVER_PATH="HTTPServer1312_installdir/htdocs/locale"
ANT_PATH="WAS35_installdir/lib"
WORK_DIR="WAS35_installdir/temp"
LOG_FILE="wcim_pm_backup.log"
INSTANCE="default"
PATH=WAS50_installdir/java/jre/bin:$PATH
WPM_PATH="WPM221_path"
```

```
export WAS_PATH
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
```

```
export PATH
export WPM_PATH
```

...

2. リモート Payments マイグレーションのシナリオの場合のみ、以下の変数をリモート Commerce Suite 5.1 マシン上の実際のパスに変更する必要があります。たとえば、以下のようになります。

```
WORK_DIR="WC551_installdir/temp"
ANT_PATH="WORK_DIR/lib"
PATH=WAS35_installdir/java/jre/bin:$PATH
```

3. 以下のように WCIM を実行して Payments インスタンスをバックアップします。

```
./wcim.sh [wcimenv_path] PM_database_password
```

ここで

- `wcimenv_path` は、マシン上で `wcimenv.sh` が存在するパスです。 `wcimenv.sh` パスが `wcimenv.sh` パスと同じであれば、`.` を使用して、現行ディレクトリーを指定します。たとえば、以下のようになります。

```
./wcim.sh . mypassword
```

4. このスクリプトは、バージョン 2.2.x Payments インスタンスのファイル (カスタム・コードを含む) を、Commerce Suite 5.1 (Payments コンポーネント) マシン上の `/backupInst` ディレクトリーにバックアップします。

WCIM はこれらのファイルを、`wcimenv.sh` で提供された作業ディレクトリーの `.../zip` サブディレクトリーに、`.zip` ファイルとしてパッケージ化します。

5. コピーします。

リモート・バックアップの場合、`instbackupwpm221.xml` ファイルも、`WC55_installdir/xml/migration` ディレクトリーから、一時作業ディレクトリーの `.../xml/migration` サブディレクトリーにコピーする必要があります。

以下のファイルは、一時作業ディレクトリーの `.../lib` サブディレクトリーにコピーする必要があります。

- `eTillConfig.jar` (`Payments_installdir/lib` から)
- `j2ee.jar` および `ant.jar` (`WAS50_installdir/lib` から)
- `xerces.jar`、`ConfigManager.jar`、および `Utilities.jar` (`WC55_installdir/lib` から)

以下のファイルは、一時作業ディレクトリーの `.../bin` サブディレクトリーにコピーしてください。

- `wcim.sh`
- `wcimenv.sh`

これらのファイルは `WC55_installdir/bin` ディレクトリーにあります。

マイグレーションでの WCIM の実行

Payments インスタンスのリモート・マイグレーションの場合、マイグレーションのために WCIM を実行する前に、必ず以下の事柄を実行してください。

- 新規の Payments データベースが存在することを確認します。このデータベースは、空でもかまいません。データは後でマイグレーションできます。

- Payment Manager 2.2.x マシンの IP アドレスまたはホスト名が WebSphere Commerce 5.5 マシンと異なる場合には、Payment Manager 2.2.x マシンの IP アドレス、ホスト名、ノード名、およびドメインのすべての出現箇所を、バックアップのための WCIM の実行時に作成された .zip ファイル内の、WebSphere Commerce 5.5 マシンに関する情報で置き換える必要があります。

たとえば、更新する必要がある可能性のあるファイルには、.properties ファイルがあります。

WCIM を実行して Payments インスタンスをバックアップした後に、以下のように WCIM を実行して、Payments インスタンスおよびインスタンス関連ファイルをマイグレーションします。

1. wcmenv.sh をカスタマイズして、必要なマイグレーション情報を提供します。たとえば、以下のようにします。

...

```
WCIM_BACKUP="false"
WCIM_INPLACE="true"
WCIM_MIGRATE_FROM="WPM221"
```

```
DB_TYPE="db2"
```

```
export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE
WAS_PATH="WAS50_installdir"
WC_PATH="WC55_installdir"
WEBSERVER_PATH="HTTPServer1326_installdir/htdocs/locale"
ANT_PATH="WAS50_installdir/lib"
WORK_DIR="WC55_installdir/temp"
LOG_FILE="wcm_pm_migration.log"
INSTANCE="default"
PATH=$WAS_PATH/java/jre/bin:$PATH
WPM_PATH="WC55_installdir/payments"
```

```
export WAS_PATH
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH
export WPM_PATH
```

...

2. wcm.sh スクリプトを実行する前に、必ず Payments データベース・ユーザーとしてログインします。さらに、Payments データベース・ユーザー ID (たとえば db2inst1) に、以下のディレクトリーに対する十分な権限があることを確認してください。

- *WC55_installdir*
- *WC55_installdir/payments*
- *your_work_dir*
- *WAS50_installdir/config*

Payment データベース・ユーザー ID を wasgroup ユーザー・グループに追加することによって、Payments データベース・ユーザー ID に必要な権限を供給することができます。たとえば、`gpasswd -a db2inst1 wasgroup` は、`db2inst1` ユーザー ID を wasgroup ユーザー・グループに追加します。

Payments のマイグレーションを完了した後、以下のディレクトリーの権限にリストアする必要があります。

```
chmod 775 -Rf WC55_installdir/payments
chmod 775 -Rf WAS50_installdir/config
```

3. Payments インスタンスをマイグレーションするために WCIM を実行するときは、`wcim.sh` スクリプトを以下のように実行します。

```
./wcim.sh wcimenv_path PM_database_password
```

Payments マイグレーションの場合、以下は必須の パラメーターです。

- `wcimenv_path` は、マシン上で `wcimenv.sh` が存在するパスです。 `wcimenv.sh` パスが `wcimenv.sh` パスと同じであれば、`."` を使用して、現行ディレクトリーを指定します。たとえば、以下のようにします。

```
./wcim.sh . mypassword
```

- `PM_database_password` は、 WebSphere Commerce Payments データベース・パスワードで、 WebSphere Commerce Payments インスタンスのマイグレーション時に使用されます。

4. WCIM を実行してマイグレーションする場合は、以下のようにします。

- WebSphere Commerce 5.5 インスタンス基本ファイルを、作業ディレクトリーの `.../migrate` サブディレクトリーにコピーします。(これが Payments = cassettes 関連ファイル用に WebSphere Commerce 5.5 より前の EAR 情報を追加するための新規の EAR テンプレートになります。)
- コピーした jar ファイルを、作業ディレクトリーの `.../migrate` サブディレクトリーにアンパックします。
- インスタンス・マイグレーションを以下のように実行します (これは Payments インスタンス固有のもので)。
 - WebSphere Commerce 5.5 デプロイメント記述子を更新します。つまり、マイグレーション済みインスタンス内から、不明の `.jar` ファイルを、 Payments Manifest.MF ファイルに追加します。
 - 既存の Payments インスタンス (`instance.xml`) からの構成情報の収集を開始します。この情報は、後で Payments マイグレーションでマイグレーション済みインスタンスを作成するときに使用します。マイグレーション・プロセスで現在の Payments 情報が読み取られ、その情報を基に新規インスタンスと WebSphere Application Server 情報が作成されます。この時点ではテーブルは作成されず、EAR ファイルもデプロイされません。
 - `instance` ファイルをマイグレーションします。
 - Web サーバー構成ファイルをマイグレーションします。
 - WCIM の実行後に必要となる残りのマイグレーション・ステップがあれば、それを表示します。

仮想ホスト・ポートの更新

WCIM で Payments インスタンスのマイグレーションを実行した後、必要に応じて WebSphere Application Server 管理コンソール内の仮想ホスト・ポートを変更します。たとえば、仮想ホスト「VH_PYM_migrated_payments_instance」と「VH_commerce_instance」が同じポート（たとえばポート 80）を使用している場合、ポート競合が起こります。

この競合をなくすためには、「VH_PYM_migrated_payments_instance」からポート 80 を除去し (Payments はもはやポート 80 を使用しないため)、「migrated_payments_instance_Commerce_Payments_App」の Web モジュールを「VH_commerce_instance」へマップする必要があります。

Commerce Commerce Suite 5.1 では、ポート 80 と 443 はデフォルト・ホストの下にのみ定義されており、Commerce および Payment Manager の両方がそのデフォルト・ホストを使用してこれらのポートにアクセスします。

WebSphere Commerce 5.5 では、ポート 80 と 443 は VH_commerce_instance に属し、Payments はポート 5432 と 5433 を別の仮想ホストと共に使用します。マイグレーションされた WebSphere Commerce 5.5 環境では、マイグレーションされた Payments インスタンスがポート 80 を使用し続けますが、これは Payments インスタンスのマイグレーション・プロセス中に自動的に作成される新しい仮想ホストにマップされます。マイグレーションされた Payments インスタンス用に、継続してポート 80 を使用する場合、このインスタンスを VH_commerce_instance (たとえば、VH_demo) にマップする必要があります。そうでない場合、「VH_PYM_migrated_payments_instance」のポート番号を変更しなければなりません。

Payments のマイグレーション後には、ポートの競合を起こす可能性のある仮想ホストを手動でチェックし、更新する必要があります。

Payments データベースのマイグレーション

Payments データ・マイグレーション・スクリプト (migratepaymentsdb) を以前の Payments データベースに対して実行する前に、23 ページの『データベースのバックアップ』の説明に従って、そのデータベースをバックアップすることをお勧めします。

Payments データベース・マイグレーション・スクリプト (migratepaymentsdb) を、前の Payments データベースに対して実行する場合 (下記のように)、スクリプトは、

1. データベース情報を収集して、バージョン、リリース・レベル、データベース・タイプを判別します。
2. バージョン、リリース・レベル、およびタイプに基づいて、マイグレーション前のステップを実行します。
3. Payments スキーマを WebSphere Commerce 5.5 レベルに更新します。
4. バージョンとリリースに基づいて、Payments データ・マイグレーション・コマンドを実行します。これは Payments フレームワークと IBM カセットについて行われるほか、マイグレーション・スクリプトの実行前にインストールされていたサード・パーティーのカセットについても行われます。

DB2 データベース

以下のステップは、DB2 データベースのマイグレーション・スクリプトを実行する方法を示しています。

1. コマンド・プロンプト・ウィンドウを開きます。
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 5.5 がインストールされている bin サブディレクトリに切り替えます。たとえば、WC55_installdir/bin などです。
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。

```
./migratepaymentsdb.sh db2 db_name db_userID password schema_owner
```

各パラメーターの意味は以下のとおりです。

- db2 は、DB2 データベースをマイグレーションすることを示します。
- db_name は、マイグレーションする Payment Manager データベースです。
- db_userID は、マイグレーションする Payment Manager データベースに接続するためのユーザー ID です。
- password は、マイグレーションする Payment Manager データベースに接続するためのユーザー ID です。
- schema_owner は、DB2 データベースのデータベース・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。

Payments データベース・マイグレーションの完了後、Payments サーバーを正常に開始するためには、以下のようにして、データベース構成を更新することにより、データベース・ログ・ファイルのサイズを増やす必要があります。

