

IBM WebSphere Commerce



## Payments プログラミング・ガイドとリファレンス

バージョン 5.5



IBM WebSphere Commerce



## Payments プログラミング・ガイドとリファレンス

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、165ページの『付録 D. 特記事項』に記載されている情報をお読みください。

本書の内容は、IBM WebSphere Commerce Payments バージョン 5.5、および新版で特に指定のない限り、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。製品のレベルに応じた適切な版を使用していることをご確認ください。

RSA Data Security, Inc. のセキュリティー・ソフトウェアが含まれています。

Copyright © 1994 RSA Data Security, Inc. All rights reserved.

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  
Payments Programming Guide and Reference  
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 2003.8

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 1997, 2003. All rights reserved.

© Copyright IBM Japan 2003

# 目次

本書について . . . . .	v
本書の表記規則 . . . . .	v
本書で使用する用語 . . . . .	vi
追加情報 . . . . .	vii
WebSphere Commerce . . . . .	vii
WebSphere Application Server . . . . .	ix
DB2 Universal Database . . . . .	ix

## 第 1 部 概要 . . . . . 1

### 第 1 章 WebSphere Commerce Payments の概念 . . . . . 3

WebSphere Commerce Payments の用語を理解する . . . . .	3
リリース 5.5 の新しい点 . . . . .	5

## 第 2 部 プログラミング・ガイド . . . . . 9

### 第 2 章 WebSphere Commerce Payments コマンド . . . . . 11

WebSphere Commerce Payments 要求 . . . . .	11
HTTP 本体 . . . . .	12
文字セットの問題 . . . . .	13
通信 . . . . .	13
WebSphere Commerce Payments 応答 . . . . .	14
コマンドのフォーマット設定 . . . . .	14
WebSphere Commerce Payments コマンドのセキュリティ	
ティー . . . . .	16
ユーザー . . . . .	16
役割ベースのアクセス制御 . . . . .	16
役割の許可に関する表 . . . . .	18

### 第 3 章 キャッシャー . . . . . 21

キャッシャーの概要 . . . . .	21
キャッシャー・プロファイル . . . . .	22
統合システムの設計 . . . . .	23
キャッシャー・プロファイルの管理 . . . . .	23
マーチャント番号のマッピング . . . . .	23
オーダー番号のマッピング . . . . .	24
プロファイルの設計 . . . . .	24
AVS . . . . .	26
トレース . . . . .	26
エラー・ログ . . . . .	26
キャッシャー・プロファイルの作成 . . . . .	26
プロファイルの基本構造 . . . . .	27
WebSphere Commerce Payments プロファイル中の	
構成 . . . . .	27
選択ステートメント . . . . .	28
CollectPayment . . . . .	28
Command . . . . .	28
購入ページの情報 . . . . .	29

Parameter . . . . .	29
統合システムの作成 . . . . .	30
プロファイルの構築 . . . . .	31
必要なファイルの組み込み . . . . .	32
キャッシャー・オブジェクトの作成 . . . . .	33
CollectPayment . . . . .	33
WebSphere Commerce Payments でのオーダーの作	
成 - issueCommand() . . . . .	34
オーダーの状況の検査 - checkPayment() . . . . .	34
BuyPageInformation の使用 . . . . .	35
トレース . . . . .	35
例外 . . . . .	35
拡張の作成 . . . . .	36
SampleCheckout アプリケーション . . . . .	36
概要 . . . . .	36
要件 . . . . .	37
構成 . . . . .	38
SampleCheckout プロファイル . . . . .	38

### 第 4 章 クライアント API ライブラリー (CAL) . . . . . 41

CAL 共用クラス . . . . .	41
PaymentServerClient の作成 . . . . .	42
CAL 使用時の SSL サポートのために iSeries を準	
備する . . . . .	43
WebSphere Commerce Payments コマンドの送出 . . . . .	44
HTTP ヘッダーで追加情報を指定する . . . . .	45
WebSphere Commerce Payments からの応答の処理 . . . . .	45
戻されるオブジェクトの処理 . . . . .	46
PaymentServerClient をクローズする . . . . .	46
サンプル CAL プログラム . . . . .	46
CAL が必要とするファイルのインストール . . . . .	47
まだ WebSphere Commerce Payments をインストー	
ルしていないマシンの場合 . . . . .	48

### 第 5 章 イベント通知 . . . . . 49

イベントのタイプと内容 . . . . .	49
状態変更イベント . . . . .	50
カセット固有イベント . . . . .	51
ネットワーク管理イベント . . . . .	51
イベントの登録 . . . . .	51
Event ListenerURL パラメーター . . . . .	52

## 第 3 部 プログラミング・リファレンス . . . . . 53

### 第 6 章 WebSphere Commerce Payments コマンド・リファレンス . . . . . 55

照会コマンド . . . . .	55
About . . . . .	57

AcceptPayment . . . . .	59
AmountExp10 キーワードの使用法 . . . . .	59
Approve . . . . .	61
ApproveReversal . . . . .	61
BatchClose . . . . .	62
BatchOpen . . . . .	63
BatchPurge . . . . .	64
CancelOrder . . . . .	65
CassetteControl . . . . .	65
CloseOrder . . . . .	66
CreateAccount . . . . .	67
CreateMerchant . . . . .	69
CreateMerchantCassetteObject . . . . .	70
CreateMerEventListener . . . . .	71
CreatePaySystem . . . . .	71
CreateSNMEventListener . . . . .	72
CreateSystemCassetteObject . . . . .	73
DeleteAccount . . . . .	74
DeleteBatch . . . . .	74
DeleteMerchant . . . . .	75
DeleteMerchantCassetteObject . . . . .	76
DeleteMerEventListener . . . . .	76
DeletePaySystem . . . . .	77
DeleteSNMEventListener . . . . .	78
DeleteSystemCassetteObject . . . . .	78
Deposit . . . . .	79
DepositReversal . . . . .	80
ModifyAccount . . . . .	80
ModifyCassette . . . . .	82
ModifyMerchant . . . . .	83
ModifyMerchantCassetteObject . . . . .	84
ModifyMerEventListener . . . . .	85
ModifyPayServer . . . . .	85
ModifyPaySystem . . . . .	86
ModifySNMEventListener . . . . .	87
ModifySystemCassetteObject . . . . .	87
ModifyUserStatus . . . . .	88
QueryAccounts . . . . .	89
QueryBatches . . . . .	90
QueryCassette . . . . .	92
QueryCredits . . . . .	93
QueryEventListeners . . . . .	94
QueryMerchants . . . . .	95
QueryOrders . . . . .	96
QueryPayments . . . . .	100
QueryPaymentServer . . . . .	101
QueryPaySystems . . . . .	101
QueryUsers . . . . .	102
オプション・パラメーター . . . . .	102
パラメーターの有効な組み合わせ . . . . .	103
アクセス制御の詳細 . . . . .	105

ReceivePayment . . . . .	105
Refund . . . . .	107
RefundReversal . . . . .	108
SetUserAccessRights . . . . .	109
マーチャント管理者のアクセス制御規則 . . . . .	110

## 第 7 章 WebSphere Commerce

### Payments データ . . . . . 111

WebSphere Commerce Payments 決済オブジェクト . . . . .	111
Order . . . . .	112
Order の状態 . . . . .	114
Payment . . . . .	115
Payment の状態 . . . . .	117
スプリット・ペイメント . . . . .	118
AVS 共通コード . . . . .	118
Credit . . . . .	119
Credit の状態 . . . . .	120
Batch . . . . .	120
Batch の状態 . . . . .	122

### WebSphere Commerce Payments About オブジェクト . . . . . 122

Payment Server About . . . . .	122
Cassette About . . . . .	122

### WebSphere Commerce Payments 管理オブジェクト . . . . . 123

Payment Server . . . . .	123
Cassette . . . . .	124
Merchant . . . . .	125
Payment System . . . . .	125
Account . . . . .	126
Event Listener . . . . .	127
User . . . . .	128

## 第 4 部 付録 . . . . . 129

### 付録 A. WebSphere Commerce

#### Payments 戻りコード . . . . . 131

1 次戻りコード . . . . .	131
2 次戻りコード (汎用) . . . . .	134

#### 付録 B. ISO 通貨コード . . . . . 151

#### 付録 C. Request For Comments

#### (RFC) の入手方法 . . . . . 163

#### 付録 D. 特記事項 . . . . . 165

商標 . . . . .	166
--------------	-----

#### 用語集 . . . . . 167

#### 索引 . . . . . 175

---

## 本書について

本書は、WebSphere Commerce の WebSphere® Commerce Payments コンポーネントと通信および対話するアプリケーションを開発するプログラマーを対象としています。WebSphere Commerce Payments とともに使用する決済カセットを開発するプログラマーもまた、本書を参考にすることができます。


**注:** IBM® WebSphere Commerce Payments for Multiplatforms (これ以降、WebSphere Commerce Payments といいます) は、以前は IBM WebSphere Payment Manager for Multiplatforms と呼ばれていました。バージョン 3.1.3 以降は決済アプリケーションの名前が WebSphere Commerce Payments に変更され、本書での製品の呼称が変更されました。以前の製品の呼称が本書に残っている場合もありますが、これは、以前のバージョンの製品を指します。

---


## 本書の表記規則


本書では、以下の書体の規則を使用します。


- **太文字**は、フィールド名、アイコン名、またはメニュー選択項目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、またはコマンドを示します。
- **モノスペース (Monospace)** は、そのとおりに入力すべきテキストの例、ファイル名、およびディレクトリー・パスと名前です。
- **イタリック** は、その言葉を強調します。さらに、実際のシステムに合わせて適切な値に置換すべき名前もまた、イタリックによって表されます。以下のような名前は、それぞれ説明に従って実際のシステム値に置換してください。


 **Windows** は、Windows® オペレーティング環境に特有の情報を示します。

 **AIX** は、AIX® に特有の情報を示します。

 **Solaris** は、Solaris オペレーティング環境に特有の情報を示します。

 **400** は、IBM iSeries™ 400 (これまでは AS/400® といいました) に特有の情報を示します。

 **Linux** は、Intel® ワークステーション上の Linux と、IBM eServer iSeries、pSeries™、zSeries™、および S/390® の各システムの Linux を指します。

 **UNIX** は、UNIX® プラットフォーム (AIX、Solaris、Linux) に特有の情報を示します。

*WC\_installdir* は、以下のような WebSphere Commerce デフォルト・インストール・パスです。

 **AIX** /usr/lpp/WebSphere/CommerceServernn

Linux Solaris /opt/WebSphere/CommerceServernn

Windows drive:¥WebSphere¥CommerceServernn

400 /QIBM/ProdData/CommerceServernn

`Payments_installdir` は、以下のような Payment Server デフォルト・インストール・パスです。

AIX /usr/lpp/WebSphere/CommerceServernn/payments

Linux Solaris /opt/WebSphere/CommerceServernn/payments

Windows drive:¥WebSphere¥CommerceServernn¥payments

400 /QIBM/ProdData/CommercePayments/Vnn

---

## 本書で使用する用語

本書で使用している決済カセット、マーチャント・サーバー、および *Payment Gateway* といった用語は、なじみが薄いかもしれません。本書や、WebSphere Commerce Payments のその他の資料で使用されている用語の定義については、本書付録の用語集を参照してください。また、WebSphere Commerce のオンライン・ヘルプにも用語の説明が含まれています。

WebSphere Commerce Payments の資料で使用されている次の用語は、WebSphere Commerce のオンライン・ヘルプや出版物で使用されている他の用語に類似のものです。

### ストア とマーチャント

WebSphere Commerce においてストアの語は、オンライン・ストアのことを指して使用されます。オンライン・ストアでは、インターネットのさまざまなテクノロジーを使用することにより、商品を販売したりサービスを提供したりします。WebSphere Commerce Payments の場合、ストアはマーチャントと同じことです。たとえば、本書でマーチャントの設定、あるいはマーチャントの追加という個所があれば、ストアの設定やストア追加のことを指しています。

### サイト管理者 と Payments 管理者

サイト管理者は WebSphere Commerce において、WebSphere Commerce とそれに伴うソフトウェアおよびハードウェアのインストール、構成作業、および保守を担当するものとして定義された役割です。通常、この役割はアクセス権や許可を制御し、管理作業を実行するための最高の権限が与えられます。

同じように、WebSphere Commerce の決済コンポーネントにおいて、Payments 管理者には、決済機能を実行するための最高の権限が与えられます。サイト管理者は Payments 管理者タスクを実行できますが、Payments 管理者はすべてのサイト管理者タスクを実行できるわけではありません。



---

## 追加情報

WebSphere Commerce および決済コンポーネントに関するさらに詳しい情報は、さまざまな形式でさまざまな情報源から入手できます。

### WebSphere Commerce

WebSphere Commerce 情報の入手先:

- オンライン・ヘルプ
- PDF ファイル
- Web サイト

#### オンライン・ヘルプの使用

WebSphere Commerce オンライン情報には、WebSphere Commerce のカスタマイズ、管理、および再構成についての情報が含まれています。

WebSphere Commerce Payments のオンライン・ヘルプには、決済コンポーネントに付随するグラフィカル・ユーザー・インターフェースの使用法に関する情報が含まれています。Payments のオンライン・ヘルプを表示するには、ユーザー・インターフェース・パネルの右上隅にある疑問符のアイコンをクリックします。

#### 印刷可能なドキュメンテーションの入手方法

WebSphere Commerce のオンライン情報の一部は、システム上で PDF ファイルの形式でも入手できます。それを表示したり印刷したりするには、Adobe Acrobat Reader を使用します。さらに、WebSphere Commerce Payments の資料も PDF ファイルで提供されています。Acrobat Reader は、Adobe Web サイトから無料でダウンロードできます。その Web アドレスは以下のとおりです。

<http://www.adobe.com>

PDF ファイルには、WebSphere Commerce のオンライン・ヘルプから、また WebSphere Commerce の製品情報に関する Web サイトからアクセスできます。

#### WebSphere Commerce 製品情報 Web サイトの閲覧

WebSphere Commerce 製品情報は、WebSphere Commerce テクニカル・ライブラリー Web サイト (<http://www.ibm.com/software/commerce/wscom/library/lit-tech.html>) でご覧いただけます。

本書のコピー、および本書の改訂バージョンが、この Web サイトから PDF ファイルとして入手できます。

#### WebSphere Commerce Payments のその他の資料および Web サイト

以下の資料は、WebSphere Commerce の決済コンポーネントに関する情報です。

- 「*WebSphere Commerce* インストール・ガイド」は、WebSphere Commerce Payments をプラットフォームにインストールおよび構成する方法を説明しています。

- 「*WebSphere Commerce 管理ガイド*」には、概念についての説明、および構成マネージャー・ユーザー・インターフェースを使って *WebSphere Commerce Payments* を構成する方法が示されています。
- 「*WebSphere Commerce Payments OfflineCard Cassette 補足*」には、支払い情報の記録に使われる決済カセットについての情報が示されています。マーチャントは *WebSphere Commerce Payments* ユーザー・インターフェースを使用して、記録された支払い情報を後で手動で処理することができます。
- 「*WebSphere Commerce Payments CustomOffline Cassette 補足*」は、手動決済トランザクション (代金引換払い、後払い、その他のマーチャント定義による方法) に関する情報を管理するために使用できる決済カセットについて説明しています。
- 「*WebSphere Commerce Payments Cassette for VisaNet Supplement*」には、*WebSphere Commerce Payments* を使用して *VisaNet* システムにアクセスする方法が説明されています。インストールと構成に関する情報も含まれています。
- 「*WebSphere Commerce Payments Cassette for BankServACH Supplement*」には、*WebSphere Commerce Payments* を使用して、*BankServ* ゲートウェイを経由し、*Automated Clearing House (ACH)* ネットワークにアクセスする方法が説明されています。インストールと構成に関する情報も含まれています。
- 「*WebSphere Commerce Payments Cassette for Paymentech Supplement*」には、*WebSphere Commerce Payments* を使用して、ニューハンプシャー州にある *Paymentech Salem* (クレジット・カードおよびデビット・カードのトランザクションを扱う処理センター) にアクセスする方法が説明されています。インストールと構成に関する情報も含まれています。

資料はすべて、PDF 形式で提供されています。

*WebSphere Commerce Payments* の詳細については、以下の Web サイトを参照してください。

- <http://www.ibm.com/software/webservers/commerce/payment/> には、*Commerce* 決済処理ソフトウェアについての詳細情報が掲載されています。その中には、*WebSphere Commerce Payments* とともに使用できる決済カセットについての情報も含まれます。
- <http://www.ibm.com/software/webservers/commerce/payments/support.html> には、現在の *WebSphere Commerce Payments* 技術情報と、最新の *WebSphere Commerce Payments* 資料へのリンクが掲載されています。
- <http://www.ibm.com/software/webservers/commerce/payment/paymentcards.html> には、*WebSphere Commerce Payments* カセットの開発に関する情報が掲載されています。

*Commerce* のサポート情報およびダウンロード情報は、以下の Web サイトでご覧いただけます。

- <http://www.ibm.com/software/commerce/wscom/support/index.html>
- <http://www.ibm.com/software/commerce/wscom/downloads/index.html>

## **WebSphere Application Server**

WebSphere Application Server についての情報は WebSphere Application Server Web サイト (<http://www.ibm.com/software/webservers/appserv>) でご覧いただけます。

## **DB2 Universal Database**

DB2 Universal Database についての情報は、Web サイト <http://www.ibm.com/software/data/db2/udb> でご覧いただけます。



---

## 第 1 部 概要



---

## 第 1 章 WebSphere Commerce Payments の概念

WebSphere Commerce Payments は、プロトコル固有のカセット を使用して、複数の異なる決済方法をサポートできる汎用フレームワークを提供します。マーチャントでは、決済および管理コマンドを使用してオーダーを処理することができます。WebSphere Commerce Payments は、汎用コマンドをプロトコル固有の決済要求に変換し、それを決済ゲートウェイまたはセキュア Web サーバーなどの適切な宛先に転送します。WebSphere Commerce Payments はその取引をリレーショナル・データベースに記録します。

すべての WebSphere Commerce Payments の統合により、下層の決済カセットによって記述されたとおりに、オーダー作成呼び出しが発行されます。多くのマーチャント・システムの場合はこれで十分であり、他のすべてのタスクは WebSphere Commerce Payments インターフェースを介して行われます。付加的な WebSphere Commerce Payments 金融コマンドをその他の既存のビジネス・フォーマットに、より緊密に組み込むことを望むマーチャントは、Approve、Deposit、および BatchClose などの付加的なコマンドを送出することができます。

---

## WebSphere Commerce Payments の用語を理解する

本書では、以下の用語および概念が使用されています。本書、および他の WebSphere Commerce Payments 資料で使われているこのほかの用語の定義については、本書に付属の用語集を参照してください。これらの用語は、WebSphere Commerce オンライン・ヘルプでも説明されています。

**バッチ** 一緒に清算される決済とクレジットの集合。

### バイヤー

マーチャントからインターネットで購入を行う人物。

### キャッシュャー

コードを変更しなくても、マーチャント・ソフトウェアが、新規のカセットを最大限に活用できるようにするコンポーネント。キャッシュャーはカセットごとに決済オプション・プロファイルを使用して、必要なカセット固有パラメーター、およびマーチャント・ソフトウェア環境からその情報を収集するメソッドを記述します。

### カセット

WebSphere Commerce Payments フレームワークにプラグインして、特定の電子決済システムをサポートするソフトウェア・パッケージ。カセットは、IBM 製でも他社製でも構いません。例としては、VisaNet、BankServACH、および Paymentech 用の IBM 製カセットが取り上げられています。

### クレジット

クレジットは、マーチャントが銀行に、バイヤーに対して金銭をリファンドするように指示する場合の、マーチャントと銀行と間の対話を表します。

## イベント・リスナー

重要なイベントの発生時やオブジェクト状態の変更時に通知を受け取れることを望む、WebSphere Commerce Payments に登録されている者。

## フレームワーク

さまざまな決済システムを使用するさまざまなマーチャント・サーバーが、WebSphere Commerce Payments に対して同じ汎用コマンドを送出して同じ汎用データを使用することを可能にする、WebSphere Commerce Payments の一部。WebSphere Commerce Payments は、プロトコル固有のカセットを使用して、汎用呼び出しをプロトコル固有のメッセージに変換します。

## マーチャント

インターネット・ショッピング・プレゼンスによるビジネス。マーチャントは、WebSphere Commerce Payments をマーチャント・ソフトウェアと統合させます。

**注:** この用語は、WebSphere Commerce のストア と似ています。

WebSphere Commerce Payments では、ストアはマーチャント と同義です。たとえば、本書でマーチャントの設定、あるいはマーチャントの追加という個所があれば、ストアの設定やストア追加のことを指しています。

## マーチャント・ソフトウェア

WebSphere Commerce Payments を使用してインターネットでの決済の処理と管理を行う、マーチャント・インターネット・ビジネスをサポートするソフトウェア。通常このソフトウェアには、WebSphere Commerce Payments の他に、カタログのブラウズ、ショッピング・カートの管理、および発注を行うための、Web ベースのソフトウェアが組み込まれています。マーチャントのビジネスの組み込みのレベルによって、在庫管理、出荷、および会計ソフトウェアも組み込まれることがあります。

## Payments 管理者

WebSphere Commerce において、**サイト管理者** は WebSphere Commerce で定義された役割であり、WebSphere Commerce および関連するソフトウェアとハードウェアのインストール、構成、保守を行います。通常、この役割はアクセス権や許可を制御し、管理作業を実行するための最高の権限が与えられます。

WebSphere Commerce の決済コンポーネントでは、**Payments 管理者** には Payment 機能を実行する際の最高の権限が与えられます。サイト管理者は Payments 管理者タスクを実行できますが、Payments 管理者はすべてのサイト管理者タスクを実行できるわけではありません。

## オーダー

WebSphere Commerce Payments のオーダーとは、1 つの決済メソッドを使って 1 つ以上の決済を行うために、バイヤーから取得する認証です。

**決済** 決済は、オーダーの全部または一部を承認して取り込むための、マーチャントと金融機関との間の 1 つの対話を表します。金銭はバイヤーからマーチャントに移動します。



**レلمム** ユーザーおよびそれらのユーザーを認証する単一メソッドのレジストリー。ユーザーは、リソースへのアクセスを認可される前に、レلمムで定義されていなければなりません。

---

## リリース 5.5 の新しい点

以前に WebSphere Commerce Payments バージョン 2.2 (またはそれ以降) にインストールされたすべてのカセット (IBM 提供または他社提供) は、WebSphere Commerce Payments バージョン 5.5 を正常にインストールした後も引き続き機能するはずですが。

WebSphere Commerce Payments をインストールする前に、ご使用のプラットフォームに関する「*WebSphere Commerce* インストール・ガイド」を参照してください。

### ディレクトリー・ファイル構造の変更点

WebSphere Commerce Payments のディレクトリー・ファイル構造が一部変更されました。本書にはこれらの変更点が示され、一部のファイル名も変更されています。たとえば、以下のとおりです。

- `etillCal.zip` パッケージの名前が `eTillCal.jar` に変わりました。
- `Payments_installdir/include` サブディレクトリーの位置は、`Payments_installdir/wc.mpf.ear/Payments.war/include` に変更されました。
- `instances` サブディレクトリーを除いて、`iSeries` のディレクトリー構造はワークステーション・プラットフォームの構造と一致するようになりました。
- `SampleCheckout` アプリケーションは、WebSphere Commerce Payments EAR ファイル内部の独立した WAR ファイルに格納されています。このため、このアプリケーションにアクセスするには、`host_name:port/webapp/PaymentManager/SampleCheckout` ではなく、`host_name:port/webapp/SampleCheckout` を指定する必要があります。

### インストールと構成に関する変更点

WebSphere Commerce Payments 専用のインストール・プログラムはもはや存在しません。WebSphere Commerce のコンポーネントの 1 つとして、WebSphere Commerce インストール・プログラムを介してインストールされます (WebSphere Commerce のインストールについては、「*WebSphere Commerce* インストール・ガイド」を参照)。インストールの後、WebSphere Commerce 構成マネージャーを使って Payments インスタンスを構成する必要があります。

構成マネージャーを使用すると、決済コンポーネントのインスタンスを含む WebSphere Commerce インスタンスを構成および管理することができます。構成マネージャーによって、Payments インスタンスの作成、更新、削除のほか、インスタンスの開始と停止、インスタンス・パスワードの変更、インスタンス用カセットの追加と除去が可能です。インスタンスの作成に関する詳細は、「*WebSphere Commerce* インストール・ガイド」を参照してください。「*WebSphere Commerce* 管理ガイド」には、WebSphere Commerce で構成タスクを行う方法について追加情報が掲載されています。

## IBM 提供のカセット

Cassette for SET™ および Cassette for CyberCash はもはやサポートされません。WebSphere Commerce Payments に付属のカセットは以下のとおりです。

- OfflineCard Cassette
- CustomOffline Cassette
- Cassette for BankServACH
- Cassette for Paymentech
- Cassette for VisaNet

## デフォルト・ポートの除去

WebSphere Commerce Payments 用に指定されるデフォルト・ポート (以前は 80) は存在しなくなりました。ポートは、WebSphere Commerce 構成マネージャーを介して指定されます。

## メッセージ機能とトレース機能の変更点

WebSphere Commerce Payments では、システム・メッセージやトレース出力を生成するために、これまでの独自の機能に代わって WebSphere Application Server のメッセージ機能およびトレース機能を使用するようになりました。この変更によって問題判別データがより整合性のある形で提供され、WebSphere 環境でデータを収集および理解するのが容易になりました。

- メッセージの変更点は以下のとおりです。

メッセージは、WebSphere Application Server 管理コンソールで Log Analyzer を使って表示できます。Log Analyzer を使用すれば、`WAS_installdir/logs/instancename_Commerce_Payments_Server` ディレクトリ内の `activity.log` ファイルに含まれているメッセージを表示できます。以前は、デフォルトで Payments ログ・パス・ディレクトリー (`Payments_installdir/logs`) 内の `PMError` ファイルにメッセージが書き込まれました。

加えて、問題判別を支援する WebSphere Commerce 症状データベースも使用できます。WebSphere Log Analyzer を使用すれば、(Payments メッセージを含む) Commerce システム・メッセージの詳細情報を表示し、メッセージに関する説明、および推奨されるユーザー応答アクションを見ることができます。Log Analyzer で WebSphere Commerce ログを扱う方法、および症状データベースについては、「*WebSphere Commerce 管理ガイド*」で説明されています。さらに、Log Analyzer の詳細情報については WebSphere Application Server InfoCenter も参照してください。

- トレースの変更点は以下のとおりです。

トレースを使用可能化するためにこれまで使用していたトレース・パネルは、WebSphere Commerce Payments グラフィカル・ユーザー・インターフェースに表示されなくなりました。

トレース・テキストの書き込み先ファイルを制御するには、WebSphere Application Server トレース・サービスを使用して、トレース・データの出力先を定義します (Payments ログ・ディレクトリー内の `PMTrace1.log` および `PMTrace2.log` ファイル以外を指定できます)。PMTrace ログ・ファイルはサポートされなくなりました。

## ModifyPayServer コマンドおよび ModifyCassette コマンドの変更点

WebSphere Commerce Payments は新たに WebSphere のメッセージ機能とトレース機能を利用するようになったため、API コマンド ModifyPayServer および ModifyCassette が変更されました。Payment Server (PSPaymentServer) オブジェクトは、属性 logPath、traceFileSize、および traceSetting をサポートしなくなりました。

決済コンポーネントのグローバル・プロパティを変更する ModifyPayServer コマンドは、オプションのキーワード LOGPATH、TRACEFILESIZE、および TRACESETTING をもはやサポートしません。

カセットのプロパティを変更する ModifyCassette コマンドは、オプションの TRACESETTING キーワードをもはやサポートしません。

加えて、エラー・ログ・パスおよびエラー・ログに関連した汎用 2 次戻りコード 508、614 が除去されました。2 次戻りコードは 134 ページの『2 次戻りコード (汎用)』に記載されています。

WebSphere JRas メッセージおよび診断トレース機能とトレース・コンポーネントについて、および WebSphere Commerce Payments のトレース・レベルを設定する方法については、「*WebSphere Commerce 管理ガイド*」を参照してください。



## 第 2 部 プログラミング・ガイド

WebSphere Commerce Payments にはいくつかのプログラミング・インターフェースがあり、それによってご使用のシステムにこの製品を統合することができます。以下の図はこれらのインターフェースを示しています。

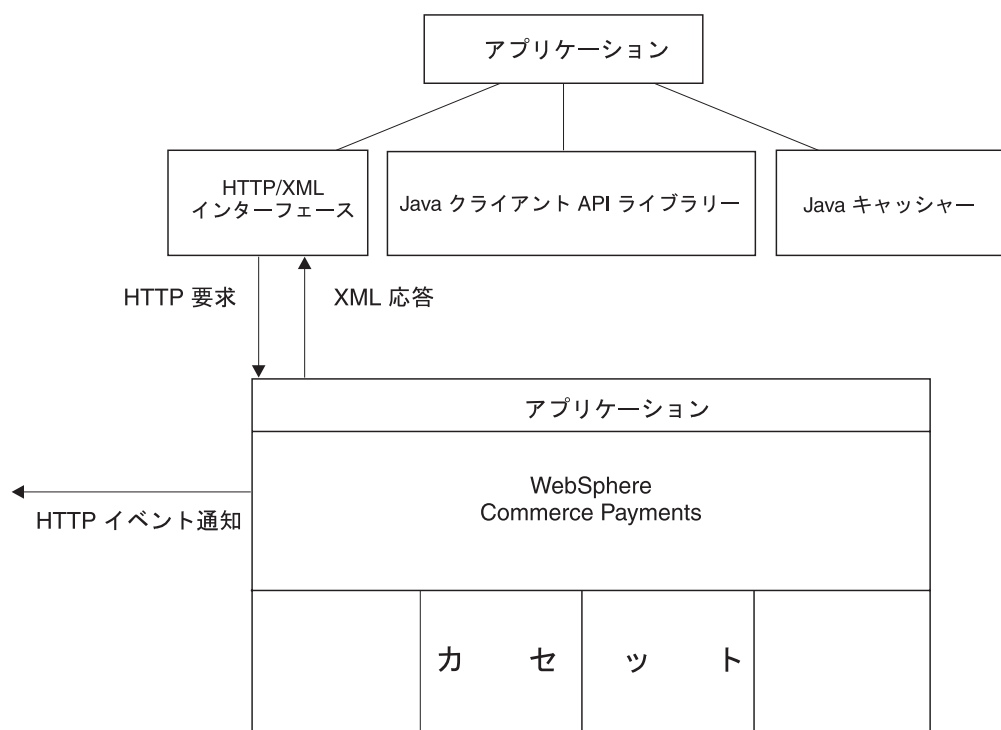


図 1. プログラミング・インターフェース

WebSphere Commerce Payments の中心的な概念は、複数の決済システムを管理するフレームワークを備えながらも、ユーザーには単一のインターフェースを示すということです。WebSphere Commerce Payments は、決済カセットの概念を導入していますが、これは単一の決済システムをサポートする 1 つのプラグイン・ソフトウェアです。WebSphere Commerce Payments は、着信決済要求を関連するカセットに送り、応答は可能な限り決済システムに依存しないものになります。したがって、既存の統合ソフトウェアをほとんどあるいはまったく妨げることなく、新規のカセットをシステムに追加することができます。この資料は、WebSphere Commerce Payments とそのカセットに統合するアプリケーションによって使用できるプログラミング・インターフェースについて説明しています。カセット・プログラミング・インターフェースにより、開発者は WebSphere Commerce Payments 用の新しいカセットを作成することができます。このインターフェースについては、<http://www.ibm.com/software/webservers/commerce/payments/download.html>にある IBM WebSphere Commerce Payments Cassette Kit 情報で詳しく説明されています。

WebSphere Commerce Payments の主なプログラミング・インターフェースは、HTTP および XML 標準に基づくものです。WebSphere Commerce Payments は、コマンドを HTTP POST 要求として受け入れ、HTTP 応答に XML 文書を組み込んで戻します。すべての決済処理機能と、ほぼすべての管理機能に対するコマンドがあります。インターフェースは HTTP および XML 標準を使用しているため、WebSphere Commerce Payments コマンドは、さまざまなプログラム言語から呼び出すことが可能です。11 ページの『第 2 章 WebSphere Commerce Payments コマンド』では、これらの要求および応答を作成する方法を示し、WebSphere Commerce Payments コマンドのすべてのセットと、その必須およびオプション・パラメータをリストしています。重要なこととして、使用するそれぞれのカセットの補足資料を参照して、各要求に指定できる追加のキーワードと、各応答に提供される追加のカセット XML データを理解する必要があることに注意してください。

Java™ クライアント API ライブラリー (CAL) が WebSphere Commerce Payments には備えられており、これによって Java ソフトウェアと WebSphere Commerce Payments とを簡単に統合することができます。CAL を使用して、Java 要求を作成して、Java 応答オブジェクトを処理することができます。CAL は HTTP 要求の作成を扱い、その中の XML 応答を構文解析します。クライアント API ライブラリーについては、41 ページの『第 4 章 クライアント API ライブラリー (CAL)』で説明しています。

WebSphere Commerce Payments オーダーを作成する際に、決済カセット固有の情報を提供する必要があります。その情報に基づいて決済が処理されます。カセット固有のデータの例として、クレジットカード番号、小切手番号、保証人 ID、および有効期限があります。オーダー作成を処理するために作成するコードが、特定のカセットだけをサポートするようにハードコーディングされている場合、新しいカセットをシステムに追加するときに再コード化する必要があります。この問題を避けるために、WebSphere Commerce Payments にはキャッシュと呼ばれる Java ベースのライブラリーの機能が備えられています。キャッシュはプロファイル (XML ドキュメント) を使用して、オーダー作成用にカセットで必要とされるすべてのパラメータを記述します。そうすれば、キャッシュを使用して WebSphere Commerce Payments オーダーを作成する場合に、特定のカセットに固有のオーダー作成コードを作成する必要がなくなります。キャッシュについては、21 ページの『第 3 章 キャッシュ』で説明しています。

WebSphere Commerce Payments は、あるイベントが発生したときにアラートを出す、イベント通知メカニズムを提供しています。サポートされているイベント・トリガーとして、WebSphere Commerce Payments とそのカセットの開始または停止、特定のマーチャントまたは特別なイベント (特定のカセットによって定義されている) に属するオーダーの状態に変更があります。WebSphere Commerce Payments に、実行したいイベントを通知することができます。イベントが起動されると、WebSphere Commerce Payments は HTTP POST メッセージを作成して、それを指定の URL に送ります。イベント通知を処理するには、サーブレットまたは CGI プログラム (これをイベント・リスナーという) を作成する必要があります。イベント通知メカニズムについては、49 ページの『第 5 章 イベント通知』で説明しています。

---

## 第 2 章 WebSphere Commerce Payments コマンド

マーチャント・ビジネス・ソフトウェアは、WebSphere Commerce Payments に対して管理コマンド、決済コマンド、および照会コマンドを送出できます。これらのコマンドは、キーワードと値の対で構成されます。(コマンドの表については、55 ページの『第 6 章 WebSphere Commerce Payments コマンド・リファレンス』を参照。) コマンドの実行は、要求を発行して応答を待つという方式で行われます。WebSphere Commerce Payments 要求は HTTP POST メッセージの形式になります。WebSphere Commerce Payments 応答には、HTTP 形式の XML 文書が組み込まれます。(XML オブジェクトと、関連するフィールドに関する詳細な説明については、111 ページの『第 7 章 WebSphere Commerce Payments データ』を参照してください。) この章では、HTTP POST、WebSphere Commerce Payments との通信、および XML 出力について説明します。

---

### WebSphere Commerce Payments 要求

HTTP POST メッセージが作成されて WebSphere Commerce Payments に送信されることにより、マーチャント・ソフトウェアから WebSphere Commerce Payments にコマンドが送られます。HTTP POST メッセージと同様に、コマンドもヘッダーと本体で構成されます。以下に WebSphere Commerce Payments コマンドの例を示します。

```
POST /webapp/PaymentManager/PaymentServlet HTTP/1.1
Connection: Keep-Alive
Accept: application/xml
PM-Accept-Language: en-US
Authorization: Basic YWRtaW46YWRtaW4=
Host: localhost
User-Agent: Java PaymentServerClient
Content-Encoding: 8859_1
Content-Length: 187
Content-Type: application/x-www-form-urlencoded

OPERATION=ACCEPTPAYMENT&ETAPIVERSION=3&PAYMENTTYPE=OfflineCard
&MERCHANTNUMBER=123456789&ORDERNUMBER=91600886&AMOUNT=500
&CURRENCY=840&%24EXPIRY=200212&%24PAN=501555000033019&%24BRAND=ROBO
```

**注:** 上記の例で HTTP 本体を改行しているのは、印刷や表示の形式を考慮したためです。構文は 1 行で構成する必要があります。

WebSphere Commerce Payments コマンドでは、ヘッダーは、複数の指定されたキーワードと値の対が、特定の形式でエンコードされている必要があります。HTTP ヘッダーには、これらの値が指定された、以下のフィールドがなければなりません。

```
POST /webapp/PaymentManager/PaymentServlet HTTP/1.1
Connection: Keep-Alive
Accept: application/xml
Content-Encoding: 8859_1
Content-Type: application/x-www-form-urlencoded
```

さらに、ヘッダーには、計算値が指定された、以下の追加フィールドがなければなりません。

**Host: <PaymentServer host>**

これは WebSphere Commerce Payments の TCP/IP ホスト名にする必要があります。

**Content-Length: <length>**

HTTP 本体の長さ (バイト単位)

**Authorization: Basic <authorization-string>**

与信ストリングは、1 つのコロン文字 (「:」) で区切られたユーザー ID とパスワードのストリングで構成されます。ストリングは base64 エンコードでエンコードされている必要があります。

```
authorization-string=base64-user-pass
base64-user-pass=<base64encoding of user-password,
except not limited to 76 char/line>
user-pass=user:"password
userid=*<TEXT excluding ":">
password=*TEXT
```

オプションで、HTTP ヘッダーに PM-Accept-Language という HTTP ヘッダーを含めることができます。このヘッダーは、WebSphere Commerce Payments が応答メッセージで戻りコード・メッセージを示すために使う言語を指示します。HTTP RFC 2068 に従って、PM-Accept-Language: Locales を指定する必要があります。WebSphere Commerce Payments がサポートするロケールには、次のものがあります。**pt** (ブラジル・ポルトガル語)、**en** (英語)、**fr** (フランス語)、**de** (ドイツ語)、**it** (イタリア語)、**ja** (日本語)、**ko** (韓国語)、**zh** (中国語 (簡体字))、**es** (スペイン語)、**zh\_TW** (中国語 (繁体字))。複数のロケールを PM-Accept-Language HTTP ヘッダーに指定できますが、WebSphere Commerce Payments では最初に指定されているロケールだけを使用することに注意してください。PM-Accept-Language ヘッダーが送信されないと、WebSphere Commerce Payments をインストールしたマシンのロケールが WebSphere Commerce Payments で使用されます。

クライアントまたはマーチャントのプログラマーは、追加のヘッダー・フィールドを指定して、WebSphere Commerce Payments 通信に最小限必要な機能以外の HTTP 機能を使用することもできます。これらのフィールドの解釈は、WebSphere Commerce Payments がインストールされているネットワーク環境と Web サーバーによって異なります。

---

## HTTP 本体

WebSphere Commerce Payments コマンドの本体は、キーワードと値の対の集合から成り、HTTP コンテンツ・タイプ application/x-www-form-urlencoded で指定されたエンコードを使用してフォーマット設定されます。

キーワードは複数回組み込むことができます (たとえば、1 つのオーダー照会コマンドに複数のオーダー番号を指定できます)。

コマンド本体の形式は、以下の規則に従っていなければなりません。

- WebSphere Commerce Payments コマンドのパラメーターとその関連引き数 (WebSphere Commerce Payments コマンドのキーワードと値の対) は、等号 (=) で区切る。



- キーワードと値の対と、他のキーワードと値の対とはアンパーサンド (&) で区切る。
- キーワードと値は URL エンコードされる。これはバイナリー・データを WebSphere Commerce Payments に送信する際の方式でもあります。URL エンコードの規則は以下のとおりです。
  - スペース文字 (16 進数 0X20 ASCII 文字) はすべて「+」文字 (16 進数 0x2B 文字) に置き換えられる。
  - 個々のキーワードと値のバイトのうち、英数字 US-ASCII 文字にマップされないものはすべてエスケープされなければならない。該当する個々のバイトは、エスケープ・シーケンス「%HH」(HH は文字 (バイト) の ASCII コードを表す 2 つの 16 進数字) に置き換えられます。
- キーワードには大文字小文字の区別はありません。値には大文字小文字の区別があります。

## 文字セットの問題

WebSphere Commerce Payments キーワードはすべて US-ASCII 文字セットで指定されます。HTTP POST 本体を URL エンコードする前に、値を UTF-8 文字セットでエンコードしなければなりません。たとえば、Unicode 文字 0x3053 は、UTF-8 では 0xE3、0x81、0x93 と表されます。この値が URL エンコードされると、%E3%81%93 になります。