```
db2 update db cfg for your_payments_db using logfilsiz 10000
```

更新を行った後、DB2 を再始動してください。

JDBC プロバイダーの検査

Payments サーバーを開始する前に、WebSphere Application Server 管理コンソールで JDBC プロバイダーをチェックして、データ・ソースが現行の環境に合わせて正しく構成されていることを検査します。データ・ソースがヌルの場合、以下のようにして、開始する前にそれを手動で追加する必要があります。

1. WebSphere Application Server 管理コンソールに再びログオンします。管理コンソールから、「リソース (Resources)」を拡張表示して、「JDBC プロバイダー (JDBC Providers)」をクリックします。WebSphere Commerce Payments データ・ソースが表示されない場合には、以下のステップを実行します。

- 「JDBC プロバイダー (JDBC Providers)」パネルから、「新規 (New)」をクリックし、ドロップダウン・リストから適切な JDBC プロバイダー (たとえば、DB2 データベースの DB2 JDBC プロバイダー) を選択します。
- 「JDBC プロバイダー名 (JDBC Provider name)」および「JDBC クラスパス (JDBC classpath)」フィールドを構成します。「適用 (Apply)」および「OK」をクリックして、変更を保管します。
- 「構成 (Configuration)」パネルの「追加プロパティ (Additional Properties)」セクションで、「データ・ソース バージョン 4 (Data Sources Version 4)」→「新規 (New)」をクリックします。
- 「名前 (Name)」、DB2 データベースの「JNDI 名 (JNDI Name)」 (たとえば、jdbc/migrated_default Commerce Payments Datasource)、「データベース名」、データベースの「デフォルト・ユーザー ID およびデフォルト・パスワード (Default user ID and Default Password)」フィールドを構成します。
- 「適用 (Apply)」および「OK」をクリックします。
- 変更を保管します。

Payments マイグレーションの使用シナリオ

以下のシナリオは、WebSphere Commerce 5.5 への遷移での Payments だけに関するフローを示したものです。

以前の Payment Manager のマイグレーション

説明 Payment Manager 2.2.x を、WebSphere Commerce Payments へマイグレーションします。

実行担当者

サイト管理者

前提条件

- WebSphere Application Server 5.0 をインストール済みである。
- 以前の WebSphere Application Server 構成を保管済みである。
- 以前の Payment Manager マシンと WebSphere Commerce 5.5 マシンの両方で WCIM が使用可能である。
- 必要なサービスをすべて開始済みである。
- 以前の Payment Manager および Commerce データベースにアクセスできる。

メイン・フロー

以前の Payment Manager マシン (リモート・システム) から、以下のようになります。

1. `wcimenv.sh` および `wcim.sh` を、WebSphere Commerce 5.5 マシンの `WC55_installdir/bin` ディレクトリーから、Commerce Suite 5.1 マシンの一時作業ディレクトリーにコピーします。
2. リモート・バックアップの場合、`instbackupwpm221.xml` ファイルも、`WC55_installdir/xml/migration` ディレクトリーから、一時作業ディレクトリーの `.../xml/migration` サブディレクトリーにコピーする必要があります。

以下のファイルも、一時作業ディレクトリーの `.../lib` サブディレクトリーにコピーする必要があります。

- `eTillConfig.jar` (`Payments_installdir/lib` から)
 - `xerces.jar`、`j2ee.jar`、および `ant.jar` (`WAS50_installdir/lib` から)
 - `ConfigManager.jar` および `Utilities.jar` (`WC55_installdir/lib` から)
3. `wcimenv.sh` を編集して、前の `Payments` インスタンスをバックアップするのに必要な情報を提供します。 `WCIM_BACKUP=true` および `WCIM_INPLACE=false` と指定します。バックアップを正常に行うために必要な詳細情報は、40 ページの『バックアップでの `WCIM` の実行』を参照してください。
 4. `wcim.sh` コマンドを実行して、インスタンスをバックアップします。
 5. `WebSphere Commerce 5.5` マシンの作業ディレクトリーにある `zip` サブディレクトリーに、`backup.zip` ファイルをコピーします。

`WebSphere Commerce 5.5` マシンから、以下のようにします。

1. `WebSphere Commerce 5.5` マシン上の `wcimenv.sh` を編集します。
`WCIM_BACKUP=false` および `WCIM_INPLACE=true` を指定します。マイグレーションを正常に行うために必要な詳細情報は、43 ページの『マイグレーションでの `WCIM` の実行』を参照してください。
2. `WCIM` を実行するには、必ず `Payments` データベース・ユーザーとしてログインします。
3. コマンド行から以下の `WCIM` スクリプトを実行して、`Payments` データベース・パスワードを渡します。

```
./wcim.sh wcimenv_path payments_db_password
```
4. `WCIM` でマイグレーション・ログ・ファイルが表示されたときは、ログを検査してマイグレーションを検証する必要があります。
5. `WCIM` で、`Payments` に必要な残りのマイグレーション・ステップが表示されます。
6. `Payments` データ・マイグレーション・スクリプトを呼び出して、以前の `Payments` データベースを現行レベルにマイグレーションします。スクリプトを呼び出す構文については、100 ページの『`Payments` データベースのマイグレーション』を参照してください。

終了後の状態

以前の `Payment Manager 2.2.x` が、`WebSphere Commerce Payments` に正常にマイグレーションされています。

Payments の 1 インスタンスを指す複数の WebSphere Commerce インスタンス

説明 複数の `WebSphere Commerce` インスタンスが 1 つの `Payments` インスタンスを指している環境で、`Payment Manager 2.2.1` を、`WebSphere Commerce Payments` へマイグレーションします。

実行担当者

サイト管理者

前提条件

- WebSphere Application Server 5.0 をインストール済みである。
- 以前の WebSphere Application Server 構成を保管済みである。
- WebSphere Commerce 5.5 インストールの一部として WebSphere Commerce Payments をインストール済みである。
- 前の Payment Manager マシンと WebSphere Commerce 5.5 マシンの両方で WCIM が使用可能である (注: これらは同一マシンにインストールされている場合があります)。
- 必要なサービスをすべて開始済みである。
- 以前の Payment Manager および Commerce データベースにアクセスできる。

メイン・フロー

1. Payments をマイグレーションします。(この方法はシナリオによって異なります。上記の該当するケースを参照してください。)
2. すべての WebSphere Commerce インスタンスが、新規にマイグレーションした Payments インスタンスを指すようにします。
3. マイグレーションする WebSphere Commerce インスタンスを選択します。

終了後の状態

以前の Payment Manager 2.2.x が、WebSphere Commerce Payments に正常にマイグレーションされています。

Payments サブシステムのマイグレーション考慮事項

WebSphere Commerce 5.5 には、Commerce Suite 5.1 から、以下の変更が行われています。

- すべての SET プロファイル (WCS51_SET_MIA および WCS51_SET_Wallet) は WebSphere Commerce 5.5 データベース・テーブルに含まれますが、ランタイムで使用不可になります。これらのプロファイルを使用可能にする場合は、IBM にご連絡ください。
- Cybercash がサポートされなくなるので、WebSphere Commerce 5.5 では使用不可になります。追加情報については、IBM にお問い合わせください。
- 新規のポリシーとして、以下の Paymentech が導入されました。

表 7. Paymentech プロファイル

ポリシー ID	ポリシー名	プロファイル名
-9980	Paymentech	WC_Paymentech

以前に提供されていた Payments カセットのプロファイルで WebSphere Commerce 5.5 でも引き続きサポートされるものは、すべて以下のように更新されていますので注意してください。

- 「Payment Manager」への参照は、すべて除去されました。

- プロファイルに含まれる DTD パスは、プラットフォーム間の整合性のために、*WC55_installdir/xml/PaymentManager/profile.dtd* 内ではなく、現行ディレクトリ内の DTD を参照するようになりました。

Commerce Suite 5.1 で提供されたプロファイルを変更している場合、マイグレーション済み Payments インスタンスでもその変更が必要なときは、更新済みの同じ名前の WebSphere Commerce 5.5 プロファイルにその変更を再び適用してください。

データ・マイグレーションの処理

データ・マイグレーション・スクリプトを以前のデータベースに対して実行すると、以下のステップが実行されます。

1. データベース・マイグレーションによって、*WC55_installdir/instances/default/xml/payment* ディレクトリーにあるすべての標準プロファイルと *profile.dtd* ファイルが、*WC55_installdir/instances/instance_name/xml/payment* ディレクトリーへコピーされます。 *.../default/xml/payment* ディレクトリーにある WebSphere Commerce 5.5 用のプロファイルの名前は、以下のとおりです。
 - WC_Paymentech (WebSphere Commerce 5.5 での新規)
 - WC51_BankServACH.profile
 - WC51_CustomOffline_BillMe.profile
 - WC51_CustomOffline_COD.profile
 - WC51_OfflineCard.profile
 - WC51_VisaNet.profile
 - WC51_VisaNet_PCard.profile
 - WCS51_CustomOffline.profile
 - WCS51_OfflineCard.profile
2. 元の Commerce Suite 5.1 *.../instances/instance_name/xml/payment* ディレクトリーにあるすべてのファイルが、WebSphere Commerce 5.5 のインストール・ツリーの同じディレクトリーにコピーされます。ただし、以下のファイルは除きます。
 - WC51_BankServACH.profile
 - WC51_CustomOffline_BillMe.profile
 - WC51_CustomOffline_COD.profile
 - WC51_CyberCash.profile
 - WC51_OfflineCard.profile
 - WC51_SET_MIA.profile
 - WC51_SET_MIA_PCard.profile
 - WC51_SET_Wallet.profile
 - WC51_VisaNet.profile
 - WC51_VisaNet_PCard.profile
 - WCS51_CustomOffline.profile
 - WCS51_CyberCash.profile
 - WCS51_OfflineCard.profile

- WCS51_SET_MIA.profile
 - WCS51_SET_Wallet.profile
3. 上記のステップ 2 でファイルがコピーされた場合、 Commerce Suite 5.1 .../xml/PaymentManager ディレクトリーの profile.dtd が、 WebSphere Commerce 5.5 のインストール・ツリーの同じディレクトリーにコピーされません。
- これは、 Commerce Suite 5.1 バージョンのプロファイルは、 profile.dtd が .../xml/PaymentManager に置かれていることを想定しているためです。更新済み WebSphere Commerce 5.5 のプロファイルは、 profile.dtd ファイルがプロファイルと同じディレクトリーに置かれていることを想定しています。
4. POLICY テーブル内の支払ビジネス・ポリシー (PolicyType_Id = 'Payment') のマイグレーション時に、 PROPERTIES 列にストリング「cassetteName=SET」または「cassetteName=CyberCash」が含まれる支払ポリシー・エントリーについては、 ENDTIME 列が CURRENT TIME に設定されます。(これは実質的には、このポリシーが非アクティブであることをマークしています。)
- ```
UPDATE POLICY SET ENDTIME = CURRENT TIME
WHERE PolicyType_Id = 'Payment'
AND (Properties LIKE '%cassetteName=SET%' OR
 Properties LIKE '%cassetteName=CyberCash%')
```
5. PAYMTHDSUP テーブルのマイグレーション時に、セット内に PayMthd\_Id が含まれるエントリーは除外されます。
- ```
SELECT PayMthd_Id FROM PAYMTHD
where PAYMTHD.ProfileName like '%SET%'
or PAYMTHD.ProfileName like '%CyberCash%')
```

このステップで除外された PAYMTHDSUP のエントリーは、マイグレーション・スクリプトによってログに記録されます。

WebSphere Commerce 5.5 は Cassette for SET と Cassette for CyberCash をサポートしなくなったため、 WebSphere Commerce 5.5 へのマイグレーションで以下のプロファイルはコピーされません。これらのカセットとプロファイルが引き続き必要な場合は、 IBM サービスに連絡してサポートを受けてください。

- WC51_CyberCash.profile
- WC51_SET_MIA.profile
- WC51_SET_MIA_PCard.profile
- WC51_SET_Wallet.profile
- WCS51_CyberCash.profile
- WCS51_SET_MIA.profile
- WCS51_SET_Wallet.profile

Payments サブシステムのマイグレーションに関する追加の考慮事項

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 ¹	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 ¹	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 ²	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 Commerce Payments 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 5.5 には、元々は WebSphere Commerce 5.4 で導入されたビジネス・ポリシー およびビジネス・ポリシー・コマンド という概念があります。

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 5.5 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 5.5 支払ポリシーは、WebSphere Commerce Payments 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 で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 5.5 へのマイグレーション時、WebSphere Commerce 5.5 の新しい機能や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 5.5		
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	-9991	CustomOffline_COD	WC51_CustomOffline_COD
		-9990	CustomOffline_BillMe	WC51_CustomOffline_BillMe
200	WCS51_OfflineCard	200	OfflineCard	WC51_OfflineCard
300	WCS51_SET_MIA ¹			
400	WCS51_SET_Wallet ¹			
500	WCS51_CyberCash ²	600	VisaNet	WC51_VisaNet
		601	VisaNet_PCard	WC51_VisaNet_PCard
		700	BankServACH	WC51_BankServACH
		-9980	Paymentech ³	WC_Paymentech ³

注:

- ¹ すべての SET プロファイルは WebSphere Commerce 5.5 データベース・テーブルに含まれますが、ランタイムで使用不可になります。これらのプロファイルを使用可能にする場合は、IBM にご連絡ください。
- ² Cybercash がサポートされなくなるので、WebSphere Commerce 5.5 では使用不可になります。追加情報については、IBM にお問い合わせください。
- ³ WebSphere Commerce 5.5 での新規。

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 5.5 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 5.5 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 5.5 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">
```

以下に置き換えます。

```
<input type=hidden name="policyId" value="200">
```

そうしない場合、PickPatches などの一部の機能が Commerce アクセラレーターで機能しないことがあります。

また、Commerce Suite 5.1 で ProfileCassetteAccountDataBean Data Bean を使用している場合は、WebSphere Commerce 5.5 では UsablePaymentTCListDataBean Data Bean を使用するように切り替える必要があります。ProfileCassetteAccountDataBean Data Bean は、Commerce Suite 5.1 テーブルの PAYMTHD、PAYMTHDDSC、および PAYMTHDSUP を使用します。UsablePaymentTCListDataBean Data Bean は、新規の WebSphere Commerce 5.5 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 5.5 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 5.5 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 Commerce Payments 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。

注:

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 5.5 Commerce アクセラレーターを使用します。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 Data 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 5.5 のオンライン・ヘルプを参照してください。WebSphere Commerce 5.5 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 5.5 ビジネス・ポリシー・コマンドへのマイグレーション

WebSphere Commerce 5.5 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 5.5 は、WebSphere Commerce 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 Commerce Payments に基づく支払ポリシーをサポートし、他のセットは、WebSphere Commerce Payments 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 Commerce Payments 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。

- DoPaymentPMCmdImpl
- CheckPaymentAcceptPMCmdImpl
- DoDepositPMCmdImpl
- DoRefundPMCmdImpl
- DoCancelPMCmdImpl

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 5.5 のオンライン・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 Commerce Payments を使用する独自のストアを作成した場合、これを WebSphere Commerce 5.5 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 Commerce Payments 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 5.5 では、DoCancelCmd は、クラス `com.ibm.commerce.payment.commands.DoCancelCmdImpl` に割り当てる必要があります。DoCancelCmdImpl インターフェースは、支払いに使用される支払ポリシーに応じて、呼び出しを DoCancelPMCmdImpl または DoCancelCLCmdImpl に経路指定します。

サンプル JavaServer ページ・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 ページ・ファイル `PayStatusPM.jsp` は、WebSphere Commerce Payments 3.1.3 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、WebSphere Commerce Payments 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、WebSphere Commerce Payments 3.1.3 では、一部のカセットは WebSphere Commerce Payments 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、WebSphere Commerce Payments 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 WebSphere Commerce Payments 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 5.5 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。(WebSphere Commerce Payments 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 5.5 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 2 部 追加のマイグレーション・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ます。たいていの場合、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。

- 115 ページの『第 9 章 メンバー・サブシステムのマイグレーション』
- 121 ページの『第 10 章 アクセス制御サブシステムの考慮事項』
- 123 ページの『第 11 章 他の WebSphere Commerce コンポーネントのマイグレーション』

第 9 章 メンバー・サブシステムのマイグレーション

この章では、メンバー・サブシステムを Commerce Suite 5.1 から WebSphere Commerce 5.5 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ます。このシナリオは、ユーザーが LDAP と WebSphere Commerce 5.5 データベースとの統合方法を決定する方法によって異なります。

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、23 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してください。

```
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 5.5 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 5.5 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 5.5 データベースへのマイグレーションを最初に行ってから、ディレクトリー・サーバーの使用に切り替える。

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、数列に続く 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 5.5 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 5.5 オンライン・ヘルプで見つけることができます。

表 8. 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。15 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	(*) を参照	(*)
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	(*) を参照	(*)
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、ORAGENTY 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	(*) を参照	(*)

表 8. 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	(*) を参照 (*)	
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	
ORGENTITY 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、それは「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、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。以下の説明は 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 5.5 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、それらのユーザーは Commerce Suite 5.1 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 5.1 がすでに認識しているユーザーです。なぜなら、そ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 5.1 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 5.5 はユーザーが 1 つだけ見つかることを期待しています。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。

- WebSphere Commerce 5.5 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 5.5 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 5.5 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 5.5 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 5.5 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 5.5 オンライン・ヘルプの『LDAP シナリオ: メンバー・リポジトリとしてのデータベース』を参照してください。

WebSphere Commerce 5.5 での Commerce Suite 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 5.5 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ことを確認します。
 - 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. 67 ページの『データベース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、70 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 5.5 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どうかを確認します。これらの接尾部は、WebSphere Commerce 5.5 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 になっていることを確認します。

第 10 章 アクセス制御サブシステムの考慮事項

WebSphere Commerce 5.5 のアクセス制御モデルは、アクセス制御ポリシーの制約に基づいています。アクセス制御ポリシーは、アクセス制御ポリシー・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、アクセス制御ポリシー・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 アクセス制御データベース・テーブルは、WebSphere Commerce 5.5 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、アクセス制御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制御テーブルに置き換えられています。詳細については、WebSphere Commerce 5.5 オンライン・ヘルプの『アクセス制御』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・アクセス制御ポリシー (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 5.5 にマイグレーションされます。
- Commerce Suite 5.1 内のデータである ACCCUSTEXC テーブルは手動でマイグレーションする必要があります。このテーブルを使用する場合は、122 ページの『ACCCUSTEXC テーブルのマイグレーション』にあるステップを参照してください。
- WebSphere Commerce 5.5 での、アクセス制御の 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制御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。

Commerce Suite 5.1 アクセス制御と WebSphere Commerce 5.5 アクセス制御の主な違いは、WebSphere Commerce 5.5 はポリシーに基づいたリソース・レベルのアクセス制御を使用するのに対し、Commerce Suite 5.1 はプログラムに基づいたリソース・レベルのアクセス制御を使用するということです。アクセス制御コードのマイグレーションについて詳しくは、「WebSphere Commerce Studio マイグレーション・ガイド」を参照してください。

ACCCUSTEXC テーブルのマイグレーション

Commerce Suite 5.1 では、指定された組織のストアの顧客がアクセスできないコントローラー・コマンドやビューを指定するために、ACCCUSTEXC テーブルが使用されました。しかし、サイト内のすべてのユーザーは、これらのコマンドやビューにアクセスが可能でした。デフォルトでは、すべてのユーザーにコマンドの実行を許可するポリシーは、`AllUsersExecuteAllSiteUserCmdResourceGroup` です。このポリシーには次のようなリソース・グループがあります：

`AllSiteUserCmdResourceGroup`。除外されるコマンドがこのリソース・グループの一部であると想定すると、マイグレーション後に同じ動作を維持するために、以下のステップを実行する必要があります。

1. `AllSiteUserCmdResourceGroup` と同様の新規のリソース・グループを作成します。ただし、この組織でアクセス可能とするべきではないコマンド (リソース) を除外します。
2. `AllUsersExecuteAllSiteUserCmdResourceGroup` と同様の新規のポリシーを作成します。ただし、ステップ 1 の新規リソース・グループを使用します。
3. `AllSiteUsersViews` と同様の新規のアクション・グループを作成します。ただし、この組織でアクセス可能とするべきではないビュー (アクション) を除外します。
4. `AllUsersExecuteAllSiteUsersViews` と同様の新規のポリシーを作成します。ただし、ステップ 3 の新規アクション・グループを使用します。
5. `ManagementAndAdministrationPolicyGroup` と同様の新規のポリシー・グループを作成します。ただし、ポリシー
`AllUsersExecuteAllSiteUserCmdResourceGroup` および
`AllUsersExecuteAllSiteUsersViews` は除外し、ステップ 2 と 4 の新規ポリシーを組み込みます。
6. この組織が、次の不要なポリシーを組み込んでいるポリシー・グループからアンサブスクライブするようにします：
`AllUsersExecuteAllSiteUserCmdResourceGroup` および
`AllUsersExecuteAllSiteUsersViews`。デフォルトでは、これらのポリシーは次のポリシー・グループに属しています：
`ManagementAndAdministrationPolicyGroup`。そのため、このポリシー・グループからアンサブスクライブする必要があると思われます。
7. この組織を、ステップ 5 の新規ポリシー・グループにサブスクライブします。

第 11 章 他の WebSphere Commerce コンポーネントのマイグレーション

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 5.5 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。

- 『ユーザー役割の構成』
- 124 ページの『ステージング・サーバーのマイグレーション』
- 125 ページの『データベース・クリーンアップ・ユーティリティーの再構成』
- 125 ページの『ルール・サーバー構成のマイグレーション』
- 127 ページの『オークション』
- 128 ページの『ビジネス・アカウントおよび契約』
- 128 ページの『配送計算コード』
- 128 ページの『edit_registration ページにおけるログオン ID の形式 (LDAP が使用される場合)』
- 129 ページの『商品アドバイザーのマイグレーション』
- 130 ページの『ATP 在庫の丸め』
- 130 ページの『デフォルトの通貨の動作』

ユーザー役割の構成

Commerce Suite 5.1 では、サイト・レベルの構成を使用していました。つまり、ユーザーがサイトのいずれかのストアに登録されれば、そのサイトの他のすべてのサイトにも暗黙的に登録されました。WebSphere Commerce 5.5 は、役割を使用して、ユーザーが特定のストアに登録されているかどうかを判別します。顧客はその役割割り当てポリシーをセットアップすることができ、これによって役割を、個々のストアか、または階層内の特定の組織の下のすべてのストアに割り当てます。

WebSphere Commerce 5.5 で同等のサイト・レベルの登録動作を確保するには、すべてのユーザーに Root Organization で Registered Customer 役割を割り当てるだけです。マイグレーション・プロセスでは、データベース内のすべての顧客にこの役割を明示的に割り当てます。したがって、既存の顧客がマイグレーション先のサイトのストアでショッピングをしようとする、そのストアへのアクセスは許可されます。新規顧客の場合、役割の割り当ては、MemberRegistrationAttributes.xml ファイルで定義された構成に基づいて動作します。このファイルは、*WC55_installdir/instances/instance_name/xml/member* にあります。

マイグレーションの場合、WebSphere Commerce 5.5 によって、このファイルの特別なバージョンが *WC55_installdir/migration/member* ディレクトリーに準備されます。このバージョンは、新規顧客用の Registered Customer 役割を、Root Organization のすべてのユーザー登録で割り当てます。希望すればこのファイルを変更して、役割割り当てポリシーを変更することができます。このファイルのセット

アップの詳細については、「WebSphere Commerce Production and Development オンライン・ヘルプ」の『MemberRegistrationAttributes XML and DTD files』のトピックを参照してください。

WebSphere Commerce 5.5 が正しく動作するように、MemberRegistrationAttributes.xml のバージョンを WC55_installdir/migration/member ディレクトリーから WC55_installdir/instances/instance_name/xml/member ディレクトリーにコピーする必要があります。

ステージング・サーバーのマイグレーション

ステージング・サーバーを WebSphere Commerce 5.5 にマイグレーションするために、事前に以下を完了させておく必要があります。

1. Commerce Suite 5.1 レベルであった場合は、6 ページの『マイグレーションのためのステージング・サーバーの準備』で説明されているとおりに、ステージ伝搬ユーティリティーまたはステージ・コピー・ユーティリティーのいずれかを実行して、ステージング・サーバーと実動サーバーを同期させる必要があります。
2. 65 ページの『第 6 章 Commerce Suite 5.1 データベースのマイグレーション』で説明されているとおりに、Commerce Suite 5.1 実動データベースを正常にマイグレーションしておく必要があります。
3. 65 ページの『第 6 章 Commerce Suite 5.1 データベースのマイグレーション』で説明されているステップに従って、Commerce Suite 5.1 ステージング・サーバー・データベースを正常にマイグレーションしておく必要があります。

ステージング・サーバーのマイグレーションを実行するには、以下のようになります。

1. ステージング・サーバーをマイグレーションする前に除去した、カスタマイズ済みトリガーを再適用します。
2. 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。

ステージング・サーバーを再構成する場合は、「WebSphere Commerce 管理ガイド」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。

3. 実動データベースからステージング・サーバー・データベースにデータをコピーして戻すには、ステージ・コピー・ユーティリティー (stagingcopy) を `-scope`

all オプションを使用して実行します。ステージ・コピー・ユーティリティーの実行方法の詳細については、「*WebSphere Commerce 管理ガイド*」の『ステージング・サーバー・コマンド』のセクションにある情報を参照してください。

データベース・クリーンアップ・ユーティリティーの再構成

データベース・マイグレーションでは、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 管理ガイド*」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。

ルール・サーバー構成のマイグレーション

以下の場合、このセクションを飛ばすことができます。

- 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 `WC55_installdir/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 5.5 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. 31 ページの『第 4 章 Commerce インスタンス構成のマイグレーション』で説明しているとおりにインスタンスをマイグレーションしたことを確認します。
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開きます。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/CommerceServer55/instances/demo/rules/ConsumerCredit_POSTER.adv</Project>
  </RulesProjectLoaderFactory>
</RulesServiceAgentFactoryFactory>
  <DeploymentType>Java</DeploymentType>
  <DeployRulesServiceClientContextFactory>
    <JavaName>com.blazesoft.server.deploy.rules.NdDeployPosterRulesServiceClientContext</JavaName>
    <SRIMappingClass>ScoredLoanApplication</SRIMappingClass>
  </DeployRulesServiceClientContextFactory>
</DeployRulesServiceConfig>
```

```

</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>
    <SrIArgumentsObjectFactory>
      <SrIClass>ScoredLoanApplication</SrIClass>
      <SrIObjectInitializerFunctional>
        <SrIName>initServiceData</SrIName>
        <SrIArgumentType>string</SrIArgumentType>
      </SrIObjectInitializerFunctional>
      <SrIArgumentsObjectFactory>
      <SrIResultExtractorFunctional>
        <SrIName>extractServiceResult</SrIName>
      </SrIResultExtractorFunctional>
    </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 5.5 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. 「ルール・サービス管理 (Rule Service Administration)」ツールに進みます。
(「ルール・サービス」>「管理」)
6. それぞれの `DeployRulesServiceConfig` タグで以下を行います。
 - a. 「ルール・サービス管理 (Rule Service Administration)」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のパスは以下のとおりです。