注: US-ASCII スtringや数値の場合は、変換は必要ありません。

---

## 通信

コマンドを WebSphere Commerce Payments に送るには、以下のようにします。

1. WebSphere Commerce Payments ホストおよびポートとの TCP 接続をオープンします。ポートは、WebSphere Commerce 構成マネージャーを使って構成されません。
2. 要求を送信して、応答を待ちます。
3. 接続をクローズします。

応答を受信しないうちに通信が失敗した場合は、WebSphere Commerce Payments コマンドが実際に実行されたかどうかは分かりません。コマンドが実行されたかどうかを判別するには、照会コマンドを送出して、コマンドが受信されて処理されているかを確認してください。

SSL を使用したい場合は、SSL 接続をサポートするように WebSphere Commerce Payments 上の Web サーバーを構成してください。Web サーバーを SSL 用に構成すると、SSL を使用してコマンドを送信できます。SSL 通信を作成するための、すべてのステップを実行できる状態になっていなければなりません。

## WebSphere Commerce Payments 応答

WebSphere Commerce Payments 応答には、HTTP 形式の XML 文書が組み込まれます。XML 文書の形式は、WebSphere Commerce Payments 文書タイプ定義 (DTD) によって定義されます。IBMPaymentServer.dtd に DTD が含まれており、このファイルは `Payments_installdir/wc.mpf.ear/Payments.war/include` サブディレクトリにあります。

すべての HTTP 応答には XML 文書が含まれており、これには `PSApiResponse` エレメントがあります。このエレメントは、1 次戻りコードと 2 次戻りコードおよびオブジェクト・カウントと追加の戻りコード・メッセージを示します。戻りコード・メッセージには、WebSphere Commerce Payments の 1 次および 2 次戻りコードの対の説明が含まれている場合があります。1 次戻りコードと 2 次戻りコードの値に関する説明は、131 ページの『付録 A. WebSphere Commerce Payments 戻りコード』を参照してください。

カセット固有のオブジェクトは、`Cassette` オブジェクトと `Cassette Configuration` エレメントを使用して表されます。個々の属性に関する詳細は、それぞれのカセットの補足に記述されています。(詳細については、カセットの補足資料を参照してください。)

照会コマンドは追加情報として、WebSphere Commerce Payments オブジェクトに関する説明と、戻されるオブジェクトの数を示します。フレームワーク・オブジェクトは、DTD (文書タイプ定義) 中と、111 ページの『第 7 章 WebSphere Commerce Payments データ』のオブジェクト定義表で説明されています。この XML 文書の DTD は以下のどちらかになります。

- 応答に組み込まれる。
- ファイル `IBMPaymentServer.dtd` 内にある。

WebSphere Commerce Payments で要求の受信、処理、および応答が正常に行われると、HTTP 状況コード 200 が戻されます。認証が失敗したり、WebSphere Application Server が稼働していなかったりしたために、他の HTTP 状況コードが Web サーバーによって戻されることもあります。この状況コードとその本体中の情報は、問題の発生源を示します。

## コマンドのフォーマット設定

`AcceptPayment` コマンドおよび `QueryOrders with Payments` コマンドを実行した場合の XML 文書の例をそれぞれ以下に示します。

### AcceptPayment

```
POST /webapp/PaymentManager/PaymentServlet HTTP/1.1
Connection: Keep-Alive
Accept: application/xml
PM-Accept-Language: en-US
Authorization: Basic YWRtaW46YWRtaW4=
Host: localhost
User-Agent: Java PaymentServerClient
Content-Encoding: 8859_1
Content-Length: 187
Content-Type: application/x-www-form-urlencoded
```

```
OPERATION=ACCEPTPAYMENT&ETAVERSION=3&PAYMENTTYPE=OfflineCard&MERCHANTNUMBER=
```

```
123456789&ORDERNUMBER=94184938&AMOUNT=500&CURRENCY=840&%24EXPIRY=
200212&%24PAN=501555000033019&%24BRAND=ROBO
```

```
-----
<?xml version="1.0" encoding="UTF-8"?>
<PSApiResult objectCount="0" primaryRC="0" secondaryRC="0">
</PSApiResult>
```

## QueryOrders with Payments

以下の例は、QueryOrder with Payment コマンドに対する応答です。応答文書には 2 つの Order オブジェクトが含まれています。

- 1 つ目の Order オブジェクトには決済が 1 つ含まれています。
- 2 つ目の Order オブジェクトには決済が含まれていません。

```
POST /webapp/PaymentManager/PaymentServlet HTTP/1.1
Connection: Keep-Alive
Accept: application/xml
PM-Accept-Language: en-US
Authorization: Basic YWRtaW46YWRtaW4=
Host: localhost
User-Agent: Java PaymentServerClient
Content-Encoding: 8859_1
Content-Length: 100
Content-Type: application/x-www-form-urlencoded
```

```
OPERATION=QUERYORDERS&ETAVERSION=3&PAYMENTTYPE=OfflineCard&MERCHANTNUMBER=
123456789&WITHPAYMENTS=1
```

```
-----
<?xml version="1.0" encoding="UTF-8"?>
<PSApiResult objectCount="2" primaryRC="0" secondaryRC="0">
  <OrderCollection size="2" withCredits="0" withPayments="1">
    <PSOrder ID="O:123456789:94184938" amount="500" amountExp10="-2"
      approvesAllowed="1" brand="ROBO" currency="840" merchantAccount="1"
      merchantNumber="123456789" merchantOriginated="1" numberOfCredits="0"
      numberOfPayments="1" orderNumber="94184938" paymentType="OfflineCard"
      state="order_refundable" timeStampCreated="966461827000"
      timeStampModified="966463091000" unapprovedAmount="0">
      <PaymentCollection size="1" withOrders="0">
        <PSPayment ID="P:123456789:94184938:1" amountExp10="-2"
          approveAmount="500" currency="840" depositAmount="0" merchantAccount="1"
          merchantNumber="123456789" orderNumber="94184938" paymentNumber="1"
          paymentType="OfflineCard" state="payment_approved"
          timeStampCreated="966463091000" timeStampModified="966463092000">
          <CassetteExtensionObject>
            </CassetteExtensionObject>
          </PSPayment>
        </PaymentCollection>
      </PSOrder>
    <CassetteExtensionObject>
      <CassetteProperty propertyId="Expiry" value="200212">
        </CassetteProperty>
      <CassetteProperty propertyId="AccountNumber" value="1">
        </CassetteProperty>
      <CassetteProperty propertyId="Brand" value="ROBO">
        </CassetteProperty>
      <CassetteProperty propertyId="AmountApproved" value="500">
        </CassetteProperty>
      <CassetteProperty propertyId="Pan" value="501555000033019">
        </CassetteProperty>
      </CassetteExtensionObject>
    </PSOrder>
    <PSOrder ID="O:123456789:92005267" amount="500" amountExp10="-2"
      approvesAllowed="1" brand="ROBO" currency="840" merchantAccount="1"
      merchantNumber="123456789" merchantOriginated="1" numberOfCredits="0"
      numberOfPayments="0" orderNumber="92005267" paymentType="OfflineCard"
      state="order_refundable" timeStampCreated="966459650000"
      timeStampModified="966459650000" unapprovedAmount="500">
```

```

<PaymentCollection size="0" withOrders="0">
</PaymentCollection>
<CassetteExtensionObject>
  <CassetteProperty propertyId="Expiry" value="200212">
  </CassetteProperty>
  <CassetteProperty propertyId="AccountNumber" value="1">
  </CassetteProperty>
  <CassetteProperty propertyId="Brand" value="ROBO">
  </CassetteProperty>
  <CassetteProperty propertyId="AmountApproved" value="0">
  </CassetteProperty>
  <CassetteProperty propertyId="Pan" value="501555000033019">
  </CassetteProperty>
</CassetteExtensionObject>
</PSOrder>
</OrderCollection>
</PSApiResponse>

```

## WebSphere Commerce Payments コマンドのセキュリティー

ユーザーが送出したコマンドを WebSphere Commerce Payments が受信すると、そのコマンドは以下のように処理されます。

- レルムによってユーザーを認証する。ユーザーは WCSRealm において定義されます。
- アクセス制御機能によってユーザーを許可する。
- コマンドを処理する。

以下の節では、コマンドのセキュリティーに関連した概念を説明します。

### ユーザー

認証は、レルムを使って行われます。レルムとは、ユーザーのレジストリーの一種で、ユーザー名、パスワード、およびその他のユーザー識別形式を使用する場合はそれも管理します。

WebSphere Commerce Payments リソースに対するアクセスを認可する場合は、その前に WCSRealm 内で WebSphere Commerce Payments ユーザーを定義しなければなりません。Payments 管理者およびマーチャント管理者は、WebSphere Commerce Payments API コマンドまたは Payments ユーザ・インタフェースの「ユーザー」ウィンドウを使用して、レルムに定義されたユーザーにアクセス権を割り当てることができます。

### 役割ベースのアクセス制御

WebSphere Commerce Payments は役割ベースのアクセス制御方式を採用し、Payments 管理者、マーチャント管理者、スーパーバイザー、およびクラークの 4 つの WebSphere Commerce Payments 役割を定義しています。これらの役割を、表 1 の役割を持つ WebSphere Commerce ユーザーに割り当ててを推奨します。

表 1. 推奨される役割の割り当て

Payments の役割	WebSphere Commerce の役割
Payments 管理者	サイト管理者
マーチャント管理者	サイト管理者

表 1. 推奨される役割の割り当て (続き)

Payments の役割	WebSphere Commerce の役割
スーパーバイザー	オペレーション・マネージャー、セールス・マネージャー
クラーク	顧客サービス・スーパーバイザー

ユーザーの役割によって、そのユーザーがどのコマンドを送出できるかが決まります。

Payments 管理者以外のユーザーを、複数のマーチャントに関連付けることができます。たとえば、1 人のマーチャント管理者は複数のマーチャントを管理できます。また、スーパーバイザーとクラークは、複数のマーチャントに対するコマンドを送出できます。 WebSphere Commerce Payments は、以下のような役割ベースのアクセスのシナリオをサポートしています。

- Payments 管理者は、すべてのマーチャントに対するすべての API コマンドを送出できる。
- マーチャント管理者は、自分と関連付けられているマーチャントに関する機能をすべて実行できる。ただし、SetUserAccessRights コマンドと QueryUsers コマンドには多少の制限事項がある (これらのコマンドについて詳しくは、102 ページの『QueryUsers』および 109 ページの『SetUserAccessRights』を参照してください)。
- スーパーバイザーとクラークは、自分と関連付けられているマーチャントに対する一部のコマンドを送出できる。

## ユーザーのアクセス許可の割り当て

ユーザーの許可 (または役割) を割り当てたり変更したりできるのは、Payments 管理者またはマーチャント管理者だけです。Payments 管理者は、任意のユーザーのアクセス権限を割り当てまたは変更することができ、(Payments 管理者役割を含めた) 任意のユーザー役割を割り当てまたは変更することができます。

マーチャント管理者は、ユーザーをマーチャント管理者、スーパーバイザー、またはクラークとして割り当てたり除去したりすることだけを行えます。また、以下のいずれかの条件が満たされている場合に限り、マーチャント管理者はその操作を行えます。

1. 現在 WebSphere Commerce Payments でアクセス権を持っていないユーザーに、複数のマーチャントに対するアクセス権を付与する場合には、マーチャント管理者は、自分がすでに (マーチャント管理者として) アクセス権を持っているマーチャントに対するアクセス権だけ を付与できます。
2. 現在 WebSphere Commerce Payments でアクセス権を持っているユーザーに、複数のマーチャントに対するアクセス権を認可する場合には、これらのユーザーに現在関連付けられているマーチャントを、割り当てを行うマーチャント管理者にも関連付ける必要があります。また、これらのユーザーに割り当てられ関連付けられるマーチャントは、割り当てを行うマーチャント管理者に関連付けられているマーチャントのサブセットでなければなりません。

たとえば、ユーザー X はマーチャント A、B および C のマーチャント管理者で、ユーザー Y は WebSphere Commerce Payments 中でアクセス権を持っていないとし

ます。X は Y をマーチャント A、B および C のマーチャント管理者として割り当てたり、マーチャント A および B のマーチャント管理者として割り当てたりすることができます。しかし、ユーザー Y が A、B および C 以外のマーチャントに対するアクセス権を持っている場合 (たとえば、ユーザー Y がマーチャント D のマーチャント管理者の場合) は、ユーザー X はユーザー Y のアクセス権を変更できません。

以下の図は、集合の形式で代表的なシナリオの一部を表しています。

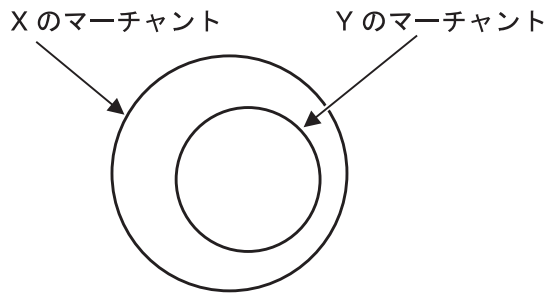


図2. アクセス権を示す例

図2 では、ユーザー X が、X のあるマーチャントに対するアクセス権限をユーザー Y に付与しています。

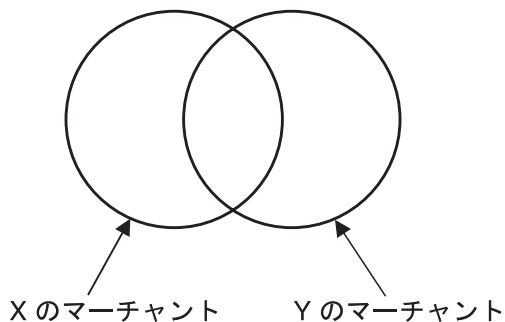


図3. 共通のマーチャントに関連付けられたユーザーを示す例

図3 では、X と Y は別々のマーチャントの集合に関連付けられていますが、両方ともいくつかの共通のマーチャントにも関連付けられています。この場合、X と Y が両者ともそれらの共通のマーチャントのマーチャント管理者であったとしても、互いに他方の許可を変更することはできません。

## 役割の許可に関する表

個々の役割には、その役割を持つユーザーが実行できる一連の操作と関連があります。個々の役割で行えることを記述する際に使用する表記を、表2 に示します。

表2. 役割で行えることを表記するフィールド値

フィールドの値	役割で行えること
Y	コマンドを実行できます。

表 2. 役割で行えることを表記するフィールド値 (続き)

フィールドの値	役割で行えること
M	ユーザーのマーチャント番号にコマンドのマーチャント番号がすべて含まれている場合に、コマンドを実行できます。コマンドにマーチャント番号が指定されていない場合は、認証は失敗します。
u	コマンドを試行しているユーザーが、そのコマンドで指定されているユーザーである場合に、コマンドを実行できます。コマンドにユーザーが指定されていない場合は、認証は失敗します。
a	SETUserAccessRights 特殊認証論理で許可されている場合に、コマンドを実行できます。SETUserAccessRights コマンドの詳細については、109 ページの『SetUserAccessRights』を参照してください。
m	QueryUsers 特殊認証論理で許可されている場合に、コマンドを実行できます。QueryUsers コマンドの詳細については、102 ページの『QueryUsers』を参照してください。
<ブランク>	この役割のユーザーはコマンドを実行できません。

以下の表は、個々の役割で行えることを示しています。コマンドの後のアスタリスク (\*) は、そのコマンドには必須のマーチャント番号パラメーターがないことを示しています。

表 3. 役割で行えること

コマンド	Payments 管理者	マーチャント 管理者	スーパー バイザー	クラーク
About*	Y	Y	Y	Y
AcceptPayment	Y	M	M	M
Approve	Y	M	M	M
ApproveReversal	Y	M	M	M
BatchClose	Y	M	M	M
BatchOpen	Y	M	M	M
BatchPurge	Y	M	M	M
CancelOrder	Y	M	M	
CassetteControl	Y	M	M	M
CloseOrder	Y	M	M	
CreateAccount	Y	M		
CreateMerchant	Y			
CreateMerchantCassetteObject*	Y	M		
CreateMerEventListener	Y	M		
CreatePaySystem	Y			
CreateSNMEventListener	Y			
CreateSystemCassetteObject*	Y			
DeleteAccount	Y	M		

表 3. 役割で行えること (続き)

コマンド	Payments 管理者	マーチャント 管理者	スーパー バイザー	クラーク
DeleteBatch	Y	M	M	M
DeleteMerchant	Y			
DeleteMerchantCassetteObject*	Y	M		
DeleteMerEventListener	Y	M		
DeletePaySystem	Y			
DeleteSNMEventListener	Y			
DeleteSystemCassetteObject*	Y			
Deposit	Y	M	M	M
DepositReversal	Y	M	M	
ModifyAccount	Y	M		
ModifyCassette*	Y			
ModifyMerchant	Y	M		
ModifyMerchantCassetteObject*	Y	M		
ModifyMerEventListener	Y	M		
ModifyPayServer*	Y			
ModifyPaySystem	Y			
ModifySNMEventListener	Y			
ModifySystemCassetteObject*	Y			
ModifyUserStatus	Y	M		
QueryAccounts	Y	M	M	M
QueryBatches	Y	M	M	M
QueryCassettes	Y			
QueryCredits	Y	M	M	M
QueryEventListeners	Y	M		
QueryMerchants	Y	M	M	M
QueryOrders	Y	M	M	M
QueryPayments	Y	M	M	M
QueryPaymentServer	Y			
QueryPaySystems	Y	M	M	M
QueryUsers	Y	m	u	u
ReceivePayment	Y	M	M	M
Refund	Y	M	M	
RefundReversal	Y	M	M	
SetUserAccessRights*	Y	a		

注: ユーザーは自分自身を更新することはできません。つまり、ユーザー "admin" は、ユーザー・パラメーターを "admin" に設定して SETUSERACCESSRIGHTS を呼び出すことはできません。



## 第 3 章 キャッシャー

この章では、WebSphere Commerce Payments キャッシャーとキャッシャー・プロファイルについて説明します。また、キャッシャーを使って WebSphere Commerce Payments を統合する場合の考慮事項も示します。

### キャッシャーの概要

キャッシャーとは、WebSphere Commerce Payments コードの一種で、クライアント・アプリケーション (マーチャント・ソフトウェアなど) によって呼び出して、WebSphere Commerce Payments のオーダーや WebSphere Commerce Payments のコマンドを作成するプロセスを単純化することができます。キャッシャーでは、プロファイルという XML ドキュメントが使用されます。このプロファイルには、特定のカセット用のコマンド (オーダーなど) を作成する方法が記述されます。そのため、クライアント・コードの作成者は、カセット固有の情報を扱うコードを作成する必要がないので、WebSphere Commerce Payments を汎用的な方法で統合することに専念できます。

キャッシャーを使用せずに WebSphere Commerce Payments のオーダーを作成することもできます。その場合プログラムは AcceptPayment と ReceivePayment API を使用することができます。しかし、キャッシャーを使用すると、その後新しいカセットをシステムに導入する際にコードを再作成する必要がなくなるので、キャッシャーの使用をお勧めします。

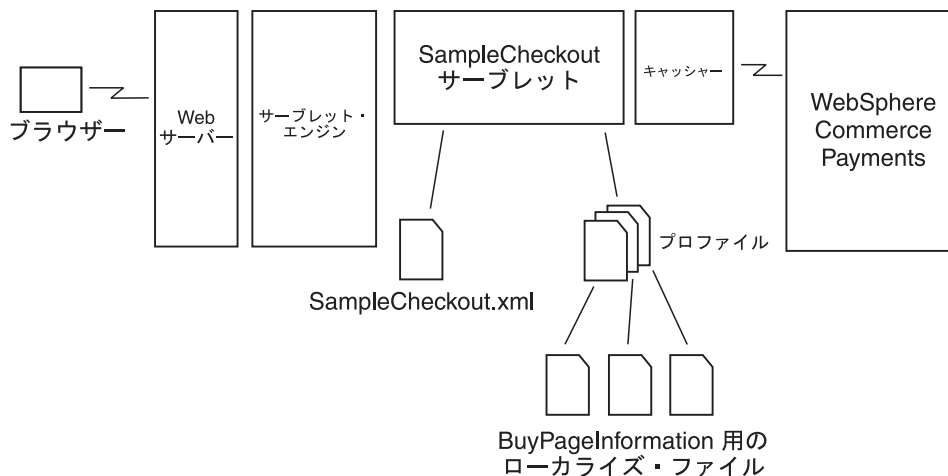


図 4. WebSphere Commerce Payments キャッシャー

キャッシャーは Java で書かれ、eTillCal.jar パッケージに含まれます。キャッシャーには、WebSphere Commerce Payments クライアント・アプリケーションによって直接呼び出せるメソッドのセットが備えられています。キャッシャー自体は、クライアント API ライブラリー (CAL) を使用して、AcceptPayment コマンド、ReceivePayment コマンド、および他のコマンドを WebSphere Commerce Payments に送信します。したがって、キャッシャーは CAL の利点をすべて活用できます。

このコードは、WebSphere Commerce Payments からリモートに操作したり、Socks サーバーを使用するように構成したり、必要に応じて SSL を介してメッセージを暗号化したりすることができます。キャッシュャーが使用するプロファイルは、キャッシュャーが実行中の場所で選択可能でなければなりません。

キャッシュャーの主要なメソッドは `collectPayment()` です。これは、WebSphere Commerce Payments のオーダーを作成するために、クライアント・アプリケーションが呼び出す必要があるメソッドです。 `collectPayment()` は、引き数としてプロファイル名、ロケール、および環境変数のリストを取ります。これはこの名前に対応するプロファイルをロードし、それを使用して `AcceptPayment` または `ReceivePayment` API 要求を構築します。環境変数は、API パラメーター値を指定するために使用されます。

WebSphere Commerce Payments のオーダーや決済の状態を照会するには、`collectPayment()` 呼び出しの後にいつでも `checkPayment()` メソッドを使用することができます。

さらに、`issueCommand()` メソッドを使用して、他の API 要求を WebSphere Commerce Payments 上で構築および発行することができます。現在のところ、`issueCommand()` を通してサポートされている API は `Deposit API` だけです。

---

## キャッシュャー・プロファイル

キャッシュャー・プロファイルとは、WebSphere Commerce Payments のコマンドを作成する方法を記述する XML 文書です。すべてのプロファイルに、以下のものが組み込まれていなければなりません。

- 必須の WebSphere Commerce Payments パラメーター
  - 必須のカセット・パラメーター
  - キャッシュャーが上記のパラメーターに対する値を指定する方法に関する指示
- プロファイルには、以下のものを組み込むこともできます。
- ウォレットを使用するかどうかの指示。このフラグは、キャッシュャーが `AcceptPayment` コマンドと `ReceivePayment` コマンドのどちらを使用するかを決定します。
  - それぞれのプロファイルにどの WebSphere Commerce Payments インスタンスを使用するかに関する指示
  - オプションの WebSphere Commerce Payments パラメーター
  - オプションのカセット・パラメーター
  - 購入ページに関する情報 (クライアント・コードにより購入ページが構築され、バイヤーの情報が収集される方法)。たとえば、特定のカセットが必要とするクレジット・カード情報を収集する HTML フォームなど
  - プロファイルに対して診断情報を使用可能にするかどうかの指示

キャッシュャー・プロファイルでは、以下の 4 つの方法でパラメーター値を指定できます。

1. プロファイル中に定数としてハードコーディングする。
2. `collectPayment()` 呼び出し、または `issueCommand()` 呼び出しに環境変数として渡す。

3. リレーショナル・データベースのフィールドからの値を使用するように指定する。
4. キャッシャー拡張コードで計算するように指定する。

ご使用のシステムが、プロファイルにとって必要なデータに簡単にアクセスできる場合、このデータをキャッシャーに `collectPayment()` 呼び出し、または `issueCommand()` 呼び出しの環境変数で渡すと実적입니다。ご使用のシステムでデータを取得することが難しい場合や、少数のプロファイルでしか必要とされない場合には、このデータを環境変数として渡すのは非効率的です。なぜなら、`collectPayment()` または `issueCommnd()` のすべての呼び出しにおいて、必要であってもなくてもこのデータを導出することになるからです。

---

## 統合システムの設計

キャッシャーを使用して WebSphere Commerce Payments と統合するコードを作成する際の考慮事項について以下に説明します。

### キャッシャー・プロファイルの管理

キャッシャーを使用する前に、ご使用のサイトでどのプロファイルを使用可能にしたいかを判別する必要があります。カセットの作成者は、用途に適したキャッシャー・プロファイルを備える必要があります。これらのプロファイルは、WebSphere Commerce Payments のプロファイル・サブディレクトリーに格納されます。ただし、カセットにプロファイルが備えられていなくても、以下のカセットに関する補足的なガイドと 26 ページの『キャッシャー・プロファイルの作成』の指示に従うと、プロファイルを簡単に作成することができます。

システムが複数のストアまたはマーチャントをサポートしている場合、アクティブに使用中のプロファイルを判別する方法を決めなければなりません。最も簡単な方法としては、すべてのマーチャントまたはストアが、1 つのプロファイルのセットを必ず使用するようにします。しかし、より複雑なシナリオでは、さまざまなマーチャントまたはストアが、さまざまなプロファイルのセットをサポートします。このような場合、これらのマーチャントやストアを、それぞれが使用するプロファイルにマップする処理をサポートしなければなりません。また、マーチャントとプロファイルとのマッピングを示すテーブルを管理するツールを備えることが必要になる場合もあります。

### マーチャント番号のマッピング

マーチャントとは、カセットと関連付けられ、オーダーが置かれるオブジェクトです。マーチャントは、最大 9 桁の数字のマーチャント番号によって識別されます。システム内のそれぞれのマーチャントのエンティティーに対して複数のマーチャント番号を作成できるので、システム中のマーチャントまたはストアのエンティティーを、WebSphere Commerce Payments でのマーチャント番号に対してマップする方法を考慮することが重要です。1 対 1 で対応しており、マーチャントまたはストアに使用する ID を最大 9 桁のストリングとして表せる場合は、WebSphere Commerce Payments のマーチャント番号をシステムに格納する必要はありません。そうでない場合は、WebSphere Commerce Payments マーチャント番号を外部キーとして格納する方法を決定しなければなりません。

## オーダー番号のマッピング

オーダーは、最大 9 桁のオーダー番号によって識別されます。それぞれのオーダー番号はマーチャント番号ごとに固有でなければならないので、理論的には、WebSphere Commerce Payments の 1 つのインスタンスが 999,999,999 のマーチャントを持ち、それぞれのマーチャントが 999,999,999 のオーダーを持つことが可能です。(もちろん現実には、この限度に達する前に、管理能力の限界を上回ってしまいます。)

WebSphere Commerce Payments のオーダーには、システムでのオーダーの使用とは正確には一致しない、固有の定義があります。それぞれの WebSphere Commerce Payments のオーダーはバイヤーからの 1 つの与信ですが、特定の支払いメソッドに対して、決済は 1 回行われる場合もあれば複数回行われる場合もあります。バイヤーが複数の決済方式を使用して 1 つのオーダーを行うと、WebSphere Commerce Payments ではそれを複数のオーダーとして表さなければなりません。一例として、クレジット・カードで頭金を支払い、残りを後から小切手で支払う場合があります。

システム中のオーダーと WebSphere Commerce Payments 中のオーダーとをマップする方法を決めます。必要に応じて、システム中のオーダー当たり 1 つまたは複数の WebSphere Commerce Payments のオーダー番号を格納する必要があります。

## プロファイルの設計

それぞれのプロファイルには WebSphere Commerce Payments およびカセットのパラメーター値を導出する方法を指定したデータが含まれているので、プロファイルは通常は統合システムごとに固有であり、変更せずに別のシステムにコピーすることはできません。この節では、統合システムでプロファイルが動作する方法を設計する際の考慮事項についてリストします。

### WebSphere Commerce Payments の構成

1 つまたは複数の WebSphere Commerce Payments インスタンスを指すようにシステムを構成するには、以下の 2 つの方法があります。

1. それぞれのキャッシャー・プロファイル内で WebSphere Commerce Payments 構成を指定する。
2. アプリケーション内で構成を指定し、すべてのキャッシャー・プロファイルでその構成を使用する。

どちらの方法も、1 つのオーダーは単一の WebSphere Commerce Payments インスタンスで管理されることを理解しておくことが重要です。したがって、オーダーを作成する際には、そのオーダーを所有する WebSphere Commerce Payments インスタンスを記録しなければなりません。プロファイル・ベースの構成を使用する場合は、オーダーと一緒にプロファイル名を格納すると、これを実行することができます。後でそのオーダーに関する決済操作を実行する際には、キャッシャーを照会して、そのオーダーを所有する WebSphere Commerce Payments インスタンスを見つけ、API 要求をそのインスタンスに送信することができます。

### プロファイル・パラメーターのソース

`collectPayment()` または `issueCommand()` のどちらかを使用できるものの、`issueCommand()` を使用することが推奨されていることを覚えておいてください。

キャッシュャーを使用する際には、プロファイルの API パラメーター値の入手先を決めなければなりません。

ご使用のシステムが、プロファイルにとって必要なデータに簡単にアクセスできる場合、このデータをキャッシュャーに `collectPayment()` 呼び出し、`issueCommand()` 呼び出しの環境変数で渡すと实际的です。ご使用のシステムでデータを取得することが難しい場合や、いくつかのプロファイルでしか必要とされない場合は、このデータを環境変数として渡すのは非効率的です。なぜなら、`collectPayment()`、`issueCommand()` へのすべての呼び出しにおいて、必要であってもなくてもこのデータを導出することになるからです。

こうした場合は、キャッシュャーにデータそのものを導出させることができます。リレーショナル・データベース中のデータを使用できる場合は、プロファイルをコーディングして、データベースを照会してデータを入手するようにキャッシュャーに指示できます。または、キャッシュャー拡張コードを作成して、パラメーター値を導出できます。この方法については、30 ページの『統合システムの作成』を参照してください。

## 購入ページの情報

キャッシュャーとプロファイルを使用すると、WebSphere Commerce Payments 統合システムは、システムを記録する必要なく、以降の決済カセットの追加をサポートできるようになります。新しいカセットは、購入ページ上に収集した別の決済情報を必要とします。クレジット・カードのカセットのセット内であっても、バイヤーに提示される購入ページは異なります。たとえば、一部のカセットはアドレス検証サービス (AVS) をサポートしており、その他はサポートしていない場合があります。

キャッシュャー・プロファイル中の情報を使用して購入ページを構築するように統合システムを作成すると、新しいカセットの購入ページを記録する必要がなくなるので、新しいカセットのサポートはかなり容易になります。

プロファイルの購入ページの情報、統合システム的设计によってすべて決まります。これには、フォームを構築してバイヤーに提示するために必要な HTML を含めたり、収集する必要のあるデータについて記述した XML ドキュメントにしたり、データを収集する Java サーバー・ページまたはアクティブ・サーバー・ページを指すポインターにしたりすることができます。確認すべきことは、キャッシュャーがプロファイルのパラメーター定義を使用して WebSphere Commerce Payments API 要求を構築する際に、バイヤーによって入力されたデータを使用できるということです。

## プロファイル・インターフェースの公開

キャッシュャーの主要な利点の 1 つとして、ご使用の統合システムで処理できるプロファイルを他人が作成できるということがあります。キャッシュャーを使用して WebSphere Commerce Payments と統合すると、プログラム・コードに変更を加える必要なしに、関係するキャッシュャー・プロファイルを備えることによって新しいカセットをサポートできます。インターフェースを公開するには、購入ページ情報、パラメーターのソース、およびプロファイルに WebSphere Commerce Payments の構成情報を含める必要があるかどうかの指定を含めます。

## AVS

WebSphere Commerce Payments カセットは、金融機関トランザクションに関するマーチャントに AVS 結果コードを戻します。これらのコードはカセット固有なので (つまり WebSphere Commerce Payments カセットごとに異なる)、WebSphere Commerce Payments は共通 AVS 結果コードのセットを備えることにより、カセット固有のコードを拡張します。共通 AVS 結果コードとカセット固有のコードとのマッピングについては、118 ページの『AVS 共通コード』を参照してください。

## トレース

キャッシュにはトレース機構が備えられており、これを使用すると、問題診断プロセスを単純化して、診断情報を独自のシステム・ログに直接書き込むことができます。この機構では、すべてのトレース情報が 1 つのログに書き込まれるので、複数のログを関連付けるむずかしさを避けることができます。この機構を使用するには、30 ページの『統合システムの作成』の指示に従ってください。ご使用のシステムでこのトレース情報を統合する必要がない場合、キャッシュに備えられている単純なトレース・クラスを使用して、診断情報をフラット・ファイルに直接書き込むことができます。

トレース情報を記録すると、わずかなパフォーマンス・オーバーヘッドが生じます。この理由から、プロファイルごとにトレースを使用可能にしたり使用不可にすることができます。Profile エlement は `enableTrace` 属性をサポートしており、これを使用してトレースを制御できます。

## エラー・ログ

キャッシュにはサービス担当者が問題診断に使用するトレース機能が備えられています。この機能では、ユーザーが参照するエラーは記録されません。その代わりに、エラー条件が検出されると、キャッシュは Java 例外をスローします。これらの例外をキャッチしてユーザーに適切に報告する作業は、ご使用のシステムが行います。

---

## キャッシュ・プロファイルの作成

WebSphere Commerce オンライン・ヘルプには、WebSphere Commerce 用の新しいキャッシュ・プロファイルを作成する方法が説明されています。

キャッシュ・プロファイルを作成するには、以下の 4 つのものがが必要です。

1. キャッシュ・プロファイルの構造に関する知識
2. WebSphere Commerce Payments の必須パラメーターとオプション・パラメーターの指定
3. 必須およびオプションのカセット・パラメーターの指定
4. キャッシュとの統合の指定

カセットの作成者がカセット・プロファイルを備えると、それらのプロファイルは WebSphere Commerce Payments のインストール先のプロファイル・ディレクトリーに格納されます。これらのプロファイルは、他のシステムで処理できるように簡単にコピーして変更することができます。入手できるテンプレート・プロファイルがない場合は、新しいプロファイルを構築しなければなりません。

## プロファイルの基本構造

キャッシュ・プロファイルは、 `profile.dtd` ドキュメント・タイプ定義を実装した XML ドキュメントです。これらのプロファイルには、以下の基本構造が含まれています。

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Profile SYSTEM "profile.dtd">
<Profile useWallet="false" enableTrace="false">
...
</Profile>
```

`useWallet` 属性を真 (`true`) に設定すると、キャッシュは `collectPayment()` のために `ReceivePayment` API 要求を構築します。偽 (`false`) に設定すると、`collectPayment()` のために `AcceptPayment` API 要求を構築します。 `enableTrace` 属性は、キャッシュでこのプロファイルが使用される際に診断情報を記録する必要があるかどうかを指示します。

## WebSphere Commerce Payments プロファイル中の構成

オプションで、プロファイルに以下のような WebSphere Commerce Payments 構成エレメントを含めることもできます。

```
<Profile useWallet="false">
...
  <PaymentManagerConfiguration
    hostname="..."
    port="..."
    userid=""
    password="..."
    useSSL="true"
    socksHostname="..."
    socksPort="..."
    dtdPath="..."
  />
...
</Profile>
```

この情報は、このプロファイルを使用する際の、キャッシュと WebSphere Commerce Payments インスタンスとの間の通信方法を指示します。 `hostname` 属性と `port` 属性は、WebSphere Commerce Payments が API 要求を `listen` するソケットを識別します。 `userid` 属性と `password` 属性は、API 要求を構築する際にキャッシュが想定する ID と認証情報を指定します。 `socksHostname` と `socksPort` はオプション属性で、使用される Socks サーバーがある場合にそのサーバーを指示します。 `useSSL` は、SSL を使用して WebSphere Commerce Payments との通信を暗号化する必要があるかどうかを指示します。オプションの `dtdPath` パラメーターは、WebSphere Commerce Payments DTD のパスを指定します。

**注:** `PaymentManagerConfiguration` エレメントは、WebSphere Commerce によってサポートされません。

## 選択ステートメント

マーチャント統合システムで、リレーショナル・データベースの照会を使用してパラメーターの値を導出することがサポートされている場合は、プロファイルに 1 つまたは複数の `SelectStatement` エレメントを含めることもできます。

```
<Profile useWallet="false">

  <SelectStatement id="..." allowMultiples="...">
    SELECT * FROM ... WHERE ...
  </SelectStatement>

  ...

</Profile>
```

エレメントの内容は、SQL 照会ステートメントを構成します。id 属性はステートメントの ID を指定します。この ID を後続の `DatabaseValue` エレメントで使用すると、このステートメントに戻って参照することができます。繰り返しのデータを戻さない SQL ステートメントを構築する際には、そのステートメントが必ず 1 行だけを戻すようにしてください。そのためには、`allowMultiples` 属性を指定しないか、`allowMultiples="false"` を指定します。このようにすると、キャッシャーは、ゼロ行や複数の行が戻されるとエラーとして報告します。

オプションの `allowMultiples` 属性が `"true"` であるということは、SQL 照会が複数のデータ行を戻す可能性があるということです。この場合、キャッシャーは、選択ステートメントを参照するそれぞれのデータベース・パラメーターごとに、複数の引き数を API 要求の中に作成します。引き数の数は、照会によって戻される行数と同じになります。各引き数は、引き数の末尾にピリオドと増分する整数を追加することによって区別されます。たとえば、`$LINEITEM` という ID のパラメーターが選択ステートメントを参照するとします。そのステートメントの `allowMultiples` が `"true"` に設定されており、SQL 照会が 3 行を戻す場合、キャッシャーは API 要求の中に、`$LINEITEM.1`、`$LINEITEM.2`、および `$LINEITEM.3` という 3 つの引き数を生成します。

## CollectPayment

`CollectPayment` エレメントには、キャッシャーを使用して `WebSphere Commerce Payments` オーダーを作成するのに必要なすべてのデータが含まれています。

```
<Profile useWallet="false">
  <CollectPayment>

  ...