```
/opt/WebSphere/CommerceServer55/instances/demo/rules/ConsumerCredit_POSTER.adv</Project
```

別々のフォルダ名を区切るには、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。

21 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 5.5 サーバーを再始動して「ルール・サービス管理 (Rule Service Administration)」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検査します。

さらに、ルール・サーバー管理コマンドのマイグレーションに関する考慮事項については、「*WebSphere Commerce Studio* マイグレーション・ガイド」を参照してください。

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 5.5 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 5.5 の新しいデフォルト契約が入れられます。WebSphere Commerce 5.5 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、ORDERITEMS 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・アカウントおよび契約

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 5.5 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 5.5 オンライン・ヘルプの『新規ビジネス・アカウントの作成』のトピックを参照してください。

StoreGroup1 の所有者は、ユーザー wcsadmin から組織 Root Organization に変更されています。インポートしたいすべてのアカウントまたは契約 XML ファイルで、PolicyReference が StoreGroup1 を参照している場合、Member エレメントは以下のものから変更する必要があります。

```
<Member>
  <User distinguishName ="uid=wcsadmin,o=Root Organization"/>
</Member>
```

これを以下のように変更します。

```
<Member>
  <Organization distinguishName ="o=Root Organization"/>
</Member>
```

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。後方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。

edit_registration ページにおけるログオン ID の形式 (LDAP が使用される場合)

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 で以下のような少しいちの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ように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

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. 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、`WC55_install/samples/pa/bin` ディレクトリー内の `createsearchspace.sh`)、*「WebSphere Commerce Studio マイグレーション・ガイド」*にある商品アドバイザー・コード・マイグレーションに関するステップに示されているとおりに、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 5.5 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 5.5 では、スケジューラー・コマンド

`PACreateSearchSpaceBatchCmd` を使用した検索スペース作成の新しい方法が導入されました。この方法では、入力として XML ファイルを使用して検索スペースを作成します。このコマンドは WebSphere Commerce 管理コンソールから実行します。さらに、検索スペースの作成、商品比較の定義、商品の探索、ストア・カテゴリー別のガイド付き販売メタフォーの作成が可能な新しいインターフェースが WebSphere Commerce アクセラレーター内に追加されました。

`PACreateSearchSpaceBatchCmd` スケジューラーは検索スペース作成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。 `WC55_installdir/samples/pa/xml` ディレクトリーでサンプル XML ファイルを検索できます。このプロセスの詳細については、「WebSphere Commerce Production 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。

2. スクリプトを実行して、マイグレーション済みの WebSphere Commerce 5.5 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
3. メタフォーを作成するための `PAXMLExportBatchCmd` スケジューラー・コマンドへの入力として使用される XML ファイルを、「*WebSphere Commerce Studio* マ

「イグレーション・ガイド」にある商品アドバイザー・コード・マイグレーションに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。

WC55_installdir/samples/pa/xml ディレクトリーでサンプル XML ファイルを検索できます。

4. PAXMLExportBatchCmd スケジューラー・コマンドを実行します。

ATP 在庫の丸め

Commerce Suite 5.1 データベースをマイグレーションする際に atp オプションを使用して ATP 在庫形式に変換する場合、RECEIPT テーブル内の数量にいくらかの丸め誤差が含まれる可能性があります。ATP 在庫に変換するとき、Commerce Suite 5.1 の INVENTORY.QUANTITY フィールドで DOUBLE と宣言されている数量は、WebSphere Commerce 5.5 の RECEIPT テーブルの QTY* フィールドで INTEGER に変換されます。

数量を訂正するためのツールを以下のように作成する必要があります。

- QuantityManager を使用して、INVENTORY.QUANTITY を単位 INVENTORY.QUANTITYMEASURE から単位 BASEITEM.QUANTITYMEASURE に変換します。この結果は X になります。
- X を BASEITEM.QUANTITYMULTIPLE で除算すると結果は Y になります。
- Y を最も近い整数に丸めると、正しい RECEIPT 数量が得られます。

以下に示すのは、そのようなツールの疑似コードの例です。

```
QuantityAmount qa = new QuantityAmount(value, oldUnits);
QuantityAmount quantityAmount = QuantityManager.getInstance().convert(qa, newUnits);

if (quantityAmount == null) {
    throw new ECSystemException(
        ECMessage._ERR_NO_CONVERSION,
        iClassName,
        methodName,
        new Object[] { oldUnits, newUnits });
}
```

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。支払通貨は以下のように決定されます。

- ストアによってサポートされる場合は、顧客の優先通貨になります。通貨がストアまたはストア・グループの CURLIST テーブルにある場合、または以下の文で定義されているように、ストアまたはストア・グループ用の言語に依存しないか、または言語に依存するデフォルト通貨である場合、ストアは通貨をサポートします。
- そうでなければ、顧客の優先通貨 (そのような通貨がある場合) を CURCVLIST テーブルでカウンター値として持つストアがサポートする通貨になります。
- それ以外の場合で、ストアまたはストア・グループ用に指定されている場合、STOREENT テーブルの SETCCURR 列で指定されている、ストア用の言語に依存しな

いデフォルト通貨になります。言語に依存しないデフォルト通貨という概念は WebSphere Commerce 5.4 で導入され、WebSphere Commerce 5.5 に引き継がれたという点に注意してください。

- それ以外の場合で、ストアまたはストア・グループ用に指定されている場合、STORELANG テーブルの SETCCURR 列で指定されている、現在の言語でのストア用の言語に依存するデフォルト通貨になります。

注:

1. 希望する通貨がない顧客や、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) がある顧客だけが、上記の新しい言語非依存デフォルト通貨の影響を受けます。顧客にサポートされている優先通貨がある場合は、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. Commerce Suite 5.1 の言語に依存するのデフォルト通貨を WebSphere Commerce 5.5 で保持するには、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。WebSphere Commerce 5.5 の言語に依存するデフォルト通貨をインプリメントする場合は、ストアまたはストア・グループの STOREENT テーブルでストアのデフォルト通貨を設定してください。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、言語に依存する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言語に依存するデフォルトの通貨の動作を保持するかを選択できます。

アクセス制御ポリシーのサブスクリプション

WebSphere Commerce 5.5 では、ポリシー・グループの概念が導入されました。各ポリシー・グループには、特定のビジネス要件またはアクセス制御要件に固有のポリシーが含まれています。たとえば、B2B ポリシー・グループには、B2B 機能およびアクセス制御の動作を必要とするストアを所有する組織に必要なポリシーが含まれます。

Commerce Suite 5.1 のストアは、マイグレーション・プロセスで作成されたカスタム・ポリシーを持つポリシー・グループに加えて、使用可能なすべての機能にアクセスできるため、マイグレーション・スクリプトはデフォルトで、ストアを所有する組織を以下のポリシー・グループにサブスクライブします。

- ManagementAndAdministrationPolicyGroup
- CommonShoppingPolicyGroup
- B2CPolicyGroup
- B2BPolicyGroup

これによって、ストアの以前の性質が保持されます。

マイグレーション後にストアを B2C ストアにしたい場合は、B2BPolicyGroup へのサブスクリプションを除去します。ストアを B2B ストアにしたい場合は、B2CPolicyGroup へのサブスクリプションを除去します。組織管理コンソールによるアクセス制御ポリシー・グループのサブスクライブおよびアンサブスクライブについては、「WebSphere Commerce 管理ガイド」を参照してください。

WebSphere Commerce 5.5 の新しいストア・タイプで他のストアを作成する場合は、別のポリシー・グループのセットをサブスクライブすることが必要な場合があ

ります。どのポリシー・グループにストアをサブスクライブするかについては、「*WebSphere Commerce* セキュリティー・ガイド」のアクセス制御の部分を参照してください。

第 3 部 付録

付録 A. 詳細情報の入手場所

WebSphere Commerce システムおよびそのコンポーネントについての詳細は、さまざまなソースからさまざまな形式で入手できます。以下に続くセクションでは、入手できる情報とそのアクセス方法を示しています。

WebSphere Commerce についての情報

以下は、WebSphere Commerce についての情報源です。

- WebSphere Commerce オンライン・ヘルプ
- WebSphere Commerce テクニカル・ライブラリー

WebSphere Commerce オンライン・ヘルプ

WebSphere Commerce オンライン情報は、WebSphere Commerce をカスタマイズ、管理、および再構成する際の主要な情報源です。WebSphere Commerce をインストールした後で、以下の URL からオンライン情報のトピックにアクセスすることができます。

`https://host_name:8000/wchelp/`

ここで、*host_name* は、WebSphere Commerce がインストールされているマシンの完全修飾された TCP/IP ホスト名です。

WebSphere Commerce テクニカル・ライブラリー

WebSphere Commerce テクニカル・ライブラリーは、以下の URL で入手できます。

`http://www.ibm.com/software/genservers/commerce/library/`

この資料のコピーおよび更新版は、WebSphere Commerce Web サイトの Library セクションから、PDF ファイルとして入手可能です。さらに、新規および更新文書も、WebSphere Commerce テクニカル・ライブラリーの Web サイトから入手できます。

WebSphere Commerce Payments についての情報

WebSphere Commerce Payments のヘルプは、次のヘルプ・アイコンをクリックして入手できます。



このヘルプ・アイコンは、WebSphere Commerce 管理コンソールおよび WebSphere Commerce アクセラレーターの WebSphere Commerce Payments ユーザー・インターフェース上、および以下の URL のスタンドアロン WebSphere Commerce Payments ユーザー・インターフェース内に表示されます。

`http://host_name:http_port/webapp/PaymentManager`

または

`https://host_name:ssl_port/webapp/PaymentManager`

ここで、変数は以下のように定義されます。

host_name

WebSphere Commerce Payments に関連した Web サーバーの完全修飾された TCP/IP ホスト名。

http_port

WebSphere Commerce Payments によって使用される HTTP ポート。デフォルトの HTTP ポートは 5432 です。

ssl_port

WebSphere Commerce Payments によって使用される SSL ポート。デフォルトの SSL ポートは 5433 です。

WebSphere Commerce Payments が SSL 対応の場合は、どちらの URL でも使用できます。WebSphere Commerce Payments が SSL 対応でない場合は、非セキュア URL (http) のみ使用できます。

ヘルプは以下の URL でも入手できます。

`http://host_name:http_port/webapp/PaymentManager/language/docenter.html`

または

`https://host_name:ssl_port/webapp/PaymentManager/language/docenter.html`

ここで、変数は以下のように定義されます。

host_name

WebSphere Commerce Payments に関連した Web サーバーの完全修飾された TCP/IP ホスト名。

http_port

WebSphere Commerce Payments によって使用される HTTP ポート。デフォルトの HTTP ポートは 5432 です。

ssl_port

WebSphere Commerce Payments によって使用される SSL ポート。デフォルトの SSL ポートは 5433 です。

language

ヘルプ・ページが表示される言語の言語コード。ほとんどの言語では 2 文字です。言語コードは以下のとおりです。

言語	コード
ドイツ語	de
英語	en
スペイン語	es
フランス語	fr
イタリア語	it
日本語	ja

言語	コード
韓国語	ko
ブラジル・ポルトガル語	pt
中国語 (簡体字)	zh
中国語 (繁体字)	zh_TW

WebSphere Commerce Payments および Payments Cassettes についての詳細は、
WebSphere Commerce テクニカル・ライブラリーで入手できます。

<http://www.ibm.com/software/genservers/commerce/library/>

IBM HTTP Server についての情報

IBM HTTP Server についての情報は、IBM HTTP Server Web サイトで入手できます。

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

資料は HTML 形式、PDF ファイル、またはその両方です。

WebSphere Application Server についての情報

WebSphere Application Server についての情報は、WebSphere Application Server InfoCenter で入手できます。

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

DB2 Universal Database についての情報

HTML 文書ファイルは、`/doc/locale/html` サブディレクトリーにあります。
locale はご使用のロケールの言語コードです (たとえば、米国英語の場合は *en*)。各国語で入手できない文書は、英語で表示されます。

入手可能な DB2 資料の完全なリスト、およびその表示または印刷方法については、「*DB2 for UNIX*概説およびインストール」を参照してください。DB2 についての追加情報が、DB2 テクニカル・ライブラリーで入手できます。

<http://www.ibm.com/software/data/db2/library/>

その他の IBM 資料

ほとんどの IBM 資料のコピーは、IBM 認定販売店または営業担当員から購入することができます。

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 Commerce Suite 5.1 データベース・スキーマを拡張した場合、提供されたデータ・マイグレーション・スクリプトを更新する必要があります。たとえば、INTEGER DEFAULT 0 および FRIEND は MBRGRP テーブルに対する外部キーを持っているため、FRIEND という新しい列を USERREG テーブルに追加するというケースを検討します。

提供されたマイグレーション・スクリプトを使用してデータ・マイグレーションを開始する前に、以下のことを実行します。

1. **DB2 データベースの場合は**、`WC55_installdir/schema/db2` ディレクトリーに進みます。
2. ファイル `wcs.schema.sql` のバックアップ・コピーを作成します。
3. `wcs.schema.sql` ファイルを編集して、CREATE TABLE USERREG ステートメントを検索します。