</CollectPayment>
</Profile>
```

## Command

`Command` エレメントには、キャッシャーを使用して `WebSphere Commerce Payments` コマンドを作成するのに必要なすべてのデータが含まれています。このコマンドは、あらゆる `WebSphere Commerce Payments` API 要求の構築と発行に使用できるものの、現時点でサポートされている API は `Deposit` だけです。

```
<Profile useWallet="false">
  <Command name="DEPOSIT">
```



```
...
</Command>
</Profile>
```

## 購入ページの情報

キャッシュが統合されるシステムは、BuyPageInformation エlementが必須かどうかを指定し、必須の場合は、取るべき形式も指定します。

```
<BuyPageInformation reference="..."
...
</BuyPageInformation>
```

BuyPageInformation Elementは、CollectPayment ElementまたはCommand Elementのどちらかの中で有効です。

オプションの参照属性は、フリー・フォームのフィールドです。このフィールドの使用法は、キャッシュを統合するシステムによって定義されます。資料を参照して、このフィールドを使用する必要があるかどうか、および使用法について調べてください。

## Parameter

Parameter Elementは、WebSphere Commerce Payments API 要求に応じてキャッシュが個々のパラメーターの値を導出する方法を指定します。

```
<name="ACCEPTPAYMENT">
...
<Parameter
  name="..."
  encoding="..."
  maxBytes="..."
  sensitive="..."
  allowNullValue="...">
...
</Parameter>
```

Parameter Elementは、CollectPayment ElementまたはCommand Elementのどちらかの中で有効です。

name 属性は、WebSphere Commerce Payments に送信される API パラメーターのキーワードの名前を示します。このElementの内容は、値が導出される方法を指示します。値が導出される方法には、定数、変数、データベース項目、および拡張の4つがあります。

オプションの encoding 属性は、パラメーターが特定の文字エンコードに従っていない場合使用されます。値は、エンコードを表す有効な Java 名です。デフォルトのエンコードは UTF8 です。

オプションの maxBytes 属性は、WebSphere Commerce Payments に渡すパラメーターのバイト数を制限するために使用されます。これは、重要でないデータの入ったパラメーターの値が長過ぎることが原因で、コマンドが失敗することのないようにする上で役に立つことがあります。

オプションの sensitive 属性が "true" に設定されていると、キャッシャーはパラメーターの値をキャッシャー・トレース・ファイルに表示しなくなります。これは、クレジット・カード番号などの機密データを不正に入手できないよう保護する上で役に立ちます。

### 定数パラメーター

定数パラメーターを使用すると、変更できないパラメーター値がプロファイル内にハードコーディングされます。

```
<Parameter name="..."><CharacterText>1</CharacterText></Parameter>
```

### 変数パラメーター

環境変数パラメーターは、キャッシャーが統合されたシステムがパラメーターの値を提供するように指定します。環境変数の値は、Parameter エlementの内容の中で、変数名を中括弧 {} で囲んで指定します。指定された変数が collectPayment() 呼び出し上に渡されなかった場合には、キャッシャーによりエラーが報告されます。

```
<Parameter name="..."><CharacterText>{var1}{var2}</CharacterText></Parameter>
```

### データベース・パラメーター

データベース・パラメーターは、リレーショナル・データベースの照会を実行し、columnName 属性で示された列中の結果を参照して値が導出されるよう指示します。statementID 属性は、その前に宣言されている SelectStatement エlementの id 属性を参照します。照会が実行されなかったり、列名が存在しなかったりすると、キャッシャーによりエラーが報告されます。

```
<Parameter name="...">  
  <DatabaseValue statementID="..." columnName="..." />  
</Parameter>
```

### 拡張パラメーター

拡張パラメーターは、カスタム作成されたプログラムを実行してパラメーター値を導出しなければならないことを指示します。ExtensionValue エlementの name 属性は、実行するプログラムの名前を示します。キャッシャー拡張の作成に関する詳細は、『統合システムの作成』を参照してください。

```
<Parameter name="...">  
  <ExtensionValue name="..." />  
</Parameter>
```

---

## 統合システムの作成

以下のトピックで、統合システムの作成に関する要件を説明します。それらは以下のとおりです。

- プロファイルの構築
- 必要なファイルの組み込み
- キャッシャー・オブジェクトの作成
- collectPayment() を使った WebSphere Commerce Payments 中でのオーダーの作成
- checkPayment() を使ったオーダーの状況の検査
- BuyPageInformation の使用
- トレース
- 例外

- 拡張の作成

注: キャッシャー用の Javadoc は、 WebSphere Commerce Payments のインストール・システムの `Payments_installdir/javadoc/cal` ディレクトリにあります。

## プロファイルの構築

23 ページの『統合システム的设计』で、ご使用のサイトで使用可能にするプロファイルを選択しました。このステップでは、独自のプロファイルを作成する場合があります。次のステップでは、これらのプロファイルを編集して、ご使用のマーチャント・ソフトウェアで使用できるようにします。このパートは、複数の部分に分けることができます。

### WebSphere Commerce Payments の構成

統合システムで複数の WebSphere Commerce Payments インスタンスを使用する場合、 WebSphere Commerce Payments の構成情報をプロファイル内に格納するように選択することもできます。この選択を行うには、プロファイル中で `PaymentManagerConfiguration` エlementを指定しなければなりません。この Elementは、 WebSphere Commerce Payments インスタンスの場所、このインスタンスで使用されるユーザー ID とパスワード、SSL を使用するかどうか、および (オプションで) Socks サーバーの情報を示します。

### Parameter と SelectStatement

プロファイル中で指定する WebSphere Commerce Payments とカセットのパラメーターの完全リストがある場合は、個々のパラメーターが値を取得する場所を指定しなければなりません。プロファイル中にハードコーディングされた定数、`collectPayment()` 呼び出し、または `issueCommand()` 呼び出し上に渡されるオーダー処理環境からの値、リレーショナル・データベース中のフィールド、またはキャッシャー拡張コードで計算された値のいずれかを値のソースにすることができます。プロファイル中のパラメーターごとに、マーチャント・ソフトウェアが関係のある値を検索する場所を定義しなければなりません。(マーチャント・ソフトウェアのインターフェースの正式な定義が公表され、 WebSphere Commerce Payments パラメーターのリストと、それらの値のマーチャント・ソフトウェア上の場所が示されていることがあります。)

たとえば、マーチャントが 1 つしかないシステム上でキャッシャーを実行する場合には、以下のようにキャッシャー・プロファイル中で `MERCHANTNUMBER` パラメーターをハードコーディングすると有効です。

```
<Parameter name="MERCHANTNUMBER"><CharacterText>1</CharacterText></Parameter>
```

WebSphere Commerce Payments の `ORDERNUMBER` パラメーターの値が、`issueCommand()` 上に渡されるマップに組み込まれるように指定するには、以下を組み込みます (キー `orderNum` を使用して関連付けます)。

```
<Parameter name="ORDERNUMBER"><CharacterText>{orderNum}</CharacterText></Parameter>
```

WebSphere Commerce Payments の `AMOUNT` パラメーターと `CURRENCY` パラメーターの値が、リレーショナル・データベースから取り出されるよう指定するには、以下を組み込みます。

```
<SelectStatement id="sql1">SELECT AMT, CUR FROM ORDER_TABLE WHERE ORDERNUMBER=
{orderNum}</SelectStatement>
<Parameter name="AMOUNT"><DatabaseValue statementID="sql1" columnName="AMT">
</Parameter>
<Parameter name="CURRENCY"><DatabaseValue statementID="sql1" columnName="CUR">
</Parameter>
```

この例で、オーダーの金額は ORDER\_TABLE の AMT という列から取り出され、通貨は CUR という列から取り出されます。パラメーターが参照する SelectStatement は、1 オーダー当たり 1 データ行になることに注意してください。SelectStatement の **orderNum** は、collectPayment() または issueCommand() に渡されるデータの中に指定されていなければなりません。

WebSphere Commerce Payments の ORDERURL パラメーターの値が、URLBuilder というキャッシュ拡張クラス中で構成されるよう指定するには、以下を組み込みます。

```
<Parameter name="ORDERURL"><ExtensionValue name="URLBuilder"></Parameter>
```

URLBuilder は、CashierExtension インターフェースを実装した Java クラスでなければなりません。URLBuilder.class は、ご使用のクラスパス中になければなりません。

## 購入ページの情報

23 ページの『統合システムの設計』には、統合システムでプロファイルの BuyPageInformation エレメントを使用する方法がいくつか説明されていました。統合システムの設計に基づいて、購入ページを生成できるようにするための情報を指定しなければなりません。マーチャント・ソフトウェアで複数の言語でのショッピングがサポートされている場合は、購入ページに関する特別の考慮事項として、ローカライズの問題を扱う必要があります。

プロファイルの編集が終わったら、次の規則に準拠したファイル名で保管するようお勧めします。

```
MerchantSoftwareNameCassetteName.profile
```

たとえば、SampleCheckout サブレットと OfflineCard カセットで使用されるプロファイルが WebSphere Commerce Payments に備えられています。その名前は SampleCheckoutOfflineCard.profile です。

## 必要なファイルの組み込み

キャッシュャーを使用するには、以下のファイルをクラスパス中に組み込んでください。

- eTillCal.jar。
- xml4j.jar。
- eTillxml4j209.jar。
- ibmjssc.jar。このファイルは、SSL サポートを使用して WebSphere Commerce Payments との通信を行う場合だけ必要です。
- JDBC ドライバー。これは Database Value パラメーターを使用する場合だけ必要です。
- キャッシュャーのプロファイル中で参照される拡張クラス。

キャッシュャーのプロファイルすべてと profile.dtd は、1 つのディレクトリーに入れる必要があります。

## キャッシュャー・オブジェクトの作成

キャッシュャー・オブジェクトを構成するには、以下の 3 つの方法があります。

1. キャッシュャーと WebSphere Commerce Payments との間の通信チャンネルに関する情報 (WebSphere Commerce Payments のホスト名とポートや、SSL を使用するかどうかなど) を指定せずにキャッシュャーを構成できます。この方式を使用する場合は、キャッシュャー・プロファイルに <PaymentManagerConfiguration> エレメントを組み込まなければならず、このエレメントに必要な情報が組み込まれます。

```
public Cashier(String profileDirectory) throws Cashier Exception
```

2. 別の方法として、キャッシュャーのコンストラクター上で WebSphere Commerce Payments の構成情報を指定できます。この場合、プロファイルに <PaymentManagerConfiguration> エレメントを組み込む必要はありません。しかし、プロファイルに <PaymentManagerConfiguration> を組み込むと、コンストラクターに基づく構成情報はオーバーライドされます。

```
public Cashier(String profileDirectory,  
               String paymentsHostname  
               String paymentsPort  
               String userid  
               String password  
               boolean useSSL) throws CashierException
```

3. 次のコンストラクターを使用すると、Socks サーバーを介して WebSphere Commerce Payments に接続できます。

```
public Cashier(String profileDirectory,  
               String paymentsHostname,  
               int paymentsPort,  
               String socksHostname,  
               int socksPort,  
               String userid,  
               String password,  
               boolean useSSL) throws CashierException
```

統合システム的设计に基づいて、使用する方式を決める必要があります。たとえば、WebSphere Commerce Payments を 1 つ使用し、管理者のパスワードを頻繁に変更する場合は、キャッシュャーのコンストラクター上に WebSphere Commerce Payments の構成を指定する方が、個々のプロファイル中の PaymentManagerConfiguration エレメントを更新するより簡単です。

キャッシュャーはマルチスレッド環境で使用しても安全です。キャッシュャーは、内部的にプロファイルのキャッシュを保持するので、(毎回新しいキャッシュャー・オブジェクトをインスタンス化するのではなく) 同一のキャッシュャー・インスタンスを再使用することにより、統合システムを最適化できます。

## CollectPayment

CollectPayment エレメントには、キャッシュャーを使用して WebSphere Commerce Payments オーダーを作成するのに必要なすべてのデータが含まれています。

```
<Profile useWallet="false">  
  <CollectPayment>
```

```
...
</CollectPayment>
</Profile>
```

## WebSphere Commerce Payments でのオーダーの作成 - issueCommand()

WebSphere Commerce Payments でオーダーを作成するには、キャッシャーの `issueCommand()` メソッドを呼び出してください。 `issueCommand()` メソッドには以下の引き数が含まれます。

- *command* は、発行するコマンドを示す定数です。キャッシャー分野の資料で、許可されているコマンドのリストを参照してください。オーダーを作成する場合、コマンドは `ACCEPTPAYMENT` か `RECEIVEPAYMENT` のどちらかでなければなりません。
- *profileName* は、WebSphere Commerce Payments のコマンドの作成に使用されるプロファイルの名前です。
- *locale* は、マーチャント・ソフトウェアがショッパーに対して提示するテキストのロケールです (オプション)。
- オーダー処理環境からの値。
- データベース接続 (オプション)。
- *queryable* は、オプションの照会インターフェース。ハッシュ・テーブルを使ってパラメーターの値のリストを戻すために使用できます。

普通は、バイヤーが選択した決済方式に基づいて、使用するプロファイルが決められます。たとえば、代金引換のオーダー用に 1 つのプロファイルを備え、クレジット・カードでのオーダー用に別のプロファイルを備えることができます。

`issueCommand()` に指定するロケールは、バイヤーがショッピングを行う際のロケールと一致している必要があります。WebSphere Commerce Payments はこの値を使用して、エラーの場合にショッパーに表示されるメッセージをローカライズして作成します。したがって、マーチャント・ソフトウェアが独自のエラー・メッセージを構成しなくても、新しいカセットを追加できます。

マーチャント・ソフトウェアがオーダーを処理している際に使用できる情報が、キャッシャーに渡される必要があります。マーチャント・ソフトウェアの種類によって、これには HTTP 要求、名前と値の対、または他の情報を構成するパラメーターが含まれます。プロファイルで使用されるデータは、マップに書き込まれ、`issueCommand()` 呼び出し上に渡される必要があります。

プロファイルに `DatabaseValue` パラメーターが組み込まれている場合は、初期設定された JDBC 接続が設けられる必要があります。キャッシャーは `issueCommand()` 呼び出し中は JDBC 接続を閉じないことに注意してください。

## オーダーの状況の検査 - checkPayment()

キャッシャーには `checkPayment()` という単純なメソッドが備えられており、これは作成済みのオーダーの状況を照会します。 `checkPayment()` を呼び出すと `CheckPaymentResponse` オブジェクトが戻され、このオブジェクトにオーダーの状態が含まれています。キャッシャーの Javadoc (WebSphere Commerce Payments に備えられている) には、この状態の値として考えられるものが記述されています。

```

CheckPaymentResponse checkPaymentResponse =
cashier.checkPayment(merchantNumber, orderNumber);

if (checkPaymentResponse.getPrimaryReturnCode() == 0 &&
    checkPaymentResponse.getSecondaryReturnCode() == 0)
{
    switch (checkPaymentResponse.getState())
    {
        case CheckPaymentResponse.APPROVED:
            ...
        case CheckPaymentResponse.MISSING:
            ...
        ...
    }
}
else
{
    ...
}

```

## BuyPageInformation の使用

キャッシャーの `getBuyPageInformation()` メソッドを呼び出すと、統合システムの設計で定義したとおりに購入ページ情報を取り出すことができます。購入ページ情報の参照を取り出すには、`getBuyPageInformationReference()` を呼び出します。

## トレース

キャッシャーにはトレース機構があり、これは統合システムを作成するのに役立ちます。トレース・クラス (`SimpleCashierTrace`) は `eTillCal.jar` 内にあり、この JAR はファイルへの書き込みを行います。その代わりに、`CashierTrace` インターフェースをインプリメントすることにより、ご使用のマーチャント・ソフトウェアの既存のトレース・クラスをキャッシャーで使用できるようにすることもできます。

プロファイルごとにトレースを使用可能にして、問題のありそうなプロファイルの診断に役立てることができます。プロファイルのトレースを使用可能にするには、`Profile` エレメント中で `enableTrace="true"` を設定します。

```

Cashier cashier = new Cashier("d:¥¥cashierProfileDirectory");
SimpleCashierTrace simpleCashierTrace =
new SimpleCashierTrace("d:¥¥cashierLogDirectory");
cashier.setTraceClass(simpleCashierTrace);

```

## 例外

キャッシャーでエラーが起きると、例外がスローされ、問題の原因が示されます。キャッシャーには、`ProfileExceptions` と `CashierExceptions` という 2 種類の例外があります。プロファイルの形式が適切でなかったり、無効だったり、論理エラーがあったりするために、そのプロファイルを使用してオーダーを作成できないといったプロファイルを検出すると、キャッシャーは `ProfileException` をスローします。キャッシャーの使用法が適切でなかったり、マーチャント・データベースへのアクセス・エラーが生じたりすると、`CashierExceptions` がスローされます。

キャッシャーを呼び出す際には、`issueCommand()`、`collectPayment()`、および `checkPayment()` が `PaymentServerCommunicationExceptions` をスローすることに注意してください。したがって、この呼び出しの前後に再試行ロジックを書き込むこともできます。

CashierExceptions と ProfileExceptions にスロー可能なオブジェクトを含めて、エラーの詳細を入手することもできます。どちらの例外にも、このスロー可能オブジェクトにアクセスするための `getNestedException()` というメソッドがあります。

## 拡張の作成

ほとんどの WebSphere Commerce Payments パラメーターの値は、定数、オーダー処理環境からの値、またはデータベース中の値を使用して取得できます。しかし、値を簡単に導出できないパラメーターもあります。たとえば、ショッパーのオーダーのテキスト記述がパラメーターにとって必要で、マーチャント・ソフトウェアに適切な形式の記述がない場合は、キャッシャー拡張をコーディングして、このパラメーターの適切な値を構築する必要が生じることがあります。

キャッシャー拡張とは、WebSphere Commerce Payments パラメーターの値を構築するためにキャッシャーで実行されるコードのことです。キャッシャー拡張を作成するには、`CashierExtension` インターフェースをインプリメントしたクラスを作成しなければなりません。このインターフェースには `getValue()` という 1 つのメソッドだけがあります。キャッシャー・プロファイルを使用する際に、そのプロファイルにキャッシャー拡張を参照する `ExtensionValue` パラメーターが含まれていると、その拡張中で `getValue()` が呼び出されます。

```
public class SampleExtension implements CashierExtension
{
    public String getValue(String keyword, Hashtable environmentValues,
        Hashtable PaymentsParameters, Connection connection,
        CashierTrace cashierTrace, Locale locale) throws CashierException
    {
        if (keyword.equals("$DATETIME"))
        {
            Date date = new Date();
            return date.toString();
        }
        else if (keyword.equals("$RANDOMNUMBER"))
        {
            return String.valueOf(Math.random());
        }
        else ...
    }
}
```

---

## SampleCheckout アプリケーション

SampleCheckout はサンプル・アプリケーションの一種で、アプリケーションがキャッシャーを使用して WebSphere Commerce Payments と統合する方法について説明します。このアプリケーションでは HTML インターフェースが使用されます。このインターフェースには URL `http://host_name:port/webapp/SampleCheckout` からアクセスできます。ソース・コードは、`Payments_installdir/samples/SampleCheckout` ディレクトリーにあります。

### 概要

SampleCheckout は単純なオーダー入力システムで、これを使用するとさまざまな決済方式を使用してオーダーを作成できます。ユーザーは基本的なオーダー情報 (オーダー番号、マーチャント番号、オーダー総額など) と、オーダーの決済の収集に使用する決済情報を入力しなければなりません。SampleCheckout を使用すると、



さまざまな決済方式をいくつでも構成できます。個々の決済方式はキャッシュ・プロファイルによってサポートされます。SampleCheckout プロファイルには、購入ページの決済情報の部分を構築するのに必要な HTML と、WebSphere Commerce Payments 中にオーダーを作成する API 要求を構築するのに必要なデータが含まれています。

SampleCheckout は、ユーザーのブラウザの言語設定を使用して購入ページを表示しようとしています。SampleCheckout は、WebSphere Commerce Payments でサポートされている言語に変換されます。ユーザーの言語設定がサポートされていない場合は、購入ページは英語で表示されます。Internet Explorer で言語を選択するには、メニュー・バーから「ツール」をクリックし、「インターネット オプション」をクリックしてから、「言語」ボタンをクリックします。Netscape Navigator の場合は、メニュー・バーから「編集」をクリックし、「設定」をクリックしてから、Navigator カテゴリの「言語」を選択します。赤いアスタリスクのマークが付いたフィールドは、入力必須フィールドです。その他のフィールドはオプションです。

SampleCheckout は、キャッシュ・プロファイルでウォレットを使用している場合でも使用できます。プロファイルにウォレットが必要な場合は、SampleCheckout は WebSphere Commerce Payments からの API 応答に HTTP ウォレットのウェイクアップ・メッセージが含まれていると想定します。

## 要件

SampleCheckout アプリケーションは、Java Server Pages (JSP) と動的な HTML テクノロジーを使用します。このアプリケーションは WebSphere Commerce Payments のインストール時に自動的にインストールされ、すべての IBM カセット用に構成されます。このアプリケーションを使用するには、JSP をサポートするサーブレット・エンジン (WebSphere Application Server など) と、動的 HTML をサポートするブラウザがなければなりません。

Web サーバーは、/webapp/SampleCheckout で始まる URI に応答して、ディレクトリ web/SampleCheckout 内のファイルを提供するように構成されていなければなりません。

ワークステーション・プラットフォームでは、サーブレット・エンジンに SampleCheckoutServlet というサーブレットが定義されて、クラスパスに eTillCal.jar、xml4j.jar、および ibmjss.jar が含まれていなければなりません。

▶ 400 クラスパスに eTillCal.jar および xml4j.jar が含まれていなければなりません。

Payments\_installdir/samples/SampleCheckout ディレクトリ内の Java ソース・コード・ファイル SampleCheckout を変更して新しいクラスを作成した場合、新しいクラス・ファイルが使われるようにするために、それらを必ず以下のディレクトリに格納してください。

Payments\_installdir/wc.mpf.ear/SampleCheckout.war/WEB-INF/classes

## 構成

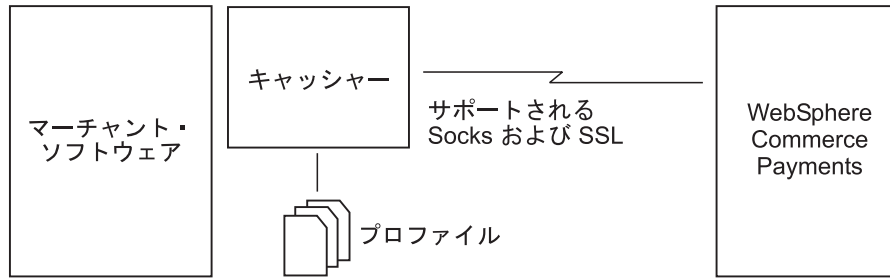


図5. WebSphere Commerce Payments キャッシュャー

SampleCheckout は、 `samples/SampleCheckout/SampleCheckout.xml` という名前の構成ファイルを使用して、以下のものを定義します。

- WebSphere Commerce Payments 構成情報 (ホスト名、ポート、Socks と SSL の使用の有無)。 SampleCheckout には WebSphere Commerce Payments 構成情報を格納するためのグローバルな方式が備えられているので、個々のプロファイルで `PaymentManagerConfiguration` エレメントを指定する必要はありません。
- 使用可能な決済方式と、個々の方式で使用されるキャッシュャー・プロファイル。以下のエレメントを `SampleCheckout.xml` の `PaymentOptionList` エレメントに追加すると、 SampleCheckout によって新しい決済方式をサポートできます。

```
<PaymentOption id="newmethod" profile="newprofname">  
  New Payment Method  
</PaymentOption>
```

`newmethod` はこの新しい決済方式の ID、 `newprofname` はこの方式をサポートするキャッシュャー・プロファイルの名前、および "New Payment Method" は SampleCheckout 購入ページに表示されるラベルです。

- SampleCheckout によってサポートされる通貨。

## SampleCheckout プロファイル

WebSphere Commerce Payments のインストール時に、 SampleCheckout プロファイルが `profiles` ディレクトリ `Payments_installdir/wc.mpf.ear/SampleCheckout.war/profiles` にインストールされます。 SampleCheckout プロファイルには、 `BuyPageInformation` エレメントが含まれていなければならない、また WebSphere Commerce Payments とこのプロファイル用に指定されているカセットに必要なすべてのパラメーターのパラメーター定義が含まれていなければならない。 WebSphere Commerce Payments 構成情報は、 SampleCheckout 構成ファイルに指定されていないので、 SampleCheckout プロファイルには必要ありません。しかし、 `PaymentManagerConfiguration` エレメントを指定すると、 `SampleCheckout.xml` で指定された構成がオーバーライドされます。

### 購入ページの情報

個々のプロファイル中の `BuyPageInformation` エレメントには、購入ページの決済情報セクションを作成するための HTML が含まれていなければならない。以下のように、個々のプロファイルの `BuyPageInformation` の内容に、 HTML の `<table>` タグと `<form>` タグが挿入されます。

```

<FORM NAME="cassetteform" METHOD="POST"
ACTION="/webapp/SampleCheckout">
  <TABLE BORDER="0" WIDTH="100%" CELSPACING="1" CELLPADDING="2">

    ... <BuyPageInformation> contents ...

  </TABLE>
</FORM>

```

SampleCheckout は、Java ResourceBundle ファイルを介して、BuyPageInformation エLEMENTの内容のローカライズをサポートします。この種のファイルには、テキストへのキーワードのマッピングが含まれているので、SampleCheckout プロファイルの作成者はこれを使用すると、BuyPageInformation エLEMENT中でテキストをハードコーディングしなくて済みます。その代わりに、SampleCheckout は実行時に、中括弧で囲まれたキーワードを、ResourceBundle 中のユーザーが要求した言語のテキストに置き換えます。SampleCheckout の BuyPageInformation エLEMENTで使用される ResourceBundle の名前は、参照属性によって示されます。たとえば、SampleCheckout プロファイルに以下のエLEMENTが含まれているとします。

```

<BuyPageInformation reference="SampleCheckoutOfflineCard">
  ...
  <p>{BPMESSAGE}</p>
  ...
</BuyPageInformation>

```

この場合、ユーザーがカナダ・フランス語の購入ページを要求していると、このアプリケーションは以下の Java ResourceBundles 中で BPMESSAGE のローカライズ済みのテキストを検索し、一致項目を検出するまで続けます。

```

SampleCheckoutOfflineCard_fr_CA.class
SampleCheckoutOfflineCard_fr_CA.properties
SampleCheckoutOfflineCard_fr.class
SampleCheckoutOfflineCard_fr.properties
SampleCheckoutOfflineCard.class
SampleCheckoutOfflineCard.properties

```

## プロファイル環境変数

以下の表は、SampleCheckout 使用時にプロファイルで使用できる変数を定義しています。これらの環境変数をプロファイル中で使用するには、変数名を中括弧で囲みます。たとえば、{merchantnumber} は、ユーザーが購入ページ上で入力したマーチャント番号に置き換えられます。

変数名	内容
merchantnumber	フォームに入力されたマーチャント番号
ordernumber	フォームに入力されたオーダー番号
currency	フォーム上で選択した、ISO 4217 通貨を表す 3 桁の数値
currencyAlpha	ISO 4217 通貨を表す 3 文字の英字の値
amount	フォームに入力された総額
amountLowestCurrUnits	最小の通貨単位に変換された総額の値。たとえば、5 US ドルは 500 セントに変換されません。

変数名	内容
paymentOption	フォーム上で選択された決済方式
個々のプロファイルの BuyPageInformation エレメント中に指定された他のフォーム変数	フォームに入力された値

---

## 第 4 章 クライアント API ライブラリー (CAL)

マーチャント・ビジネス・ソフトウェアは、WebSphere Commerce Payments に対して決済コマンド、管理コマンド、および照会コマンドを送出できます。HTTP POST メッセージを発行すると、要求が WebSphere Commerce Payments に送信されます。WebSphere Commerce Payments からの応答は、HTTP に組み込まれた XML ドキュメントの形式で受信します。Java Client API Library (CAL) は、Java プログラミング・インターフェースを提供しています。このインターフェースにより、これらのコマンドを WebSphere Commerce Payments に対して送出してその応答を受け取るマーチャント・ソフトウェアを、Java で書くことができます。CAL は、マーチャント・ソフトウェアの作成者が、HTTP 通信や XML ドキュメント構文解析についての詳細を理解していなくてもかまわないようにするラッパーを提供します。CAL は、マーチャント・プログラムが次のことを行えるようにする Java オブジェクトを提供します。

- WebSphere Commerce Payments に送信する要求を作成する
- TCP 接続または SSL 接続を介して WebSphere Commerce Payments と通信する
- WebSphere Commerce Payments からの応答を処理する

CAL を使用するように作成されたマーチャント・プログラムには、次のようないくつかのステップがあります。

- PaymentServerClient を作成する
- WebSphere Commerce Payments にコマンドを発行する
  - ハッシュ・テーブルを作成し、それにキーワードと値のペアを取り込む
  - コマンドを発行する
  - 戻りコードを処理する
  - 戻されたデータを処理する
- PaymentServerClient をクローズする

この節の残りの部分は、これらのステップについてのハイレベルの解説です。より詳しい説明は、以下のディレクトリー場所にある Javadoc を参照してください。

 [Payments\\_installdir/javadoc](#)

---

### CAL 共用クラス

CAL は、いくつかの共用クラスを含む Java クラス・ライブラリーとして構造化されています。これらのクラスは、次の 3 つのグループに分割できます。

1. **クライアント・クラス:** マーチャント・プログラムは、これらのクラスの 1 つのインスタンスを作成して WebSphere Commerce Payments と通信します。
  - PaymentServerClient: TCP 接続を介して (SOCKS サポートを使用してまたは使用しないで) WebSphere Commerce Payments と通信します。
  - PaymentServerSSLClient: SSL 接続を介して WebSphere Commerce Payments と通信します。

2. **応答クラス:** WebSphere Commerce Payments に対して送られるそれぞれのコマンドは、 `PaymentServerResponse` クラスのインスタンスを戻します。
3. **PSObject クラス:** QUERY コマンドから戻されるデータは、実際の WebSphere Commerce Payments オブジェクトを反映している `PSObjects` のセットの中に作成されます。
  - `PSObject`: これらすべてのオブジェクトのスーパークラス
  - `PSAdminObject`: すべての管理オブジェクトのスーパークラス
  - `PSOrder`: Order を表す
  - `PSPayment`: Payment を表す
  - `PSCredit`: Credit を表す
  - `PSBatch`: Batch を表す
  - `PSBatchTotal`: 特定の通貨のバッチの合計を表す
  - `PSPaymentServer`: Payment Server 管理オブジェクトを表す
  - `PSMerchant`: マーチャント管理オブジェクトを表す
  - `PSCassette`: カセット管理オブジェクトを表す
  - `PSMerchantCassetteSettings`: PaymentSystem 管理オブジェクトを表す
  - `PSAccount`: アカウント管理オブジェクトを表す
  - `PSCassetteObject`: カセットによって汎用オブジェクトに付加されたオブジェクトを表す
  - `PSCassetteConfigObject`: カセットによって汎用管理オブジェクトに付加された管理オブジェクトを表す
  - `PSAbout`: WebSphere Commerce Payments のバージョン情報およびコマンド送出者のユーザー名を提供する
  - `PSCassetteAbout`: WebSphere Commerce Payments カセットのバージョン情報を提供する

---

## PaymentServerClient の作成

`PaymentServerClient` は、マーチャント・プログラムから WebSphere Commerce Payments への接続を表しています。これは永続オブジェクトで、マーチャント・プログラムの開始時に作成され、プログラム実行中に使用および再使用され、プログラムが終了する時にクローズされます。 `PaymentServerClient` には単一ソケット接続があり、それは `PaymentServerClient` がクローズされるまで保持されます。 `PaymentServerClient` は、さまざまな通信オプションを機能させるために、いくつかの方法で作成できます。

基本的な `PaymentServerClient` は、次の 3 つの引き数で構成されます。

```
PaymentServerClient (String dtdPath, String hostName, int portNumber)
```

このコンストラクターは、TCP を使用して `hostName:portNumber` で WebSphere Commerce Payments と通信するクライアントを作成します。追加の引き数 `socksHost` と `socksPort` を基本コンストラクターに追加することができます。このようにすると、SOCKS サーバーを介して WebSphere Commerce Payments と通信するクライアントが作成されます。

```
PaymentServerClient(String dtdPath, String hostName, int portNumber,  
String socksHost, int socksPort)
```

以下の 2 つの追加コンストラクターを使えば、ハッシュ・テーブルの仕様を使用して、追加のキーワードと値のペアを HTTP ヘッダーで渡すように指定することができます。

```
PaymentServerClient (String dtdPath, String hostName, int portNumber,  
Hashtable httpHeaderFields)
```

```
PaymentServerClient (String dtdPath, String hostName, int portNumber,  
String socksHost, int socksPort, Hashtable httpHeaderFields)
```

その他の通信オプションは、`PaymentServerClient` のサブクラスを使用して作成されます。`PaymentServerSSLClient` は、SSL 接続を介して WebSphere Commerce Payments と通信します。

```
PaymentServerSSLClient(String dtdPath, String hostName, int portNumber)
```

```
PaymentServerSSLClient(String dtdPath, String hostName, int portNumber,  
String socksHost, int socksPort)
```

```
PaymentServerSSLClient(String dtdPath, String hostName, int portNumber,  
Hashtable httpHeaderFields)
```

```
PaymentServerSSLClient(String dtdPath, String hostName, int portNumber,  
String socksHost, int socksPort, Hashtable httpHeaderFields)
```

**注:** `PaymentServerClient` がインスタンス化される時に指定される `DTDPPath` は、セッションの期間中、つまり `close ()` メソッドの前に処理されるこれ以降のすべてのコマンドで使用されます。`PaymentServerClient` の `DTDPPath` はオプションであり、ヌルにすることもできますが、指定した方がパフォーマンスが向上します。

---

## CAL 使用時の SSL サポートのために iSeries を準備する

**注:** 以下の説明は、 において CAL を使用している場合にのみ当てはまります。

SSL (Secure Sockets Layer) を使用するようシステムを準備するには、デジタル証明書マネージャー・ライセンス・プログラム (5722-SS1 OS/400 — デジタル証明書マネージャー) をインストールする必要があります。

以下のどちらかの Cryptographic Access Provider ライセンス・プログラムもインストールする必要があります。

- 5722-AC2 Cryptographic Access Provider 56-Bit
- 5722-AC3 Cryptographic Access Provider 128-Bit

サーバーからクライアント認証を求められる場合、以下のプロパティを設定して、使用するデジタル証明書を指定できます。

- `os400.certificateContainer`
- `os400.certificateLabel`

上記のプロパティを設定しない場合、デフォルトのシステム証明書があるなら、それが使用されます。Java/SSL のインストールに関する iSeries の資料に関する詳細は、<http://publib.boulder.ibm.com/pubs/html/as400/infocenter.html> に記載されています。現行の iSeries のバージョンのリンクをたどり、次の順序で選択してください。

「プログラミング」、「Java」、「IBM Developer Kit for Java」、「セキュリティー」、「セキュア・ソケット・レイヤー」。

---

## WebSphere Commerce Payments コマンドの送付

PaymentServerClient クライアントの `issueCommand` メソッドは、コマンドを WebSphere Commerce Payments に送信するために使用されます。 `issueCommand` メソッドには、もっと多くのパラメーターをとるものもありますが、最低限それぞれの `issueCommand` メソッドは次のパラメーターをとります。

### WebSphere Commerce Payments API コマンド名

WebSphere Commerce Payments API コマンド名。 WebSphere Commerce PaymentsAPI コマンドのリストは、55 ページの『第 6 章 WebSphere Commerce Payments コマンド・リファレンス』を参照してください。共用インターフェース `PaymentCommandConstants` は、それぞれの API コマンドごとに定数を定義します。詳細については Javadoc を参照してください。

### WebSphere Commerce Payments コマンドとともに送信される、キーワードと値のペアのハッシュ・テーブル

この Java ハッシュ・テーブルは、指定された API コマンドとともに渡されるパラメーターを表します。ハッシュ・テーブルのキーは、API コマンド・パラメーター名を表す文字列です。値は API コマンド・パラメーター値を表します。値のタイプは次のうちの 1 つです。

- `String`: サポートされているすべての文字セットの中の、1 つのユニコード・ストリング
- `byte[]`: バイナリー・データ用のバイト配列
- `Integer`: 4 バイト整数
- `Date`: タイム・スタンプを表す Java 日付 (`java.util.Date`)
- `Boolean`: ブール値
- `Vector`: 上記のいずれかのベクトル。ベクトル値は、特別なケースです。キーワードに値のベクトルが割り当てられる場合は、キーワードは、ベクトルの中の項目ごとに 1 つずつ、HTTP 本体に複数回組み込まれます。

次のことに注意する必要があります。すなわち、CAL はこれらのキーワードと値のペアをチェックして、指定されたキーワードに対してそれらが有効であることや、値のデータ・タイプが正確であることを検査したりはしません。CAL は、すべてのキーワードと値のペアを WebSphere Commerce Payments に渡すだけです。それぞれの WebSphere Commerce Payments コマンドごとの必須パラメーターおよび任意指定パラメーターのリストについては、55 ページの『第 6 章 WebSphere Commerce Payments コマンド・リファレンス』を参照してください。共用インターフェース `PaymentCommandConstants` は、それぞれの API コマンド・パラメーター値ごとに定数を定義します。詳細については Javadoc を参照してください。

### 認証情報

WebSphere Commerce Payments がコマンドを受け取ると、`WCSRealm` を使用してユーザーを認証します。WebSphere Commerce Payments を使用するプログラムを作成するとき、WebSphere Commerce Payments はこのレルムを使うことを理解している必要があります。レルムには、WebSphere



Commerce Payments にアクセスする権限を潜在的に付与されているユーザーのリストが、それらのユーザーの認証情報とともに含まれています。レルムは、それぞれのコマンドがそれに関連したユーザー ID とパスワードのペア、もっと一般的には、他の認証信任状を含むバイト配列を持っている必要があるかどうかを記述します。WCSRealm は userId とパスワードを使って認証を行います。WCSRealm の使用は、ユーザー・インターフェースの WebSphere Commerce Payments 設定画面で確認できます。

issueCommand メソッドの基本バージョンは次のとおりです。

```
issueCommand(String command, Hashtable keywordValuePairs, String basicAuthUserid, String basicAuthPassword)
```

さらに、ハッシュ・テーブルの仕様を使用して、追加のキーワードと値のペアを HTTP ヘッダーで渡すように指定することができる、issueCommand のバージョンがあります。

```
issueCommand(String command, Hashtable keywordValuePairs, Hashtable httpHeaderPairs, String basicAuthUserid, String basicAuthPassword)
```

issueCommand メソッドは、エラーまたはその他の処理上の問題が起こる場合に例外を発します。

## HTTP ヘッダーで追加情報を指定する

HTTP ヘッダーで追加情報を指定する方法が 2 つあります。

- PaymentServerClient オブジェクトのコンストラクターを使用する。これは、WebSphere Commerce Payments に対して送出されるすべてのコマンド上で、追加パラメーターを指定させます。
- issueCommand メソッドで。これは、送出しようとしているコマンドに対してのみ、追加パラメーターを指定させます。したがって、それぞれの WebSphere Commerce Payments コマンドごとに、HTTP ヘッダーを調整することができます。この例が、**AcceptPayment** および **ReceivePayment** API コマンド上で生じます。これらのコマンドでは、コマンドの処理が成功しなかったときに、WebSphere Commerce Payments がメッセージ・テキストを戻します。メッセージ・テキストは、HTTP ヘッダー内の PM-Accept-Language タグで指定されたクライアント・アプリケーションの言語設定で、追加情報を提供します。PM-Accept-Language タグが HTTP ヘッダーで指定されていない場合には、サーバーを実行しているマシンのデフォルトの言語が使用されます。これらのメッセージに関する追加情報は、PaymentServerResponse メソッド getBuyerMessage() および getMerchantMessage() を参照してください。さらに、CAL は PM-Accept-Language タグ用のキーワードと値のペアを作成するための便利な方法を提供します。詳しくは、PaymentServerClient メソッド addLocaleToHTTPHeader を参照してください。

---

## WebSphere Commerce Payments からの応答の処理

PaymentServerResponse オブジェクトは issueCommand メソッドによって戻されます。このオブジェクトには、WebSphere Commerce Payments にコマンドを送出した結果として戻ってきた 1 次および 2 次戻りコードを、マーチャント・ソフトウェアがアクセスできるようにするメソッドが含まれます。WebSphere Commerce Payments 戻りコードのリストについては、131 ページの『付録 A. WebSphere

Commerce Payments 戻りコード』を参照してください。プログラム上のエラーが生じた場合には、Java 例外割り込みが出されます。CAL には次の 2 つのタイプの例外割り込みが定義されています。

- `PaymentServerCommunicationException`: この例外割り込みは、WebSphere Commerce Payments との通信で CAL に問題がある時に出されます。考えられる原因としては、次のことがあります。
  - CAL が無効な HTTP 応答を受信した。一般的にこれは、Payment サーブレットまたは WebServer/WebSphere 構成に何らかの問題があるという意味です。
  - CAL が `IOException` を受け取った。これは、TCP または SSL 層が `IOException` を発した (たとえば、WebSphere Commerce Payments に接続できなかったか、接続が途中で切れた) という意味です。この例外割り込みの原因が `IOException` である場合は、`IOException` は `PaymentServerCommunicationException` 内に保管されます (マーチャント・プログラマーがアクセスできるようになります)。
- `PaymentServerClientException`: これは、CAL が発する内部例外です。これは CAL 内の不良を示します。

## 戻されるオブジェクトの処理

コマンドの結果が、データの戻りになる場合 (たとえば、`QUERY` コマンド)、`PaymentServerResponse` の一部として `PSObjects` のセットが戻されます。これらのオブジェクトは、基本 WebSphere Commerce Payments オブジェクトと対応しています。これらのフィールドの解釈は、111 ページの『第 7 章 WebSphere Commerce Payments データ』で説明されています。

`PaymentServerResponse` オブジェクトには、応答で戻されたオブジェクト数を戻すメソッド `getObjectCount` が含まれています。これは、データのサイズを制限する `RETURNATMOST` を使用した照会で特に役立ちます。

---

## PaymentServerClient をクローズする

`PaymentServerClient` クラスには、`close()` メソッドが含まれます。マーチャント・プログラムは、終了する前に `close()` を呼び出す必要があります。これは、標準的な TCP または SOCKS 通信を使用している単純なプログラムにとっては、特に重要なことではありません。なぜなら、Java Runtime がこれらのリソースを終了時に終結処理するからです。しかし、SSL クライアントにとっては非常に重要です。これらのクライアントで `close()` を呼び出さないと、マーチャントのアプリケーションが再始動するときに問題が起こることがあります。マーチャント・プログラムは、SSL を使用するよう変換される可能性があるため、すべての場合に `close()` が呼び出されるようにするのはよいことです。

---

## サンプル CAL プログラム

この節では、単純な CAL プログラムの骨組みが示されています。サンプルの CAL プログラムが用意されており、次の場所から入手できます。

 `Payments_installdir/samples`

CAL を使用するように作成されたマーチャント・プログラムには、次のような 3 つの基本ステップがあります。

1. PaymentServerClient を作成する
2. WebSphere Commerce Payments にコマンドを発行する
  - a. ハッシュ・テーブルを作成し、それにキーワードと値のペアを取り込む
  - b. コマンドを発行する
  - c. 戻りコードを処理する
  - d. 戻されたデータを処理する
3. PaymentServerClient をクローズする

CAL プログラムの例を以下に示します。

```
PaymentServerClient client = new PaymentServerClient(dtdPath, hostName,
port);

while (...)
{
    Hashtable keywordValuePairs = new Hashtable();
    keywordValuePairs.put("merchantnumber","123456789");
    ... using documentation in the programming reference as a guide, fill
    in other keywordValuePairs ...