```
CREATE TABLE USERREG (  
    ...  
);
```

4. 以下のように、新しい列 FRIEND を CREATE TABLE USERREG ステートメントに追加します。

```
CREATE TABLE USERREG (  
    ...  
    FRIEND INTEGER DEFAULT 0  
);
```

ファイルの形式が維持されているかを確認してください。データベース・マイグレーション・スクリプトはファイルを構文解析して、テーブル内の列定義はそれぞれ固有の行であり、ブランク行はないと想定します。(これには大括弧が含まれます。)

5. ファイル `wcs.referential.sql` のバックアップ・コピーを同じディレクトリーに作成します。
6. ファイル `wcs.referential.sql` 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. テーブルをチェックして、作成した新しいテーブルに他にも制約があるかを確認します (このテーブルは標準の WebSphere Commerce スキーマ・テーブルを指しています)。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、除去しなければなりません。

注: マイグレーションの完了後に、これらの制約を再作成する必要があります。

8. `WC55_installdir/schema/migration/5.1.0.1` ディレクトリー内の `MapFile.txt` ファイルをチェックします。

変更したテーブルが **MapFile.txt** にリストされている場合、ソース列リストとターゲット列リストの両方に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ます。(以下の例では、表示上の理由で行が分割されていることにご注意ください。)

```
addrbook=(addrbook_id, description, displayname, member_id, type);  
          (ADDRBOOK_ID, DESCRIPTION, DISPLAYNAME, MEMBER_ID, TYPE)
```

括弧で囲まれた最初の文字ストリングのセットは、Commerce Suite 5.1 データベース内の ADDRBOOK テーブルの列を表します。括弧で囲まれた 2 番目の文字ストリングのセットは、WebSphere Commerce 5.5 データベースの ADDRBOOK テーブルの列を表します。Commerce Suite 5.1 の ADDRBOOK テーブルに FRIEND 列を追加したとします。その場合、MapFile.txt を変更して、以下に太字で示されているとおり、FRIEND 列を指定する必要もあります。

```
addrbook=(addrbook_id, description, displayname, member_id, type, friend);  
          (ADDRBOOK_ID, DESCRIPTION, DISPLAYNAME, MEMBER_ID, TYPE, FRIEND)
```

そうしないと、マイグレーション・スクリプトは FRIEND 列を WebSphere Commerce 5.5 にマイグレーションしません。

9. 65 ページの『第 6 章 Commerce Suite 5.1 データベースのマイグレーション』で説明されているとおりに、マイグレーション・スクリプトを実行します。

付録 C. WCIM ツールおよびデータ・マイグレーション・スクリプトの概要

このセクションは、WCIM (WebSphere Commerce Instance Migration) ツールとデータベース・マイグレーション・スクリプトについて説明するもので、情報提供のために含まれています。この情報に基づいてユーザーが取るべきアクションはありません。

WCIM を使用したインスタンス・マイグレーション

WCIM ツールは、以下のインスタンス・マイグレーション・ステップを実行します (wcimenv 内の WORK_DIR 変数は WC55_installdir/temp に設定されていると想定しています。WORK_DIR がユーザー定義であることに注意してください)。

1. WebSphere Commerce 5.5 インスタンス・テンプレート・ファイルを WC55_installdir/temp ディレクトリーにコピーします。
2. コピーした ZIP ファイルを WC55_installdir/temp ディレクトリーに解凍します。
3. インスタンス・マイグレーションを以下のように実行します。
 - インスタンス構成ファイル *instance_name.xml* をマイグレーションします。
 - wcs_instances ファイルをインスタンス情報で更新します。
 - 各インスタンスごとに Web サーバー構成ファイルをマイグレーションします。
 - カスタマイズ済みのストア・プロパティ・ファイルをマイグレーションします。
 - 必要な変更を実行して、JavaServer Pages レベルをバージョン 1.3 にマイグレーションします。(手動でのいくつかの追加の変更も必要になります。)
 - 解凍された Commerce Suite 5.1 インスタンス・ファイルを、WebSphere Commerce 5.5 EAR ファイル内の適切な場所にコピーします。
 - WebSphere Commerce 5.5 デプロイメント記述子を更新します。
4. JACL スクリプト・ファイルを生成して、WebSphere Application Server の構成を支援します (たとえば、仮想ホスト、クラスパス、JVM プロパティなど)。
5. WORK_DIR ディレクトリー内の wcimenv の LOG_FILE 変数で指定されたログ・ファイルを生成します。たとえば、以下のようにします。

たとえば、WORK_DIR を WC55_installdir/temp に設定した場合、ログ・ファイルは WC55_installdir/temp/logs ディレクトリーに生成されます。

WCIM ツール が JSP ファイル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 5.5 へのマイグレーション時のものです。

- Commerce Suite 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箇所を変更する必要があります。

```
<jsp:include page="<%=incfile%>"/>
```


これを以下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 5.5 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 Microsystems 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) のすべての出現箇所を変更します。

```
<Form NAME="BillAddressForm" METHOD="POST" action="<%=OrderCopy%">
```

または

```
<Form NAME=BillAddressForm METHOD="POST" action="<%=OrderCopy%">
```

これを以下のように変更します。

```
<Form NAME="BillAddressForm" METHOD="POST" action="OrderCopy">
```

- JSP 1.2 仕様では、サポートされている言語は "java" だけであると宣言されています。したがって、JSP での以下のページ言語宣言はもはや無効です。

```
<%@ page language="JAVA" %>
```

なお、WCIM ツールは、すべての `<%@ page language="JAVA" %>` を、ユーザーに代わって `<%@ page language="java" %>` に変換します。

データ・マイグレーション・スクリプト

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
- アクセス制御

Commerce Suite 5.1 と WebSphere Commerce 5.5 との間のデータベース・スキーマの変更の要約については、「WebSphere Commerce Production and Development オンライン・ヘルプ」の『このリリースでのデータベース・スキーマの変更』を参照してください。

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 テーブルの状況を以下のように設定します。
 - スクリプトは、以下に対しては状況をヌル (状況なし) 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、以下に対しては状況を承認済み (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

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・プロファイルを持つ (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ァイル・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 テーブルで、カスタム・メンバー・グループが AccessGroup (-2) の MBRGRPTYPE_ID に関連付けられている場合、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。
 - Commerce Suite 5.1 で Order Clerk または Store Developer 役割が使用されている場合、マイグレーション・スクリプトはそれらの役割を WebSphere Commerce 5.5 のユーザー定義役割にマイグレーションします。

スクリプトは、WebSphere Commerce 5.5 の ROLE テーブル内の各役割の MBRROLE テーブルにレコードを追加し、これらのレコードの 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、役割が -1 (Site administrator) に設定されたエントリーを作成します。

- すべての親および先祖に同じ役割が割り当てられるようにします。
- 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するものであり、ゲスト・ユーザーに対するものではありません。
- SQL ステートメントを生成して、ORGENITY 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。ORGENITY 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、70 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 5.5 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。ゼロ以外の値であれば、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 5.5 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 アクセラレーター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。後方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。

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 アクセラレーター・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 アクセラレーター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 5.5 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 アクセラレーター・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 5.5 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 5.5 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 5.5 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 MGTRDPSCN 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 5.5 では、(もともとは WebSphere Commerce 5.4 で導入された) 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 5.5 にマイグレーションした後に、WebSphere Commerce アクセラレーターを使用してそれを行います。WebSphere Commerce 5.5 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 5.5 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 ページ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 5.5 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 5.5 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 5.5 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、BZRPENTSTG 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 EMPSPOT 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。

アクセス制御のマイグレーション

アクセス制御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 5.5 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 5.5 ACPOLICY テーブルへのマイグレーション。
- Commerce Suite 5.1 の Order Clerk および Store Developer 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 AccessGroup 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 5.5 でも、アクセス制御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るよう推奨されています。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 5.5 役割および WebSphere Commerce 5.5 アクセス・グループにマップしています。MBRGRPCOND テーブルのCONDITIONS 列には、アクセス・グループに属するための暗黙の条件が保管されています。

表 9.

Commerce Suite 5.1 アクセス・グループ	WebSphere Commerce 5.5 役割	WebSphere Commerce 5.5 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)	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
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 5.5 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 5.5 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 5.5 ブートストラップ・アクセス制御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. Order Clerk および Store Developer 役割は、WebSphere Commerce 5.5 では使用されません。そのため、WebSphere Commerce 5.5 ブートストラップ・アクセス制御ポリシーは、これらの役割を参照しません。ただし、顧客がこれら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制御ポリシーを作成します。

以前の動作を保持するために、マイグレーション・スクリプトは、ストアを所有する組織を必須のポリシー・グループにサブスクライブします。詳しくは、131 ページの『アクセス制御ポリシーのサブスクリプション』を参照してください。

データ・マイグレーションの補足情報

以下のセクションでは、データベース・マイグレーション・ツールに関する補足的な参照情報、およびデータ・マイグレーション・プロセスのステップに関する補足的な考慮事項を説明します。

データベース準備スクリプトの戻りコード

以下のセクションでは、データベース準備スクリプトからの必須の戻りコードおよびアクションについて説明します。

必須のデータベース・プレマイグレーション項目

データベース準備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 (ストア開発者)
- 8 Merchandising Manager (取引管理マネージャー)

上記のメンバー ID が、上記のメンバー・グループに対応しているか確認する必要があります。

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。つまり、メンバー・グループ ID は上記の値から変更できません。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 5.5 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。固有であると思われる 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 5.5 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、ORGENITY 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。

323

スクリプトは MSGSTORE テーブル内に未配布メッセージがあるかどうかを検査します。

データベース・マイグレーション・スクリプトを実行する前に、msgviewer ツールを実行する必要があります。

- 324** スクリプトは、CATENTSHIP テーブルにある商品の QUANTITYMULTIPLE フィールドが、BASEITEM テーブルの QUANTITYMULTIPLE フィールドと整合しているかどうかを検査します。
- フィールド間で違いがある場合は、同じにする必要があります。
「WebSphere Commerce Production and Development オンライン・ヘルプ」で、BASEITEM テーブルの QUANTITYMULTIPLE フィールドについての説明を参照してください。
- 325** スクリプトは、CATENTSHIP テーブルのにある商品の QUANTITYMEASURE フィールドが、BASEITEM テーブルの QUANTITYMEASURE フィールドと整合しているかどうかを検査します。
- フィールド間で違いがある場合は、同じにする必要があります。
「WebSphere Commerce Production and Development オンライン・ヘルプ」で、BASEITEM テーブルの QUANTITYMEASURE フィールドについての説明を参照してください。
- 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。
- 500** いずれかのストアの LANGUAGE_ID 列がヌルです。そのストアに適切な LANGUAGE_ID を割り当ててください。更新が必要なストア・エントリーを判別するには、以下の照会を発行します。
- ```
select store_id,language_id from store
```
- 501** いずれかの契約の名前がヌルです。契約の契約名を設定します。更新が必要な契約エントリーを判別するには、以下の照会を発行します。
- ```
select contract_id,name from contract
```
- 502** いずれかの契約の MEMBER_ID 列がヌルです。契約の MEMBER_ID を設定します。更新が必要な契約を判別するには、以下の照会を発行します。
- ```
select member_id, contract_id from contract"
```
- 503** いずれかの TRADEPOSCN エントリーの NAME 列がヌルです。取引価格の NAME 列を設定します。更新が必要な TRADEPOSCN エントリーを判別するには、以下の照会を発行します。
- ```
select tradeposcn_id,name from tradeposcnt
```
- 504** いずれかの CALCODE エントリーの CODE 列がヌルです。計算コードの CODE 列を設定します。更新が必要な CALCODE エントリーを判別するには、以下の照会を発行します。
- ```
select calcode_id, code from calcode
```

## オプションのデータベース・プレマイグレーション項目

データベース準備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 5.5 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。このデータは WebSphere Commerce 5.5 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 役割をそれらの先祖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 5.5 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 5.5 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 5.5 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 5.5 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 5.5 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、アクセス制御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 5.5 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、ORGENITY 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。
- ORGENITY 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。
- 317** Commerce Suite 5.1 は Store Developer 役割をサポートしていましたが、WebSphere Commerce 5.5 では不要になり使用されなくなりました。Store Developer 役割で実行されていたタスクは、自動化されているか、または WebSphere Commerce 5.5 の Site Administrator で実行できます。ユーザーに Commerce Suite 5.1 で Store Developer 役割があり、ACCCMDGRP テーブルにエントリーがある場合、そのユーザーは、アクセス制御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。Store Developer 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 Store Developer 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 5.5 ではサポートされなくなったので、除去してください。
- 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 5.5 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 5.5 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 5.5 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** アクセス制御に関係した問題があります。アクセス制御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. 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.
  - The owner xxxx for the member group defined in ACCMBRGRP is not an organization and will not be migrated.
- 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。

## 列の長さの確認

WebSphere Commerce 5.5 では、以下の列の長さが変更されています。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、データ・マイグレーションの前に、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おいてください。たとえば、MBRGRP.DESCRPTION が 512 字より大きくないことを確認してください。列内の既存のデータが新しい長さを超える場合は、データベース準備スクリプトによってフラグ化されます。

| Table.Column               | Commerce Suite 5.1 | WebSphere Commerce 5.5 |
|----------------------------|--------------------|------------------------|
| MBRGRP.DESCRPTION          | LONG VARCHAR       | VARCHAR(512)           |
| ORGENCY.DESCRPTION         | LONG VARCHAR       | VARCHAR(512)           |
| CONTRACT.NAME              | VARCHAR(254)       | VARCHAR(200)           |
| BZRPENTSTG.VALUE           | LONG VARCHAR       | VARCHAR(4000)          |
| CALCODEDSC.LONGDESCRIPTION | LONG VARCHAR       | VARCHAR(4000)          |
| CATALOGDSC.LONGDESCRIPTION | LONG VARCHAR       | VARCHAR(4000)          |
| CATENTDSC.LONGDESCRIPTION  | LONG VARCHAR       | VARCHAR(4000)          |
| CATENTDSC.AUXDESCRIPTION1  | LONG VARCHAR       | VARCHAR(4000)          |
| CATENTDSC.AUXDESCRIPTION2  | LONG VARCHAR       | VARCHAR(4000)          |
| COLLDSC.MARKETINGTEXT      | LONG VARCHAR       | VARCHAR(4000)          |
| CONTRLRULE.RULETEXT        | LONG VARCHAR       | VARCHAR(4000)          |

| Table.Column            | Commerce Suite 5.1 | WebSphere Commerce 5.5 |
|-------------------------|--------------------|------------------------|
| FFMCENTDS.DESCRPTION    | LONG VARCHAR       | VARCHAR(4000)          |
| FORUM.DESCRPTION        | LONG VARCHAR       | VARCHAR(4000)          |
| FORUMMSG.ACTREMARK      | LONG VARCHAR       | VARCHAR(4000)          |
| FORUMMSG.MSGBODY        | LONG VARCHAR       | VARCHAR(4000)          |
| PAYSTATUS.SETSWAKEUPMSG | LONG VARCHAR       | VARCHAR(4000)          |
| STOREENTDS.DESCRPTION   | LONG VARCHAR       | VARCHAR(4000)          |
| USERPROF.DESCRPTION     | LONG VARCHAR       | VARCHAR(4000)          |
| USRTRAFFIC.PREVURL      | LONG VARCHAR       | VARCHAR(254)           |
| USRTRAFFIC.QUERYSTRING  | LONG VARCHAR       | VARCHAR(4000)          |
| USRTRAFFIC.REFURL       | LONG VARCHAR       | VARCHAR(254)           |

## ATP マイグレーションに関する考慮事項

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、8 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。詳細については、WebSphere Commerce 5.5 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、163 ページの『付録 D. 後で実行する ATP 在庫への変換』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。

### 注:

1. atp を指定して WebSphere Commerce 5.5 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ません。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 アクセラレーターを使用してベンダーを作成する必要があります。
2. migrateATP スクリプトを使用して後から ATP 在庫にマイグレーションすることを選択した場合、このスクリプトは製品カテゴリー・エントリーごとに 1 つの基本項目 (BASEITEM)、および項目カテゴリー・エントリーごとに 1 つの指定項目 (ITEMSPC) を作成します。通常これは、データ・マイグレーション時に ATP 在庫表記へのマイグレーションを選択した場合に、マイグレーション・スクリプトによって実行されます。

## データベース・マイグレーション・ログ・ファイルのチェック

マイグレーション・スクリプトの実行後に、以下のようにマイグレーション・ログ・ファイルをチェックします。

1. migratedb\_database\_name.log ファイルをチェックします。

このログ・ファイルは、WC55\_installdir/logs/migration/instance\_name 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、メッセージをアナライズして、必要であれば、データを修正してデータ・マイグレーションを再試行する必要があります。

ログ・ファイル中の以下のタイプのメッセージは無視することができます。

- 以下の SQLSTATE メッセージは、migratedb51.log 内では次のように表示されます。

```
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・メッセージは、データベース・マイグレーション・ログ内では次のように表示されます。これらのメッセージは、オプションの要約テーブルが、データの移植時に Commerce Suite 5.1 データベースにロードされなかった場合にのみ出されます。WebSphere Commerce 5.5 のデフォルトでは、WebSphere Commerce 5.5 データベースで要約テーブルにデータが移植されます (wcs.summary.sql スクリプトを使用)。したがって、これらのメッセージが出されても、無視することができます。

```
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
```

- マイグレーション・スクリプトで `wcs.dropview.sql` スクリプトが実行された場合、ビューが未定義名であれば、以下のようなメッセージが生成されることがあります。Commerce Suite 5.1 で WebSphere Catalog Manager を使用しなかった場合、これらのビューは存在しない場合があり、このメッセージを受け取ります。存在する場合、これらのビューはマイグレーション・スクリプトによって除去されます。

```
drop VIEW STOREINV
*The SQLException caught *
SQLState: 42S02
Message: [IBM][CLI Driver][DB2/NT] SQL0204N "DB2ADMIN.STOREINV" is an undefined name. SQLSTATE=42704
```

Vendor Code: -204

```
drop VIEW wcPRODDdesc
*The SQLException caught *
SQLState: 42S02
Message: [IBM][CLI Driver][DB2/NT] SQL0204N "DB2ADMIN.WCPRODDDESC" is an undefined name. SQLSTATE=42704
```

Vendor Code: -204

```
drop VIEW wcPRODship
*The SQLException caught *
SQLState: 42S02
Message: [IBM][CLI Driver][DB2/NT] SQL0204N "DB2ADMIN.WCPRODSHIP" is an undefined name. SQLSTATE=42704
```

Vendor Code: -204

.  
.  
.

- `defaultAccessControlPoliciesout.error.xml` ファイルが、`WC55_installdir/logs/migration/instance_name` ディレクトリーに生成されます。このファイルに重複したレコードに関するメッセージしか含まれていない場合は、無視できます。これらのメッセージは、レコードがすでにデータベースに存在しているため、再ロードされないということを伝えるものです。

## 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、

`WC55_installdir/logs/migration/instance_name` ディレクトリーにさまざまなログ・ファイル (拡張子 `.log`) を生成します。

Commerce Suite 5.1 データベースに対して生成されるログ・ファイルを以下に示します。

#### **preigrate\_dbname.log**

このログ・ファイルには、プレマイグレーション・スクリプトが Commerce Suite 5.1 データベースに対して実行されたときに生成されたすべてのメッセージが入っています。このログ内に、フラグが立てられたエラーまたは警告がないことを確認してください。

#### **migratedbscripts.log**

Commerce Suite 5.1 から WebSphere Commerce 5.5 へのスキーマの変更を扱うファイルの生成用のログ・ファイル。

*WC55\_installdir/schema/instances/instance\_name/migration/dbType* ディレクトリーに生成されるファイルは以下のとおりです。

- *wcs.schema.create.sql* — 既存のテーブルから一時テーブルへの名前変更、新規テーブルの作成、既存のテーブルの変更、列などの追加、および一時テーブルの内容の新規テーブルへのコピーなどを実行するステートメントが入っています。
- *wcs.temp.drop.sql* — すべての一時的テーブルを除去するステートメントが入っています。ただしステージング・テーブルは除去しません (これは手動でマイグレーションする必要があります)。
- *delwcs.constraint.sql* — すべての制約をスキーマから削除するステートメントが入っています。

#### **migratedb\_instance\_name.log**

*instance\_name* で指定されたインスタンスのブートストラップ・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。これには、実行された実際のスキーマの変更のログが入ります。たとえば、制約の除去、テーブルの名前変更、テーブルの変更、新規スキーマ・テーブルの作成、および一時テーブルから新規テーブルへのデータのコピーなどです。さらにこれは、ビジネス・ロジック関連データのすべての操作にもフラグを立てます。

#### **OrigSchema.log**

データベース内にある、元となるリリースの 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 5.5 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 5.1 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)。

#### **その他のログ・ファイル**

以下のファイルの場所は、WCALoggerConfig.xml ファイルの内容によって決まります。このファイルが *WC55\_installdir/instances/instance\_name/xml/loader* ディレクトリーにある場合は、これらのファイルの場所を決めるのに使用されます。そうでなければ、場所は *WC55\_installdir/xml/loader/WCALoggerConfig.xml* の内容によって決まります。

#### **messages.txt**

Mass Loader および ID リゾルバーのメッセージが入っています。

#### **trace.txt**

Mass Loader および ID リゾルバーのトレース・ファイルが入っています。このファイルのサイズは通常はかなり大きいことに注意してください。

### **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に、

*WC55\_installdir/logs/migration/instance\_name* に生成された、データベース・マイグレーション・ログ・ファイルの内容を検討する必要があります。

- *migratedb\_dbname.log* — Commerce Suite 5.1 の主なマイグレーション・ログ (ブートストラップ・データの作成、新規スキーマ・テーブルの作成、制約、ビジネス・ロジック関連データの操作)

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。 *migratedb.log* ファイルは以下のようになっているはずです。

```
Migration starts... Date:2003-06-30
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.
```

何らかの理由でデータベース・マイグレーションに満足できず、データベース・マイグレーションを再度実行する必要がある場合は、以下のようにする必要があります。

1. 部分的にマイグレーションされたデータベース・スキーマを除去します。
2. バックアップ・デバイスから元の Commerce Suite 5.1 スキーマをリストアします。
3. マイグレーション時に生じた問題を修正します。
4. データのプレマイグレーション・タスクから、マイグレーション・プロセスを再度開始します。

## 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。

- 以下の 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。

```
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。

```
SELECT DN FROM ORGENITY
```

```
SELECT DN FROM USERS
```

いずれかの DN エントリーが空であれば、70 ページの『識別名の更新』で説明しているように、マイグレーション後にデータベースの 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 5.5 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 CATENTREL 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 CATENTREL テーブルから除去します。 |
| 405 | マイグレーションされた USER 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。  
BUSPROF テーブル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、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。ORGENITY 内の MEMBER\_ID 列の値が、BUSPROF 内  
の ORG\_ID 列の値と異なっています。ORGENITY 内の MEMBER\_ID  
列の値が、BUSPROF 内の ORG\_ID 列の値と同じになるようにしてく  
ださい。

**500** 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 つをマスタ  
ー・カタログとして指定しておく必要があります。

72 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、choosemc.sql 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。

**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 5.5 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い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よう  
にする必要があります。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よう  
にする必要があります。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。

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、色の属性を持つアイテムがあり、その関連属性値が赤および青の場合、ATTRVALUE テーブルは以下ようになります。

| CATENTRY_ID | ATTRVALUE_ID | NAME      | STRINGVALUE | LANGUAGE_ID | ATTRIBUTE_ID |
|-------------|--------------|-----------|-------------|-------------|--------------|
| item1_id    | redval_id    | redvalue  | red         | -1          | color_id     |
| item2_id    | blueval_id1  | bluevalue | blue        | -1          | color_id     |
| item3_id    | blueval_id2  | bluevalue | blue        | -10         | color_id     |
| item4_id    | redval_id    | redvalue  | red         | -1          | color_id     |

ATTRVALUE テーブルの CATENTRY\_ID が 0 (ゼロ) に設定されている箇所に、データを取り込む必要があります。

| CATENTRY_ID | ATTRBUTEVALUE_ID | NAME      | STRINGVALUE | LANGUAGE_ID | ATTRIBUTE_ID |
|-------------|------------------|-----------|-------------|-------------|--------------|
| item1_id    | 1001             | redvalue  | red         | -1          | color_id     |
| item2_id    | 1002             | bluevalue | blue        | -1          | color_id     |
| item3_id    | 1003             | bluevalue | blue        | -10         | color_id     |
| item4_id    | 1004             | redvalue  | red         | -1          | color_id     |
| 0           | 1005             | redvalue  | red         | -1          | color_id     |
| 0           | 1006             | bluevalue | blue        | -1          | color_id     |
| 0           | 1007             | bluevalue | blue        | -10         | color_id     |

上記のテーブルは完全なテーブルではなく、例示だけを目的としている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 5.5 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 5.5 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。12 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 5.5 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、*WC55\_installdir/samples* ディレクトリーにある、WebSphere Commerce 5.5 に同梱されているサンプル・ストアを参照してください。



---

## 付録 D. 後で実行する ATP 在庫への変換

WebSphere Commerce 5.5 にマイグレーションする時にデータベース・マイグレーション・スクリプトの `atp` オプションを選択する場合、デフォルトでは、WebSphere Commerce 5.5 でサポートされる (WebSphere Commerce 5.4 で導入された) ATP 在庫表記に変換されます。在庫データを、マイグレーション・フレームワークの `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、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、データベース・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 5.5 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 5.5 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、および `STARTDATE` に現在日付を指定します。
- 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、これは以下を行います。

- 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・ウィンドウを開きます。
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. 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
```