    PaymentServerResponse response =
client.issueCommand(command,keywordValuePairs,userid,password);

    int primaryRC = response.getPrimaryRC();
    int secondaryRC = response.getSecondaryRC();
    ... process return codes ...
    String contentType = response.getContentType();
    if (contentType != null)
    ... process contentType




    Enumeration objects = response.getObjects();
    while (objects.hasMoreElements())
    {
        PSObject object = (PSObject) objects.nextElement();
        ... process object ...
    }
}

client.close();
```

---

## CAL が必要とするファイルのインストール

CAL が必要とするファイルはすべて、 WebSphere Commerce によってすでに提供されています。

   すべてのファイルは、*Payments\_installdir/etillClientSDK.zip* ディレクトリーにある圧縮ファイル *eTillCal.jar* に提供されています。



WebSphere Commerce Payments アプリケーションを実行しているシステムまたはセッションの CLASSPATH 環境変数中に、必須クラス・ライブラリーが組み込まれていることを確認してください。

## まだ WebSphere Commerce Payments をインストールしていないマシンの場合


CAL インターフェースに書き込みを行う場合、または WebSphere Commerce Payments がインストールされていないマシンから CAL プログラムを実行する場合は、以下のステップを実行してください。

Windows > UNIX > 400

1. WebSphere Commerce Payments がインストールされているマシンから、次のファイルを自分のマシンにコピーします。これらのファイルは、WebSphere Commerce Payments ディレクトリーにあります。

- eTillCal.jar
- eTillxml4j209.jar
-   ibmjsse.jar (SSL を使用している場合は、ランタイムでのみ必要です。)

eTillCal.jar、eTillxml4j209.jar、および ibmjsse.jar ファイルは、*Payments\_installdir/etillClientSDK.zip* ファイルに含まれています。iSeries 環境では ibmjsse.jar ファイルは使用されません。

2.  別の iSeries システムで稼働している CAL を通して SSL を使用する場合は、『SSL サポートのために iSeries を準備する』に列挙されているライセンス・プログラムがリモートの iSeries システムにも必要です。

**注:** iSeries 以外のシステムで稼働している CAL を通して SSL をサポートする場合は、ibmjsse.jar をコピーします。

3. システム CLASSPATH を編集して、eTillCal.jar と eTillxml4j209.jar を含めます。

---

## 第 5 章 イベント通知

WebSphere Commerce Payments のイベント通知サービスにより、マーチャント・ソフトウェア (またはネットワーク管理システムなどの非マーチャント・ソフトウェア) はイベントを `listen` し、マーチャントの営業システムで適切な処置を行うことができます。たとえばアクションの例として、オーダーが承認されたことをイベントが示す場合、出荷部門にオーダーを送ります。この機能によって、通常は `QUERY` コマンドを出して WebSphere Commerce Payments オブジェクトの状態を判別するシステムのパフォーマンスを最適化することができます。オブジェクトの状態の変更時に発生するイベントを `listen` することにより、ポーリング・ループのオーバーヘッド全体から影響を受けることなく、マーチャント・システムは活動を再開することができます。また、ネットワーク管理ソフトウェアによってイベント通知サービスを使用すれば、WebSphere Commerce Payments の状態をモニターできます。

マーチャント・ソフトウェアは、対象とする WebSphere Commerce Payments イベントを登録し、URL を指定します。イベントが発生すると、イベント通知サービスは URL によって指定された宛先へ `HTTP POST` を送信します。マーチャント・ソフトウェアには、イベントを受信する責任があります。イベントを `listen` するマーチャント・ソフトウェアには、CGI、Java サブレット、または登録時に指定されたポートを `listen` するプログラムを選択できます。

---

### イベントのタイプと内容

WebSphere Commerce Payments イベント通知サービスは、以下に示す 3 タイプのイベントを定義および送信します。

1. **状態変更イベント:** このタイプのイベントは、フレームワーク・オブジェクトの状態に変化があると送信されます。たとえば、`Order` オブジェクトの状態が「受信 (Received)」から「承認 (Approved)」へ変更された場合などです。
2. **カセット固有イベント:** カセットはこのタイプのイベントを使用することにより、カセット内で発生したイベントをマーチャントに通知できます。カセットはイベントの内容を定義します。すべてのカセットがカセット固有のイベントを実装するわけではありません。
3. **ネットワーク管理イベント:** このイベントは、WebSphere Commerce Payments が開始または停止されると送信されます。

WebSphere Commerce Payments では、フレームワークの金融オブジェクトに対しても「状態変更イベント」が提供されています。また、フレームワークの起動と停止に関連したネットワーク管理イベントも提供されています。マーチャント・ソフトウェアは、該当するカセットの付録 (補足情報) を参照して、どのカセット・イベントがサポートされているかを判別する必要があります。

どのイベントにも、以下に示す「基本的な」内容が含まれます。

- **EventType:** イベントのタイプ。
- **Timestamp:** イベントが生じた時刻。

- **ObjectID:** イベントの参照先のオブジェクトを識別する。 ObjectID は複数のフィールドで構成される場合もあります。

イベント・タイプが異なれば、そこに含まれる情報も異なる可能性があります。その点を以下の節で説明します。

## 状態変更イベント

状態変更イベント

名前	値
EventType	「1」
Object	値は以下のいずれかです。 <ul style="list-style-type: none"> <li>• Order</li> <li>• Payment</li> <li>• Credit</li> <li>• Batch</li> </ul>
<ObjectID>	ObjectID は Object タイプに従属しています。各オブジェクトはキーのセットによって識別されます。(たとえば、Order は MerchantNumber と OrderNumber によって識別されます。)
PreviousState	状態名。状態の定義については、111 ページの『WebSphere Commerce Payments 決済オブジェクト』を参照してください。
Current State	状態名。状態の定義については、111 ページの『WebSphere Commerce Payments 決済オブジェクト』を参照してください。
TransactionId	ユーザーが AcceptPayment または ReceivePayment API に指定したトランザクション ID。
OrderData1	ユーザーが AcceptPayment または ReceivePayment API に指定した補助データ。
OrderData2	ユーザーが AcceptPayment または ReceivePayment API に指定した補助データ。
OrderData3	ユーザーが AcceptPayment または ReceivePayment API に指定した補助データ。
OrderData4	ユーザーが AcceptPayment または ReceivePayment API に指定した補助データ。
OrderData5	ユーザーが AcceptPayment または ReceivePayment API に指定した補助データ。

## カセット固有イベント

カセット固有イベントの場合、以下の表に定義されている名前と値の対だけでなく、各カセットで独自の名前と値の対も定義できます。各カセットの文書では、カセット固有の名前と値の対を詳細に示すとともに、カセット固有のイベントが送信される状況を定義する規則を示します。

カセット固有イベント

名前	値
EventType	「2」
CassetteName	<CassetteName> ASCII 文字ストリングの値。
MerchantNumber	ASCII 文字の整数。

## ネットワーク管理イベント

ネットワーク管理イベント

名前	値
EventType	「3」
ComponentName	値は以下のいずれかです。ASCII 文字ストリングで指定します。 <ul style="list-style-type: none"><li>• Framework</li><li>• &lt;CassetteName&gt;</li></ul>
Status	値は以下の整数値のいずれかです。ASCII 文字ストリングで指定します。 <ul style="list-style-type: none"><li>• 「1」: (実行するように指示します)</li><li>• 「2」: (実行しないように指示します)</li></ul>

たとえば、WebSphere Commerce Payments からイベント・リスナーに、以下の内容の状態変更イベントが送信されます。

```
EVENTTYPE=1  
TIMEGENERATED=  
MERCHANTNUMBER=  
PREVIOUSSTATE=  
CURRENTSTATE=  
OBJECT=  
ORDERNUMBER=  
PAYMENTNUMBER=  
CREDITNUMBER=  
BATCHNUMBER=  
ACCOUNTNUMBER=
```

WebSphere Commerce Payments からイベント・リスナーに、以下の内容のネットワーク管理イベントが送信されます。

```
EVENTTYPE=3  
TIMEGENERATED=  
COMPONENTNAME=  
STATUS=
```

---

## イベントの登録

マーチャント・ソフトウェアがイベントを受信するためには、マーチャント・ソフトウェアを WebSphere Commerce Payments に登録する必要があります。イベント・リスナーには、マーチャントと非マーチャントの 2 つのタイプがあります。マーチャント・リスナーは、マーチャント固有のイベント (すべての状態変更イベントおよびカセット固有のイベント) だけを登録できます。非マーチャント・ソフトウェア (ネットワーク管理システムなど) は、ネットワーク管理イベントだけを登録

できます。マーチャント・ソフトウェアも非マーチャント・ソフトウェアも、同じタイプのイベントを何回でも登録できます。その場合、イベントは登録された各ロケーションにブロードキャストされます。

イベント・リスナーを登録および管理するための API コマンドについては、55 ページの『第 6 章 WebSphere Commerce Payments コマンド・リファレンス』で説明します。

---

## Event ListenerURL パラメーター

イベント・リスナーの作成時には、有効な ListenerURL が必須キーワードです。WebSphere Commerce Payments では、有効なリスナー URL は、有効な Java URL として定義されます。同じ有効なリスナー URL でも形式が異なる場合があります。たとえば、`http://foo` と `http://foo/` は同じ URL ですが、`http://foo/xx` と `http://foo/xx/` は別の URL です。WebSphere Commerce Payments コマンドにより、有効な URL が WebSphere Commerce Payments の正規の URL 形式に変換されます。これは以下の拡張を伴う有効な URL です。

- ポート番号が定義されていない場合、WebSphere Commerce Payments コマンドによりポート番号「80」が挿入される。
- ホスト名が定義されていない場合、WebSphere Commerce Payments コマンドによりホスト名「localhost」が挿入される。
- 両方とも定義されていない場合、WebSphere Commerce Payments コマンドによりポート番号「80」とホスト名「localhost」が挿入される。

この正規の URL 形式を使用することによって、QueryEventListener コマンドは、入力 URL ストリングがやや異なっても同じリスナーを戻します。たとえば、リスナーのポート番号が 80 の場合は、URL に指定されているポート番号の値に関係なく同じリスナーが戻されます。



---

## 第 3 部 プログラミング・リファレンス



---

## 第 6 章 WebSphere Commerce Payments コマンド・リファレンス

ここで説明しているコマンドのパラメーターは、フレームワークだけに適用されるものです。カセットに固有の追加のパラメーターについては、該当するカセットの付録で説明しています。たいていの場合、WebSphere Commerce Payments は重複パラメーターを検査しないことに注意してください。複数のパラメーターが指定されている場合、その最後のインスタンスが使用されます。

クライアントは、キーワードと値のペアのリストを含む HTTP POST 要求を使用して、WebSphere Commerce Payments にコマンドを送信します。この章では、以下の点について説明します。

- WebSphere Commerce Payments の金融コマンドと管理コマンド
- コマンドの説明
- 必須キーワードとオプション・キーワードのリスト
- 決済コマンドと照会コマンドに関する指針

個々のコマンドには OPERATION という名前が含まれています。OPERATION パラメーターの値は、要求されるプロシーチャーを指定します。

OPERATION 以外にも、ETAPIVERSION は API のバージョン番号を指定します。ETAPIVERSION もすべてのコマンドで必須です。

個々のコマンドに含まれるそれ以外の名前と値のペアは、OPERATION の値によって異なります。決済操作に必須の名前と値のペアが、以下の表にリストされています。それ以外の名前と値のペアに関する指針は、以下のとおりです。

- キーワード・ストリングは大文字小文字の区別がない。
- ASCII 文字中で整数として先行ゼロを使用しない。

---

### 照会コマンド

以下の一般規則が、すべての照会に適用されます。

- 個々の照会には検索修飾子の集合と稼働パラメーターの集合がある。検索修飾子によって検索基準が判別され、稼働パラメーターはコマンドの動作や出力に影響を及ぼします。
- 金融照会を行うと、照会される基本オブジェクトの「コレクション」または「キー・コレクション」が必ず戻される。KEYONLY パラメーターの設定値によって、コレクションかキー・コレクションかが決められます。
- 複数回指定することにより、一連のオーダー値を検索できるキーワードもある (たとえば、STATE=batch\_opening、STATE=batch\_open、STATE=batch\_closed)。複数インスタンスをサポートしていないパラメーターの場合は、WebSphere Commerce Payments によりエラーは戻されず、どの値が使用されるか保証もされません。

- 照会結果のサイズを制御するために、アプリケーションでは RETURNATMOST パラメーターを使用できる。RETURNATMOST は、特定の照会で戻されるオブジェクトやオブジェクト ID の数を制限します。指定値が、照会と一致する実際のオブジェクトの数より小さくても適用されます。戻すことができるオブジェクトの最大数は 10,000 です。照会の結果の詳細については、46 ページの『戻されるオブジェクトの処理』を参照してください。
- 機密データを表示させるためにユーザーが最低限持っていなければならない役割を、指定することができる。各照会コマンドについてフレームワークはユーザーの役割と最低の役割とを照合し、機密データを丸ごと戻すべきか、マスクすべきかを示す標識を QueryRequest オブジェクトに設定します。WebSphere Commerce 構成マネージャーを使って設定できる JVM システム・パラメーター (wpm.MinSensitiveAccessRole) によって、機密データを表示させるためにユーザーが持っていなければならない役割 (クラーク、スーパーバイザー、マーチャント管理者、Payments 管理者、または指定なし) を指定できます。最小アクセス役割を構成マネージャーで設定することに関する詳細は、構成マネージャーのオンライン・ヘルプを参照してください。

---

## About

ABOUT コマンドは、一般的に次の 2 つの方法で使用されます。

- WebSphere Commerce Payments が実行しているかどうかを検査する PING メカニズムとして。
- WebSphere Commerce Payments のバージョン情報、インストールされているカセットの情報、およびコマンドを実行しているユーザー名を戻すため。

ABOUT コマンドによって戻される構造化された応答の詳細については、122 ページの『Payment Server About』および 122 ページの『Cassette About』を参照してください。

ABOUT コマンドが正常に実行されると、1 次および 2 次戻りコードとして "0"、"0" が戻されます。

ABOUT コマンドは、非認証ユーザーが実行できる唯一のコマンドです。このコマンドが非認証ユーザーによって実行されると、コマンドは 1 次および 2 次戻りコードだけを戻します。

About コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「About」

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ～ 254 バイトでなければなりません。

## AcceptPayment

ACCEPTPAYMENT コマンドは、電子ウォレットを使用しない場合に Order オブジェクトを作成するために使用します。一般的に、このコマンドが正常に実行されると、オーダーは「オーダー済み」状態になります。このコマンドが失敗すると、オーダーは作成されません。このコマンドにはプロトコル固有のデータを渡してください。しかし、固有データはカセットに応じて異なります。詳細については特定のカセットの補足を参照してください。

AcceptPayment コマンドの処理中、カセットに承認ステップをオーダー作成ステップとは別個に処理させることができます。承認の実行を非同期にスケジュールするよう指示するには、非同期自動承認決済処理オプションを選択します。こうすれば、バイヤーは承認の下りるのを待たずに、元の購入要求に対する応答を受け取ることができます。

オーダーを作成する際には、資金の承認やデポジットを自動的に行うこともできます。APPROVEFLAG キーワードは Payment オブジェクトを承認するかどうかを示し、DEPOSITFLAG キーワードは Payment オブジェクトをデポジットするかどうかを示します。APPROVEFLAG か DEPOSITFLAG を指定する場合に使用する追加キーワードについては、以下の該当する表を参照してください。

### AcceptPayment コマンドの必須キーワード

必須キーワード	値
AMOUNT	ASCII 文字の 32 ビット正整数。
CURRENCY	ASCII 文字の整数。ISO 通貨コードのリストについては、付録 B の『通貨コード』を参照してください。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ～ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「AcceptPayment」
ORDERNUMBER	ASCII 文字の整数。1 ～ 999999999 でなければなりません。
PAYMENTTYPE	ASCII 文字ストリング。使用される決済カセットまたはプロトコルを指定します。たとえば、OfflineCard です。

AcceptPayment コマンドのオプション・キーワード

オプション・キーワード	値
AMOUNTEXP10	ASCII 文字の整数。桁送りする桁数を指定します。有効な値は -10 ~ 10 です。このキーワードの詳細については、59 ページの『AmountExp10 キーワードの使用法』を参照してください。
APPROVEFLAG	ASCII 文字の整数。承認を自動的に試行する必要があるかどうかを示します。デフォルトは 0 です。以下の値がサポートされています。  0 - 取引を承認しないように指示します。  1 - 取引を自動的に承認するように指示します。  2 - 取引を非同期に承認するように指示します。
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ORDERDATA1	ユーザーが指定する補助データ。1 ~ 254 バイト長の ASCII 文字ストリングとして指定されます。
ORDERDATA2	ユーザーが指定する補助データ。1 ~ 254 バイト長の UTF-8 ストリングとして指定されます。
ORDERDATA3	ユーザーが指定する補助データ。1 ~ 254 バイト長の UTF-8 ストリングとして指定されます。
ORDERDATA4	ユーザーが指定する補助データ。1 ~ 254 バイト長のバイナリー・ストリングとして指定されます。
ORDERDATA5	ユーザーが指定する補助データ。任意の長さのバイナリー・ストリングとして指定されます。
ORDERURL	オーダーの詳細を含む URL
TRANSACTIONID	ユーザーが指定するトランザクション ID。1 ~ 128 バイト長の ASCII 文字ストリングとして指定されます。

以下の表には、APPROVEFLAG=1 または 2 の場合の必須キーワードとオプション・キーワードがリストされています。

APPROVEFLAG が 1 または 2 に設定されている場合の必須キーワード

必須キーワード	値
PAYMENTAMOUNT	ASCII 文字の 32 ビット正整数。
PAYMENTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。

APPROVEFLAG が 1 または 2 に設定されている場合のオプション・キーワード

オプション・キーワード	値
DEPOSITFLAG	ASCII 文字によるブール値。デポジットを自動的に試行する必要があるかどうかを示します。このフラグが有効なのは、APPROVE=1 (オーダーが自動的に承認される) のときだけです。以下の値がサポートされています。  0 - 資金を自動的にデポジットしてはならない。 1 - 資金を自動的にデポジットする必要がある。

DEPOSITFLAG=1 の場合、以下のキーワードはオプションです。

DEPOSITFLAG が 1 に設定されている場合のオプション・キーワード

オプション・キーワード	値
BATCHNUMBER	この決済が処理されるバッチを識別します。1 ~ 999999999 でなければなりません。

## AmountExp10 キーワードの使用法

金額に関する値はすべて金額 (Amount)、通貨 (Currency)、および指数 (Exponent) で表されます。たとえば、\$5.00 USD (U.S. ドル) は Amount=500、Currency=840 (USD の ISO 通貨コード)、および AmountExp10 =-2 で表されます。

現行の ISO 通貨にはすべて 1 つの有効な指数値があるので、通貨から指数を判断できます。付録 B の『通貨コード』で示されているように、WebSphere Commerce Payments には通貨から指数へのマッピング・テーブルがあります。オーダーの作成時 (つまり RECEIVEPAYMENT または ACCEPTPAYMENT コマンドの実行時) には、常にマーチャント・ソフトウェアにより AMOUNT キーワードと CURRENCY キーワードを両方とも指定しなければなりません。ISO テーブル中で認識されている通貨を指定すると、対応する指数が使用されます。認識されていない (つまり ISO テーブル中にない) 通貨を指定する場合は、指数を指定するには追加のパラメーター (AMOUNTEXP10) が必要です。AMOUNTEXP10 パラメーターを使用すると、将来の通貨をサポートする際に柔軟に対応できます。

API で指定されている AMOUNTEXP10	マッピング・テーブル中にある CURRENCY
真 (True)	真 (True)  マッピング・テーブル中にある指数が AMOUNTEXP10 パラメーターに渡される場合は、その指数が使用されます。  テーブル中のものとは違う指数が渡される場合は、パラメーター・エラーになります。
真 (True)	偽 (False)  AMOUNTEXP10 パラメーターに渡される指数が使用されます。

API で指定されている <b>AMOUNTEXP10</b>	マッピング・テーブル中にある <b>CURRENCY</b>
偽 (False)	真 (True)  マッピング・テーブルから導出された指数が使用されます。
偽 (False)	偽 (False)  「パラメーター検出不能 (parameter not found)」エラーになります。

## Approve

APPROVE コマンドは、マーチャントが金融システムに、バイヤーに購入許可を与えるべきかどうか問い合わせるために使用します。たとえば、クレジット・カード・システムの場合、このコマンドを実行するとクレジット・カードの与信が行われます。

APPROVE コマンドは、既存のオーダー用に新しい Payment オブジェクトを作成します。このコマンドは、オーダーが「オーダー済み」状態か「リファンド可能」状態にある場合に有効です。正常実行されると、DEPOSITFLAG が 1 に設定されている場合は決済は「承認」、「デPOSIT」、または「クローズ」状態になります。失敗すると、決済は「拒否」状態になります。

決済を承認する際に、自動的にデPOSITすることもできます。DEPOSITFLAG キーワードは、Payment オブジェクトをデPOSITする必要があることを示します。DEPOSITFLAG が 1 に設定されている場合の追加キーワードについては、以下の該当する表を参照してください。

Approve コマンドの必須キーワード

必須キーワード	値
AMOUNT	ASCII 文字の 32 ビット正整数。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「Approve」
ORDERNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
PAYMENTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。

Approve コマンドのオプション・キーワード

オプション・キーワード	値
DEPOSITFLAG	承認された決済を自動的にデPOSITする必要があるかどうかを示します。デフォルトは 0 です。以下の値がサポートされています。  0 - 資金を自動的にデPOSITしてはならない。  1 - 資金を自動的にデPOSITする必要がある。



Approve コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

DEPOSITFLAG=1 の場合は以下のキーワードはオプションです。

DEPOSITFLAG が 1 に設定されている場合のオプション・キーワード

オプション・キーワード	値
BATCHNUMBER	この決済が処理されるバッチを識別します。最大 9 文字の数値文字列です。1 ~ 999999999 でなければなりません。

## ApproveReversal

ApproveReversal コマンドは、承認された決済金額を修正します。たとえば、決済が「承認の有効期限切れ」状態になったら、ApproveReversal コマンドを使用して、新しい承認を得たり、決済を無効にしたりすることができます。ApproveReversal は、決済が「承認」状態にある場合に有効です。ApproveReversal が正常実行された場合に、指定された金額が「0」であれば、決済が「無効」状態に移行します。指定された金額が「0」でなければ、決済は「承認」状態のままで、承認額が変更されません。

ApproveReversal コマンドの必須キーワード

必須キーワード	値
AMOUNT	ASCII 文字の 32 ビット正整数でなければなりません。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字文字列「ApproveReversal」
ORDERNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
PAYMENTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。

ApproveReversal コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## BatchClose

BATCHCLOSE コマンドは、バッチをクローズし、Batch オブジェクトを「クローズ」状態に移行します。このバッチに関連した Payment オブジェクトと Credit オブジェクトもすべて「クローズ」状態に移行します。このコマンドは、以下の場合に限り使用できます。

- バッチが「オープン」状態にある。
- アカウントでマーチャントによるバッチのクローズを許可している。
- マーチャント制御属性が真に設定されている。

BatchClose コマンドの必須キーワード

必須キーワード	値
BATCHNUMBER	最大 9 文字の数値ストリングです。1 ~ 999999999 でなければなりません。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「BatchClose」

BatchClose コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
FORCE	有効な値は「0」と「1」です。値「1」は、金融操作が失敗した場合でも、ローカルのクローズを実行することを示します。

## BatchOpen

BATCHOPEN コマンドは、Batch オブジェクトを作成し、正常に作成するとそのバッチを「オープン」状態にします。このコマンドは、マーチャントがバッチをオープンすることをアカウントで許可している場合だけ使用できます。

**注:** 1 つのマーチャント (123456789) と 2 つのアカウント (acct#1、acct#2) のシナリオで、BatchOpen に acct#1、batch#1 を指定して発行すると、バッチはオープンされます。BatchOpen に acct#2、batch#1 を指定して送信すると、BatchOpen は失敗し、以下のメッセージが表示されます。