4. WebSphere Commerce 5.5 がインストールされている bin サブディレクトリに切り替えます。たとえば、WC55\_installdir/bin などです。
5. 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 instance_name
db2insthome host_name
```

ここで

- *db\_name* は、マイグレーションする Commerce Suite 5.1 データベースです (たとえば、mall)。
- *db\_userID* は、マイグレーションする 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、マイグレーションする 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)。
- *instance\_name* は、ご使用の Commerce Suite 5.1 インスタンスの名前です (たとえば、demo)。
- *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)。
- *host\_name* は、マシンの完全修飾ホスト名です (たとえば、myhost.montreal.ca)。

たとえば、在庫データを新規の 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 demo /home/db2inst1/sqllib
myhost.montreal.ca
```

- 6. *WC55\_installdir/logs/migration/instance\_name* ディレクトリーに生成される *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. サンプル JSP の更新

マイグレーションの後に WebSphere Commerce 5.5 でご使用のストアのさまざまな面を正しく機能させるために、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_MINIMUMLLENGTH_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"> </td>

<td align="left" valign="top" colspan="3" class="categoryspace">
<%=infashiontext.getString("REGISTRATION")%>
<hr width="580" noshade align="left">
*<%=infashiontext.getString("REQUIRED_FIELDS3")%></td>
</tr><tr>

```

```

<td align="left" valign="top" width="400" class="topspace">
<%
if (strErrorMessage != null)
{
 //We have error message.
 %>
<p><%=strErrorMessage%>

</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">
*<%=infashiontext.getString("EMAIL2")%></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">
*<%=infashiontext.getString("PASSWORD3")%></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">
*<%=infashiontext.getString("VERIFY_PASSWORD3")%></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">
*<%=infashiontext.getString("LAST_NAME3")%></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">
<%=infashiontext.getString("FIRST_NAME3")%></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%= ECUserConstants.EC_ADDR_FIRSTNAME %>" value="<%=strFirstName%>" type="text"></td>
</tr>
<% } else { %>

<tr>
<td align="right" valign="middle">
<%=infashiontext.getString("FIRST_NAME3")%></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">
*<%=infashiontext.getString("LAST_NAME3")%></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">
 </td>
<td align="left" valign="top" class="categoryspace">
<table cellpadding="4" cellspacing="0" border="0">
<tr>
<td align="left" valign="middle" bgcolor="#FFCC99">
<%=infashiontext.getString("SUBMIT")%>
</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">
<%=infashiontext.getString("YOUR_PRIVACY")%>

<%=infashiontext.getString("PRIVACY_STATEMENT")%>

<a href="PrivacyView?langId=<%=languageId%>&storeId=<%=storeId%>&catalogId=<%=catalogId%>">
<%=infashiontext.getString("LEARN_MORE")%>
</td></tr></table>
</td>

</tr></table>
</td>

</tr></table>
<%
infile = "/" + storeDir + "/footer.jsp";
%>

<jsp:include page="<%=infile%>" 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(EConstants.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");
 }
}
}

```



```

<table cellpadding="4" cellspacing="0" border="0">
<tr>
<td align="left" valign="middle" bgcolor="#FFCC99">
<A href="UserRegistrationForm?storeId=<%=storeId%>&langId=<%=languageId%>&catalogId=<%=catalogId%>">
<%=infashiontext.getString("CHANGE_INFO")%></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><%=strErrorMessage%>

<%
}
}%>
<table cellpadding="0" cellspacing="0" border="0">
<tr>
<td align="left" valign="top">
<%=infashiontext.getString("UPDATE_ADDRESS1")%>

</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%>">
<%=infashiontext.getString("EDIT_ADD")%></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.
PASSWORD_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 F. 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。

1. **問題:** DB2 データベースの場合、 WebSphere Commerce 組織管理コンソールからアクセス・グループを削除しようとする、 SystemOut.log ファイルに以下のようなエラーが記録される。

```
[7/21/03 13:39:28:892 CST] 289b7b86 ExceptionUtil E CNTR0020E: Non-application exception occurred while processing method "remove" on bean "BeanId(WC_demo#Member-MemberManagementData.jar#MemberGroup, com.ibm.commerce.user.objects.MemberKey@5425b554)". Exception data: com.ibm.websphere.cpi.CPIException: ; nested exception is: com.ibm.ejs.cm.portability.ResourceAllocationException: [IBM] [CLI Driver] [DB2/LINUX390] SQL0973N Not enough storage is available in the "PCKCACHESZ" heap to process the statement. SQLSTATE=57011
```

**解決法:** 以下の SQL ステートメントを使用して、パッケージ・キャッシュ・サイズを 4096 (4 KB ページ) あるいはそれ以上の値に増やす必要があります。

```
db2 update database configuration for database_name using pckcachesz 4096
```

2. **問題:** DB2 データベースの場合、 WebSphere Commerce 組織管理コンソールからアクセス・グループを削除しようとする、 SystemOut.log ファイルに以下のようなエラーが記録される。

```
[8/21/03 0:31:41:413 PDT] 12264319 ExceptionUtil E CNTR0020E: Non-application exception occurred while processing method "remove" on bean "BeanId(WC_demo#Member-MemberManagementData.jar#MemberGroup, com.ibm.commerce.user.objects.MemberKey@5425b5ab)". Exception data: com.ibm.websphere.cpi.CPIException: ; nested exception is:
```

```
COM.ibm.db2.jdbc.DB2Exception: [IBM] [CLI Driver] [DB2/LINUX] SQL0101N The statement is too long or too complex. SQLSTATE=54001
```

```
COM.ibm.db2.jdbc.DB2Exception: [IBM] [CLI Driver] [DB2/LINUX] SQL0101N The statement is too long or too complex. SQLSTATE=54001
```

**解決法:** 以下の SQL ステートメントを使用して、 SQL ステートメント・ヒープを 60000 (4 KB ページ) あるいはそれ以上の値に増やす必要があります。

```
db2 update database configuration for database_name using stmtheap 60000
```

3. **問題:** WebSphere Application Server 5.0 でアプリケーション・サーバーを始動すると、以下のようなエラーが戻される。

```
7e171f0 BeanMetaData E CNTR0035E: EJB container caught com.ibm.ejs.container.ContainerException: ; nested exception is: com.ibm.websphere.cpi.CPIException: Persister creation failed.The bean's J2EEName is WC_demo#Catalog-ProductManagementData.jar#Package. The data source name is jdbc/WebSphere Commerce DB2 DataSource demo.; nested exception is: com.ibm.ejs.persistence.EJSPersistenceException: ; nested exception is: java.sql.SQLException: SQLAllocEnv() Error: rc = -1 com.ibm.websphere.cpi.CPIException: Persister creation failed.The bean's J2EEName is WC_demo#Catalog-ProductManagementData.jar#Package. The data source name is jdbc/WebSphere Commerce DB2 DataSource demo.; nested exception is: com.ibm.ejs.persistence.EJSPersistenceException: ; nested exception is: java.sql.SQLException: SQLAllocEnv() Error: rc = -1 com.ibm.ejs.persistence.EJSPersistenceException: ; nested exception is: java.sql.SQLException: SQLAllocEnv() Error: rc = -1 java.sql.SQLException: SQLAllocEnv() Error: rc = -1 at COM.ibm.db2.jdbc.DB2BaseDataSource.loadDLL(DB2BaseDataSource.java:316)
```

**解決法:** DB2 データベースの場合、マイグレーションされた WebSphere Commerce アプリケーション・サーバーを始動する前に、DB2 インスタンス・ユーザー ID (たとえば db2inst1) でログインし、以下のスクリプトを実行します。

```
.instance_home/sql1lib/db2profile
```

ここで *instance\_home* は、DB2 インスタンスのホーム・ディレクトリーです (たとえば /home/db2inst1)。このコマンドは、アプリケーション・サーバーを始動する前に実行する必要があります。

4. **問題:** DB2 データベースの場合に、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、以下のエラーが戻される。

```
SQL0964C The transaction log for the database is full SQLSTATE=57011
```

**解決法:** DB2 ログ・ファイルのサイズを、データベースに合わせて大きくする必要があります。以下の SQL ステートメントを使用して、DB2 ログ・ファイルのサイズを最低でも 1000 ブロックに設定する必要があります。しかし、データベースが大規模な場合、ログ・ファイルのサイズを 10000 ブロックまたはそれ以上に変更することが必要になる場合があります。

```
db2 update database configuration for data_base_name using logfilsiz 10000
```

(ログ・ファイルのサイズは DB2 Control Center 内で変更することもできます。)

5. **問題:** CatalogDataBean という Access Bean を使用するストア、または WebFashion サンプル・ストアに基づくストアがあるときに、商品などのカタログ・グループをホーム・ページに表示できなくなった。

**解決法:** Commerce Suite 5.1 では、CatalogDataBean は、現行ストアに属さないカタログ・グループを表示します。つまり、表示すべきでないカタログ・グループが表示されます。WebSphere Commerce 5.5 では、この振る舞いは変更されて、現行ストアに属さないカタログ・グループが表示されないようになりました。

マイグレーション後にホーム・ページにカタログ・グループを表示する場合は、以下のステップを実行します。

- a. 以下の SQL を実行します。

```
select * from cattogrp where CATALOG_ID= your_store_id
```

- b. この結果から、ホーム・ページに欠落しているカタログの *catgroup\_id* の値 (たとえば 11111) が分かります。

- c. 以下の SQL を実行します。

```
insert into storecgrp (catgroup_id,storeent_id) values (11111,your_store_id)
```

- d. ストアを起動します。これでホーム・ページで商品を探ることができるようになるはずです。

6. **問題:** 実稼働マシン上でのマイグレーション中にソフトウェアをアップグレードした後に、IBM HTTP Server を再始動できません。

**解決法:** httpd.conf 構成ファイルから plug-in-cfg.xml ファイルへのパスを調べます。プラグインのパスが以前の IBM HTTP Server の古い位置を指している場合、WebSphere Application Server 暫定修正は、WebSphere Commerce 5.5 のインストール時にシステムに正しく適用されていない可能性があります。(これ

は Web サーバーが、 WebSphere Commerce 5.5 のインストール時にシャットダウンされていなかったことが原因の可能性があります。)

WebSphere Application Server 暫定修正 PQ72596 によって、プラグインの位置は正しいパス `WAS50_installdir/plugins/plugin-cfg.xml` に変更されます。この修正を正しく適用してください。

7. **問題:** キャンペーン・データのマイグレーション中に、データベース・マイグレーション・ログで以下のようなエラーを受け取ります。

```
*** Migrating INITIATIVE ***
Query: select * from keys where tablename='initiative'

...
Query: select * from campaign where name='End of Season Sale.' and storeent_id=10051
Query: select * from bzrpent where path='
/UserTemplate/Store10051/Campaign/Campaign10101/Initiative/Initiative10001_working'
Query: select * from bzrpentstg where bzrpent_id=10009 order by sequence
Query: select * from intvmpe where initiative_id=10001
Query: select * from mpe where mpe_id=10051
Query: select * from emspot where name='StoreHomePage' and storeent_id = 10051
MigrateCampaigns [Fatal Error] :240:2019: Attribute name "instan" must be followed by the
' = ' character.
Attribute name "instan" must be followed by the ' = ' character.
Campaign 10101: failed to migrate initiative 10001 ...
possible error in the rule xml parsing, please check the log for details.
Execute update keys set counter=10050 where tablename='initiative'
Execute update keys set counter=10000 where tablename='intvsched'

*** INITIATIVE migrated and committed ***
```

**解決法:** データベース内のデータ・マイグレーション・ログおよびキャンペーン・データを分析します。たとえば、BZRPENTSTG テーブルに壊れたデータを見つけた場合、以下のような `select` を発行することができます。

```
select * from bzrpentstg where bzrpent_id=10009
```

この `select` が下記のような結果になった場合、

```
.....
10006 <value>com.ibm.commerce.marketingcenter.object.rule.SelectAttributeWithValue</value>
.....
</value>
</item>
</instance-list>
<instan //this element key word has been cut off,
//so migration for campaign initiative has failed.
// The following is next record

10007 ce ref='Class'>
<value>com.ibm.commerce.marketingcenter.object.
CatalogEntryDistinctiveRecommendationActionEndMark</value>
</instance>

.....
```