```
Tue Jun22 13:04:31 EDT 1999 CEPFW0715: Batch ID 299 already exists for Merchant 123456789 and account 2.
```

後者のテストが失敗する理由は、指定の BatchNumber のバッチは一度に 1 つしかシステムに存在できないからです。

#### BatchOpen コマンドの必須キーワード

必須キーワード	値
OPERATION	ASCII 文字ストリング「BatchOpen」
ACCOUNTNUMBER	ASCII 文字の整数。この値は、マーチャントに決済機関を指示するための固有の ID です。この値は、WebSphere Commerce Payments の構成済み AccountNumber 値と一致しなければなりません。1 ~ 999999999 でなければなりません。
BATCHNUMBER	最大 9 文字の数値ストリングです。1 ~ 999999999 でなければなりません。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
PAYMENTTYPE	決済カセットまたはプロトコルを識別する ASCII 文字ストリング。

#### BatchOpen コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## BatchPurge

BATCHPURGE コマンドは、バッチをすべて消去し、Batch オブジェクトを「オープン」状態に戻します。このバッチに関連した Payment オブジェクトと Credit オブジェクトはすべてバッチから除去されます。その際、Payment オブジェクトは「承認」状態に戻され、Credit オブジェクトは「無効」状態に戻されます。このコマンドは、PurgeAllowed 属性が真に設定されている場合だけ使用できます。

#### BatchPurge コマンドの必須キーワード

必須キーワード	値
BATCHNUMBER	最大 9 文字の数値ストリングです。1 ~ 999999999 でなければなりません。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)

BatchPurge コマンドの必須キーワード

必須キーワード	値
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「BatchPurge」

BatchPurge コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。 このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。 DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## CancelOrder

CANCELORDER コマンドは、オーダーを「キャンセル」状態に移行します。  
CancelOrder コマンドは、以下の基準を満たすオーダーに対して呼び出すことができます。

- 関連した決済またはクレジットがない。または
- 関連した決済またはクレジットは、それぞれの「リセット」、「無効」、「承認の有効期限切れ」、または「拒否」状態にある。

オーダーが「キャンセル」状態になると、CancelOrder 以外の操作はすべて無効になります。オプション・パラメーター DELETEDORDER を「1」に設定すると、オーダーは取り除かれます。関連した決済とクレジットもすべて削除されます。カセットごとに固有のオブジェクトも削除されます。

CancelOrder コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「CancelOrder」
ORDERNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。

#### CancelOrder コマンドのオプション・キーワード

オプション・キーワード	値
DELETEORDER	<p>オーダーとすべての補助オブジェクトを削除する必要があることを示します。デフォルトは「0」です。以下の値がサポートされています。</p> <p>0 - オブジェクトを削除してはならない。</p> <p>1 - オブジェクトを削除する必要がある。</p>
DTDPATH	<p>ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。</p>

## CassetteControl

CASSETTECONTROL コマンドは、どの汎用コマンドにも対応しない、カセットごとに固有のファンクションを実行するために使用します。CASSETTECONTROL はフレームワークでは解釈されず、カセットに渡されます。

このコマンドの使用法に関する詳細は、該当するカセットの付録を参照してください。

#### CassetteControl コマンドの必須キーワード

必須キーワード	値
CASSETTECOMMAND	ASCII 文字のコマンド名。最大長は 1000 バイトです。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「CassetteControl」。
PAYMENTTYPE	ASCII 文字ストリング。使用される決済カセットまたはプロトコルを指定します。

#### CassetteControl コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	<p>ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。</p>

## CloseOrder

CloseOrder コマンドは、オーダーを「クローズ」状態に移行します。 CloseOrder コマンドは、以下の基準を満たすオーダーに対して呼び出すことができます。

- 少なくとも 1 つの決済またはクレジットがある。および
- オーダーに関連したすべての決済およびクレジットが、それぞれ「クローズ」状態にある。

オーダーが「クローズ」状態になると、CancelOrder 以外の操作はすべて無効になります。オプション・パラメーター DELETEORDER を「1」に設定すると、データベースは取り除かれるので、「クローズ」状態のオーダーに対して CloseOrder を呼び出せます。決済とクレジットはクローズされていなければなりません。

CloseOrder の必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「CloseOrder」。
ORDERNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。

CloseOrder コマンドのオプション・キーワード

オプション・キーワード	値
DELETEORDER	オーダーとすべての補助オブジェクトを削除する必要があることを示します。デフォルトは 0 です。以下の値がサポートされています。  0 - オーダーとすべての補助オブジェクトを削除してはならない。  1 - オーダーとすべての補助オブジェクトを削除する必要がある。
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## CreateAccount

CREATEACCOUNT コマンドは、指定された Payment System オブジェクト用の Account オブジェクトを作成します。

CreateAccount コマンドの必須キーワード

必須キーワード	値
ACCOUNTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。  新しいアカウントに ID を指定します。
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。  新しいアカウントに ID を指定します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。  新しいアカウントに ID を指定します。
OPERATION	ASCII 文字ストリング「CreateAccount」

CreateAccount コマンドのオプション・キーワード

オプション・キーワード	値
ACCOUNTTITLE	ヌルまたは 1 ~ 254 バイトの UTF-8 文字ストリング。値を指定すると、渡された値は、Account オブジェクトの「AccountTitle」に指定されている値と置き換わります。
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字 (「0」は偽、「1」は真)。値を指定すると、渡された値は、Account オブジェクトの「Enabled」フィールドと置き換わります。  Account オブジェクトをアクティブにする必要があるかどうかを示します。
FINANCIALINSTITUTION	ヌルまたは 1 ~ 254 バイトの UTF-8 文字ストリング。値を指定すると、渡された値は、Account オブジェクトの「Financial Institution」に指定されている値と置き換わります。

CreateAccount コマンドのオプション・キーワード

オプション・キーワード	値
APAPPROVEFLAG	AcceptPayment 用の承認フラグ。「0」、「1」、または「2」の ASCII 文字。デフォルトは「0」です。  「0」は、取引を承認しないように指示します。  「1」は、取引を自動的に承認するように指示します。  「2」は、取引を非同期に承認するように指示します。
RPAPPROVEFLAG	ReceivePayment 用の承認フラグ。「0」、「1」、または「2」の ASCII 文字。デフォルトは「0」です。  「0」は、取引を承認しないように指示します。  「1」は、取引を自動的に承認するように指示します。  「2」は、取引を非同期に承認するように指示します。
APDEPOSITFLAG	「0」または「1」の ASCII 文字（「0」は偽、「1」は真）。APAPPROVEFLAG が定義されており、0 に設定されていない場合にのみ指定します。そうでない場合は、PRC_INVALID_PARAMETER_COMBINATION_、RC_AP_DEPOSITFLAG が戻されます。
RPDEPOSITFLAG	「0」または「1」の ASCII 文字（「0」は偽、「1」は真）。RPAPPROVEFLAG が定義されており、0 に設定されていない場合にのみ指定します。そうでない場合は、PRC_INVALID_PARAMETER_COMBINATION_、RC_RP_DEPOSITFLAG が戻されます。
APPROVALEXPIRATION	決済の承認後、何日経過したら承認の有効期限が切れるかを示す整数値。このフィールドは構成可能な承認有効期限をサポートしています。つまり、この設定により、一定時間の経過後に、アカウントに関係した決済承認の有効期限が切れるかどうかは制御されます。値 0 は、有効期限がないことを意味します。決済の承認の有効期限が切れると、「承認の有効期限切れ」状態になります。  注: カセットは、この設定とは無関係に決済の承認を有効期限切れにすることができますが、このパラメータにより、フレームワークはカセットのために決済の承認の有効期限を検出できるようになります。「承認の有効期限切れ」状態の説明は、『Payment の状態』を参照してください。

注: APAPPROVEFLAG および RPAPPROVEFLAG 値は、API 承認フラグに非ゼロ、非空値が含まれるとき、API 承認フラグによって取って代わられます。



## CreateMerchant

CREATEMERCHANT コマンドは Merchant オブジェクトを作成します。

CreateMerchant コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。 新しい Merchant オブジェクトの ID を指定します。
OPERATION	ASCII 文字ストリング「CreateMerchant」。

CreateMerchant コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字 (「0」は偽、「1」は真)。値を指定すると、渡された値は、Merchant オブジェクトの「Enabled」フィールドと置き換わります。 Merchant オブジェクトをアクティブにする必要があるかどうかを示します。
MERCHANTTITLE	ヌルまたは 1 ~ 128 バイトの UTF-8 文字ストリング。値を指定すると、その値は渡されて、Merchant オブジェクトの「MerchantTitle」に指定されている値と置き換わります。

## CreateMerchantCassetteObject

CREATEMERCHANTCASSETTEOBJECT コマンドを使用して、OBJECTNAME キーワードに指定されたタイプの、カセットごとに固有のオブジェクトを作成します。

このコマンドの使用法に関する詳細は、該当するカセットの付録を参照してください。

CreateMerchantCassetteObject コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。 MerchantCassette オブジェクトの ID を指定します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。 MerchantCassette オブジェクトの ID を指定します。

CreateMerchantCassetteObject コマンドの必須キーワード

必須キーワード	値
OBJECTNAME	ASCII 文字ストリング。カセットに指定されている値。 MerchantCassette オブジェクトの ID を指定します。最大長は 1000 バイトです。
OPERATION	ASCII 文字ストリング「CreateMerchantCassetteObject」

CreateMerchantCassetteObject コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字（「0」は偽、「1」は真）。値を指定すると、渡された値は、MerchantCassette オブジェクトの「Enabled」フィールドと置き換わります。 MerchantCassette オブジェクトを使用可能にする必要があるかどうかを示します。

## CreateMerEventListener

CREATEMEREVENTLISTENER コマンドはマーチャント・イベントの listener を作成します。

CreateMerEventListener コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」（WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す）
EVENTTYPE	イベント・タイプを識別する、ASCII 文字の整数。イベントの値は以下のとおりです。 1: 状態変更イベント 2: カセットごとに固有のイベント
LISTENERURL	イベントの表示場所 ( <a href="http://www.merchant.com/webapp/PaymentManager/eventReceiver888">http://www.merchant.com/webapp/PaymentManager/eventReceiver888</a> など) を識別する ASCII 文字ストリング。WebSphere Commerce 構成マネージャーを使ってポート番号を指定する必要があります。1 ~ 256 文字の有効な URL。
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「CreateMerEventListener」。

#### CreateMerEventListener コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ～ 254 バイトです。
SOCKSHOST	Socks サーバーのホスト名。このパラメーターは、Socks サーバーを介して送信されるイベントだけに必須です。最大長は 256 バイトです。
SOCKSPORT	Socks サーバーのポート番号。このパラメーターは、SOCKSHOST を指定した場合だけ使用します。デフォルトは 1080 です。SocksPort パラメーターの値 (非 nul 文字) は、1 ～ 65535 の無符号 16 ビット正整数でなければなりません。

#### EventType を 2 に設定した場合の必須キーワード

必須キーワード	値
CASSETTENAME	カセットの名前を識別する ASCII 文字ストリング (最大 64 バイト)。カセット・イベントを登録する場合に必須です。パラメーターの制限はありませんが、既存のカセットと一致していなければなりません。一致していないと失敗します。

## CreatePaySystem

CREATEPAYSYSTEM コマンドは、Payment System オブジェクトを作成し、指定されたカセットの使用許可を指定されたマーチャントに割り当てます。

#### CreatePaySystem コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ～ 64 バイトの ASCII 文字ストリング。 新しい PaySystem オブジェクトの ID を指定します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ～ 99999999 でなければなりません。 新しい PaySystem オブジェクトの ID を指定します。
OPERATION	ASCII 文字ストリング「CreatePaySystem」。

#### CreatePaySystem コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ～ 254 バイトでなければなりません。

CreatePaySystem コマンドのオプション・キーワード

オプション・キーワード	値
ENABLED	「0」または「1」の ASCII 文字（「0」は偽、「1」は真）。値を指定すると、渡された値は、PaySystem オブジェクトの「Enabled」フィールドと置き換わります。  PaySystem オブジェクトをアクティブにする必要があるかどうかを示します。

---

## CreateSNMEventListener

CREATESNMEVENTLISTENER コマンドは、システム・ネットワーク管理イベントの listener を作成します。

CreateSNMEventListener コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」（WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す）
EVENTTYPE	「3」（SNM イベント・タイプを示す）他の値は将来の利用のために予約済み。
LISTENERURL	イベントの表示場所（ <a href="http://www.merchant.com/webapp/PaymentManager/eventReceiver888">http://www.merchant.com/webapp/PaymentManager/eventReceiver888</a> など）を識別する ASCII 文字ストリング。WebSphere Commerce 構成マネージャーを使ってポート番号を指定する必要があります。1 ～ 256 文字の有効な URL。
OPERATION	ASCII 文字ストリング「CreateSNMEventListener」。

CreateSNMEventListener コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ～ 254 バイトでなければなりません。
SOCKSHOST	Socks サーバーのホスト名。このパラメーターは、Socks サーバーを介して送信されるイベントだけに必須です。パラメーター値は有効な整数でなければなりません（指定する場合）。最大長は 256 バイトです。
SOCKSPORT	Socks サーバーのポート番号。このパラメーターは、SOCKSHOST を指定した場合だけ使用します。デフォルトは 1080 です。SocksPort パラメーターの値（非ヌル文字）は、1 ～ 65535 の無符号 16 ビット正整数でなければなりません。

---

## CreateSystemCassetteObject

CREATESYSTEMCASSETTEOBJECT コマンドは、OBJECTNAME キーワードに指定されたタイプの、カセットごとに固有のオブジェクトを作成します。

このコマンドの使用法に関する詳細は、該当するカセットの付録を参照してください。

CreateSystemCassetteObject コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。 SystemCassette オブジェクトの ID を指定します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
OBJECTNAME	ASCII 文字ストリング。カセットに指定されている値。 SystemCassette オブジェクトの ID を指定します。
OPERATION	ASCII 文字ストリング「CreateSystemCassetteObject」。

CreateSystemCassetteObject コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメータの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメータを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字（「0」は偽、「1」は真）。値を指定すると、渡された値は、SystemCassette オブジェクトの「Enabled」フィールドと置き換わります。  SystemCassette オブジェクトをアクティブにする必要があるかどうかを示します。

## DeleteAccount

DELETEACCOUNT コマンドは、指定された Account オブジェクトとその補助オブジェクトすべてを削除します。

DeleteAccount コマンドの必須キーワード

必須キーワード	値
ACCOUNTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。MERCHANTNUMBER および CASSETTENAME と共に、このコマンドのターゲットの Account オブジェクトを固有に識別します。
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。MERCHANTNUMBER および ACCOUNTNUMBER と共に、このコマンドのターゲットの Account オブジェクトを固有に識別します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)

## DeleteAccount コマンドの必須キーワード

必須キーワード	値
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。CASSETTENAME および ACCOUNTNUMBER と共に、このコマンドのターゲットの Account オブジェクトを固有に識別します。
OPERATION	ASCII 文字ストリング「DeleteAccount」。

## DeleteAccount コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

---

## DeleteBatch

DELETEBATCH コマンドは、指定されたバッチをデータベース表からプルーニングします。DELETEBATCH コマンドは、バッチが「クローズ」状態にある場合だけ有効です。

## DeleteBatch コマンドの必須キーワード

必須キーワード	値
BATCHNUMBER	ASCII 文字の整数。この決済が割り当てられるバッチの番号を識別します。1 ~ 999999999 でなければなりません。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「DeleteBatch」。

## DeleteBatch コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## DeleteMerchant

DELETEMERCHANT コマンドは、指定された Merchant オブジェクトとその補助オブジェクトすべてを削除します。

DeleteMerchant コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。このコマンドのターゲットの Merchant オブジェクトを使用します。
OPERATION	ASCII 文字ストリング「DeleteMerchant」。

DeleteMerchant コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## DeleteMerchantCassetteObject

DELETEMERCHANTCASSETTEOBJECT コマンドは、オブジェクト名で指定されるタイプの Cassette オブジェクトを削除します。

このコマンドの使用法に関する詳細は、該当するカセットの付録を参照してください。

DeleteMerchantCassetteObject コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。 MERCHANTNUMBER、OBJECTNAME、およびプロトコル・データ・パラメーターと共に、このコマンドのターゲットの MerchantCassette を固有に識別します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。 CASSETTENAME、OBJECTNAME、およびデータ・パラメーターと共に、このコマンドのターゲットの MerchantCassette を固有に識別します。
OBJECTNAME	カセットに指定されている ASCII 文字ストリング値。 CASSETTENAME、MERCHANTNUMBER、およびプロトコル・データ・パラメーターと共に、このコマンドのターゲットの MerchantCassette オブジェクトを固有に識別します。最大長は 1000 バイトです。
OPERATION	ASCII 文字ストリング「DeleteMerchantCassetteObject」。

DeleteMerchantCassetteObject コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## DeleteMerEventListener

DELETEMEREVENTLISTENER コマンドは、MerEventListener オブジェクトを削除します。

DeleteMerEventListener コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	カセットの名前を識別する ASCII 文字ストリング (最大 64 バイト)。カセット・イベントを登録する場合に必須です。パラメーターの制限はありませんが、既存のカセットと一致していなければなりません。一致していないと失敗します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
EVENTTYPE	イベント・タイプを識別する、ASCII 文字の整数。イベントの値は以下のとおりです。  1: 状態変更イベント  2: カセットごとに固有のイベント
LISTENERURL	イベントの表示場所 ( <a href="http://www.merchant.com/webapp/PaymentManager/eventReceiver888">http://www.merchant.com/webapp/PaymentManager/eventReceiver888</a> など) を識別する ASCII 文字ストリング。WebSphere Commerce 構成マネージャーを使ってポート番号を指定する必要があります。パラメーターの制限はありません。
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「DeleteMerEventListener」。

DeleteMerEventListener コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。



---

## DeletePaySystem

DELETEDPAYSYSTEM コマンドは、指定された Payment System オブジェクトとその補助オブジェクトすべてを削除します。

DeletePaySystem コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。 MERCHANTNUMBER と共に、このコマンドのターゲットの MerchantCassetteSettings オブジェクトを固有に識別します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 99999999 でなければなりません。 CASSETTENAME と共に、このコマンドの MerchantCassetteSettings オブジェクトを固有に識別します。
OPERATION	ASCII 文字ストリング「DeletePaySystem」。

DeletePaySystem コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

---

## DeleteSNMEventListener

DELETESNMEVENTLISTENER コマンドは、指定されたシステム・ネットワーク管理イベントの listener を削除します。

DeleteSNMEventListener コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
EVENTTYPE	「3」(SNM イベント・タイプを示す。他の値は将来の利用のために予約済み。)
LISTENERURL	イベントの表示場所 ( <a href="http://www.merchant.com/webapp/PaymentManager/eventReceiver888">http://www.merchant.com/webapp/PaymentManager/eventReceiver888</a> など) を識別する ASCII 文字ストリング。WebSphere Commerce 構成マネージャーを使ってポート番号を指定する必要があります。1 ~ 256 文字の有効な URL。
OPERATION	ASCII 文字ストリング「DeleteSNMEventListener」。

DeleteSNMEventListener コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

---

## DeleteSystemCassetteObject

DELETESYSTEMCASSETTEOBJECT コマンドは、オブジェクト名で指定されるタイプの Cassette オブジェクトを削除します。

このコマンドの使用法に関する詳細は、該当するカセットの付録を参照してください。

DeleteSystemCassetteObject コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。OBJECTNAME およびプロトコル・データ・パラメーターと共に、このコマンドのターゲットの SystemCassette オブジェクトを固有に識別します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OBJECTNAME	ASCII 文字ストリング。カセットに指定されている値。CASSETTENAME およびプロトコル・データ・パラメーターと共に、このコマンドのターゲットの SystemCassette オブジェクトを固有に識別します。最大長は 1000 バイトです。
OPERATION	ASCII 文字ストリング「DeleteSystemCassetteObject」。

DeleteSystemCassetteObject コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

---

## Deposit

DEPOSIT コマンドは、指定された決済をバッチと関連付けてから、以前にその決済に関して承認されていたお金をデポジットします。DEPOSIT コマンドは、「承認」状態でデポジットの操作が行われる場合に有効です。

正常に実行されると、指定された決済は「デポジット」状態に移行します。

#### Deposit コマンドの必須キーワード

必須キーワード	値
AMOUNT	ASCII 文字の 32 ビット整数でなければなりません。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「Deposit」。
ORDERNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
PAYMENTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。

#### Deposit コマンドのオプション・キーワード

オプション・キーワード	値
BATCHNUMBER	この決済が処理されるバッチを識別します。 1 ~ 999999999 でなければなりません。
DTDPATH	ローカルに保管されている DTD へのパス。 このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。 DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## DepositReversal

DEPOSITREVERSAL コマンドは、決済とバッチとの関連付けを解除します。このコマンドは、決済が「デポジット」状態にある場合に有効です。正常に実行されると、決済は「承認」状態または「無効」状態に移行し、デポジットの金額は「0」にリセットされます。

#### DepositReversal コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「DepositReversal」。
ORDERNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
PAYMENTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。

DepositReversal コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。 このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。 DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## ModifyAccount

MODIFYACCOUNT コマンドは、指定された Account オブジェクトの属性を変更するために使用されます。

ModifyAccount コマンドの必須キーワード

必須キーワード	値
ACCOUNTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。  MERCHANTNUMBER および CASSETTENAME と共に、このコマンドのターゲットの Account を固有に識別します。
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。  MERCHANTNUMBER および ACCOUNTNUMBER と共に、このコマンドのターゲットの Account を固有に識別します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。  CASSETTENAME および ACCOUNTNUMBER と共に、このコマンドのターゲットの Account を固有に識別します。
OPERATION	ASCII 文字ストリング「ModifyAccount」。

ModifyAccount コマンドのオプション・キーワード

オプション・キーワード	値
ACCOUNTTITLE	ヌルまたは 1 ~ 254 バイトの UTF-8 文字ストリング。 値を指定すると、渡された値は、Account オブジェクトの「AccountTitle」に指定されている値と置き換わります。
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字（「0」は偽、「1」は真）。値を指定すると、渡された値は、Account オブジェクトの「Enabled」フィールドと置き換わります。  Account オブジェクトをアクティブにする必要があるかどうかを示します。

ModifyAccount コマンドのオプション・キーワード

オプション・キーワード	値
FINANCIALINSTITUTION	ヌルまたは 1 ~ 255 バイトの UTF-8 文字ストリング。値を指定すると、その値は渡されて、Account オブジェクトの「FinancialInstitution」に指定されている値と置き換わります。
APAPPROVEFLAG	AcceptPayment 用の承認フラグ。「0」、「1」、または「2」の ASCII 文字。デフォルトは「0」です。  「0」は、取引を承認しないように指示します。  「1」は、取引を自動的に承認するように指示します。  「2」は、取引を非同期に承認するように指示します。
RPAPPROVEFLAG	ReceivePayment 用の承認フラグ。「0」、「1」、または「2」の ASCII 文字。デフォルトは「0」です。  「0」は、取引を承認しないように指示します。  「1」は、取引を自動的に承認するように指示します。  「2」は、取引を非同期に承認するように指示します。
APDEPOSITFLAG	「0」または「1」の ASCII 文字（「0」は偽、「1」は真）。APAPPROVEFLAG が定義されており、0 に設定されていない場合のみ指定します。そうでない場合は、PRC_INVALID_PARAMETER_COMBINATION_、RC_AP_DEPOSITFLAG が戻されます。
RPDEPOSITFLAG	「0」または「1」の ASCII 文字（「0」は偽、「1」は真）。RPAPPROVEFLAG が定義されており、0 に設定されていない場合のみ指定します。そうでない場合は、PRC_INVALID_PARAMETER_COMBINATION_、RC_RP_DEPOSITFLAG が戻されます。
APPROVALEXPIRATION	決済の承認後、何日経過したら承認の有効期限が切れるかを示す整数値。このフィールドは構成可能な承認有効期限をサポートしています。つまり、この設定により、一定時間の経過後に、アカウントに関係した決済承認の有効期限が切れるかどうか制御されます。値 0 は、有効期限がないことを意味します。決済の承認の有効期限が切れると、「承認の有効期限切れ」状態になります。 <b>注:</b> カセットは、この設定とは無関係に決済の承認を有効期限切れにすることができますが、このパラメーターにより、フレームワークはカセットのために決済の承認の有効期限を検出できるようになります。「承認の有効期限切れ」状態の説明は、117 ページの『Payment の状態』を参照してください。

**注:** APAPPROVEFLAG および RPAPPROVEFLAG 値は、API 承認フラグに非ゼロ、非空値が含まれるとき、API 承認フラグによって延期済みにされます。

## ModifyCassette

MODIFYCASSETTE コマンドは、指定された Cassette オブジェクトの属性を修正するため使用します。

ModifyCassette コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。このコマンドのターゲットの Cassette オブジェクトを識別します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「ModifyCassette」。

ModifyCassette コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字 (「0」は偽、「1」は真)。値を指定すると、渡された値は、Cassette オブジェクトの「Enabled」フィールドと置き換わります。  Cassette オブジェクトをアクティブにする必要があるかどうかを示します。

## ModifyMerchant

MODIFYMERCHANT コマンドは、指定された Merchant オブジェクトの属性を修正します。

ModifyMerchant コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。  このコマンドのターゲットの Merchant オブジェクトを識別します。
OPERATION	ASCII 文字ストリング「ModifyMerchant」。

### ModifyMerchant コマンドのオプション・キーワード

必須キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ～ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字 (「0」は偽、「1」は真)。値を指定すると、渡された値は、Merchant オブジェクトの「Enabled」フィールドと置き換わります。  Merchant オブジェクトをアクティブにする必要があるかどうかを示します。
MERCHANTTITLE	ヌルまたは 1 ～ 128 バイトの UTF-8 文字ストリング。値を指定すると、渡された値は、Merchant オブジェクトの「MerchantTitle」に指定されている値と置き換わります。

## ModifyMerchantCassetteObject

MODIFYMERCHANTCASSETTEOBJECT コマンドは、オブジェクト名で指定されるタイプの Cassette オブジェクトの属性を修正します。

このコマンドの使用法に関する詳細は、該当するカセットの付録を参照してください。

### ModifyMerchantCassetteObject コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ～ 64 バイトの ASCII 文字ストリング。  MerchantCassette オブジェクトの ID を指定します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ～ 99999999 でなければなりません。  CASSETTENAME、OBJECTNAME、およびプロトコル・データ・パラメーターと共に、このコマンドのターゲットの MerchantCassette オブジェクトを固有に識別します。
OBJECTNAME	ASCII 文字ストリング。カセットに指定されている値。  CASSETTENAME、MERCHANTNUMBER、およびプロトコル・データ・パラメーターと共に、このコマンドのターゲットの MerchantCassette を固有に識別します。最大長は 1000 バイトです。
OPERATION	ASCII 文字ストリング「ModifyMerchantCassetteObject」。

ModifyMerchantCassetteObject コマンドのオプション・キーワード

オプション・キーワード	値
DTDSPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPath の長さは 1 ～ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字 (「0」は偽、「1」は真)。値を指定すると、渡された値は、MerchantCassette オブジェクトの「Enabled」フィールドと置き換わります。  MerchantCassette オブジェクトを使用可能にする必要があるかどうかを示します。

## ModifyMerEventListener

MODIFYMEREVENTLISTENER コマンドは、指定された MerEventListener オブジェクトを修正します。

ModifyMerEventListener コマンドの必須キーワード

必須キーワード	値
ENABLED	1 (真) または 0 (偽) に設定できます。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
EVENTTYPE	イベント・タイプを識別する、ASCII 文字の整数。イベントの値は以下のとおりです。  1: 状態変更イベント  2: カセットごとに固有のイベント
LISTENERURL	イベントの表示場所 ( <a href="http://www.merchant.com/webapp/PaymentManager/eventReceiver888">http://www.merchant.com/webapp/PaymentManager/eventReceiver888</a> など) を識別する ASCII 文字ストリング。WebSphere Commerce 構成マネージャーを使ってポート番号を指定する必要があります。パラメーターの制限はありません。
MERCHANTNUMBER	ASCII 文字の整数。1 ～ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「ModifyMerEventListener」。

ModifyMerEventListener コマンドのオプション・キーワード

オプション・キーワード	値
DTDSPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPath の長さは 1 ～ 254 バイトでなければなりません。



EventType を 2 に設定した場合の必須キーワード

必須キーワード	値
CASSETTENAME	カセットの名前を識別する ASCII 文字ストリング (最大 64 バイト)。カセット・イベントを修正する場合に必須です。パラメーターの制限はありません。既存のカセットと一致していなければなりません。

---

## ModifyPayServer

MODIFYPAYSERVER コマンドは、Payment Server オブジェクトのグローバル属性を修正します。

ModifyPayServer コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「ModifyPayServer」。

ModifyPayServer コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字 (「0」は偽、「1」は真)。値を指定すると、渡された値は、PayServer オブジェクトの「Enabled」フィールドと置き換わります。  PaymentServer オブジェクトをアクティブにする必要があるかどうかを示します。
ETILLHOSTNAME	ヌルまたは 1 ~ 254 文字の ASCII 文字ストリング。値を指定すると、渡された値は、PaymentServer オブジェクトの「ETillHostname」フィールドと置き換わります。  非ヌル値は、WebSphere Commerce Payments へのメッセージ送信時に送信する必要がある DNS ホスト名を示します。ヌル値は、DNS を検索して値を判別する必要があることを示します。

---

## ModifyPaySystem

MODIFYPAYSYSTEM コマンドは、指定された Payment System オブジェクトを修正します。

### ModifyPaySystem コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。  MERCHANTNUMBER と共に、このコマンドのターゲットの Payment System オブジェクトを固有に識別します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。  CASSETTENAME と共に、このコマンドのターゲットの PaymentSystem オブジェクトを固有に識別します。
OPERATION	ASCII 文字ストリング「ModifyPaySystem」。

### ModifyPaySystem コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字 (「0」は偽、「1」は真)。値を指定すると、渡された値は、ModifyPaySystem オブジェクトの「Enabled」フィールドと置き換わります。  ModifyPaySystem オブジェクトをアクティブにする必要があるかどうかを示します。

## ModifySNMEventListener

MODIFYSNMEVENTLISTENER コマンドは、System Network Management Event Listener オブジェクトを修正します。

### ModifySNMEventListener コマンドの必須キーワード

必須キーワード	値
ENABLED	1 (真) または 0 (偽) に設定できます。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
EVENTTYPE	3: SNM イベント・タイプを示します。他の値は将来の利用のために予約済み。
LISTENERURL	イベントの表示場所 ( <a href="http://www.merchant.com/webapp/PaymentManager/eventReceiver888">http://www.merchant.com/webapp/PaymentManager/eventReceiver888</a> など) を識別する ASCII 文字ストリング。WebSphere Commerce 構成マネージャーを使ってポート番号を指定する必要があります。1 ~ 256 文字の有効な URL。
OPERATION	ASCII 文字ストリング「ModifySNMEventListener」。

ModifySNMEventListener コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## ModifySystemCassetteObject

MODIFYSYSTEMCASSETTEOBJECT コマンドは、オブジェクト名で指定されるタイプの Cassette オブジェクトの属性を修正します。

このコマンドの使用法に関する詳細は、該当するカセットの付録を参照してください。

ModifySystemCassetteObject コマンドの必須キーワード

必須キーワード	値
CASSETTENAME	1 ~ 64 バイトの ASCII 文字ストリング。  OBJECTNAME およびプロトコル・データ・パラメーターと共に、このコマンドのターゲットの SystemCassette オブジェクトを固有に識別します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OBJECTNAME	ASCII 文字ストリング。カセットに指定されている値。  CASSETTENAME およびプロトコル・データ・パラメーターと共に、このコマンドのターゲットの SystemCassette オブジェクトを固有に識別します。最大長は 1000 バイトです。
OPERATION	ASCII 文字ストリング「ModifySystemCassetteObject」。

ModifySystemCassetteObject コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ENABLED	「0」または「1」の ASCII 文字 (「0」は偽、「1」は真)。値を指定すると、渡された値は、SystemCassette オブジェクトの「Enabled」フィールドと置き換わります。  SystemCassette オブジェクトをアクティブにする必要があるかどうかを示します。

## ModifyUserStatus

このコマンドは、WebSphere Commerce Payments に対するアクセス権のあるユーザーの状況を変更します。この機能に対してアクセス制御権があるのは、Payments 管理者とマーチャント管理者のみです。マーチャント管理者は、自分のマーチャントのユーザーの「ユーザー状況を修正する」ことだけを行えます。

ModifyUserStatus コマンドの必須キーワード

必須キーワード	値
ENABLED	1 (真) または 0 (偽) に設定できます。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	文字ストリング形式の数値のマーチャント番号。このキーワードは、マーチャント管理者が要求を発行する場合に必須です。
OPERATION	ASCII 文字ストリング「ModifyUserStatus」。
USER	ユーザー ID 文字を含むバイト配列。1 ~ 80 文字の ASCII 文字ストリング。
ROLE	それぞれの WebSphere Commerce Payments 役割に割り当てられる値。指定された値については、103 ページの表 5 を参照してください。

## QueryAccounts

QUERYACCOUNTS コマンドは、Account オブジェクトのコレクションを XML 形式で戻します。

QueryAccounts コマンドの必須キーワードと稼働パラメーター

必須キーワード	複数使用できるかどうか	値
ETAPIVERSION	N	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	N	ASCII 文字ストリング「QueryAccounts」。

QueryAccounts コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

QueryAccounts コマンドの検索修飾子

オプション・キーワード	複数使用できるかどうか	値
ACCOUNTNUMBER	Y	アカウント番号。ASCII 文字の整数。1 ~ 999999999 でなければなりません。
CASSETTENAME	Y	カセットの名前。最大長 64 バイトの ASCII 文字ストリング。
MERCHANTNUMBER	Y	マーチャント番号。ASCII 文字の整数。1 ~ 999999999 でなければなりません。

## QueryBatches

QUERYBATCHES コマンドは、WebSphere Commerce Payments Batch オブジェクトまたはバッチ・キーのコレクションを戻します。

QueryBatches コマンドの必須キーワードと稼働パラメーター

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「QueryBatches」。

QueryBatches コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ASCII 文字ストリング。ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
KEYSONLY	1: 実際のオブジェクトが戻されるのではなく、固有のバッチ ID のリスト (「オーダー番号: バッチ番号」の形式) が戻される必要があります。 0: 完全なオブジェクトが戻されます。
RETURNATMOST	この呼び出しによって戻される、オブジェクトか固有のクレジット ID の最大数を指定します。この値を指定すると、指定の照会呼び出しにより戻されるデータの量をアプリケーションで制御できます。ASCII 文字の 32 ビット正整数。
WITHCREDITS	1: 関連した PSCredit オブジェクトをすべて見つけ、Batch オブジェクトと共に保持する必要があります。 0: クレジットは戻されません。

QueryBatches コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
WITHPAYMENTS	1: 関連した PSPayment オブジェクトをすべて見つけ、 Batch オブジェクトと共に保持する必要があります。  0: 決済は戻されません。

QueryBatches コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
ACCOUNTNUMBER	Y	マーチャントのアカウントとその金融機関。ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
BALANCESTATUS	Y	以下の値の 1 つを含む ASCII 文字ストリング。  "batch_not_yet_balanced"  "batch_balanced"  "batch_out_of_balance"
BATCHNUMBER	Y	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
CLOSEALLOWED	N	1: マーチャントがクローズできるバッチだけを戻す必要があります。  0: 金融機関がクローズするバッチだけを戻す必要があります。このパラメーターを指定しなかったり、他の値を指定したりすると、両方のタイプのバッチが戻されます。
CLOSEBEGINTIME	N	バッチのクローズ時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
CLOSEENDTIME	N	バッチのクローズ時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
MERCHANTNUMBER	Y	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
MODIFYBEGINTIME	N	バッチの変更時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。

### QueryBatches コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
MODIFYENDTIME	N	バッチの変更時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
OPENBEGINTIME	N	バッチのオープン時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
OPENENDTIME	N	バッチのオープン時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
PAYMENTTYPE	Y	ASCII 文字ストリング。決済カセットまたはプロトコルを指定します。値は最大長 10 バイトまでです。
STATE	Y	以下の値の 1 つを含む ASCII 文字ストリング。 <ul style="list-style-type: none"> <li>• "batch_opening"</li> <li>• "batch_open"</li> <li>• "batch_closing"</li> <li>• "batch_closed"</li> </ul>

## QueryCassette

QUERYCASSETTE コマンドは、Cassette オブジェクトの集合を XML 形式で戻します。

QueryCassettes コマンドの必須キーワードと稼働パラメーター

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「QueryCassettes」。

QueryCassettes コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

QueryCassettes コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
CASSETTENAME	Y	カセットの名前。最大長 64 バイトの ASCII 文字ストリング。

## QueryCredits

QUERYCREDITS コマンドは、WebSphere Commerce Payments の Credit オブジェクトのコレクションか、固有の決済 ID (「オーダー番号: クレジット番号」の形式) のコレクションを戻します。

QueryCredits コマンドの必須キーワードと稼働パラメーター

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「QueryCredits」。

QueryCredits コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ASCII 文字ストリング。ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
KEYSONLY	1: 実際のオブジェクトが戻されるのではなく、固有のクレジット ID のリスト (「マーチャント番号: オーダー番号: クレジット番号」の形式) が戻される必要があります。  0: 完全なオブジェクトが戻されます。
RETURNATMOST	この呼び出しによって戻される、オブジェクトか固有のクレジット ID の最大数を指定します。この値を指定すると、指定の照会呼び出しにより戻されるデータの量をアプリケーションで制御できます。ASCII 文字の整数。32 ビットの正整数。
WITHORDERS	1: POrder オブジェクトを見つけ、Credit オブジェクトと共に戻す必要があります。  0: Credit オブジェクトだけが戻されます。



## QueryCredits コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
BATCHNUMBER	Y	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
BRAND	Y	カスタマーの支払いメソッドのブランド。 ASCII 文字ストリング。
CREATEBEGINTIME	N	クレジットの作成時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
CREATEENDTIME	N	クレジットの作成時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