Commerce Suite 5.1 データベースのテーブル BZRPENTSTG にある壊れたレコードを手動で変更して、`<instance ref='Class'>` スtringが同じレコードにあるようにし、データベース・マイグレーションを再実行します。

---

## ロギングとトレースの使用可能化

WebSphere Commerce 5.5 のロギング・サブシステムは WebSphere Application Server のロギング・インフラストラクチャーを使用します。これによりシステム全体でログの一貫性が得られるので管理が単純になり、また WebSphere Application Server に付属するツールを利用できるので問題判別能力が向上します。ログ・ファ

イルは WebSphere Commerce アプリケーション・インフラストラクチャーと WebSphere Application Server で共用されるため、この 2 つの間のログ・レコードの相関は自動的に得られます。

Commerce Suite 5.1 のログを直接使用するツールを作成した場合は、WebSphere Application Server のロギング・インフラストラクチャーを使うようにそのツールを変更する必要があります。WebSphere Commerce のスタンドアロン・アプリケーション (構成マネージャーや dbclean ユーティリティなど) の場合、ログおよびトレースは、新しい WebSphere Application Server JRas 形式ではなく以前のロギング形式を使用します。

詳細については、WebSphere Application Server バージョン 5.0 Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>) で、使用可能なロギング・インフラストラクチャーによる問題の診断と修正に関するセクションを参照してください。

## ロギング

以降のセクションではロギングについて説明します。

### • ロギングの重大度

Commerce Suite 5.1 では、ERROR、INFORMATION、DEBUG、STATUS、WARNING の 5 種類の重大度があります。一方、WebSphere Application Server のロギング機能では、ERROR、INFORMATION、および WARNING の 3 種類のロギングがあります。以下の表では、WebSphere Application Server のログ・タイプとのマッピングを示します。

表 10. ログの重大度タイプのマッピング

| Commerce Suite 5.1 のロギングでの重大度タイプ | WebSphere Commerce 5.5 のロギング (WebSphere Application Server の JRas 拡張機能を使用) での重大度タイプ |
|----------------------------------|-------------------------------------------------------------------------------------|
| ERROR または ERR                    | TYPE_ERROR または TYPE_ERR                                                             |
| INFORMATION または INFO             | TYPE_INFORMATION または TYPE_INFO                                                      |
| DEBUG                            | TYPE_INFORMATION または TYPE_INFO                                                      |
| WARNING または WARN                 | TYPE_WARNING または TYPE_WARN                                                          |
| STATUS                           | TYPE_INFORMATION または TYPE_INFO                                                      |

ERROR、INFORMATION、および WARNING タイプは変更されず、WebSphere Application Server の JRas 拡張機能の対応するタイプにマップされることに注意してください。STATUS と DEBUG は、INFORMATION タイプとしてログに記録されます。

### • 出力ファイルの場所

デフォルトでは、出力ログ・ファイルの名前は activity.log であり、`WAS50_installdir/logs` ディレクトリーに置かれます。ファイル activity.log はバイナリー・フォーマットであるため、このファイルを読むには WebSphere Application Server ログ・アナライザーを使用する必要があります。ログ・アナライザーの使用については、後で説明します。

いずれかのトレース・コンポーネントが使用可能になっていると、WebSphere Application Server JRes はログ・エントリを、トレース・エントリとともにプレーン・テキスト形式でトレース出力ファイルに書き込みます。

- ログの構成

ログを構成すると、ログ重大度を使用可能または使用不可にしたり、ファイル名を変更したりすることができます。WebSphere Application Server 管理コンソールからログを構成するには、以下のようにします。

1. 「**トラブルシューティング (Troubleshooting)**」 → 「**ログとトレース (Logs and Trace)**」 → 「**WC\_instance\_name**」 → 「**IBM 保守ログ (IBM Service Logs)**」 の順にクリックします。
2. 「**IBM 保守ログ (IBM Service Logs)**」をクリックします。「構成」パネルで「**メッセージ選別 (Message Filtering)**」をクリックしてログ重大度を使用可能にします。サーバーを始動する前に、「ファイル名」フィールドでデフォルトのログ出力ファイルを変更することができます。

**注:** WebSphere Application Server 管理コンソールから WC\_instance\_name サーバーを動的に構成するには、WebSphere Application Server ネットワーク・デプロイメントをインストールしておく必要があります。インストールされていない場合は、WC\_instance\_name サーバーを構成後に再始動する必要があります。

## ログ・アナライザーの使用

ログ・アナライザーは、WebSphere Application Server 5.0 のインストール時にデフォルトでインストールされます。ログ・アナライザーを起動して使用するには、以下のようにします。

- 「**スタート**」 → 「**プログラム**」 → 「**IBM WebSphere**」 → 「**Application Server v5.0**」 → 「**ログ・アナライザー**」の順にクリックします。
- ランタイム・ログのトラブルシューティングにログ・アナライザーを使用するには、以下のようにします。
  1. 分析するログ・ファイルを開きます。

「ログ・アナライザー」ウィンドウで、「**ファイル**」 → 「**オープン**」の順に選択して、分析するログ・ファイルを開きます。WebSphere Commerce と WebSphere Application Server のログは、WAS50\_installdir/logs にあります。
  2. (オプション) 左側のパネルのメッセージの順序を選択するには、以下のようにします。

「ログ・アナライザー」ウィンドウで、「**ファイル**」 → 「**設定 (Preferences)**」の順に選択し、左側のパネルの「**ログ (Logs)**」を強調表示して、右側のパネルの「**ソートの順序 (Sorting Sequence)**」テーブルで、左側のパネルにメッセージを表示する順序を選択します。
  3. メッセージを分析するには、以下のようにします。
    - a. タイム・スタンプを拡張表示して、分析するメッセージを表示します。
    - b. メッセージを強調表示して右マウス・ボタンでクリックし、「**分析 (Analyze)**」を選択します。結果は右側のパネルのテーブルに表示されます。



## トレース

以降のセクションではトレースについて説明します。

### • トレースの構成

トレース・エントリーは、リング・バッファと呼ばれるメモリー内の循環バッファに保管されます。トレースを表示するには、リング・バッファをファイルにダンプする必要があります。WebSphere Application Server 管理コンソールの GUI で、リング・バッファのサイズを設定したり、リング・バッファをダンプしたり、出力ファイル名を指定してトレース・エントリーをファイルに送ったりすることができます。デフォルトでは、トレース出力ファイル、`trace.log` は、`WAS50_installdir/logs/WC_instance_name/` に置かれます。

WebSphere Application Server 管理コンソールからトレースを構成するには、以下のようにします。

1. 「**トラブルシューティング (Troubleshooting)**」 → 「**ログとトレース (Logs and Trace)**」 → 「`WC_instance_name`」 → 「**診断トレース (Diagnostic Trace)**」 の順にクリックします。
2. 「**IBM 保守ログ (IBM Service Logs)**」 をクリックします。サーバーを始動する前に「構成」パネルを使用します。トレースの仕様を直接入力することができます。たとえば、以下のようにします。

```
com.ibm.websphere.commerce.WC_SERVER=all=
enabled:com.ibm.websphere.commerce.WC_RAS=all=enabled
```

(なお、上記の行が 2 行に分かれているのは記載スペース上の理由によります。実際は 1 行にする必要があります。)

トレースの出力形式と出力ファイル名を指定することもできます。

3. WebSphere Application Server ネットワーク・デプロイメントがインストールされていない場合は、上記のステップで示したトレース・ストリングを手動で入力してください。WebSphere Application Server ネットワーク・デプロイメントがインストールされている場合は、WebSphere Application Server の GUI を使って、以下のように「構成とランタイム (Configuration and Runtime)」タブでトレースを構成することができます。
  - a. 「**変更**」ボタンをクリックして、グラフィカル・トレース・インターフェースを使用するトレース・コンポーネントを使用可能にします。
  - b. `com.ibm.websphere.commerce` が表示されるまで拡張表示します。
  - c. 使用可能にするトレース・コンポーネントを右マウス・ボタンでクリックして、「**すべて**」を選択します。

なお、トレース・コンポーネントは、「**適用**」ボタンをクリックすることにより、サーバーを再始動しなくても動的に使用可能にすることができます。

### • トレース・コンポーネント

表 11. トレース・コンポーネントのマッピング

| Commerce Suite 5.1 の |                                                    |
|----------------------|----------------------------------------------------|
| トレース・コンポーネント         | WebSphere Application Server の JRas 拡張機能トレース・ロガー   |
| SERVER               | <code>com.ibm.websphere.commerce.WC_SERVER</code>  |
| CATALOG              | <code>com.ibm.websphere.commerce.WC_CATALOG</code> |

表 II. トレース・コンポーネントのマッピング (続き)

| Commerce Suite 5.1 の<br>トレース・<br>コンポーネント | WebSphere Application Server の<br>JRas 拡張機能トレース・ロガー |
|------------------------------------------|-----------------------------------------------------|
| DATASOURCE                               | com.ibm.websphere.commerce.WC_DATASOURCE            |
| ORDER                                    | com.ibm.websphere.commerce.WC_ORDER                 |
| USER                                     | com.ibm.websphere.commerce.WC_USER                  |
| COMMAND                                  | com.ibm.websphere.commerce.WC_COMMAND               |
| CF                                       | com.ibm.websphere.commerce.WC_CF                    |
| NEGOTIATION                              | com.ibm.websphere.commerce.WC_NEGOTIATION           |
| RAS                                      | com.ibm.websphere.commerce.WC_RAS                   |
| DB                                       | com.ibm.websphere.commerce.WC_DB                    |
| METAPHOR                                 | com.ibm.websphere.commerce.WC_METAPHOR              |
| SCHEDULER                                | com.ibm.websphere.commerce.WC_SCHEDULER             |
| DEVTOOLS                                 | com.ibm.websphere.commerce.WC_DEVTOOLS              |
| TOOLSFRAMEWORK                           | com.ibm.websphere.commerce.WC_TOOLSFRAMEWORK        |
| RULESYSTEM                               | com.ibm.websphere.commerce.WC_RULESYSTEM            |
| MERCHANDISING                            | com.ibm.websphere.commerce.WC_MERCHANDISING         |
| MARKETING                                | com.ibm.websphere.commerce.WC_MARKETING             |
| REPORTING                                | com.ibm.websphere.commerce.WC_REPORTING             |
| TRANSPORT_ADAPTER                        | com.ibm.websphere.commerce.WC_TRANSPORT_ADAPTER     |
| SVT                                      | com.ibm.websphere.commerce.WC_SVT                   |
| PERFMONITOR                              | com.ibm.websphere.commerce.WC_PERFMONITOR           |
| MESSAGING                                | com.ibm.websphere.commerce.WC_MESSAGING             |
| STOREOPERATIONS                          | com.ibm.websphere.commerce.WC_STOREOPERATIONS       |
| CACHE                                    | com.ibm.websphere.commerce.WC_CACHE                 |
| EVENT                                    | com.ibm.websphere.commerce.WC_EVENT                 |
| EJB                                      | com.ibm.websphere.commerce.WC_EJB                   |
| CURRENCY                                 | com.ibm.websphere.commerce.WC_CURRENCY              |
| CATALOGTOOL                              | com.ibm.websphere.commerce.WC_CATALOGTOOL           |
| CONTRACT                                 | com.ibm.websphere.commerce.WC_CONTRACT              |
| UBF                                      | com.ibm.websphere.commerce.WC_UBF                   |
| BI                                       | com.ibm.websphere.commerce.WC_BI                    |
| INVENTORY                                | com.ibm.websphere.commerce.WC_INVENTORY             |
| UTF                                      | com.ibm.websphere.commerce.WC_UTF                   |
| RFQ                                      | com.ibm.websphere.commerce.WC_BI                    |
| EXCHANGE                                 | com.ibm.websphere.commerce.WC_INVENTORY             |
| ACCESSCONTROL                            | com.ibm.websphere.commerce.WC_ACCESSCONTROL         |
| AC_UNITTEST                              | com.ibm.websphere.commerce.WC_AC_UNITTEST           |
| APPROVAL                                 | com.ibm.websphere.commerce.WC_APPROVAL              |
| COLLABORATION                            | com.ibm.websphere.commerce.WC_COLLABORATION         |
| THREAD                                   | com.ibm.websphere.commerce.WC_THREAD                |



---

## 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 以外の製品とプログラムの操作またはサービス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。

本文書中において IBM プログラム・プロダクトについて言及している場合、当該 IBM プログラム・プロダクト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。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-0032  
東京都港区六本木 3-2-31  
IBM World Trade Asia Corporation  
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。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更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の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証できません。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します。

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。

- AIX
- CrossWorlds
- DB2
- IBM
- iSeries
- Lotus
- Net.Commerce
- QuickPlace
- Sametime
- SecureWay
- WebSphere

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

SET および SET ロゴは、SET Secure Electronic Transaction LLC の商標です。

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

UNIX は、The Open Group の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等はそれぞれ各社の商標です。









Printed in Japan