CREDITNUMBER	Y	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
CURRENCY	N	金額値の ISO 4217 通貨コード。 ASCII 文字の整数。長さは正確に 3 文字でなければならず、必要に応じて先行ゼロを含めなければなりません。
MAXAMOUNT	N	クレジット金額の最大値。通貨の値も指定しなければなりません。 ASCII 文字の 32 ビット正整数。
MERCHANTNUMBER	Y	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
MINAMOUNT	N	クレジット金額の最小値。通貨の値も指定しなければなりません。 ASCII 文字の 32 ビット正整数。
MODIFYBEGINTIME	N	クレジットの変更時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
MODIFYENDTIME	N	クレジットの変更時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
ORDERNUMBER	Y	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
PAYMENTTYPE	Y	ASCII 文字ストリング。決済カセットまたはプロトコルを指定します。値は最大長 10 文字までです。
REFERENCENUMBER	Y	マーチャントに割り当てられた、このクレジットの参照番号。 ASCII 文字ストリング。

### QueryCredits コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
STATE	Y	以下の値の 1 つを含む ASCII 文字ストリング。 <ul style="list-style-type: none"> <li>"credit_reset"</li> <li>"credit_refunded"</li> <li>"credit_closed"</li> <li>"credit_declined"</li> <li>"credit_void"</li> <li>"credit_pending"</li> </ul>

## QueryEventListeners

QUERYEVENTLISTENERS コマンドは、Event Listener オブジェクトのコレクションを戻します。

### QueryEventListeners コマンドの必須キーワード

必須キーワード	複数使用できるかどうか	値
ETAPIVERSION	N	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	N	ASCII 文字ストリング 「QueryEventListeners」。

### QueryEventListeners コマンドのオプション稼働パラメーター

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

### QueryEventListeners コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
CASSETTENAME	Y	1 ~ 64 バイトの ASCII 文字ストリング。
EVENTTYPE	Y	ASCII 文字の整数。値は 1 ~ 3 でなければなりません。 <ul style="list-style-type: none"> <li>1 = 状態変更イベント</li> <li>2 = カセット・イベント</li> <li>3 = ネットワーク管理イベント</li> </ul>

QueryEventListeners コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
LISTENERURL	Y	イベントの表示場所 ( <a href="http://www.merchant.com/webapp/PaymentManager/eventReceiver888">http://www.merchant.com/webapp/PaymentManager/eventReceiver888</a> など) を識別する ASCII 文字ストリング。 WebSphere Commerce 構成マネージャーを使ってポート番号を指定する必要があります。パラメーターの制限はありません。
MERCHANTNUMBER	Y	ASCII 文字の整数。値は 1 ~ 999999999 でなければなりません。

## QueryMerchants

QUERYMERCHANTS コマンドは、Merchant オブジェクトのコレクションを戻します。

QueryMerchants コマンドの必須キーワードと稼働パラメーター

必須キーワード	複数使用できるかどうか	値
ETAPIVERSION	N	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	N	ASCII 文字ストリング「QueryMerchants」。

QueryMerchants コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ASCII 文字ストリング。ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

QueryMerchants コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
MERCHANTNUMBER	Y	マーチャント番号。マーチャント番号を指定しないと、WebSphere Commerce Payments に定義されているすべてのマーチャントについて、PS マーチャントのエレメントが戻されます。ASCII 文字の整数。1 ~ 999999999 でなければなりません。

## QueryOrders

QUERYORDERS コマンドは、PSOrder オブジェクトかオーダー番号のコレクションを戻します。

QueryOrders コマンドの必須キーワードと稼働パラメーター

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「QueryOrders」。

QueryOrders コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ASCII 文字ストリング。ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
KEYSONLY	1: 実際のオブジェクトが戻されるのではなく、オーダー番号とマーチャント番号のリストだけが戻される必要があります。 0: 完全なオブジェクトが戻されます。
RETURNATMOST	この呼び出しによって戻される、オブジェクトかオーダー番号の最大数を指定します。この値を指定すると、指定の照会呼び出しにより戻されるデータの量をアプリケーションで制御できます。ASCII 文字の 32 ビット正整数。
WITHCREDITS	1: 関連した PSCredit オブジェクトをすべて見つけ、Order オブジェクトと共に保持する必要があります。 0: クレジットは戻されません。
WITHPAYMENTS	1: 関連した PSPayment オブジェクトをすべて見つけ、Order オブジェクトと共に保持する必要があります。 0: 決済は戻されません。

QueryOrders コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
ACCOUNTNUMBER	Y	マーチャントのアカウントとその金融機関。ASCII 文字の整数。1 ~ 999999999 でなければなりません。

## QueryOrders コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
APPROVESALLOWED	N	以下の値がサポートされています。 1: このオーダーに承認コマンドを使用できます。 0: このオーダーに承認コマンドを使用できません。
BRAND	Y	カスタマーの支払いメソッドのブランド。 ASCII 文字ストリング。
CREATEBEGINTIME	N	オーダーの作成時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
CREATEENDTIME	N	オーダーの作成時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
CURRENCY	N	金額値の ISO 4217 通貨コード。 ASCII 文字の整数。長さは正確に 3 文字でなければならず、必要に応じて先行ゼロを含めなければなりません。
MAXAMOUNT	N	オーダー金額の最大値。通貨の値も指定しなければなりません。 ASCII 文字の 32 ビット正整数。
MAXUNAPPROVEDAMOUNT	N	まだ承認されていないオーダー金額の最大値。通貨の値も指定しなければなりません。 ASCII 文字の 32 ビット正整数。
MERCHANTNUMBER	Y	マーチャント番号。 ASCII 文字の整数でなければなりません。値は 1 ~ 999999999 でなければなりません。
MINAMOUNT	N	オーダー金額の最小値。通貨の値も指定しなければなりません。 ASCII 文字の 32 ビット正整数。
MINUNAPPROVEDAMOUNT	N	まだ承認されていないオーダー金額の最小値。通貨の値も指定しなければなりません。 ASCII 文字の 32 ビット正整数。
MODIFYBEGINTIME	N	オーダーの変更時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。

### QueryOrders コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
MODIFYENDTIME	N	オーダーの変更時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
ORDERDATA1	N	ユーザーが指定する補助データ。1 ~ 254 バイト長の ASCII 文字ストリングとして指定されます。
ORDERNUMBER	Y	ASCII 文字の整数。1 ~ 99999999 でなければなりません。
PAYMENTTYPE	Y	決済のタイプ。決済カセットまたはプロトコルを識別します。ASCII 文字の整数。最大長は 10 バイトです。
STATE	Y	以下の値の 1 つを含む ASCII 文字ストリング。 <ul style="list-style-type: none"> <li>• "order_requested"</li> <li>• "order_ordered"</li> <li>• "order_refundable"</li> <li>• "order_rejected"</li> <li>• "order_pending"</li> </ul>
TRANSACTIONID	N	ユーザーが指定するトランザクション ID。1 ~ 128 バイト長の ASCII 文字ストリングとして指定されます。

## QueryPayments

QUERYPAYMENTS コマンドは、WebSphere Commerce Payments の Payment オブジェクトのコレクションか、固有の決済 ID (「オーダー番号: 決済番号」の形式) のコレクションを戻します。

QueryPayments コマンドの必須キーワードと稼働パラメーター

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	ASCII 文字ストリング「QueryPayments」。

QueryPayments コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
BATCHNUMBER	Y	ASCII 文字の整数。1 ~ 999999999 でなければなりません。

## QueryPayments コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
BRAND	Y	カスタマーの支払いメソッドのブランド。ASCII 文字ストリング。
CREATEBEGINTIME	N	決済の作成時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
CREATEENDTIME	N	決済の作成時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
CURRENCY	N	金額値の ISO 4217 通貨コード。ASCII 文字の整数。長さは正確に 3 文字でなければならず、必要に応じて先行ゼロを含めなければなりません。
MAXAPPROVEAMOUNT	N	承認額の最大値。通貨の値も指定しなければなりません。ASCII 文字の 32 ビット正整数。
MAXDEPOSITAMOUNT	N	デポジット済み金額の最大値。通貨の値も指定しなければなりません。ASCII 文字の 32 ビット正整数。
MERCHANTNUMBER	Y	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
MINAPPROVEAMOUNT	N	承認額の最小値。通貨の値も指定しなければなりません。ASCII 文字の 32 ビット正整数。
MINDEPOSITAMOUNT	N	デポジット済み金額の最小値。通貨の値も指定しなければなりません。ASCII 文字の 32 ビット正整数。
MODIFYBEGINTIME	N	決済の変更時間の下限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
MODIFYENDTIME	N	決済の変更時間の上限として使用される日時。照会結果に組み込まれます。この値は、ASCII 10 進数字で、1970 年 1 月 1 日の深夜 (24 時間刻時の 00:00:00:000) からのミリ秒として指定されます。
ORDERNUMBER	Y	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
PAYMENTNUMBER	Y	ASCII 文字の整数。1 ~ 999999999 でなければなりません。

QueryPayments コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
PAYMENTTYPE	Y	ASCII 文字の整数。決済カセットまたはプロトコルを識別します。最大長は 10 バイトです。
REFERENCENUMBER	Y	マーチャントに割り当てられた、この決済の参照番号。 ASCII 文字ストリング。
STATE	Y	以下の値の 1 つを含む ASCII 文字ストリング。 <ul style="list-style-type: none"> <li>• "payment_reset"</li> <li>• "payment_approved"</li> <li>• "payment_deposited"</li> <li>• "payment_closed"</li> <li>• "payment_declined"</li> <li>• "payment_void"</li> <li>• "payment_pending"</li> </ul>

QueryPayments コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ASCII 文字ストリング。ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。 DTDPath の長さは 1 ~ 254 バイトでなければなりません。
KEYSONLY	1: 実際のオブジェクトが戻されるのではなく、固有の決済 ID のリスト (「マーチャント番号: オーダー番号: 決済番号」の形式) が戻される必要があります。  0: 完全なオブジェクトが戻されます。
RETURNATMOST	この呼び出しによって戻される、オブジェクトか固有の決済 ID の最大数を指定します。この値を指定すると、指定の照会呼び出しにより戻されるデータの量をアプリケーションで制御できます。 ASCII 文字の 32 ビット正整数。
WITHORDERS	1: PSOrder オブジェクトを見つけ、Payment オブジェクトと共に戻す必要があります。  0: オーダーは戻されません。

## QueryPaymentServer

QUERYPAYMENTSERVER コマンドは Payment Server オブジェクトを戻します。



QueryPaymentServer コマンドの必須キーワードと稼働パラメーター

必須キーワード	複数使用できるかどうか	値
ETAPIVERSION	N	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	N	ASCII 文字ストリング 「QueryPaymentServer」。

QueryPaymentServer コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## QueryPaySystems

QUERYPAYSYSTEMS コマンドは、Payment System オブジェクトのコレクションを戻します。

QueryPaySystems コマンドの必須キーワードと稼働パラメーター

必須キーワード	複数使用できるかどうか	値
ETAPIVERSION	N	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
OPERATION	N	ASCII 文字ストリング 「QueryPaySystems」。

QueryPaySystems コマンドのオプション稼働パラメーター

オプションの稼働パラメーター	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

QueryPaySystems コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
CASSETTENAME	Y	カセット名。ASCII 文字ストリング。最大長は 64 バイトです。

QueryPaySystems コマンドの検索修飾子

オプションの検索修飾子	複数使用できるかどうか	値
MERCHANTNUMBER	Y	マーチャント番号。ASCII 文字の整数。1 ~ 999999999 でなければなりません。

## QueryUsers

QUERYUSERS コマンドは、User オブジェクトのコレクションを戻します。

### オプション・パラメーター

#### MerchantNumber

QUERYUSERS を MerchantNumber で実行するとそのマーチャントに関連するすべてのユーザーが戻されます。

**Filter** QUERYUSERS コマンドによって、管理者はユーザー・フィルターを指定することによってユーザーを照会します。フィルターは、ユーザー・レジストリー全体のサブセットを識別するために WCSRealm クラスによって使用されます。WCSRealm では、フィルターはユーザー名の文字サブストリングを指定できます。たとえば、QUERYUSERS を呼び出してフィルター Smi を渡すと、その結果として、Smith、Smitty、および Jones-Smittinger を含むユーザーのリストが生成されます。WCSRealm ではユーザー・フィルターの太文字小文字が区別されることに注意してください。フィルター・パラメーターは、戻されるユーザーを選別するフィルターを指定します。詳細については、103 ページの『パラメーターの有効な組み合わせ』を参照してください。

デフォルトでは、WCSRealm は管理者以外のすべてのユーザーをフィルターで選別します。このフィルターは、WebSphere Commerce の管理ユーザー・クラス用の追加的なフィルターです。

マーチャント管理者が追加のユーザー ID を必要とする場合、それらは Payments 管理者によって作成されて割り当てられる必要があることに注意してください。

以下の表は、QUERYUSERS コマンドのコマンド構文の詳細を記載しています。

表4. QueryUsers コマンドのオプション・キーワード

オプション・キーワード	複数使用できるかどうか	値
ETAPIVERSION	N	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	Y	文字ストリング形式の数値のマーチャント番号。
OPERATION	N	ASCII 文字ストリング「QueryUsers」。

表 4. QueryUsers コマンドのオプション・キーワード (続き)

オプション・キーワード	複数使用できるかどうか	値
ROLE	N	それぞれの WebSphere Commerce Payments 役割に割り当てられる値。指定された値については、以下の表 5 を参照してください。
USER	N	最大長は 80 バイトです。これはユーザー名です。
RETURNATMOST	N	ASCII 文字の整数。32 ビットの正整数。戻されるユーザーの最大数は 10000 です。
FILTER	N	最大長 128 バイトの UTF-8 文字ストリング。

表 5. 役割値と指定

値	意味	マーチャント固有の役割かどうか
0	Payments 管理者	N
1	マーチャント管理者	Y
2	スーパーバイザー	Y
3	クラーク	Y

## パラメーターの有効な組み合わせ

以下の表は、QUERYUSERS コマンドのすべてのパラメーターの組み合わせの例を示しています。さらにこれは、そのパラメーターの組み合わせのコマンドを出すことができるユーザー、および戻される結果についても示しています。

たいていの場合、WebSphere Commerce Payments は重複パラメーターを検査しないことに注意してください。複数のパラメーターが指定されている場合、その最後のインスタンスが使用されます。

表 6. QueryUsers 用の有効なパラメーターの組み合わせ

パラメーターの組み合わせ	有効か?	だれが発行できるか? (注 *)	非許可ユーザーを戻す
パラメーター指定なし	はい	PA	はい
MERCHANTNUMBER	はい	PA/MA	いいえ
ROLE	はい	PA	いいえ
USER	はい	制限なし	はい
MERCHANTNUMBER + ROLE	はい	PA/MA	いいえ
MERCHANTNUMBER + USER	はい	制限なし	いいえ
ROLE + USER	はい	制限なし	いいえ
MERCHANTNUMBER + ROLE + USER	はい	制限なし	いいえ
FILTER	はい	PA	はい
FILTER + MERCHANTNUMBER	はい	PA/MA	いいえ
FILTER + ROLE	はい	PA	いいえ
FILTER + MERCHANTNUMBER + ROLE	はい	PA/MA	いいえ
FILTER + USER	はい、ただしフィルターは無視される	制限なし	はい
FILTER + MERCHANTNUMBER + USER	はい、ただしフィルターは無視される	制限なし	いいえ
FILTER + ROLE + USER	はい、ただしフィルターは無視される	制限なし	いいえ
FILTER + MERCHANTNUMBER + USER + ROLE	はい、ただしフィルターは無視される	制限なし	いいえ

\*PA = Payments 管理者、 MA = マーチャント管理者

### パラメーターの組み合わせ

QUERYUSERS パラメーターの組み合わせのいくつかのキーポイントは以下のとおりです。

- Username が指定されると、フィルターは無視されます。
- 非許可ユーザーを戻すには、以下のメソッドのいずれか 1 つだけを使用することができます。
  1. Username なしでフィルターを使用する
  2. パラメーターを指定しない
  3. Username だけで照会する

**有効** パラメーターの組み合わせが QUERYUSERS パラメーター・テーブルで有効と定義されていても、特定の照会は無効のままである場合があります。たとえば、マーチャント管理者は Role および Username パラメーターを指定して照会を発行できるとしても、指定したユーザー名がマーチャント管理者のユーザー名である場合しか、その照会は許可されません (つまり、マーチャント管理者は自分自身を照会している)。 QUERYUSERS コマンドのアクセス制御の詳細については、105 ページの『アクセス制御の詳細』を参照してください。

### 非許可ユーザーを戻す

非許可ユーザーを戻す 列は、指定したパラメーターの組み合わせが、レルム内にいるが、WebSphere Commerce Payments の使用は許可されていないユーザーを戻すことができることを示しています。これによって、Payments 管理者は、単一のユーザーを照会し、そのユーザーに WebSphere Commerce Payments アクセスを割り当てることができます。QUERYUSERS へのすべての呼び出しは、許可されているユーザーを戻す場合があることに注意してください。

レルムは、特にフィルターがあまり制限を課さないような場合に、レルム内のすべての一致するユーザーを戻さないように選択できることに注意してください。こうした場合に、上記のメソッドでは、User objectCount を、一致するレルム・ユーザーの合計数に設定します。これは結果として、QUERYUSERS の呼び出し元に、結果が完了しておらず、さらに制限的な検索フィルターを適用するべきことを指示します。

## アクセス制御の詳細

照会が許可されるかどうかは、照会発行者の役割には依存していません。たとえば、

### Payments 管理者

Payments 管理者は、任意の組み合わせのパラメーターを指定して照会を発行することができます。

### マーチャント管理者

マーチャント管理者が照会できるのは、以下のユーザーだけです。

- マーチャント管理者によって管理されるマーチャント番号 (複数の場合あり) に関連したユーザー

さらに、マーチャント管理者は、照会要求において以下の要件を満たすことも求められます。

- 少なくとも 1 つの MerchantNumber を指定する必要がある、指定するすべてのマーチャント番号は、そのマーチャント管理者に関連したマーチャントに属している必要があります。ただし、マーチャント管理者が自分自身を照会する場合は、マーチャント番号は必要ありません。
- Role パラメーターを指定する場合は、これに Payments 管理者の役割を含めてはなりません。

### スーパーバイザーおよびクラーク

他のすべての役割に対して、ユーザーは自分自身を照会することができます。この場合、フィルターが指定されると、フィルターは無視されます。

---

## ReceivePayment

RECEIVEPAYMENT コマンドは、電子ウォレットが関係している場合に、オーダーを作成するために使用されます。正常に実行されると、オーダー・オブジェクトは「要求」状態に移行します。それからウォレット通信でオーダーが完了し、「オーダー済み」状態に移行します。

オーダーを作成する際には、資金の承認やデポジットを自動的に行うこともできます。ウォレット通信が行われてオーダーが「オーダー済み」状態になると、APPROVEFLAG キーワードと DEPOSITFLAG キーワードにより、Payment オブジェクトを自動的にデポジットして承認する必要があることが示されます。APPROVEFLAG か DEPOSITFLAG を指定する場合に使用する追加キーワードについては、以下の該当する表を参照してください。

表 7. ReceivePayment コマンドの必須キーワード

必須キーワード	値
AMOUNT	ASCII 文字の 32 ビット正整数でなければなりません。
CURRENCY	ASCII 文字の整数。ISO 通貨コードのリストについては、付録 B の『通貨コード』を参照してください。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「ReceivePayment」。
ORDERNUMBER	ASCII 文字の整数。1 ~ 999999999 でなければなりません。
PAYMENTTYPE	ASCII 文字ストリング。使用される決済カセットまたはプロトコルを指定します。たとえば、OfflineCard などです。

表 8. ReceivePayment コマンドのオプション・キーワード

オプション・キーワード	値
AMOUNTEXP10	ASCII 文字の整数。桁送りする桁数を指定します。キーワードの詳細については、59 ページの『AmountExp10 キーワードの使用法』を参照してください。
APPROVEFLAG	ASCII 文字の整数。承認を自動的に試行する必要があるかどうかを示します。デフォルトは 0 です。以下の値がサポートされています。  0 - 取引を承認しないように指示します。  1 - 取引を自動的に承認するように指示します。  2 - 取引を非同期に承認するように指示します。
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。
ORDERDATA1	ユーザーが指定する補助データ。1 ~ 254 バイト長の ASCII 文字ストリングとして指定されます。
ORDERDATA2	ユーザーが指定する補助データ。1 ~ 254 バイト長の UTF-8 ストリングとして指定されます。
ORDERDATA3	ユーザーが指定する補助データ。1 ~ 254 バイト長の UTF-8 ストリングとして指定されます。
ORDERDATA4	ユーザーが指定する補助データ。1 ~ 254 バイト長のバイナリー・ストリングとして指定されます。

表 8. *ReceivePayment* コマンドのオプション・キーワード (続き)

オプション・キーワード	値
ORDERDATA5	ユーザーが指定する補助データ。任意の長さのバイナリー・ストリングとして指定されます。
ORDERURL	オーダーの詳細を含む URL
TRANSACTIONID	ユーザーが指定するトランザクション ID。 1 ~ 128 バイト長の ASCII 文字ストリングとして指定されます。

以下の表には、APPROVEFLAG=1 または 2 の場合の必須キーワードとオプション・キーワードがリストされています。

表 9. *APPROVEFLAG* が 1 または 2 に設定されている場合の必須キーワード

必須キーワード	値
PAYMENTAMOUNT	ASCII 文字の 32 ビット正整数。
PAYMENTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。

表 10. *APPROVEFLAG* が 1 または 2 に設定されている場合のオプション・キーワード

オプション・キーワード	値
DEPOSITFLAG	ASCII 文字によるブール値。デポジットを自動的に試行する必要があるかどうかを示します。このフラグが有効なのは、APPROVE=1 (オーダーが自動的に承認される) のときだけです。以下の値がサポートされています。  0 - 資金を自動的にデポジットしてはならない。 1 - 資金を自動的にデポジットする必要がある。

DEPOSITFLAG=1 の場合、以下のキーワードはオプションです。

表 11. *DEPOSITFLAG* が 1 に設定されている場合のオプション・キーワード

オプション・キーワード	値
BATCHNUMBER	この決済が処理されるバッチを識別します。 1 ~ 999999999 でなければなりません。

## Refund

REFUND コマンドは、Credit オブジェクトを作成するために使用されます。このコマンドは、マーチャントからカード所有者にお金を戻したい場合に使用されます。REFUND コマンドは、指定されたオーダーが「リファンド可能」状態にある場合に有効です。

正常に実行されると、「リファンド」状態か「クローズ」状態の Credit オブジェクトが作成されます。正常に実行されないと、「拒否」状態の Credit オブジェクトが作成されます。

表 12. Refund コマンドの必須キーワード

必須キーワード	値
AMOUNT	ASCII 文字の 32 ビット正整数でなければなりません。
CREDITNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。このクレジットに割り当てられた番号を示します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「Refund」。
ORDERNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。

表 13. Refund コマンドのオプション・キーワード

オプション・キーワード	値
BATCHNUMBER	暗黙バッチの場合にオプション。最大 9 文字の数値ストリングです。この決済が処理されるバッチを識別します。 1 ~ 999999999 でなければなりません。
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、外部 DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。 DTDPath の長さは 1 ~ 254 バイトでなければなりません。

## RefundReversal

REFUNDREVERSAL コマンドは、既存の Credit オブジェクトを無効にするために使用します。このコマンドは、「リファンド」状態にある Credit オブジェクトに対して操作を行います。REFUNDREVERSAL 呼び出しが正常に実行されると、Credit オブジェクトは「無効」状態に移行します。正常に実行されない場合は、Credit オブジェクトは「リファンド」状態のままです。

表 14. RefundReversal コマンドの必須キーワード

必須キーワード	値
CREDITNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。このクレジットに割り当てられた番号を示します。
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、 2.2.x、 3.1.x、 および 5.5.x を示す)
MERCHANTNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。
OPERATION	ASCII 文字ストリング「RefundReversal」。
ORDERNUMBER	ASCII 文字の整数。 1 ~ 999999999 でなければなりません。



表 15. RefundReversal コマンドのオプション・キーワード

オプション・キーワード	値
DTDPATH	ローカルに保管されている DTD へのパス。このパラメーターの値は、既存の DTD の場所を指定するために XML 文書内で使用されます。このパラメーターを指定しないと、完全な DTD が内部 DTD として戻されます。DTDPATH の長さは 1 ~ 254 バイトでなければなりません。

## SetUserAccessRights

SETUSERACCESSRIGHTS コマンドは、ユーザーのアクセス権を設定、変更、除去するときに使用します。ただし、このコマンドを使用して、ユーザー認証のために使用している WCSRealm からユーザーを作成したり除去したりすることはできません。SetUserAccessRights コマンドを使用する前に、ユーザーがすでに WCSRealm に追加されていることを確認してください。WebSphere Commerce でユーザーを追加する方法や、アクセス管理についての詳細は、「WebSphere Commerce 基本」または WebSphere Commerce オンライン・ヘルプを参照してください。

### ユーザー・アクセスの追加

ユーザーのアクセス権を追加するには、最初にそのユーザーを WCSRealm に追加し、それから SetUserAccessRights コマンドを送出します。

### ユーザー・アクセスの除去

ユーザーのアクセス権を除去するには、SetUserAccessRights コマンドを送出して最初にユーザーのアクセス権を除去し、その後そのユーザーを WCSRealm から除去します。

表 16. SetUserAccessRights コマンドの必須キーワード

必須キーワード	値
ETAPIVERSION	「3」(WebSphere Commerce Payments または以前の製品の API バージョン 2.1.x、2.2.x、3.1.x、および 5.5.x を示す)
MERCHANTNUMBER	文字ストリング形式の数値のマーチャント番号。このキーワードは、マーチャントごとに固有の役割が指定されている場合に必須です。マーチャント番号は、1 ~ 999999999 でなければなりません。Payments 管理者以外のユーザーの場合は、複数のキーワードと値の対を指定することができます。
OPERATION	ASCII 文字ストリング「SetUserAccessRights」。
ROLE	文字ストリング形式の数値。
USER	最大長 40 バイトの ASCII 文字ストリング。(ユーザーは自分自身を更新することはできないことに注意してください。つまり、ユーザー "admin" は、ユーザー・パラメーターを "admin" に設定して SETUSERACCESSRIGHTS を呼び出すことはできません。)

ユーザーのアクセス権を設定または変更するには、コマンドで役割およびマーチャント番号 (複数の場合あり) を指定します。ユーザーが複数のマーチャントの役割を持つようにユーザーのアクセス権を設定または変更するには、マーチャント番号のキーワードと値の対を複数回繰り返す必要があります。与えられている役割

がマーチャントに固有のものである場合 (103 ページの表 5 を参照) は、マーチャント番号 (複数の場合あり) も必ず指定しなければならず、与えられている役割がマーチャントに固有のものでない場合、指定することはできません。

注:

1. Role パラメーターが指定されていない場合は、このコマンドを使用してユーザーのアクセス権を除去できます。この場合、WebSphere Commerce Payments はマーチャント番号を無視します (それがコマンドに指定された場合でも)。
2. ユーザーは自分自身を更新することはできません。つまり、ユーザー "admin" は、ユーザー・パラメーターを "admin" に設定して SETUSERACCESSRIGHTS を呼び出すことはできません。

## マーチャント管理者のアクセス制御規則

Payments 管理者およびマーチャント管理者だけが、ユーザーの許可 (または役割) を割り当て、または変更できます。Payments 管理者は、任意のユーザーのアクセス権限を割り当てまたは変更することができ、(Payments 管理者役割を含めた) 任意のユーザー役割を割り当てまたは変更することができます。マーチャント管理者は、ユーザーをマーチャント管理者、スーパーバイザー、またはクラークとして割り当てることも除去することもできますが、ユーザーの許可を Payments 管理者の許可に割り当て、または変更することはできません。さらに、マーチャント管理者は、17 ページの『ユーザーのアクセス許可の割り当て』に概説されている条件下でのみ許可を割り当て、または変更することができます。

---

## 第 7 章 WebSphere Commerce Payments データ

この章では、WebSphere Commerce Payments Framework の決済および管理オブジェクトと状態について説明します。オブジェクトは、実世界のエンティティを表す WebSphere Commerce Payments が保守するデータの集合です。すべてのオブジェクトは定義され、フィールド名、構文および説明を示すためのテーブルがあります。オブジェクトの状態は、その特定のオブジェクトの法的な処置についての情報を提供します。オブジェクトの現在の状態は、QUERY コマンドを使用して検索できます。付加的なテーブルには、特定のオブジェクトの取り得る状態と、その状態が意味すること、およびその状態に対してどのコマンドが有効であるかということの説明がリストされます。

---

### WebSphere Commerce Payments 決済オブジェクト

WebSphere Commerce Payments は、決済プロトコルが何であるかにかかわらず、すべての電子決済用に以下のフレームワーク・オブジェクトを定義します。

- Order
- Payment
- Credit
- Batch

WebSphere Commerce Payments は、すべての電子決済の決済データを表すのに、オーダー、決済、およびクレジット という用語を使用します。オーダーは、バイヤーが商品またはサービスをオーダーするときに、バイヤーとマーチャントとの間でデータが流れる結果として作成されるオブジェクトです。取引は、オーダーのライフ・サイクルの間に、マーチャントと金融機関との間で流れます。これらの取引は、決済 (消費者からマーチャントに送られる代金) と、クレジット (商品に欠陥がある場合などに消費者に返却される代金) の 2 つの大きなカテゴリーに分類できます。オーダーの処理が進むにつれて、Payment および Credit オブジェクトは、作成、変更、および破棄されます。

WebSphere Commerce Payments によって使用される別の種類のオブジェクトは、Batch オブジェクトです。バッチは、営業日の最後におけるすべての決済のデポジットなど、1 つのグループとしての複数の取引を表します。WebSphere Commerce Payments の Batch オブジェクトは、一連の取引の記録を残します。たとえば、金融機関がマーチャントにその週の取引の締めを行うように要求した場合、マーチャントは現在のバッチを終了して新しいバッチを開始します。それら 2 つのバッチの Batch オブジェクトは、バッチの新しい状態を反映するようになります。

Order、Payment、Credit、および Batch オブジェクトには、それぞれ関連した状態があります。オブジェクトの状態は、そのオブジェクトについてどの処置が許可されているかを決定します。オブジェクトの状態は、そのオブジェクトに対して最後に実行されたアクションまたはコマンド によって決まります。

それぞれの WebSphere Commerce Payments Framework オブジェクトは、その属性またはフィールドによって定義されます。以下のセクションのオブジェクト・テーブルは、それぞれのフレームワーク・オブジェクトの表示フィールド名、フィールド構文、およびフィールドの説明を示しています。さらに、オブジェクト状態テーブルでは、オブジェクトが想定する状態と、それらの状態のフィールドの説明を表示します。

## Order

オーダーは、マーチャント (受取人) が代金を収集するために、バイヤー (支払人) について必要なすべての指示と情報を表します。マーチャントは、代金を一度に、または一定期間に渡って収集することができますが、追加情報を入手するためにバイヤーのもとに戻る必要はありません。必要な情報は、すべてオーダーにあります。WebSphere Commerce Payments Order オブジェクトは、オーダーに含まれるデータを説明しています。それぞれのオーダーに関連する決済は、ゼロでも複数でも構いません。Order オブジェクトの属性は、次のとおりです。

表 17. PSOrderObject 属性

フィールド名	構文	説明
merchantNumber	数値トークン、1 ~ 9 桁の長さ	オーダーを作成したマーチャントを識別する番号。
orderNumber	数値トークン、1 ~ 9 桁の長さ	マーチャントによって割り当てられた、オーダーを固有に識別する番号。
merchantOriginated	0 または 1 (ブール)	AcceptPayment を使用してオーダーが作成された場合、値は 1 (真) です。ReceivePayment を使用してオーダーが作成された場合、値は 0 (偽) です。
amount	整数	発注で使われる特定の通貨の最小単位でオーダー金額を指定します。このフィールドは、AmountExp10 と組み合わせて使用されると、指定された通貨でオーダー全体の金額を指定します。
amountExp10	整数	通貨を反映するために小数点を桁送りする桁数を指定します。たとえば、金額が 2325 で、通貨コードが U.S. ドルを表し、AmountExp10 が -2 である場合には、取引金額は U.S. ドルで \$23.25 になります。
currency	整数	通貨を表す ISO コード。たとえば、840 は U.S. ドルを表す数字コードであり、392 は日本の円を表す数字コードです。
paymentType	文字ストリング	発注するために使用される決済カセットまたはプロトコル (たとえば、OfflineCard) を識別します。
timeStampCreated	日付	このオーダー項目が作成された時刻。GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
timeStampModified	日付	このオーダー項目が最後に変更された時刻。GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。

表 17. PSOrderObject 属性 (続き)

フィールド名	構文	説明
state	文字ストリング	オーダーの状態。 <ul style="list-style-type: none"> <li>• order_requested</li> <li>• order_ordered</li> <li>• order_refundable</li> <li>• order_rejected</li> <li>• order_pending</li> <li>• order_canceled</li> <li>• order_closed</li> </ul>
approvesAllowed	0 または 1 (ブール)	このオーダーについて approve コマンドが適法かどうかを示すフラグ。
unapprovedAmount	整数	オーダーの金額から、そのオーダーについて承認されたすべての決済の金額を引いたもの。
numberOfPayments	整数	このオーダーに関連した決済の数。
numberOfCredits	整数	このオーダーに関連したクレジットの数。
brand	文字ストリング	クレジット・カード。発注するのに使用された決済カードのブランド (たとえば、VISA または MasterCard)。
orderURL	URL	マーチャントの営業システムの中のオーダーについての情報を指すのによく使用されるマーチャント定義の URL。
merchantAccount	数値トークン、1 ~ 9 桁の長さ	このオーダーを処理するために使用されるアカウントの番号。 オーダーが「オーダー済み」状態になる前に割り当てられる。
transactionId	文字ストリング、ASCII 文字で 1 ~ 128 の長さ	顧客のトランザクション ID。この値が存在するのは、AcceptPayment コマンドまたは ReceivePayment コマンドに非 nul の TRANSACTIONID 値が指定された場合のみです。
orderData1	文字ストリング、ASCII 文字で 1 ~ 254 の長さ	この値が存在するのは、AcceptPayment コマンドまたは ReceivePayment コマンドに非 nul の ORDERDATA1 値が指定された場合のみです。
orderData2	UTF-8 ストリング、1 ~ 254 バイト長	この値が存在するのは、AcceptPayment コマンドまたは ReceivePayment コマンドに非 nul の ORDERDATA2 値が指定された場合のみです。
orderData3	UTF-8 ストリング、1 ~ 254 バイト長	この値が存在するのは、AcceptPayment コマンドまたは ReceivePayment コマンドに非 nul の ORDERDATA3 値が指定された場合のみです。
orderData4	バイナリー・ストリング、1 ~ 254 バイト長	この値が存在するのは、AcceptPayment コマンドまたは ReceivePayment コマンドに非 nul の ORDERDATA4 値が指定された場合のみです。
orderData5	バイナリー・ストリング、任意の長さ	この値が存在するのは、AcceptPayment コマンドまたは ReceivePayment コマンドに非 nul の ORDERDATA5 値が指定された場合のみです。

注: 数値トークン は、長さが 1 ~ 9 桁の数値ストリングとして定義されます。

## Order の状態

オブジェクトの状態は、そのオブジェクトについてどの処置が有効であるかを決定します。オブジェクトの状態は、そのオブジェクトに対して最後に実行されたアクションまたはコマンドによって決まります (たとえば、承認された決済は「承認」状態になります)。

オーダーは、以下のいずれかの状態になります。

状態	説明
要求中	<p>バイヤーがオーダーを完了するために必要なすべての情報をまだ提供していない仮の状態。この状態について有効なコマンドは、次のとおりです。</p> <ul style="list-style-type: none"><li>• CancelOrder</li></ul>
オーダー済み	<p>消費者 - マーチャント・サーバー - WebSphere Commerce Payments のオーダー・メッセージのフローが正常に完了したことを示します。これで WebSphere Commerce Payments は、決済に対してコマンドを実行できるようになりました。この状態について有効なコマンドは、次のとおりです。</p> <ul style="list-style-type: none"><li>• CloseOrder。オーダーに決済またはクレジットが関連付けられている場合、CloseOrder が使用可能になるまですべての決済またはクレジットはクローズ状態でなければなりません。オーダーに決済またはクレジットが関連付けられていない場合、CloseOrder は無効です。</li><li>• CancelOrder。次のどちらかが該当する場合。<ul style="list-style-type: none"><li>- オーダーに決済またはクレジットが関連付けられていない。</li><li>- すべての決済およびクレジットが「リセット (Reset)」、「無効 (Void)」、「承認の有効期限切れ (ApprovalExpired)」、または「拒否 (Declined)」状態のいずれかである。</li></ul></li><li>• Approve</li><li>• ApproveReversal</li><li>• Deposit</li><li>• DepositReversal</li></ul>

状態	説明
リファンド可能	<p>これで WebSphere Commerce Payments は、決済およびクレジットに対してコマンドを実行できるようになりました。オーダーの状態が、「オーダー済み」から「リファンド可能」に移動する時期は、決済支払いメソッドによって異なります。この状態について有効なコマンドは、次のとおりです。</p> <ul style="list-style-type: none"> <li>• CloseOrder。オーダーに決済またはクレジットが関連付けられている場合、CloseOrder が使用可能になるまですべての決済またはクレジットはクローズ状態であればなりません。オーダーに決済またはクレジットが関連付けられていない場合、CloseOrder は無効です。</li> <li>• Approve</li> <li>• ApproveReversal</li> <li>• Deposit</li> <li>• DepositReversal</li> <li>• Refund</li> <li>• RefundReversal</li> </ul>
拒否	<p>消費者 - マーチャントの購買フローの際に、問題が生じたことを示します。この状態について有効なコマンドは、次のとおりです。</p> <ul style="list-style-type: none"> <li>• CancelOrder</li> </ul>
保留	<p>WebSphere Commerce Payments がオーダーに対してコマンドを実行している時は、オーダーは「保留」状態になります。この状態のオーダーについて有効なコマンドはありません。</p>
キャンセル	<p>このオーダーはキャンセルされました。この状態について有効なコマンドは、次のとおりです。</p> <ul style="list-style-type: none"> <li>• DELETEORDER フラグを使用可能にした CancelOrder (これは、データベースからオーダーを削除します)。</li> </ul>
クローズ	<p>このオーダーはクローズされました。この状態について有効なコマンドは、次のとおりです。</p> <ul style="list-style-type: none"> <li>• CloseOrder</li> <li>• DELETEORDER フラグを使用可能にした CancelOrder (これは、データベースからオーダーを削除します)。</li> </ul>

## Payment

Payment オブジェクトは、オーダーの全部または一部を承認するための、金融機関に対するマーチャントによる要求を表します。

多くの場合、オーダーが発生して集金が与信されると、その全額は一度の決済で集金されます。決済のシステムによっては、1 つのオーダー (つまり一連の決済指示) で与信された金額を、ビジネス・モデルに応じて複数回の決済で集金することもあります。オーダーごとに、0 個以上の決済が存在する可能性があります。Payment オブジェクトの属性は、次のとおりです。

表 18. PSPaymentObject 属性

フィールド名	構文	説明
merchantNumber	数値トークン、1 ~ 9 桁の長さ	オーダーを作成したマーチャントを識別する番号。
orderNumber	数値トークン、1 ~ 9 桁の長さ	マーチャントによって割り当てられた、オーダーを固有に識別する番号。このフィールドは、オーダー・テーブルの orderNumber と一致します。
paymentNumber	数値トークン、1 ~ 9 桁の長さ	あるオーダー内の特定の決済を示す固有の識別番号。
paymentType	文字ストリング	発注するために使用される決済カセットまたはプロトコル (たとえば、 VisaNet または OfflineCard) を識別します。
approvedAmount	整数	決済のために承認されたオーダーの金額。
amount	整数	オーダーで使われる特定の通貨の最小単位で決済金額を指定します。このフィールドは、 AmountExp10 と組み合わせて使用されると、指定された通貨で決済の金額を指定します。
amountExp10	整数	通貨を反映するために小数点を桁送りする桁数を指定します。たとえば、金額が 2325 で、通貨コードが U.S. ドルを表し、 AmountExp10 が -2 である場合には、取引金額は U.S. ドルで \$23.25 になります。
currency	整数	決済を行うために使用される通貨。通貨を表す ISO コード。たとえば、840 は U.S. ドルを表す数字コードであり、392 は日本の円を表す数字コードです。
timeStampCreated	日付	この決済項目が作成された時刻。 GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
timeStampModified	日付	この決済項目が最後に変更された時刻。 GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
state	文字ストリング	決済の状態。 <ul style="list-style-type: none"> <li>• payment_reset</li> <li>• payment_approved</li> <li>• payment_deposited</li> <li>• payment_pending</li> <li>• payment_declined</li> <li>• payment_void</li> <li>• payment_closed</li> <li>• payment_approvaexpired</li> </ul>
batchNumber	数値トークン、1 ~ 9 桁の長さ	バッチを識別する番号。  決済がデポジットされる時に割り当てられます。
referenceNumber	文字ストリング	決済を識別するために金融機関が使用するプレーン・テキスト ID。
depositAmount	整数	この決済についてデポジットされた金額 (承認された金額とは異なる場合もある)。  デポジットされる時に割り当てられます。
merchantAccount	数値トークン、1 ~ 9 桁の長さ	このオーダーを処理するために使用されるアカウントを識別する番号。



表 18. PSPaymentObject 属性 (続き)

フィールド名	構文	説明
order	IDREF	この決済に関連したオーダーを表す XML 要素。
approveTime	日付	この決済項目が最後に承認された時刻。
approvalExpiry	日付	決済の承認の有効期限が切れる時刻。ヌル値は、有効期限がないことを意味します。

## Payment の状態

決済は、以下のいずれかの状態になります。

状態	説明	有効なコマンド
リセット	決済が作成されても、まだ処理されていないと、「リセット」状態になります。	Approve コマンドがまだ完了していないため、この状態の決済に有効なコマンドはありません。
承認	approve コマンドが正常に完了すると、決済は「承認」状態になります。クレジット・カードの場合、「承認」状態は決済が許可されたことを意味します。	<ul style="list-style-type: none"> <li>ApproveReversal</li> <li>Deposit</li> </ul>
デPOSIT済み	deposit または auto-deposit コマンドが正常に完了すると、決済は「デPOSIT済み」状態になります。決済の種類がクレジット・カードの場合、「デPOSIT済み」状態は決済が資金化されたことを意味します。	DepositReversal
クローズ	決済に関連したバッチがクローズされたとき、「デPOSIT済み」状態の決済は「クローズ」状態になります。決済が「クローズ」状態のとき、金融上の取引は完了しています。代金はデPOSITされており、決済は変更できません。	この状態の決済に有効なコマンドはありません。
拒否	approve コマンドが金融上の理由で拒否されると、決済は「拒否」状態になります。	Approve
無効	金額がゼロに対して ApproveReversal コマンドが正常に完了すると、決済は「無効」状態になります。	Approve
保留	この決済に対してあるコマンドが現在実行されています。	この状態の決済に有効なコマンドはありません。

承認の有効期限切れ	指定された承認時間が経過した後、あるいは決済与信の有効期限が切れたことをカセットが検出した場合、決済は「承認」状態から「承認の有効期限切れ」状態に移行します。これはオプションの状態であり、カセットによってはサポートされない場合があります。	ApproveReversal
-----------	---	-----------------

## スプリット・ペイメント

顧客がオンライン・カタログ・ストアにアクセスし、\$80 の商品をオーダーしたとします。マーチャントが在庫を調べたところ、出荷可能な在庫は \$60 相当の量に過ぎないことが分かりました。マーチャントは当面 \$60 を徴収し、残りの \$20 については残りのオーダーに応じるときに徴収することを希望します。WebSphere Commerce Payments は、顧客が支払情報を一度に提示し (\$80 全体)、マーチャントが支払金額を長期に渡って徴収する (当面は \$60 を徴収し、\$20 は後で徴収) という支払システムをサポートしています。これをスプリット・ペイメントと言います。

## AVS 共通コード

使用しているカセットが WebSphere Commerce Payments の共通 AVS コードをサポートする場合に、**commonAVSCode** パラメーターを照会すれば、カセットに依存せずに AVS 結果を判別することができます。

共通 AVS 結果コードからカセット結果コードへのマッピングは以下のようになります。

共通 AVS コード	PM 定数名	説明
4	AVS_OTHER_RESPONSE	この定数は、使用不可のアドレス情報、使用不可のシステム (おそらくタイムアウトが原因)、サポートされていないカード・タイプ、および不適格なトランザクションを AVS 戻りコードにマップします。クレジット・カード・プロセッサから、システムに関連した他の何らかの応答を受け取りました。
3	AVS_NO_MATCH	番地も郵便番号も一致しません。
2	AVS_POSTALCODE_MATCH	5 桁または 9 桁の郵便番号は一致しますが、番地は一致しません。
1	AVS_STREETADDRESS_MATCH	番地は一致しますが、郵便番号は一致しません。
0	AVS_COMPLETE_MATCH	この定数は、AVS の 5 桁と 9 桁の両方の郵便番号、および番地をマップします。どちらも完全に一致します。

## Credit

Credit オブジェクトを作成する WebSphere Commerce Payments コマンドは、Refund と呼ばれます。Credit オブジェクトは、1 つのオーダー (つまり、決済の同意) オブジェクトの中で識別される金額に対して作成される 1 つのクレジットを識別します。オーダーごとに、0 個以上のクレジットが存在する可能性があります。Credit オブジェクトの属性は、次のとおりです。

表 19. PSCreditObject 属性

フィールド名	構文	説明
merchantNumber	数値トークン、1 ~ 9 桁の長さ	オーダーを作成したマーチャントを識別する番号。
orderNumber	数値トークン、1 ~ 9 桁の長さ	マーチャントによって割り当てられた、オーダーを固有に識別する番号。このフィールドは、オーダー・テーブルの orderNumber と一致します。
creditNumber	数値トークン、1 ~ 9 桁の長さ	あるオーダー内の特定のクレジットを示す固有の識別番号。
paymentType	文字ストリング	発注するために使用される決済カセットまたはプロトコル (たとえば、VisaNet または OfflineCard) を識別します。
amount	整数	オーダーに使われる特定通貨の最小単位で、クレジット金額を識別します。このフィールドを AmountExp10 と組み合わせて使用すると、指定された通貨でクレジットの金額を指定します。
amountExp10	整数	通貨を反映するために小数点を桁送りする桁数を指定します。たとえば、金額が 2325 で、通貨コードが U.S. ドルを表し、AmountExp10 が -2 である場合には、取引金額は U.S. ドルで \$23.25 になります。
currency	整数	このクレジットを発行するために使用された通貨。通貨を表す ISO コード。たとえば、840 は U.S. ドルを表す数字コードであり、392 は日本の円を表す数字コードです。
timeStampCreated	日付	このクレジット項目が作成された時刻。GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
timeStampModified	日付	このクレジット項目が最後に変更された時刻。GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
state	文字ストリング	クレジットの状態。 <ul style="list-style-type: none"> <li>• credit_reset</li> <li>• credit_refunded</li> <li>• credit_pending</li> <li>• credit_declined</li> <li>• credit_void</li> <li>• credit_closed</li> </ul> クレジットの状態の詳細については、120 ページの『Credit の状態』を参照してください。
batchNumber	数値トークン、1 ~ 9 桁の長さ	バッチを識別する番号。 決済がデポジットされるときに割り当てられます。
referenceNumber	文字ストリング	決済を識別するために金融機関が使用するプレーン・テキスト ID。

表 19. PSCreditObject 属性 (続き)

フィールド名	構文	説明
merchantAccount	数値トークン、1 ~ 9 桁の長さ	このオーダーを処理するために使用されるアカウントの番号。

## Credit の状態

クレジットは、以下のいずれかの状態になります。

状態	説明
リセット	クレジットが作成されても、まだ処理されていないと、「リセット」状態になります。この状態のクレジットについて有効なコマンドはありません。
リファンド	refund コマンドが正常に完了すると、クレジットは「リファンド」状態になります。この状態について有効なコマンドは、次のとおりです。 <ul style="list-style-type: none"> <li>• RefundReversal</li> </ul>
クローズ	クレジットに関連したバッチがクローズされたとき、「リファンド」状態のクレジットは「クローズ」状態になります。クレジットが「クローズ」状態のとき、金融トランザクションは完了しています。代金はリファンドされており、クレジットは変更できません。「クローズ」状態のクレジットについて有効なコマンドはありません。
拒否	refund コマンドが金融上の理由で拒否されると、クレジットは「拒否」状態になります。この状態について有効なコマンドは、次のとおりです。 <ul style="list-style-type: none"> <li>• Refund</li> </ul>
無効	金額がゼロに対して RefundReversal コマンドが正常に完了すると、クレジットは「無効」状態になります。この状態について有効なコマンドは、次のとおりです。 <ul style="list-style-type: none"> <li>• Refund</li> </ul>
保留	このクレジットに対してあるコマンドが現在実行されています。この状態のクレジットについて有効なコマンドはありません。

## Batch

バッチは、金融機関が 1 つの単位として処理する金融上の取引 (決済とクレジット) の集合です。Batch は、Account およびマーチャントと関連付けられています。アカウントには、バッチがゼロでも複数あっても構いません。Batch オブジェクトの属性は、次のとおりです。

表 20. PSBatchObject 属性

フィールド名	構文	説明
merchantNumber	数値トークン、1 ~ 9 桁の長さ	バッチを所有するマーチャントの番号。
merchantAccount	数値トークン、1 ~ 9 桁の長さ	バッチに関連付けられたアカウント番号。

表 20. PSBatchObject 属性 (続き)

フィールド名	構文	説明
batchNumber	数値トークン、1 ~ 9 桁の長さ	バッチを識別する番号。 決済がデポジットされるときに割り当てられます。
purgeAllowed	0 または 1 (ブール)	マーチャントがこのバッチを除去することが有効かどうかを示すフラグ。この値が 1 (yes) の場合は、マーチャントは BatchPurge コマンドを使用してこのバッチを除去できます。値が 0 (no) の場合は、マーチャントはこのバッチを除去できません。
forceAllowed	0 または 1 (ブール)	Force オプションが設定された BatchClose コマンドをマーチャントが送出することが有効かどうかを示すフラグ。この値が 1 (yes) の場合は、マーチャントはこのコマンドを送出できます。
paymentType	文字ストリング	発注するために使用される決済カセットまたはプロトコル (たとえば、VisaNet または OfflineCard) を識別します。
merchantControl	0 または 1 (ブール)	マーチャントがこのバッチを制御することが有効かどうかを示すフラグ。値が 1 (真) の場合は、マーチャントがこのバッチを清算します。(マーチャントは、BatchClose コマンドを使用してバッチを明示的に終了することによって、バッチを清算します。) 値が 0 (偽) の場合は、マーチャントはこのバッチを清算するのに何も行いません。
timeStampOpened	日付	このバッチが (マーチャントか金融機関のどちらかによって) 開かれた時刻。GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
timeStampClosed	日付	このバッチが (マーチャントか金融機関のどちらかによって) 閉じられた時刻。GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
timeStampModified	日付	このバッチが最後に変更された時刻。GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
状態	文字ストリング	バッチの状態。 <ul style="list-style-type: none"> <li>• batch_opening</li> <li>• batch_open</li> <li>• batch_closing</li> <li>• batch_closed</li> </ul> バッチの状態の詳細については、122 ページの『Batch の状態』を参照してください。
batchStatus	文字ストリング	このバッチの収支計算状況。 <ul style="list-style-type: none"> <li>• <b>batch_not_yet_balanced:</b> このバッチに対して、収支計算はまだ実行されていません。</li> <li>• <b>batch_balanced:</b> バッチの収支計算が行われ、すべてについての同意が得られています。</li> <li>• <b>batch_out_of_balance:</b> バッチの収支計算が行われ、すべてについての同意が得られていません。</li> </ul>

## Batch の状態

バッチは、以下のいずれかの状態になります。

状態	説明
オープン中	バッチは現在開かれています。「オープン中」状態のバッチに対して有効なコマンドはありません。
オープン	「オープン」状態のバッチには、決済とクレジットを追加することができます。この状態について有効なコマンドは、次のとおりです。 <ul style="list-style-type: none"><li>• CloseBatch。merchantControl が真である場合のみ。</li></ul>
クローズ中	バッチは現在清算中です。この状態のバッチについて有効なコマンドはありません。
クローズ	「クローズ」状態のバッチは、清算済みです。この状態について有効なコマンドは、次のとおりです。 <ul style="list-style-type: none"><li>• DeleteBatch</li></ul>

---

## WebSphere Commerce Payments About オブジェクト

WebSphere Commerce Payments は、以下の About オブジェクトを定義します。

- Payment Server About
- Cassette About

それぞれの WebSphere Commerce Payments About オブジェクトは、その属性またはフィールドによって定義されます。以下のセクションのオブジェクト・テーブルは、それぞれの About オブジェクトのフィールド名、フィールド構文、およびフィールドの説明を示しています。

### Payment Server About

Payment Server About オブジェクトは、WebSphere Commerce Payments のバージョンを含んでいます。Payment Server 属性は、次のとおりです。

フィールド名	構文	説明
version	文字ストリング	WebSphere Commerce Payments のバージョン。
userName	文字ストリング	About コマンドを実行するユーザーの名前。

### Cassette About

Cassette About オブジェクトは、カセットのバージョン情報を含んでいます。Payment Server 属性は、次のとおりです。

フィールド名	構文	説明
cassette	文字ストリング	カセットの決済システム名。
version	文字ストリング	カセットのバージョン。

## WebSphere Commerce Payments 管理オブジェクト

WebSphere Commerce Payments は、Payments の管理用に以下のフレームワーク・オブジェクトを定義します。

- Payment Server
- Cassette
- Merchant
- Payment System
- Account
- Event Listener
- User

それぞれの WebSphere Commerce Payments 管理オブジェクトは、その属性またはフィールドによって定義されます。以下のセクションのオブジェクト・テーブルは、それぞれの管理オブジェクトの表示フィールド名、フィールド構文、およびフィールドの説明を示しています。

### Payment Server

Payment Server オブジェクトは、WebSphere Commerce Payments の状態を説明しています。Payment Server の属性は次のとおりです。

表 21. PSPaymentServer オブジェクト属性

フィールド名	構文	説明
paymentServerHostname	文字ストリング	WebSphere Commerce Payments がインストールされているコンピューターのホスト名。
realmName	文字ストリング	WebSphere Commerce Payments によって現在使用されているレルムの名前。
numberOfOrderCommands	整数	WebSphere Commerce Payments が再始動してから、WebSphere Commerce Payments に対して実行されたオーダー・コマンドの数。
numberOfPaymentCommands	整数	WebSphere Commerce Payments が再始動してから、WebSphere Commerce Payments に対して実行された決済コマンドの数。
numberOfAdminCommands	整数	WebSphere Commerce Payments が再始動してから、WebSphere Commerce Payments に対して実行された管理コマンドの数。
numberOfQueryCommands	整数	WebSphere Commerce Payments が再始動してから、WebSphere Commerce Payments に対して実行された照会コマンドの数。
changesPending	ブール、XML 0 または 1	変更が WebSphere Commerce Payments に適用されたかどうかを示すフラグ。ここで、0= 偽、1= 真です。これらの変更は、WebSphere Commerce Payments が再始動されると有効になります。
enabled	ブール、XML 0 または 1	WebSphere Commerce Payments が使用可能になっているかどうか (つまり、書き込み可能かどうか) を示すフラグ。ここで、0= 偽、1= 真です。

表 21. PSPaymentServer オブジェクト属性 (続き)

フィールド名	構文	説明
active	ブール、XML 0 または 1	WebSphere Commerce Payments がアクティブになっているかどうか (つまり、使用できるようになっているかどうか) を示すフラグ。ここで、0= 偽、1= 真です。
valid	ブール、XML 0 または 1	WebSphere Commerce Payments が有効になっているかどうか (つまり、正しく構成されているかどうか) を示すフラグ。ここで、0= 偽、1= 真です。
paymentServerMsgs	文字ストリング	マーチャントの決済設定に関連したエラー、警告、または通知メッセージを識別する、WebSphere Commerce Payments によって生成されるメッセージ・コードのコンマ区切りのリスト。

## Cassette

Cassette オブジェクトは、WebSphere Commerce Payments にインストールされているカセットの状態を説明しています。Cassette オブジェクトの属性は、次のとおりです。

表 22. PSCassetteObject 属性

フィールド名	構文	説明
cassette	文字ストリング	カセットの名前 (たとえば、VisaNet または OfflineCard)。
companyPkgName	文字ストリング	カセット (カセットの Java パッケージ名を識別するために使用される) を開発した会社の名前。
changesPending	0 または 1 (ブール)	変更がカセットに適用されたかどうかを示すフラグ。ここで、0= 偽、1= 真です。これらの変更は、カセットを再始動してから有効になります。
enabled	0 または 1 (ブール)	カセットが使用可能になっているかどうか (つまり、カセットが書き込み可能かどうか) を示すフラグ。ここで、0= 偽、1= 真です。
active	0 または 1 (ブール)	カセットがアクティブになっているかどうか (つまり、カセットが使用できるようになっているかどうか) を示すフラグ。ここで、0= 偽、1= 真です。
valid	0 または 1 (ブール)	カセットが有効になっているかどうか (つまり、カセットが正しく構成されているかどうか) を示すフラグ。ここで、0= 偽、1= 真です。
cassetteMsgs	文字ストリング	XDM クライアント・アプリケーションのカセットに関連したエラー、警告、または通知メッセージを識別する、カセットによって生成されるメッセージ・コードのコンマ区切りのリスト。
paymentServerMsgs	文字ストリング	カセットに関連したエラー、警告、または通知メッセージを識別する、WebSphere Commerce Payments によって生成されるメッセージ・コードのリスト (コンマ区切り)。



## Merchant

Merchant オブジェクトは、WebSphere Commerce Payments を使用するよう定義されているマーチャントの状態を記述します。マーチャントの属性は、次のとおりです。

表 23. PSMerchantObject 属性

フィールド名	構文	説明
merchantNumber	数値トークン、1 ~ 9 桁の長さ	オーダーを作成したマーチャントを識別する番号。
merchantName	文字ストリング	マーチャントの名前。これは、WebSphere Commerce Payments ユーザー・インターフェースに、分かりやすい表示情報を提供するオプションのフィールドです。
changesPending	0 または 1 (ブール)	変更がカセットに適用されたかどうかを示すフラグ。ここで、0= 偽、1= 真です。これらの変更は、マーチャントが再度使用可能になると有効になります。
enabled	0 または 1 (ブール)	カセットが使用可能になっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。
active	0 または 1 (ブール)	カセットがアクティブになっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。
valid	0 または 1 (ブール)	カセットが有効になっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。
paymentServerMsgs	文字ストリング	マーチャントに関連したエラー、警告、または通知メッセージを識別する、 WebSphere Commerce Payments によって生成されるメッセージ・コードのコンマ区切りのリスト。

## Payment System

Payment System オブジェクトは、マーチャントがカセットについて行った設定を説明しています。カセット設定の属性は、次のとおりです。

表 24. PSMerchantCassetteSettingsObject 属性

フィールド名	構文	説明
cassette	文字ストリング	カセットの名前 (たとえば、VisaNet または OfflineCard)。
merchantNumber	数値トークン、1 ~ 9 桁の長さ	マーチャントを識別する番号。
changesPending	0 または 1 (ブール)	変更がカセットに適用されたかどうかを示すフラグ。ここで、0= 偽、1= 真です。これらの変更は、このマーチャントについてカセットが再始動されると有効になります。
enabled	0 または 1 (ブール)	カセットが使用可能になっているかどうか (つまり、カセットが書き込み可能かどうか) を示すフラグ。ここで、0= 偽、1= 真です。
active	0 または 1 (ブール)	カセットがアクティブになっているかどうか (つまり、カセットが使用できるようになっているかどうか) を示すフラグ。ここで、0= 偽、1= 真です。
valid	0 または 1 (ブール)	カセットが有効になっているかどうか (つまり、カセットが正しく構成されているかどうか) を示すフラグ。ここで、0= 偽、1= 真です。

表 24. *PSMerchantCassetteSettingsObject* 属性 (続き)

フィールド名	構文	説明
paymentServerMsgs	文字ストリング	決済システムに関連したエラー、警告、または通知メッセージを識別する、WebSphere Commerce Payments によって生成されるメッセージ・コードのコンマ区切りのリスト。

## Account

マーチャント Account オブジェクトは、マーチャントが金融機関との間で保持しているアカウントの状態を説明しています。Account の属性は、次のとおりです。

表 25. *PSMerchantAccountObject* 属性

フィールド名	構文	説明
cassette	文字ストリング	カセットの名前 (たとえば、VisaNet または OfflineCard)。
merchantNumber	数値トークン、1 ~ 9 桁の長さ	マーチャントを識別する番号。
merchantAccountNumber	数値トークン、1 ~ 9 桁の長さ	アカウントを識別する番号。この番号はローカルに (つまり、ホスティング・サービス・プロバイダーまたはマーチャント管理者によって) 作成され、追跡のために使用されます。
merchantAccountName	文字ストリング	アカウントの名前。これは、WebSphere Commerce Payments ユーザー・インターフェースに、分かりやすい表示情報を提供するオプションのフィールドです。
financialInstName	文字ストリング	金融機関の名前。これは、WebSphere Commerce Payments ユーザー・インターフェースに、分かりやすい表示情報を提供するオプションのフィールドです。
changesPending	0 または 1 (ブール)	変更がカセットに適用されたかどうかを示すフラグ。ここで、0= 偽、1= 真です。これらの変更は、アカウントが再始動されると有効になります。
enabled	0 または 1 (ブール)	カセットが使用可能になっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。
active	0 または 1 (ブール)	カセットがアクティブになっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。
valid	0 または 1 (ブール)	カセットが有効になっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。
cassetteMsgs	文字ストリング	アカウントまたは XDM クライアント・アプリケーションに関連したエラー、警告、または通知メッセージを識別する、カセットによって生成されるメッセージ・コードのコンマ区切りのリスト。
paymentServerMsgs	文字ストリング	アカウントに関連したエラー、警告、または通知メッセージを識別する、WebSphere Commerce Payments によって生成されるメッセージ・コードのリスト (コンマ区切り)。
apApproveFlag	数値トークン、1 ~ 9 桁の長さ	AcceptPayment 用の承認フラグ

表 25. PSMerchantAccountObject 属性 (続き)

フィールド名	構文	説明
apDepositFlag	0 または 1 (ブール)	0= 偽、1= 真。 AcceptPayment 用のデポジット・フラグ。 apApproveFlag が定義されており、0 に設定されていない場合のみ指定します。
rpApproveFlag	数値トークン、1 ~ 9 桁の長さ	ReceivePayment 用の承認フラグ
rpDepositFlag	0 または 1 (ブール)	0= 偽、1= 真。 ReceivePayment 用のデポジット・フラグ。 rpApproveFlag が定義されており、0 に設定されていない場合のみ指定します。
approvalExpiration	数値トークン、1 ~ 9 桁の長さ	決済が承認されたときから決算の承認の有効期限が切れるまでの日数を示す値。

## Event Listener

Event Listener オブジェクトは、登録された WebSphere Commerce Payments イベントの状態を説明しています。 Event Listener の属性は、次のとおりです。

表 26. PSEventListenerObject 属性

フィールド名	構文	説明
eventType	文字ストリング	モニターされているイベントのタイプ。
listenerURL	文字ストリング	イベント・タイプに定義されている URL。 WebSphere Commerce Payments イベント通知モデルは、特定のイベント・タイプに定義されているリスナー URL に送信されるメッセージを準備します。単一のイベント・タイプに複数の URL を定義できます。
timeRegistered	日付	マーチャントがイベント・タイプを登録した時刻。 GMT (グリニッジ標準時) 1970 年 1 月 1 日 0 時から経過したミリ秒の数値。
socksHost	文字ストリング	WebSphere Commerce Payments からイベント通知を受信している Socks サーバーのホスト名。 Socks サーバーを使用していない場合は、この値は null です。デフォルトは null です。
socksPort	文字ストリング	WebSphere Commerce Payments からイベント通知を受信している Socks サーバーのポート。 Socks サーバーを使用していない場合は、この値は null です。デフォルトは null です。
merchantNumber	数値トークン、1 ~ 9 桁の長さ	マーチャントを識別する番号。
changesPending	0 または 1 (ブール)	変更がカセットに適用されたかどうかを示すフラグ。ここで、0= 偽、1= 真です。これらの変更は、カセットを再始動してから有効になります。使用されていません。
enabled	0 または 1 (ブール)	カセットが使用可能になっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。
active	0 または 1 (ブール)	カセットがアクティブになっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。使用されていません。
valid	0 または 1 (ブール)	カセットが有効になっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。使用されていません。

表 26. *PSEventListenerObject* 属性 (続き)

フィールド名	構文	説明
paymentServerMsgs	文字ストリング	イベント・タイプに関連したエラー、警告、または通知メッセージを識別する、WebSphere Commerce Payments によって生成されるメッセージ・コードのコンマ区切りのリスト。

## User

User オブジェクトは、WebSphere Commerce Payments に定義されているユーザーの状態を説明しています。User の属性は、次のとおりです。

表 27. *PSUserObject* 属性

フィールド名	構文	説明
userName	文字ストリング	ユーザーの名前。
configuration	文字ストリング	ユーザー構成。
roleIDs	文字ストリング	ユーザーに定義されている役割 ID (つまり、クラーク、スーパーバイザー、マーチャント管理者、または WebSphere Commerce Payments 管理者)。
merchantNumber	数値トークン、1 ~ 9 桁の長さ	マーチャントを識別する番号。この値は WebSphere Commerce Payments 管理者以外のすべての役割に対して設定されます。QueryUsers コマンドの結果は、複数のマーチャントに対するアクセス権を持つユーザーを戻す場合があることに注意してください。この場合、WebSphere Commerce Payments はマーチャント番号を戻します。その際のマーチャント番号のリストは次のような構文になっています。 m1, m2, m3, . . .
changesPending	0 または 1 (ブール)	変更がユーザーに適用されたかどうかを示すフラグ。ここで、0= 偽、1= 真です。これらの変更は、マーチャントについてカセットが再始動されると有効になります。使用されていません。
enabled	0 または 1 (ブール)	カセットが使用可能になっているかどうかを示すフラグ。ここで、0= 偽、1= 真 (使用可能) です。
active	0 または 1 (ブール)	カセットがアクティブになっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。使用されていません。
valid	0 または 1 (ブール)	カセットが有効になっているかどうかを示すフラグ。ここで、0= 偽、1= 真です。使用されていません。
paymentServerMsgs	文字ストリング	ユーザーに関連したエラー、警告、または通知メッセージを識別する、WebSphere Commerce Payments によって生成されるメッセージ・コードのコンマ区切りのリスト。
objectCount		実際に一致したオブジェクトの数。

---

## 第 4 部 付録



## 付録 A. WebSphere Commerce Payments 戻りコード

戻りコードには、1 次戻りコード と 2 次戻りコード の両方が含まれます。

- 1 次戻りコード (PRC) は、WebSphere Commerce Payments の基本的な応答を説明します。

1 次戻りコードはコマンドごとに戻されます。

- 2 次戻りコード (SRC) は、追加情報を提供します。 WebSphere Commerce Payments は 2 つのタイプの汎用 SRC を定義しています。それらは、すべての PRC に共通のセットと、特定の PRC に固有のセットです。

SRC は、それぞれのコマンドで渡されるオプションの secondaryrc 構造体で戻されます。

プロトコル・カセットの作成者は、プロトコル固有のコードを使用してセットを拡張することができます。これらのコードに関する情報については、該当するカセットの補足情報を参照してください。

### 1 次戻りコード

次の表は、WebSphere Commerce Payments の 1 次戻りコード (PRC) を示しています。この表には、特定の 2 次戻りコード (SRC) がある PRC がリストされています。複数の PRC に関連する SRC は、134 ページの『2 次戻りコード (汎用)』に示されています。

表 28. 1 次戻りコード (PRC)

1 次戻りコード	値	説明
PRC_OPERATION_SUCCESS	0	操作は正常に終了しました。追加情報のためにゼロでない 2 次戻りコード (SRC) が提供される場合があります。
PRC_OPERATION_PENDING	1	API 呼び出しはまだ完了していません。それは WebSphere Commerce Payments エンティティーが使用可能になるまで保留にされています。 SRC によって、操作が保留されているリソースが示されています。
PRC_UNDEFINED_OBJECT	2	指定されたオブジェクトは見つかりませんでした。オブジェクトは SRC によって示されています。
PRC_PARAMETER_NOT_FOUND	3	必須パラメーターが見つかりませんでした。パラメーターは SRC によって示されています。
PRC_PARAMETER_TOO_SHORT	4	必須パラメーターが短すぎます。パラメーターは SRC によって示されています。

表 28. 1 次戻りコード (PRC) (続き)

1 次戻りコード	値	説明
PRC_PARAMETER_TOO_LONG	5	必須パラメーターが長すぎます。パラメーターは SRC によって示されています。
PRC_PARAMETER_FORMAT_ERROR	6	必須パラメーターの形式が不適切です。パラメーターは SRC によって示されています。
PRC_PARAMETER_VALUE_ERROR	7	必須パラメーターの値が誤りです。パラメーターは SRC によって示されています。
PRC_DUPLICATE_OBJECT	8	重複するオブジェクトが存在します。SRC によって示されているように、この決済番号の決済はすでに存在します。
PRC_PARAMETER_MISMATCH	9	パラメーターのミスマッチが生じました。パラメーターは SRC によって示されています。
PRC_INPUT_ERROR	10	入力ストリームの解析中にエラーが生じました。コマンドまたはそのパラメーターの 1 つの長さが無効です。
PRC_VERB_NOT_VALID_IN_PRESENT_STATE	11	このアクションを実行するためにはオブジェクトが適切な状態にありません。特定のオブジェクトは SRC によって示されています。
PRC_COMMUNICATION_ERROR	12	WebSphere Commerce Payments で通信エラーが発生しました。
PRC_INTERNAL_ETILL_ERROR	13	WebSphere Commerce Payments に予期しない内部エラーが生じました。
PRC_DATABASE_ERROR	14	データベース通信エラーが生じました。
PRC_CASSETTE_ERROR	15	カセット固有のエラーが生じました。資料については、カセットの補足情報を参照してください。
PRC_UNSUPPORTED_API_VERSION	17	アプリケーション・プログラムによって使用されている API のバージョンは、WebSphere Commerce Payments によってサポートされるバージョンよりも新しいものです。
PRC_OBSOLETE_API_VERSION	18	アプリケーションによって使用されている API のバージョンは、WebSphere Commerce Payments によってサポートされなくなっています。アプリケーション・プログラムをアップグレードして、廃止されたファンクションまたは機能を置換する新しいファンクションを使用してください。
PRC_AUTOAPPROVE_FAILED	19	ReceivePayment または AcceptPayment での自動認証が失敗しました。



表 28. 1 次戻りコード (PRC) (続き)

1 次戻りコード	値	説明
PRC_AUTODEPOSIT_FAILED	20	ReceivePayment または AcceptPayment での自動デポジットが失敗しました。
PRC_CASSETTE_NOTRUNNING	21	カセットが実行されていません。
PRC_CASSETTE_NOTVALID	22	カセットが無効です。
PRC_UNSUPPORTED_IN_SYSPLEX	23	この操作は、シスプレックス環境ではサポートされていません。
PRC_PARAMETER_NULL_VALUE	24	パラメーターにヌル値があります。
PRC_XML_ERROR	30	XML 文書に誤りがあります。
PRC_COREQUISITE_PARAMETER_NOT_FOUND	31	もう一方のパラメーターが指定される時には、このパラメーターも指定する必要があります。
PRC_INVALID_PARAMETER_COMBINATION	32	API コマンドに指定されたパラメーターの組み合わせは許可されていません。
PRC_BATCH_ERROR	33	バッチ操作に関連したエラーが発生しました。
PRC_FINANCIAL_FAILURE	34	操作が金融上の理由により失敗しました。
PRC_SERVLET_INIT_ERROR	50	サーブレットの初期化の際にエラーが発生しました。
PRC_AUTHENTICATION_ERROR	51	ユーザー認証中にエラーが発生しました。
PRC_AUTHORIZATION_ERROR	52	ユーザー与信中にエラーが発生しました。
PRC_UNHANDLED_EXCEPTION	53	未処理の (ヌル・ポインターなど) 例外が発生しました。
PRC_DUPLICATE_PARAMETER_VALUE_NOT_ALLOWED	54	この API コマンドではパラメーターを複数回指定できません。
PRC_COMMAND_NOT_SUPPORTED	55	コマンド名が、有効な WebSphere Commerce Payments コマンドとして認識されません。
PRC_CRYPTO_ERROR	56	暗号機能のキーに関連したエラー。
PRC_NOT_ACTIVE	57	管理オブジェクトがアクティブになっていません。
PRC_PARAMETER_NOT_ALLOWED	58	パラメーターは指定できません。
PRC_DELETE_ERROR	59	オブジェクトを削除できませんでした。
PRC_WEBSHERE	60	WebSphere/WebServer に関連したエラーが発生しました。
PRC_SUPPORTED_IN_SYSPLEX_ADMIN_ONLY	61	シスプレックス管理者として指定された WebSphere Commerce Payments の場合にのみ、要求はシスプレックス・モードでサポートされます。
PRC_REALM	62	レルムに関連したエラーが発生しました。

## 2 次戻りコード (汎用)

表 29. 汎用 2 次戻りコード (SRC)

2 次戻りコード	値	説明
RC_NONE	0	追加情報はありません。
RC_INITIALIZATION_MESSAGE	1	戻りデータ・バッファに初期化メッセージが入っています。このバッファは、このルーチンの呼び出し元によって解放されなければなりません。
RC_INPUT_ERROR_TOO_LONG	2	入力ストリームが最大長を超えています。
RC_INPUT_ERROR_UNKNOWN_COMMAND	3	コマンドが無効です。
RC_UNEXPECTED	4	予期しないエラーが発生しました。
RC_COMMUNICATION_ERROR_INPUT	5	WebSphere Commerce Payments が、マーチャント・サーバーからのデータの読み取り時に例外を受け取りました。
RC_API_INITIALIZE_FAILURE	6	API の初期化が失敗しました。
RC_MERCHANTNUMBER	110	応答はマーチャント番号パラメーターを参照しています。
RC_ORDERNUMBER	111	応答はオーダー番号パラメーターを参照しています。
RC_PAYMENTNUMBER	112	応答は PAYMENTNUMBER パラメーターを参照しています。
RC_CREDITNUMBER	113	応答は CREDITNUMBER パラメーターを参照しています。
RC_BATCHNUMBER	114	応答は BATCHNUMBER パラメーターを参照しています。(注: 旧バージョンでは、この戻りコードは BATCHID パラメーターを参照していました。)
RC_ACCOUNTNUMBER	115	応答は ACCOUNTNUMBER パラメーターを参照しています。
RC_PAYMENTTYPE	116	応答は PAYMENTTYPE パラメーターを参照しています。
RC_AMOUNT	117	応答は AMOUNT パラメーターを参照しています。
RC_AMOUNTEXP10	118	応答は AMOUNTEXP10 パラメーターを参照しています。
RC_CURRENCY	119	応答は CURRENCY パラメーターを参照しています。
RC_OD	120	応答はオーダー記述パラメーターを参照しています。
RC_CHARSET	121	応答は文字セット・パラメーターを参照しています。
RC_SUCCESSURL	122	応答は成功 URL パラメーターを参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_FAILURL	123	応答は失敗 URL パラメーターを参照しています。
RC_CANCELURL	124	応答はキャンセル URL パラメーターを参照しています。
RC_APPROVEFLAG	125	応答は承認フラグ・パラメーターを参照しています。
RC_PAYMENTAMOUNT	126	応答は決済金額パラメーターを参照しています。
RC_SPLITFLAG	127	応答は分割許可のパラメーターを参照しています。
RC_DEPOSITFLAG	128	応答はデPOSIT・フラグ・パラメーターを参照しています。
RC_PROTOCOLDATA	129	応答はプロトコル・データ・パラメーターを参照しています。
RC_ORDERURLS	130	応答はオーダー URL パラメーターを参照しています。
RC_SERVICEURL	131	応答はサービス URL パラメーターを参照しています。
RC_CASSETTECOMMAND	132	応答はカセット・コマンド・パラメーターを参照しています。
RC_USERNAME	133	応答はユーザー・パラメーターを参照しています。
RC_EVENTTYPE	134	応答はイベント・タイプ・パラメーターを参照しています。
RC_WITHCREDITS	135	応答は withCredits パラメーターを参照しています。
RC_CREATEBEGINTIME	136	応答は作成開始時刻パラメーターを参照しています。
RC_CREATEENDTIME	137	応答は作成終了時刻パラメーターを参照しています。
RC_MINAMOUNT	138	応答は最低金額パラメーターを参照しています。
RC_MAXAMOUNT	139	応答は最高金額パラメーターを参照しています。
RC_RETURNATMOST	140	応答は「最大戻し」パラメーターを参照しています。
RC_KEYSONLY	141	応答はキー専用パラメーターを参照しています。
RC_DTDPATH	143	応答は dtd パス・パラメーターを参照しています。
RC_REFERENCENUMBER	144	応答は参照番号パラメーターを参照しています。
RC_WITHORDERS	145	応答は withOrders パラメーターを参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_MESSAGES	146	応答はメッセージ・キーを参照しています。
RC_OPENBEGINTIME	147	応答はバッチ・オープン開始時刻パラメーターを参照しています。
RC_OPENENDTIME	148	応答はバッチ・オープン終了時刻パラメーターを参照しています。
RC_CLOSEBEGINTIME	149	応答はバッチ・クローズ開始時刻パラメーターを参照しています。
RC_CLOSEENDTIME	150	応答はバッチ・クローズ終了時刻パラメーターを参照しています。
RC_STATUS	151	応答は状況パラメーターを参照しています。
RC_CLOSEALLOWED	153	応答はクローズ許可のパラメーターを参照しています。
RC_WITHPAYMENTS	154	応答は withPayments パラメーターを参照しています。
RC_TIMEREGISTERED	155	応答はタイム・レジスター・パラメーターを参照しています。
RC_MINAPPROVEAMOUNT	156	応答は最低承認額パラメーターを参照しています。
RC_MAXAPPROVEAMOUNT	157	応答は最高承認額パラメーターを参照しています。
RC_MINDEPOSITAMOUNT	158	応答は最低デPOSIT額パラメーターを参照しています。
RC_MAXDEPOSITAMOUNT	159	応答は最高デPOSIT額パラメーターを参照しています。
RC_ORDERURL	160	応答はオーダー URL パラメーターを参照しています。
RC_MODIFYBEGINTIME	161	応答は変更開始時刻パラメーターを参照しています。
RC_MODIFYENDTIME	162	応答は変更終了時刻パラメーターを参照しています。
RC_DELETEORDER	165	応答はオーダーの削除パラメーターを参照しています。
RC_MINUNAPPROVEDAMOUNT	166	応答は最低未認証金額パラメーターを参照しています。
RC_MAXUNAPPROVEDAMOUNT	167	応答は最高未認証金額パラメーターを参照しています。
RC_APPROVESALLOWED	168	応答は認証許可のパラメーターを参照しています。
RC_PURGEALLOWED	169	応答は PURGEALLOWED パラメーターを参照しています。
RC_MAXBATCHSIZE	170	応答は \$MAXBATCHSIZE パラメーターを参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CHECK_CASSETTE_STATUS	171	詳細については、カセット固有のデータを検査してください。
RC_FORCE	172	応答は FORCE パラメーターを参照しています。 BATCHCLOSE コマンドに 応答して戻される場合があります。 1 次戻りコードで記述されているエラー が、ブール・パラメーター FORCE を 参照していることを示します。
RC_AP_APPROVEFLAG	173	応答は acceptPayment 承認フラグ・パ ラメーターを参照しています。
RC_AP_DEPOSITFLAG	174	応答は acceptPayment デPOSIT・フ ラグ・パラメーターを参照していま す。
RC_RP_APPROVEFLAG	175	応答は receivePayment 承認フラグ・パ ラメーターを参照しています。
RC_RP_DEPOSITFLAG	176	応答は receivePayment デPOSIT・フ ラグ・パラメーターを参照していま す。
RC_APPROVALEXPIRATION	177	応答は ApprovalExpiration パラメーター を参照しています。
RC_MERCHANTPAYSYS	202	応答はマーチャント決済システムを参 照しています。
RC_ACCOUNT	203	応答はアカウントを参照しています。
RC_ORDER	204	応答はオーダー・エンティティを参 照しています。
RC_PAYMENT	205	応答は決済エンティティを参照して います。
RC_CREDIT	206	応答はクレジット・エンティティを 参照しています。
RC_BATCH	207	応答はバッチ・エンティティを参照 しています。
RC_BRAND	208	応答はカード・ブランドを参照してい ます。
RC_STATE	209	応答は状態を参照しています。
RC_MULTIPLE_BATCHES	211	応答は Batch オブジェクトを参照して います。
RC_AUTOMATIC_CREATION	212	自動バッチ・オープン中にエラーが発 生しました。
RC_BATCH_EMPTY	213	バッチが空です。決済またはクレジット が含まれていないバッチのクローズ が試行されました。これがエラー状態 であるかどうかを決めるのは、カセッ トです。
RC_COMMTYPE	215	応答は通信タイプを参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_PAYMENTGROUPNAME	216	応答は決済グループ名を参照しています。
RC_ADMINHOSTNAME	217	応答は管理ホスト名を参照しています。
RC_NDHOSTNAME	218	応答は Net.Dispatcher ホスト名を参照しています。
RC_PLEXNAME	219	応答はシスプレックス名を参照しています。
RC_UNKNOWN_ETILL_HOST	301	指定の WebSphere Commerce Payments ホストは無効です。
RC_HOSTNAME_NOT_VALID	303	WebSphere Commerce Payments ホスト名パラメーターに誤りがあります。
RC_HOST_IP_ADDRESS_UNAVAILABLE	306	ホスト IP アドレスが見つかりません。
RC_SOCKET_STARTUP_FAILURE	307	ソケット・ライブラリーを初期化できません。
RC_HANDLE_REQUIRED	308	PaymentServerHandle はこの API で必須です。
RC_COMMUNICATION_ERROR	309	通信エラーが発生しました。
RC_RESERVED_BITS_SET_IN_FLAGS	310	将来の利用のために予約されているビットがゼロではありません。それらはゼロでなければなりません。
RC_TIME_PERIOD_INVALID	311	TimePeriod で指定された値が無効です。
RC_PROTOCOL_DATA_KEYWORD_INVALID	312	プロトコル・データの中のキーワードが無効です。
RC_AMOUNT_RANGE_INVALID	313	金額の範囲が無効です。
RC_SOCKET_CREATION_FAILED	320	WebSphere Commerce Payments と通信するためのソケットをオープンできませんでした。TCP/IP ソケット・リソースが使い尽くされている可能性があります。
RC_CONNECTION_TO_PAYMENT_SERVER_FAILED	321	以前に etInitializeAPI() 呼び出しで指定されたポートとアドレスを使用して、WebSphere Commerce Payments へのネットワーク接続をオープンできませんでした。
RC_SEND_OF_DATA_ON_SOCKET_FAILED	322	WebSphere Commerce Payments とのネットワーク接続で、データを送信できませんでした。WebSphere Commerce Payments によって接続がクローズされている可能性があります。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_RECEIVE_OF_DATA_ON_SOCKET_FAILED	323	WebSphere Commerce Payments とのネットワーク接続で、データを受信できませんでした。 WebSphere Commerce Payments によって接続がクローズされている可能性があります。
RC_ERROR_CHECKING_FOR_READ_DATA	324	WebSphere Commerce Payments とのネットワーク接続で、読み取りの準備ができていたデータをチェックできませんでした。 WebSphere Commerce Payments によって接続がクローズされている可能性があります。
RC_SOCKET_CLOSE_FAILED	325	ソケットのクローズに失敗しました。
RC_ENCODING_EXCEPTION	400	エンコード・エラーが発生しました。
RC_UNSUPPORTED_DOCUMENT_TYPE	401	この XML 文書タイプはサポートされていません。
RC_EMPTY_DOCUMENT	402	文書が空です。
RC_MISSING_ORDER_COLLECTION	403	オーダーのコレクションがありません。
RC_DOCUMENT_TOO_LARGE	404	XDM 照会によって生成された XML 文書が大き過ぎます。検索基準を改善して照会を再試行してください。
RC_SERVLET_INIT_EXCEPTION	500	サーブレットの初期化中にエラーが発生しました。
RC_CANNOT_FIND_PROPERTY_FILE	501	プロパティ・ファイルが見つかりません。
RC_ERROR_LOADING_PROPERTY_FILE	502	プロパティ・ファイルのロード中にエラーが発生しました。
RC_ERROR_JBCDRIVER_NAME	503	応答は JDBC ドライバー名を参照しています。
RC_ERROR_JDBCURL	504	応答は JDBC URL を参照しています。
RC_ERROR_DBOWNER	505	応答はデータベース所有者を参照しています。
RC_ERROR_DBUSERID	506	応答はデータベース・ユーザー ID を参照しています。
RC_ERROR_DBPASSWORD	507	応答はデータベース・パスワードを参照しています。
RC_ERROR_HOSTNAME	509	応答はホスト名を参照しています。
RC_ERROR_PSENGINE_PORTNUMBER	510	応答は WebSphere Commerce Payments エンジン・ポート番号を参照しています。
RC_ERROR_LOADING_JBCDRIVER	511	JDBC ドライバーのロード中にエラーが発生しました。
RC_ERROR_CONNECTING_DATABASE_OR_EXEC_SQL	512	データベースへの接続中か SQL 文の実行中にエラーが発生しました。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_ERROR_INIT_ERROR_LOG	513	エラー・ログの初期化中にエラーが発生しました。
RC_ERROR_LOADING_CASSETTE	514	カセットのロード中にエラーが発生しました。
RC_ERROR_ROOT_PASSWORD	515	ルート・パスワードが無効です。
RC_ERROR_MAXDBCONNECTIONS	516	応答はデータベース接続の最大数を参照しています。
RC_ERROR_MINSENSITIVEACCESSROLE	517	応答は、重要な金融データを表示するために最低限必要な役割を参照しています。
RC_NEW_PASSWORD	518	パラメーターは新しいパスワードを参照しています。
RC_DATA_SOURCE	519	パラメーターはデータ・ソース名を参照しています。
RC_OPERATION	530	応答は操作パラメーターを参照しています。
RC_ETAPIVERSION	531	応答は etApiVersion パラメーターを参照しています。
RC_AUTHENTICATED_USER_NOT_GIVEN	553	WebSphere Commerce Payments コマンドについては、認証されたユーザーはいません。
RC_USER_NOT_AUTHORIZED	554	指定のユーザーには、要求された操作を実行する許可は与えられていません。
RC_ERROR_PROTECTION_REALM_NOT_SPECIFIED	555	PaymentServlet.properties ファイルの中の ProtectedRealm 設定値に名前が指定されていません。
RC_SPECIFIED_REALM_UNKNOWN	556	PaymentServlet.properties ファイルの中で指定されている領域は不明です。
RC_REALMCLASS	557	応答は eTill.RealmClass プロパティを参照しています。
RC_PAYSERVER_ADMIN	600	応答は WebSphere Commerce Payments 管理エンティティを参照しています。
RC_CASSETTE_ADMIN	601	応答はカセット管理エンティティを参照しています。
RC_MERCHANT_ADMIN	602	応答はマーチャント管理エンティティを参照しています。
RC_PAYMENTSYSYSTEM_ADMIN	603	応答は決済システム管理エンティティを参照しています。
RC_ACCOUNT_ADMIN	604	応答はアカウント管理エンティティを参照しています。
RC_ETILLHOSTNAME	611	応答は ETILLHOSTNAME パラメーターを参照しています。



表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTENAME	615	応答は CASSETTENAME パラメーターを参照しています。
RC_MERCHANTTITLE	616	応答は MERCHANTTITLE パラメーターを参照しています。
RC_ACCOUNTTITLE	617	応答は ACCOUNTTITLE パラメーターを参照しています。
RC_FINANCIALINSTITUTION	618	応答は FINANCIALINSTITUTION パラメーターを参照しています。
RC_OBJECTNAME	619	応答は OBJECTNAME パラメーターを参照しています。
RC_ENABLED	620	応答は ENABLED パラメーターを参照しています。
RC_EVENT_LISTENER	621	応答は EVENTLISTENER オブジェクトを参照しています。
RC_LISTENERURL	622	応答は LISTENERURL パラメーターを参照しています。
RC SOCKSPORT	623	応答は SOCKSPORT パラメーターを参照しています。
RC_ROLE	624	応答はユーザー役割パラメーターを参照しています。
RC_USER	625	応答は User オブジェクトを参照しています。
RC_USER_NOT_ENABLED	626	応答はユーザー (このユーザーは使用可能ではない) を参照しています。
RC_USER_MISCONFIGURED	627	応答は User オブジェクト (このユーザーは WebSphere Commerce Payments の権限を持っています。このユーザーは正しく構成されていません) を参照しています。
RC_KEY_TAMPERED	628	暗号鍵が変更されています。
RC_KEY_NOT_EXIST	629	指定のコンポーネントの暗号鍵が存在しません。
RC SOCKSHOST	630	応答は SOCKSHOST パラメーターを参照しています。
RC_ENCRYPT_ENCRYPTION_KEY_FAILED	631	暗号鍵の暗号化が失敗しました。
RC_DECRYPT_ENCRYPTION_KEY_FAILED	632	暗号鍵の暗号解除が失敗しました。
RC_ENCRYPTION_KEY_TYPE_NOT_SUPPORTED	633	この暗号鍵タイプはサポートされていません。
RC_VALIDATE_ENCRYPTION_KEY_FAILED	634	暗号鍵の妥当性検査が失敗しました。
RC_GENERATE_ENCRYPTION_KEY_FAILED	635	暗号鍵の生成が失敗しました。
RC_NOT_ACL_OWNER	636	このユーザーは ACL の所有者ではありません。
RC_BAD_REALM	637	レルム・エラーが発生しました。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_NO_SUCH_ACL	638	ACL が定義されていません。
RC_LAST_ACL_OWNER	639	このユーザーは ACL の最後の所有者です。
RC_NO_SUCH_USER	640	このユーザーは WebSphere レルムに定義されていません。
RC_FILTER	642	応答は FILTER パラメーターを参照しています。
RC_TRANSACTIONID	643	応答は TRANSACTIONID パラメーターを参照しています。
RC_ORDERDATA1	644	応答は ORDERDATA1 パラメーターを参照しています。
RC_ORDERDATA2	645	応答は ORDERDATA2 パラメーターを参照しています。
RC_ORDERDATA3	646	応答は ORDERDATA3 パラメーターを参照しています。
RC_ORDERDATA4	647	応答は ORDERDATA4 パラメーターを参照しています。
RC_ORDERDATA5	648	応答は ORDERDATA5 パラメーターを参照しています。
RC_SERVICE_POOL	649	応答は、サービス・スレッド・プール・サイズ wpm.spoolsize を参照しています。
RC_INVALID_CHANGEPASSWORD_STATE	650	PM パスワードの変更は、WebSphere Commerce Payments Application Server の始動直後に限って有効です。
RC_ASYNCHAPPDELAY	651	応答は wpm.AsynApproveDelayTimeInSecs パラメーターを参照しています。
RC_APPEXPDELAY	652	応答は wpm.ApprovalExpirationDelayTimeInMins パラメーターを参照しています。
RC_PROTOCOL_POOL	653	応答は、プロトコル・スレッド・プール・サイズ wpm.ppoolsize を参照しています。
RC_CASSETTE_PCARD_SHIPPINGAMOUNT	900	応答は、購入カード・データの出荷量パラメーターを参照しています。
RC_CASSETTE_PCARD_DUTYAMOUNT	901	応答は、購入カード・データの税額パラメーターを参照しています。
RC_CASSETTE_PCARD_DUTYREFERENCE	902	応答は、購入カード・データの税参照パラメーターを参照しています。
RC_CASSETTE_PCARD_NATIONALTAXAMOUNT	903	応答は、購入カード・データの国税額パラメーターを参照しています。
RC_CASSETTE_PCARD_NATIONALTAXRATE	904	応答は、購入カード・データの国税率パラメーターを参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTE_PCARD_LOCALTAXAMOUNT	905	応答は、購入カード・データの地方税額パラメーターを参照しています。
RC_CASSETTE_PCARD_OTHERTAXAMOUNT	906	応答は、購入カード・データのその他の税の金額パラメーターを参照しています。
RC_CASSETTE_PCARD_TOTALTAXAMOUNT	907	応答は、購入カード・データの税額合計パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTTAXID	908	応答は、購入カード・データのマーチャントの税 ID パラメーターを参照しています。
RC_CASSETTE_PCARD_ALTERNATETAXID	909	応答は、購入カード・データの代替税 ID パラメーターを参照しています。
RC_CASSETTE_PCARD_TAXEXEMPTINDICATOR	910	応答は、購入カード・データの免税標識パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTDUTYTARIFFREFERENCE	911	応答は、購入カード・データのマーチャント関税参照パラメーターを参照しています。
RC_CASSETTE_PCARD_CUSTOMERDUTYTARIFFREFERENCE	912	応答は、購入カード・データの顧客関税参照パラメーターを参照しています。
RC_CASSETTE_PCARD_SUMMARYCOMMODITYCODE	913	応答は、購入カード・データの要約商品コード・パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTTYPE	914	応答は、購入カード・データのマーチャント・タイプ・パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTCOUNTRYCODE	915	応答は、購入カード・データのマーチャント国別コード・パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTCITYCODE	916	応答は、購入カード・データのマーチャント市区町村コード・パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTSTATEPROVINCE	917	応答は、購入カード・データのマーチャント都道府県コード・パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTPOSTALCODE	918	応答は、購入カード・データのマーチャント郵便番号パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTLOCATIONID	919	応答は、購入カード・データのマーチャントのロケーション ID パラメーターを参照しています。
RC_CASSETTE_PCARD_MERCHANTNAME	920	応答は、購入カード・データのマーチャント名パラメーターを参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTE_PCARD_SHIPFROMCOUNTRYCODE	921	応答は、購入カード・データの配送元 国別コード・パラメーターを参照して います。
RC_CASSETTE_PCARD_SHIPFROMCITYCODE	922	応答は、購入カード・データの配送元 市区町村コード・パラメーターを参照 しています。
RC_CASSETTE_PCARD_SHIPFROMSTATEPROVINCE	923	応答は、購入カード・データの配送元 都道府県コード・パラメーターを参照 しています。
RC_CASSETTE_PCARD_SHIPFROMPOSTALCODE	924	応答は、購入カード・データの配送元 郵便番号パラメーターを参照していま す。
RC_CASSETTE_PCARD_SHIPFROMLOCATIONID	925	応答は、購入カード・データの配送元 ロケーション ID パラメーターを参照 しています。
RC_CASSETTE_PCARD_SHIPTOCOUNTRYCODE	926	応答は、購入カード・データの配送先 国別コード・パラメーターを参照して います。
RC_CASSETTE_PCARD_SHIPTOCITYCODE	927	応答は、購入カード・データの配送先 市区町村コード・パラメーターを参照 しています。
RC_CASSETTE_PCARD_SHIPTOSTATEPROVINCE	928	応答は、購入カード・データの配送先 都道府県コード・パラメーターを参照 しています。
RC_CASSETTE_PCARD_SHIPTOPOSTALCODE	929	応答は、購入カード・データの配送先 郵便番号パラメーターを参照していま す。
RC_CASSETTE_PCARD_SHIPTOLOCATIONID	930	応答は、購入カード・データの配送先 ロケーション ID パラメーターを参照 しています。
RC_CASSETTE_PCARD_MERCHANTORDERNUMBER	931	応答は、購入カード・データのマーチ ャントオーダー番号パラメーターを参 照しています。
RC_CASSETTE_PCARD_CUSTOMERREFERENCENUMBER	932	応答は、購入カード・データの顧客参 照番号パラメーターを参照していま す。
RC_CASSETTE_PCARD_ORDERSUMMARY	933	応答は、購入カード・データのオーダ ー要約パラメーターを参照していま す。
RC_CASSETTE_PCARD_CUSTOMERSERVICEPHONE	934	応答は、購入カード・データのお客様 サービス電話パラメーターを参照して います。
RC_CASSETTE_PCARD_DISCOUNTAMOUNT	935	応答は、購入カード・データの割引額 パラメーターを参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTE_PCARD_SHIPPINGNATIONALTAXRATE	936	応答は、購入カード・データの配送国税率パラメーターを参照しています。
RC_CASSETTE_PCARD_SHIPPINGNATIONALTAXAMOUNT	937	応答は、購入カード・データの配送国税額パラメーターを参照しています。
RC_CASSETTE_PCARD_NATIONALTAXINVOICEREFERENCE	938	応答は、購入カード・データの国税送り状参照パラメーターを参照しています。
RC_CASSETTE_PCARD_PRINTCUSTOMERSERVICEPHONE NUMBER	939	応答は、購入カード・データの印刷お客様サービス電話番号パラメーターを参照しています。
RC_CASSETTE_ITEM_COMMODITYCODE	940	応答は品目データの商品コード・パラメーターを参照しています。
RC_CASSETTE_ITEM_PRODUCTCODE	941	応答は品目データの製品コード・パラメーターを参照しています。
RC_CASSETTE_ITEM_DESCRIPTOR	942	応答は品目データの記述子パラメーターを参照しています。
RC_CASSETTE_ITEM_QUANTITY	943	応答は品目データの数量パラメーターを参照しています。
RC_CASSETTE_ITEM_SKU	944	応答は品目データの SKU パラメーターを参照しています。
RC_CASSETTE_ITEM_UNITCOST	945	応答は品目データの単価パラメーターを参照しています。
RC_CASSETTE_ITEM_UNITOFMEASURE	946	応答は品目データの計測単位パラメーターを参照しています。
RC_CASSETTE_ITEM_NETCOST	947	応答は品目データの純費用パラメーターを参照しています。
RC_CASSETTE_ITEM_DISCOUNTAMOUNT	948	応答は品目データの割引額パラメーターを参照しています。
RC_CASSETTE_ITEM_DISCOUNTINDICATOR	949	応答は品目データの割引標識パラメーターを参照しています。
RC_CASSETTE_ITEM_NATIONALTAXAMOUNT	950	応答は品目データの国税額パラメーターを参照しています。
RC_CASSETTE_ITEM_NATIONALTAXRATE	951	応答は品目データの国税率パラメーターを参照しています。
RC_CASSETTE_ITEM_NATIONALTAXTYPE	952	応答は品目データの国税タイプ・パラメーターを参照しています。
RC_CASSETTE_ITEM_LOCALTAXAMOUNT	953	応答は品目データの地方税額パラメーターを参照しています。
RC_CASSETTE_ITEM_LOCALTAXRATE	954	応答は品目データの地方税率パラメーターを参照しています。
RC_CASSETTE_ITEM_OTHERTAXAMOUNT	955	応答は品目データのその他の税の金額パラメーターを参照しています。
RC_CASSETTE_ITEM_TOTALCOST	956	応答は品目データの総費用パラメーターを参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTE_FUNCTION_NOT_SUPPORTED	1000	カセットはこのコマンドをサポートしていません。
RC_CASSETTE_UNSPECIFIED_ERROR	1001	カセットはこのコマンドをサポートしていません。
RC_CASSETTE_BATCH_ID	1002	バッチ ID が、(1) 禁止されているときに指定されたか、または (2) 要求されているときに指定されていません。
RC_CASSETTE_REFUND_AMOUNT_NOT_ZERO	1003	カセットでは、リファンドの完全な取り消しのみが許可されます (つまり、金額はゼロでなければなりません)。
RC_CASSETTE_OPERATION_FAILED	1004	操作に金融上の障害が生じました。
RC_CASSETTE_ENCRYPTION_ERROR	1008	カセットがプロトコル・メッセージを構成または処理している間に、暗号化エラーが発生しました。
RC_CASSETTE_DECRYPTION_ERROR	1009	カセットがプロトコル・メッセージを構成または処理している間に、暗号化解除エラーが発生しました。
RC_CASSETTE_IMPLICIT_BATCHES_ONLY	1010	アカウントに関連した金融プロセッサ以外に BATCHOPEN または BATCHCLOSE コマンドだけが、バッチ処理を制御します。
RC_CASSETTE_BATCH_CURRENCY	1011	バッチ内のすべての取り引きについての通貨は同じでなければなりません。
RC_CASSETTE_BATCH_AMOUNTEXP10	1012	バッチ内のすべての取り引きについての金額指数は同じでなければなりません。
RC_CASSETTE_BRAND	1014	応答はブランド・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_PAN	1015	応答は PAN パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_EXPIRY	1016	応答は有効期限パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_DEPOSIT_AMOUNT_NOT_ZERO	1017	このアカウントでは、デポジットの完全な取り消しのみが許可されます (つまり、金額はゼロでなければなりません)。
RC_CASSETTE_COMMUNICATION_ERROR	1018	カセットとカセットが通信するエンティティとの間で通信エラーが発生しました。
RC_CASSETTE_INTERMEDIATE_RESPONSE_NULL	1019	カセットは、カセットが通信するエンティティからヌル応答を受信しました。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTE_INTERMEDIATE_RESPONSE_UNEXPECTED	1020	カセットは、カセットが通信するエンティティから予期しない応答を受信しました。
RC_CASSETTE_BATCH_ERROR	1021	バッチ関連のエラーが発生しました。
RC_CASSETTE_BATCH_BALANCE_ERROR	1022	このバッチについて計算された収支合計が、WebSphere Commerce Payments と金融機関で一致しませんでした。
RC_CASSETTE_APPROVE_NO_DEPOSIT	1040	自動デPOSITで APPROVE を処理している間に、カセットは、承認を正常に完了させましたが、デPOSITを正常に完了させることはできませんでした。
RC_CASSETTE_DECLINED	1041	金融機関は、不明の理由により要求を拒否しました。
RC_CASSETTE_DECLINED_EXPIRY	1042	金融機関は、満期の値のために要求を拒否しました。
RC_CASSETTE_DECLINED_INSTRUMENT	1043	金融機関は、購買の手段 (クレジット・カード、小切手、または何であれこのカセットの決済プロトコルが使用する手段) に問題があるために要求を拒否しました。
RC_CASSETTE_AVSDATA	1051	応答は AVS パラメーターのグループ (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AVS_COUNTRYCODE	1052	応答は AVS 国別コード・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AVS_STREETADDRESS	1053	応答は AVS 番地パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AVS_CITY	1054	応答は AVS 市町村パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AVS_STATEPROVINCE	1055	応答は AVS 都道府県パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AVS_POSTALCODE	1056	応答は AVS 郵便番号パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AVS_LOCATIONID	1057	応答は AVS 場所 ID パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_CARDHOLDERNAME	1058	応答はカード所有者名パラメーター (プロトコル・データの中で指定されている) を参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTE_MAXBATCHSIZE	1059	応答は最大バッチ・サイズ・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_CURRENCY	1060	応答は通貨パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_HUMAN_INTERVENTION_REQUIRED	1061	操作が完全にまたは部分的に失敗しました。この障害を解決するには、ユーザーの介入が必要です。
RC_CASSETTE_DECLINED_APPROVAL_EXPIRED	1062	決済の承認は有効期限が切れました。正常にデPOSITする前に、決済金額の新規の承認を入手する必要があります。カセットが ApproveReversal をサポートする場合、それを使用して、既存の決済用の新規の承認を入手します。サポートしていない場合、Approve を使用して、新規の承認済み決済を作成して、それに続いてデPOSITします。
RC_CASSETTE_AMOUNT_WOULD_EXCEED_ORDER_AMOUNT	1063	指定した金額の承認が原因で、すべての決済の累積金額が、元のオーダー金額を超える場合があります。
RC_CASSETTE_VERSION	1064	データベース・テーブルで指定されているカセットのバージョンが、最大長を超えています。
RC_CASSETTE_CARDVERIFYCODE	1065	応答は指定したカード確認コードを参照しています。
RC_CASSETTE_AUTHCODE	1066	応答は権限コードを参照しています。
RC_CASSETTE_DECLINECODE	1067	応答は拒否コードを参照しています。
RC_REALM_INIT_ERROR	1068	定義されたレルムを初期化できませんでした。
RC_REALM_OPERATION_ERROR	1069	定義されたレルムの使用中にエラーが発生しました。
RC_CASSETTE_SHIPPINGDATA	1071	応答は出荷住所パラメーターのグループ (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_SHIP_COUNTRYCODE	1072	応答は出荷国別コード・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_SHIP_STREETADDRESS	1073	応答は出荷番地パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_SHIP_CITY	1074	応答は出荷市町村パラメーター (プロトコル・データの中で指定されている) を参照しています。



表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTE_SHIP_STATEPROVINCE	1075	応答は出荷都道府県パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_SHIP_POSTALCODE	1076	応答は出荷郵便番号パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_BILLINGDATA	1081	応答は請求住所パラメーターのグループ (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_BILL_COUNTRYCODE	1082	応答は請求国別コード・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_BILL_STREETADDRESS	1083	応答は請求番地パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_BILL_CITY	1084	応答は請求市町村パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_BILL_STATEPROVINCE	1085	応答は請求都道府県パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_BILL_POSTALCODE	1086	応答は請求郵便番号パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_ACCEPTPAYMENTAUTOAPPROVE	1087	応答は AcceptPayment でのマーチャント・アカウントの承認フラグを参照しています。
RC_ACCEPTPAYMENTAUTODEPOSIT	1088	応答は AcceptPayment でのマーチャント・アカウントのデポジット・フラグを参照しています。
RC_RECEIVEPAYMENTAUTOAPPROVE	1089	応答は ReceivePayment でのマーチャント・アカウントの承認フラグを参照しています。
RC_RECEIVEPAYMENTAUTODEPOSIT	1090	応答は ReceivePayment でのマーチャント・アカウントのデポジット・フラグを参照しています。
RC_CASSETTE_COUNTRYCODE	1092	応答は国別コード・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_STREETADDRESS	1093	応答は番地パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_CITY	1094	応答は市町村パラメーター (プロトコル・データの中で指定されている) を参照しています。

表 29. 汎用 2 次戻りコード (SRC) (続き)

2 次戻りコード	値	説明
RC_CASSETTE_STATEPROVINCE	1095	応答は都道府県パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_POSTALCODE	1096	応答は郵便番号 (zip) パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AVSCODE	1097	応答は AVS コード・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AUTHCODE_AND_DECLINEREASON	1098	競合するプロトコル・データがこの API コマンドで指定されました。
RC_CASSETTE_BATCHCLOSETIME	1099	応答はバッチ・クローズ時間パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_METHOD	1100	応答は決済メソッド・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_FIBATCHID	1101	応答は金融機関バッチ識別パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AUXILIARY1	1102	応答は 1 次補助テキスト・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_AUXILIARY2	1103	応答は 2 次補助テキスト・パラメーター (プロトコル・データの中で指定されている) を参照しています。
RC_CASSETTE_DECLINEREASON	1104	応答は指定した権限理由を参照しています。
RC_CASSETTE_BUYERNAME	1105	応答はバイヤーの名前を参照しています。
RC_CASSETTE_STREETADDRESS2	1106	応答は番地の 2 行目を参照しています。
RC_CASSETTE_PHONENUMBER	1107	応答は電話番号を参照しています。
RC_CASSETTE_EMAILADDRESS	1108	応答は電子メールアドレスを参照しています。
RC_CASSETTE_CHECKROUTINGNUMBER	1109	応答は小切手のルーティング番号を参照しています。
RC_CASSETTE_CHECKINGACCOUNTNUMBER	1110	応答は当座預金番号を参照しています。

## 付録 B. ISO 通貨コード

ISO 4217 通貨コードのリストを以下に示します。これらの値を CURRENCY パラメーターとともに使用してください。

国/地域	英字コード	数値コード	指数変換	通貨
アフガニスタン	AFA	004	-2	アフガニスタン・アフガニ
アルバニア	ALL	008	-2	アルバニア・レク
アルジェリア	DZD	012	-2	アルジェリア・ディナール
米領サモア	USD	840	-2	米国ドル
アンドラ公国	ESP、FRF、ADP	724、250、020	0、-2、0	スペイン・ペセタ、フランス・フランク、アンドラ・ペセタ
アンゴラ	AOA	973	-2	カンザ
アンギラ島	XCD	951	-2	東カリブ・ドル
アンティグア・バーブーダ	XCD	951	-2	東カリブ・ドル
アルゼンチン	ARS	032	-2	アルゼンチン・ペソ
アルメニア	AMD	051	-2	アルメニア・ドラム
アルバ島	AWG	533	-2	アルバ島ギルダー
オーストラリア	AUD	036	-2	オーストラリア・ドル
オーストリア	ATS	040	-2	オーストリア・シリング
アゼルバイジャン	AZM	031	-2	アゼルバイジャン・マナート
バハマ諸島	BSD	044	-2	バハマ・ドル
バーレーン	BHD	048	-3	バーレーン・ディナール
バングラデシュ	BDT	050	-2	バングラデシュ・タカ
バルバドス	BBD	052	-2	バルバドス・ドル

国/地域	英字コード	数値コード	指数変換	通貨
ベラルーシ	BYB、RYR	112、974	0	ベラルーシ・ルーブル
ベルギー	BEF	056	0	ベルギー・フラン
ベリーズ	BZD	084	-2	ベリーズ・ドル
ベナン	XOF	952	0	CFA フラン (BCEAO)
バーミューダ	BMD	060	-2	バーミューダ・ドル
ブータン	INR、BTN	356、064	-2、-2	インド・ルピー、ヌルタム
ボリビア	BOB、BOV	068、984	-2、-2	ボリビア・ノ、Mvdol
ボスニア & ヘルツェゴビナ	BAM	977	-2	兌換可能マルク
ボツワナ	BWP	072	-2	プーラ
ブーベ島	NOK	578	-2	ノルウェー・クローネ
ブラジル	BRL	986	-2	ブラジル・レアル
英領インド洋植民地	USD	840	-2	米国ドル
ブルネイ (ダルサラーム国)	BND	096	-2	ブルネイ・ドル
ブルガリア	BGL、BGN	100、975	-2、-2	レフ、ブルガリア・レフ
ブルキナファソ	XOF	952	0	CFA フラン (BCEAO)
ブルンジ	BIF	108	0	ブルンジ・フラン
カンボジア	KHR	116	-2	カンボジア・リエル
カメルーン	XAF	950	0	CFA フラン (BEAC)
カナダ	CAD	124	-2	カナダ・ドル
カーボバルデ	CVE	132	-2	カーボバルデ・エスクード
ケイマン諸島	KYD	136	-2	ケイマン諸島ドル
中央アフリカ共和国	XAF	950	0	CFA フラン (BEAC)
チャド	XAF	950	0	CFA フラン (BEAC)

国/地域	英字コード	数値コード	指数変換	通貨
チリ	CLP、CLF	152、990	0、0	チリ・ペソ (Unidades de fomento)
中国	CNY	156	-2	人民元
中国 (香港特別行政区)	HKD	344	-2	香港ドル
中国 (マカオ特別行政区)	MOP	446	-2	パタカ
クリスマス島	AUD	036	-2	オーストラリア・ドル
ココス (キーリング) 諸島	AUD	036	-2	オーストラリア・ドル
コロンビア	COP	170	-2	コロンビア・ペソ
コモロ	KMF	174	0	コモロ・フラン
コンゴ	XAF	950	0	CFA フラン (BEAC)
コンゴ民主共和国	CDF	976	-2	コンゴ・フラン
クック諸島	NZD	554	-2	ニュージーランド・ドル
コスタリカ	CRC	188	-2	コスタリカ・コロン
コートジボアール	XOF	952	0	CFA フラン (BCEAO)
クロアチア	HRK	191	-2	クロアチア・クン
キューバ	CUP	192	-2	キューバ・ペソ
キプロス	CYP	196	-2	キプロス・ポンド
チェコ共和国	CZK	203	-2	チェコ・コルナ
デンマーク	DKK	208	-2	デンマーク・クローネ
ジブチ	DJF	262	0	ジブチ・フラン
ドミニカ国	XCD	951	-2	東カリブ・ドル
ドミニカ共和国	DOP	214	-2	ドミニカ・ペソ
東チモール	TPE、IDE	626、360	0、-2	チモール・エスクード、ルピア

国/地域	英字コード	数値コード	指数変換	通貨
エクアドル	ECS、 ECV	218、 983	-2、 -2	スークレ、 Unidad de Valor Constante (UVC)
エジプト	EGP	818	-2	エジプト・ ポンド
エルサルバドル	SVC	222	-2	エルサルバドル・ コロン
赤道ギニア	XAF	950	0	CFA フラン (BEAC)
エリトリア	ERN	232	-2	ナクファ
エストニア	EEK	233	-2	クルーン
エチオピア	ETB	230	-2	エチオピア・ ビル
フェロー諸島	DKK	208	-2	デンマーク・ クローネ
欧州連合 (ECU)	XEU	954	-2	ユーロ
欧州連合 (ユーロ)	EUR	978	-2	欧州通貨単位
フォークランド諸島	FKP	238	-2	フォークランド 諸島ポンド
フィジー	FJD	242	-2	フィジー・ドル
フィンランド	FIM	246	-2	フィンランド・ マルッカ
フランス	FRF	250	-2	フランス・ フラン
仏領ギアナ	FRF	250	-2	フランス・ フラン
仏領ポリネシア	XPF	953	0	CFP フラン
仏領極南諸島	XPF	953	0	CFP フラン
ガボン	XAF	950	0	CFA フラン (BEAC)
ガンビア	GMD	270	-2	ダラーシ
グルジア	GEL	981	-2	ラリ
ドイツ	DEM	276	-2	ドイツ・マルク
ガーナ	GHC	288	-2	ガーナ・ セディー
ジブラルタル	GIP	292	-2	ジブラルタル・ ポンド
ギリシャ	GRD	300	0	ドラクマ

国/地域	英字コード	数値コード	指数変換	通貨
グリーンランド	DKK	208	-2	デンマーク・クローネ
グラナダ	XCD	951	-2	東カリブ・ドル
グアドループ	FRF	250	-2	フランス・フラン
グアム島	USD	840	-2	米国ドル
グアテマラ	GTQ	320	-2	グアテマラ・ケツァール
ギニア	GNF	324	0	ギニア・フラン
ギニアビサオ	GWP、XOF	624、952	-2、0	ギニアビサオ・ペソ、CFA フラン (BCEAO)
ガイアナ	GYD	328	-2	ガイアナ・ドル
ハイチ	HTG、USD	332、840	-2、-2	ハイチ・グールド、米国ドル
ヘアド島・マクドナルド諸島	AUD	036	-2	オーストラリア・ドル
教皇庁 (バチカン市国)	ITL	380	0	イタリア・リラ
ホンジュラス	HNL	340	-2	ホンジュラス・レンピラ
ハンガリー	HUF	348	-2	フォリント
アイスランド	ISK	352	-2	アイスランド・クローナ
インド	INR	356	-2	インド・ルピー
インドネシア	IDR	360	-2	インドネシア・ルピア
国際通貨基金	XDR	960	N.A.	SDR
イラン	IRR	364	-2	イラン・リアル
イラク	IQD	368	-3	イラク・ディナール
アイルランド	IEP	372	-2	アイルランド・ポンド
イスラエル	ILS	376	-2	新イスラエル・シェケル
イタリア	ITL	380	0	イタリア・リラ

国/地域	英字コード	数値コード	指数変換	通貨
ジャマイカ	JMD	388	-2	ジャマイカ・ドル
日本	JPY	392	0	円
ヨルダン	JOD	400	-3	ヨルダン・ディナール
カザフスタン	KZT	398	-2	カザフスタン・テンゲ
ケニア	KES	404	-2	ケニア・シリング
キリバス	AUD	036	-2	オーストラリア・ドル
北朝鮮	KPW	408	-2	北朝鮮ウォン
韓国	KRW	410	0	韓国ウォン
クウェート	KWD	414	-3	クウェート・ディナール
キルギスタン	KGS	417	-2	キルギスタン・ソン
ラオス	LAK	418	-2	ラオス・キープ
ラトビア	LVL	428	-2	ラトビア・ラット
レバノン	LBP	422	-2	レバノン・ポンド
レソト	ZAR、LSL	710、426	-2、-2	ランド、ロチ
リベリア共和国	LRD	430	-2	リベリア・ドル
リビア・アラブ国	LYD	434	-3	リビア・ディナール
リヒテンシュタイン	CHF	756	-2	スイス・フラン
リトアニア	LTL	440	-2	リトアニア・リータス
ルクセンブルグ	LUF	442	0	ルクセンブルグ・フラン
マケドニア (旧ユーゴスラビア共和国)	MKD	807	-2	マケドニア・デナリ
マダガスカル	MGF	450	0	マダガスカル・フラン
マラウイ	MWK	454	-2	クワチャ
マレーシア	MYR	458	-2	マレーシア・リンギット
モルジブ	MVR	462	-2	モルジブ・ルフィア



国/地域	英字コード	数値コード	指数変換	通貨
マリ	XOF	952	0	CFA フラン (BCEAO)
マルタ	MTL	470	-2	マルタ・リラ
マーシャル諸島	USD	840	-2	米国ドル
マルチニーク島	FRF	250	-2	フランス・フラン
モーリタニア	MRO	478	-2	モーリタニア・オーギア
モーリシャス	MUR	480	-2	モーリシャス・ルピー
メキシコ	MXN、MXV	484、979	-2、-2	メキシコ・ペソ、Mexican Unidad de Inversion (UDI)
ミクロネシア	USD	840	-2	米国ドル
モルドバ共和国	MDL	498	-2	モルドバ・レウ
モナコ	FRF	250	-2	フランス・フラン
モンゴル	MNT	496	-2	モンゴル・トゥグリク
モントセラト島	XCD	951	-2	東カリブ・ドル
モロッコ	MAD	504	-2	モロッコ・ディルハム
モザンビーク	MZM	508	-2	モザンビーク・メティカル
ミャンマー	MMK	104	-2	ミャンマー・チャット
ナミビア	ZAR、NAD	710、516	-2、-2	ランド、 ナミビア・ドル
ナウル	AUD	036	-2	オーストラリア・ドル
ネパール	NPR	524	-2	ネパール・ルピー
オランダ領アンティル諸島	ANG	532	-2	オランダ領アンティル諸島 ギルダー
オランダ	NLG	528	-2	オランダ・ギルダー
ニューカレドニア	XPF	953	0	CFP フラン
ニュージーランド	NZD	554	-2	ニュージーランド・ドル

国/地域	英字コード	数値コード	指数変換	通貨
ニカラグア	NIO	558	-2	ニカラグア・コルドバ・オーロウ
ニジェール	XOF	952	0	CFA フラン (BCEAO)
ナイジェリア	NGN	566	-2	ナイジェリア・ナイラ
ニウエ島	NZD	554	-2	ニュージーランド・ドル
ノーフォーク島	AUD	036	-2	オーストラリア・ドル
北マリアナ諸島	USD	840	-2	米国ドル
ノルウェー	NOK	578	-2	ノルウェー・クローネ
オマーン	OMR	512	-3	オマーン・リアル
パキスタン	PKR	586	-2	パキスタン・ルピー
パラオ	USD	840	-2	米国ドル
パナマ	PAB、USD	590、840	-2、-2	バルボア、米国ドル
パプアニューギニア	PGK	598	-2	パプアニューギニア・キナ
パラグアイ	PYG	600	0	パラグアイ・グアラニ
ペルー	PEN	604	-2	ペルー・ヌエボ・ソル
フィリピン	PHP	608	-2	フィリピン・ペソ
ピトケアン島	NZD	554	-2	ニュージーランド・ドル
ポーランド	PLN	985	-2	ポーランド・ズローティ
ポルトガル	PTE	620	0	ポルトガル・エスクード
プエルトリコ	USD	840	-2	米国ドル
カタール	QAR	634	-2	カタール・リアル
レユニオン島	FRF	250	-2	フランス・フラン
ルーマニア	ROL	642	-2	ルーマニア・レウ
ロシア	RUR、RUB	810、643	-2、-2	ロシア・ルーブル、ロシア・ルピー

国/地域	英字コード	数値コード	指数変換	通貨
ルワンダ	RWF	646	0	ルワンダ・フラン
セントクリストファー・ネイビス	XCD	951	-2	東カリブ・ドル
セントルシア	FRF	951	-2	東カリブ・ドル
サンピエール・エ・ミクロン島	XCD	250	-2	フランス・フラン
セントビンセント・グレナディーン諸島	XCD	951	-2	東カリブ・ドル
セントヘレナ島	SHP	654	-2	セントヘレナ島ポンド
西サモア	WST	882	-2	タラ
サンマリノ	ITL	380	0	イタリア・リラ
サントメ・プリンシペ	STD	678	-2	サントメ・プリンシペ・ドブラ
サウジアラビア	SAR	682	-2	サウジ・リヤール
セネガル	XOF	952	0	CFA フラン (BCEAO)
セイシェル諸島	SCR	690	-2	セイシェル・ルピー
シエラレオネ	SLL	694	-2	シエラレオネ・レオーネ
シンガポール	SGD	702	-2	シンガポール・ドル
スロバキア	SKK	703	-2	スロバキア・コルナ
スロベニア	SIT	705	-2	スロベニア・トウラ
ソロモン諸島	SBD	090	-2	ソロモン諸島ドル
ソマリア	SOS	706	-2	ソマリア・シリング
南アフリカ	ZAR	710	-2	南アフリカ・ランド
スペイン	ESP	724	0	スペイン・ペセタ
スリランカ	LKR	144	-2	スリランカ・ルピー
スーダン	SDP	736	-2	スーダン・ディナール

国/地域	英字コード	数値コード	指数変換	通貨
スリナム	SRG	740	-2	スリナム・ギルダ
スバルバル諸島・ヤンマイエン島	NOK	578	-2	ノルウェー・クローネ
スワジランド	SZL	748	-2	スワジランド・リランゲニ
スウェーデン	SEK	752	-2	スウェーデン・クローナ
スイス	CHF	756	-2	スイス・フラン
シリア・アラブ共和国	SYP	760	-2	シリア・ポンド
台湾	TWD	901	-2	新台湾ドル
タジキスタン	TJR	762	0	タジク・ルーブル
タンザニア連合共和国	TZS	834	-2	タンザニア・シリング
タイ	THB	764	-2	タイ・バーツ
トーゴ	XOF	952	0	CFA フラン (BCEAO)
トケラウ諸島	NZD	554	-2	ニュージーランド・ドル
トンガ	TOP	776	-2	トンガ・パアンガ
トリニダード・トバコ	TTD	780	-2	トリニダード・トバコ・ドル
チュニジア	TND	788	-3	チュニジア・ディナール
トルコ	TRL	792	0	トルコ・リラ
トルクメニスタン	TMM	795	-2	マナート
タークス諸島・カイコス諸島	USD	840	-2	米国ドル
ツバル	AUD	036	-2	AUD
ウガンダ	UGX	800	2	ウガンダ・シリング
ウクライナ	UAH	980	-2	グリブナ
アラブ首長国連邦	AED	784	-2	UAE ディルハム
英国	GBP	826	-2	英国ポンド
アメリカ合衆国	USD、USS、USN	840、998、997	-2、-2、-2	米国ドル、(Same day) (Next day)

国/地域	英字コード	数値コード	指数変換	通貨
米領太平洋諸島	USD	840	-2	米国ドル
ウルグアイ	UYU	858	-2	ペソ・ウルグアイ
ウズベキスタン	UZS	860	-2	ウズベキスタン・サム
バヌアツ	VUV	548	0	バヌアツ・バーツー
ベネズエラ	VEB	862	-2	ベネズエラ・ボリバル
ベトナム	VND	704	-2	ベトナム・ドン
英領バージン諸島	USD	840	-2	米国ドル
英領バージン諸島	USD	840	-2	米国ドル
ウォリスフツナ	XPF	953	0	CFP フラン
西サハラ	MAD	504	-2	モロッコ・ディルハム
イエメン	YER	886	-2	イエメン・リアル
ユーゴスラビア	YUN	891	-2	ユーゴスラビア・ディナール
ザイール	ZRN	180	-2	不明
ザンビア	ZMK	894	-2	ザンビア・クワチャ
ジンバブエ	ZWD	716	-2	ジンバブエ・ドル



---

## 付録 C. Request For Comments (RFC) の入手方法

Request for comment (RFC) は、新しいプロトコルを提供し、一連のインターネット・プロトコルに関する規格を確立する文書です。すべての RFC のハードコピーは、ネットワーク情報センター (NIC) から個別に、あるいは予約購入により、入手することができます。これらの文書は、下記の宛先から入手することができます。

Government Systems, Inc.  
Attn: Network Information Center  
14200 Park Meadow Drive  
Suite 200  
Chantilly, VA 22021

次の URL から RFC にアクセスすることができます。

<http://www.cis.ohio-state.edu/hypertext/information/rfc.html> (英語)





---

## 付録 D. 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、本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品、プログラムまたはサービスの操作性の評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む。) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権の許諾については、下記の宛先に書面にてご照会ください。

〒106-0032  
東京都港区六本木 3-2-31  
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。

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 Corporation  
Department TL3B/Building 503  
PO Box 12195

3039 Cornwallis Road  
Research Triangle Park, NC 27709-2195

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せんが、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 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。

本書には、日常のビジネス・オペレーション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。

---

## 商標

以下は、IBM Corporation の商標です。

- AIX
- AS/400
- DB2
- IBM
- IBM Payment Server
- iSeries
- pSeries
- OS/400
- WebSphere
- zSeries

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。

Microsoft、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、SET ロゴ、SET Secure Electronic Transaction および Secure Electronic Transaction は、SET Secure Electronic Transaction LLC の商標です。

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。

MMX または Pentium は、Intel Corporation の米国およびその他の国における商標です。

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または登録商標です。

## 用語集

この用語集は、WebSphere Commerce Payments の資料で使われている技術用語を定義したものです。最新の IBM Dictionary of Computing は、Web サイト

<http://www.ibm.com/ibm/terminology/goc/gocmain.htm> でご覧いただけます。

### [ア行]

**アカウント (account).** アカウントとは、マーチャントと、そのマーチャントのトランザクションを処理する金融機関との関係のこと。決済カセットごとに、複数のアカウントを設けることができる。

**アドレス検証サービス (AVS) (Address Verification Service (AVS)).** IBM e-commerce において、カードの所有者を認証するためにマーチャントが使用するクレジット・カードやデビット・カードの体系。マーチャントはカード所有者の住所を要求し、AVS を使用してカード所有者本人かどうかを確認する。

**アプレット (applet).** Web サーバーから取り出し、Web ブラウザーで実行できる、Java プログラム言語で書かれたアプリケーション・プログラム。アプレットへの参照は、グラフィックス・ファイルと同じように、Web ページのマークアップで表示される。また、ブラウザは、グラフィックス・ファイルを取り出すのと同じ方法で、アプレットを取り出す。セキュリティのため、アプレットのアクセス権限は 2 つの方法に制限される。まず、アプレットは、そのアプレットを実行しているクライアントのファイルシステムにアクセスすることができない。また、アプレットがネットワークを介して行える通信は、アプレットのダウンロード元であるサーバーへの通信だけに制限される。「サーブレット」と対比。

**暗号化 (encryption).** (1) コンピューター・セキュリティにおいて、元のデータを入手できないようにするか、または復号処理を使用しないと入手できないようにするかのいずれかの方法で、データを解明困難な形式に変換する処理。(2) 無許可アクセスを防止するため、簡単には解読できない形式にデータを変換すること。特にデータ送信において利用される。

**イベント (event).** (1) ある部分に発生した変化を表すもの。その変化により、それに関連する他の部分が、変化のあった部分に関する何らかの情報についての通知を受け取ることになる。たとえば、プッシュボタンがクリ

ックされると、そのプッシュボタンはそのことに関する通知を出すことによりイベントを生成し、それにより別の部分がウィンドウを表示することになる。(2) システム・リソース、ネットワーク・リソース、またはネットワーク・アプリケーションで生じる、重要な状態の変化。イベントは、問題の発生、問題の解決、またはタスクの正常な完了などについて生成される。

**イベント・リスナー (event listener).** IBM e-commerce において、関心のあるイベントの通知を待ち、そのイベントに反応するコンピューター・プログラム。

**インターネット (internet).** (1) TCP/IP において、複数のネットワークを相互に接続して、1 つの大きなネットワークとして機能するようにしたもの。(2) インターネットのプロトコルを使用する相互接続ネットワークの集合。汎用アクセスが可能なインターネットを指す場合には、大文字の「I」を使い Internet と表記する。インターネットであっても、アクセスが制限されている(特定の企業または組織だけがアクセスできる、など)場合には、公共インターネットに接続されているかどうかにかかわらず、イントラネットと呼ぶことが多い。

**ウェルノウン・ポート (well-known port).** インターネットのプロトコルにおいて、伝送制御プロトコル (TCP) やユーザー・データグラム・プロトコル (UDP) などのトランスポート・レベル・プロトコルによって使用される特定の機能をアドレッシングする、一連の割り当て済みプロトコル・ポート番号のうちの 1 つ。たとえば、ファイル転送プロトコル (FTP) やシンプル・メール転送プロトコル (SMTP) は、ウェルノウン・ポート番号を使用する。

**ウォレット (wallet).** ユーザーが認証済みマーチャントに対する承認済み決済を公衆ネットワーク経由で実行したり、決済カードのアカウントや購入を管理したりするためのソフトウェア。

**オーダー (order).** WebSphere Commerce Payments においてオーダーは、マーチャント (被支払者) が代金を収集できるようにするために、消費者 (支払者) が必要とするすべての指示および情報を表す。

**オーダー金額 (order amount).** オーダーの金額。

**オーダー調達 (order fulfillment).** IBM e-commerce において、決済済みのオーダーに対して発送または配達する責任があるマーチャント・システム。オーダー調達シ

ステムは、WebSphere Commerce Payments に問い合わせ、どの商品を出荷するかを決定するものと見なされる。

**オーダーの検索 (order search).** 定義済みの一連の特性に基づいて単一のオーダーまたはオーダー・グループを検索すること。

**オンライン・カタログ (online catalog).** オンライン・ストアでの表示や購入のために利用できるカタログ・グループやカタログ・エントリーの総称。

## [カ行]

**カード所有者 (cardholder).** e-commerce において、有効な決済カード・アカウントを所有し、e-commerce をサポートするソフトウェアを使用する人。

**カード・プロセッサ (card processor).** マーチャントがトランザクション要求を送信する先の決済機関のエージェント。カード・プロセッサには、マーチャントがトランザクションを処理するのに利用できる管理用および組織的なインフラストラクチャーの大半が用意されている。

**鍵 (key).** コンピューター・セキュリティーにおいて、データを暗号化または復号するために暗号化アルゴリズムで使用されるシンボルのシーケンス。「秘密鍵」および「公開鍵」を参照。

**鍵リング (key ring).** コンピューター・セキュリティーにおいて、公開鍵、秘密鍵、トラステッド・ルート、および証明書を含むファイル。

**カセット (cassette).** (1) e-commerce において、e-commerce に関係する他のソフトウェア・コンポーネントに簡単にインストールし、これらのコンポーネントの機能を拡張することができる Java のクラスおよびインターフェースの集合からなるソフトウェア・コンポーネント。(2) IBM e-commerce における、WebSphere Commerce Payments の概念の 1 つ。WebSphere Commerce Payments では、さまざまな決済形式をサポートできる枠組みが用意されている。WebSphere Commerce Payments カセットは、IBM またはサード・パーティーのベンダーによって書かれ、WebSphere Commerce Payments の枠組みに含まれるさまざまな決済プロトコル (VisaNet や BankServACH など) をサポートしている。したがって、WebSphere Commerce Payments はさらに別のプロトコルをサポートできる拡張可能な製品である。

**完全修飾ドメイン・ネーム (FQDN) (fully qualified domain name (FQDN)).** インターネットのプロトコルにおいて、ドメイン・ネームのすべてのサブネームを含

むホスト・システムの名前。完全修飾ドメイン・ネームの例は、mycomputer.city.company.com。「ホスト名」を参照。

**強制クローズ (force).** IBM e-commerce における、WebSphere Commerce Payments の 1 つの用語。バッチの清算が試みられる。調整ステップが失敗しても、バッチが WebSphere Commerce Payments でクローズされることはない (ただし、収支がとれていないかまたは金融機関でクローズされていない可能性はある)。

**均衡 (balanced).** IBM e-commerce における、WebSphere Commerce Payments の Batch オブジェクトの属性の 1 つ。バッチは正常に均衡を保っている。総合計が一致している。

**金融機関 (financial institution).** (1) 信用または管理の延長、貸付、両替、または金銭の発行など、広範囲にわたる顧客主導の資金の取引や伝送を円滑に行えるようにする信用機関。たとえば、銀行または銀行が指定する機関。(2) IBM e-commerce では、銀行、住宅金融組合、および信用組合は金融機関の例である。金融サービスを提供する機関。

**金融ネットワーク (financial network).** IBM e-commerce において、決済カードのトランザクション処理を以前から実行しているカードの処理機関、決済機関、カードの発行者、およびその他の機関の集合体。

**クラーク (Clerk).** IBM e-commerce における、WebSphere Commerce Payments の概念の 1 つ。WebSphere Commerce Payments には、4 つの異なるアクセス権がある。クラークは、マーチャントごとに定義される最低のアクセス・レベルである。

**クライアント (client).** (1) サーバーから提供される共有サービスを受け取る機能単位。たとえば、Web サーバーに対して HTML 文書を要求するパーソナル・コンピュータは、そのサーバーのクライアントである。(2) 一般にサーバーと呼ばれる他のコンピュータ・システムまたはプロセスのサービスを要求する、コンピュータ・システムまたはプロセス。複数のクライアントが共通のサーバーへのアクセスを共用することもできる。

**クレジット (credit).** 有効な資金化メッセージが出た後に、マーチャントがカード所有者に (決済機関や発行機関を通して) 返金する必要がある場合に送信されるトランザクション。たとえば、商品が返品されたり不良品であった場合などが考えられる。

**クレジット回数 (number of credits).** クレジットは、有効な資金化メッセージが出た後に、マーチャントがカード所有者に (決済機関や発行機関を通して) 返金

する必要がある場合に送信されるトランザクションである。たとえば、商品が返品されたり不良品であった場合などが考えられる。クレジットは、オーダーに関連付けられたすべての決済の合計数まで作成できる。オーダーごとに、0 件以上のクレジットが存在する。

**クローズ (closed).** オーダーに関連した決済が「デポジット」状態から「クローズ」状態に変わるとき (つまり、決済に関連付けられたバッチがクローズする場合) に、オーダーは「クローズ」状態になる。オーダーが「クローズ」状態になると、金銭的取引は完了する。代金はデポジットされ、オーダーを変更できなくなる。この状態では、オーダーへのコマンドは許可されていない。

**ゲートウェイ (gateway).** ローカル・データ・ネットワークを別のネットワークにつなぐための機能単位。

**決済 (payment).** 決済とは、マーチャントが金融機関に対して、オーダーの全部または一部を承認するよう要求すること。多くの場合、オーダーが発生して集金が与信されると、その全額が一度の決済で集金される。ただし決済のシステムによっては、1 つのオーダー (つまり一連の決済指示) で与信された金額を、ビジネス・モデルに応じて複数回の決済で集金することもある。

**決済カード (payment card).** (1) クレジット・カード、デビット・カード、およびキャッシュ・カードの総称で、金融機関によって発行され、所有者と金融機関との関係を反映するもの。(2) e-commerce において、クレジット・カード、デビット・カード、またはチャージ・カードであって、(a) 金融機関によって発行され、所有者と金融機関との関係を示し、(b) 認証された認証局がそのカードに関する証明書を発行できるもの。

**決済回数 (number of payments).** 決済とは、マーチャントが金融機関に対して、オーダーの全部または一部を承認するよう要求すること。多くの場合、オーダーが発生して集金が与信されると、その全額が一度の決済で集金される。ただし決済のシステムによっては、1 つのオーダー (つまり一連の決済指示) で与信された金額を、ビジネス・モデルに応じて複数回の決済で集金することもある。オーダーごとに 0 件以上の決済が存在する。

**決済額 (payment amount).** マーチャントがこのオーダーでデポジットした決済金額の合計。

**決済カセット (payment cassette).** 電子決済プロトコルを実装するカセット。

**決済機関 (acquirer).** e-commerce において、取引に関連する金融データをマーチャントから受け取り、その取引を許可する金融機関 (または金融機関のエージェント)。

**決済ゲートウェイ (payment gateway).** (1) 決済ゲートウェイ・コンポーネントは、決済機関または指定したサード・パーティーが実行する製品であり、マーチャントの与信や決済メッセージ (カード所有者による決済に関する指示を含む) を処理し、専用の金融ネットワークとのインターフェースをとる。(2) e-commerce において、マーチャントと決済機関の間のトランザクションを処理するエンティティ。

**決済の無効化 (void payment).** IBM e-commerce において、決済操作を破棄またはキャンセルすることを意味する用語。

**公開鍵 (public key).** (1) セキュア通信において、対応する秘密鍵によって暗号化されたメッセージを暗号化解除するために使用されるアルゴリズム・パターン。公開鍵は、対応する秘密鍵によってのみ暗号化解除されるよう、メッセージを暗号化するためにも使用される。ユーザーは、暗号化されたメッセージをやり取りする相手に対して公開鍵を送る。(2) コンピューター・セキュリティにおいて、すべてのユーザーが使用できる鍵。

**コマース・サービス・プロバイダー (CSP) (commerce service provider (CSP)).** マーチャントのショッピング・サイトのホストとして機能し、マーチャントのために決済を処理するインターネット・サービス・プロバイダー。

**コンストラクター (constructor).** プログラム言語で、クラスと同じ名前を持ち、そのクラスのオブジェクトの作成および初期化に使用されるメソッド。

## [サ行]

**サーバー (server).** (1) ネットワークを介して 1 つまたは複数のクライアントにサービスを提供する機能単位。(2) ネットワークを通じて、他のコンピューターに共有サービスを提供するコンピューター。ファイル・サーバー、プリント・サーバー、メール・サーバーなど。

**サーブレット (servlet).** Web サーバー上で実行される、Java プログラム言語で書かれたアプリケーション・プログラム。サーブレットは、グラフィックス・ファイルの参照の場合と同じように、Web ページ用のマークアップで表示される。Web サーバーは、サーブレットを実行し、実行の結果 (何らかの結果が得られた場合) を Web ブラウザーに送信する。「アプレット」と対比。

**資金化 (capture).** 決済機関が顧客の金融機関からの支払を受け取ったり、支払額を送金する処理。資金化は、預金が利用可能な状態であり振替が実行されるということの保証となる。

**収支 (balance).** IBM e-commerce における、WebSphere Commerce Payments の Batch オブジェクトの属性の 1 つ。バッチがクローズされたときに、マーチャントと金融機関がバッチの内容に同意したかどうかを示す。

**収支状況 (balance status).** IBM e-commerce における、WebSphere Commerce Payments の Batch オブジェクトの属性の 1 つ。バッチの収支状況は、均衡または不均衡のいずれか。

**承認 (approve).** IBM e-commerce における、WebSphere Commerce Payments の 1 つの用語。マーチャントは、これを発行して Payment オブジェクトを作成する。クレジット・カードのプロトコルを実装したカセットの場合、この用語は与信に相当する（「与信」を参照）。カセットによっては、別の承認処理を実装しているものもある。

**証明書 (certificate).** (1) コンピューター・セキュリティにおいて、公開鍵を証明書の所有者の ID と結合するデジタル文書。これによって、証明書の所有者を認証することができる。証明書は認証局 (CA) によって発行される。(2) SETCo. では、公開鍵のユーザーを識別するために、信用機関（通常はカード所有者の金融機関）によってデジタル署名されている証明書。SET は、以下の証明書タイプを定義する。

- 署名
- 鍵の暗号化
- 証明書署名
- CRL 署名

**除去 (purge).** IBM e-commerce における、WebSphere Commerce Payments の 1 つの用語。関連付けられているすべての決済およびクレジットを、Batch オブジェクトから除去すること。その際、Batch オブジェクトが作成されたばかりであるかのように扱う。

**スーパーバイザー (Supervisor).** マーチャントのすべての決済処理機能を実行できる。

**スレッド (thread).** プロセスを制御するコンピューター命令のストリーム。マルチスレッド化されたプロセスは、命令のうちの 1 ストリーム (1 スレッド) によって開始され、その後で、タスクを実行するための他の命令ストリームを作成することができる。

**スレッド・プール (thread pool).** コンピューター・プログラムによって使用されているか、あるいは使用可能なスレッド。

**清算 (settle).** IBM e-commerce における、WebSphere Commerce Payments の 1 つの用語。Batch オブジェクトのクローズと資金の振り込みを試みること。清算手順の一部として、マーチャントと金融機関が資金の転送に同意していることを確認するため、いくつかの調整または清算ステップがある場合がある（カセットや金融機関の方針によって決まる）。調整ステップが失敗すると、バッチはオープン状態のままになる可能性がある。

**正当性 (integrity).** コンピューター・セキュリティにおいて、相手に到着する情報が送り出された情報と同じであるという保証。

**選択して販売 (sale selected).** 承認するオーダーを選択して承認し、関連する決済をデポジット状態に直接移行する。販売機能は、決済に対する承認およびデポジットを自動的に実行する。

**専用回線 (leased line).** 顧客の端末をネットワーク上の専用ポートに接続するため、通信事業者からリースにより顧客に提供されている電話回線。

**ソケット (socket).** プロセス間またはアプリケーション・プログラム間での通信のためにネットワークのトランスポート・サービスによって提供されるエンドポイント。

## [夕行]

**定期循環払い (recurring payments).** カードの所有者が開始する決済トランザクションの種類で、複数の与信の処理をマーチャントに認める。以下のような、2 種類の定期循環払いがある。

1. 定額の複数回払い
2. 定期的な請求

## [ナ行]

**認証 (authentication).** (1) コンピューター・セキュリティにおいて、メッセージが変更または破壊されていないことの検証。(2) コンピューター・セキュリティにおいて、ユーザーを特定すること、またはあるオブジェクトにアクセスする権限がユーザーにあるかどうかを判断すること。(3) 個人を特定する処理。普通、ユーザー ID とパスワードを使用する。セキュリティ・システムにおいて、認証と与信は異なる。認証は、単にその個人が、その主張通りの人物であることを確認するだけであり、その個人のアクセス権を定義するわけではない。

## [八行]

**発行者 (issuer).** (1) 決済カード・ブランドのカード所有者に固有の主要アカウント番号 (PAN) を発行する金融機関またはその代理店。 (2) e-commerce において、個人に対して決済カードを発行する金融機関。カード発行機関は、それ自体の認証局 (CA) として機能したり、第三者とサービスに関する契約を結んだりすることもできる。

**バッチ (batch).** (1) 決済トランザクションの集まり。資金化、クレジット、資金化取り消し、およびクレジット取り消しを 1 つのグループとして実行する場合など。バッチは、決済機関の金融システムに対して 1 つの単位として送信される。バッチ処理の使用に関するビジネス・ガイドラインは、信用調査機関によって定められる。マーチャントも、それらのガイドラインに沿ったポリシーを設定する。 (2) IBM e-commerce において、基本的な WebSphere Commerce Payments オブジェクトの 1 つがバッチである。バッチには、Payments オブジェクトや Credit オブジェクトが関連付けられる。資金の転送は、バッチをクローズするときに行われる。 (3) 共に処理または伝送される、レコードやデータ処理ジョブのグループ。

**バッチの清算 (settle batches).** バッチの清算は、バッチ (決済およびリファンド) の処理を決済処理プログラムに依頼するために使用する。1 つのバッチを清算するか、複数のバッチを清算するかを選択することができる。

**バッチ番号 (batch number).** バッチを識別する番号。支払額の入金時に WebSphere Commerce Payments がバッチに割り当てる番号。

**販売 (sale).** クレジット・カード業界では、トランザクションが与信され、2 段階のプロセスを使用せずに一度に資金化するようにマークが付けられると、販売が発生する。

**秘密鍵 (private key).** (1) セキュア通信において、対応する公開鍵によってのみ暗号化解除できるよう、メッセージを暗号化するために使用されるアルゴリズム・パターン。秘密鍵は、対応する公開鍵によって暗号化されたメッセージを暗号化解除するためにも使用される。秘密鍵はユーザーのシステム上に保存され、パスワードによって保護される。 (2) コンピューター・セキュリティにおいて、所有者のみが知っている鍵。

**ファイアウォール (firewall).** あるネットワークと別のネットワークとの接続を保護および制御する機能単位。ファイアウォールは、(a) 望まれない、または許可されていない通信トラフィックが、保護されているネットワ

ーク内に進入しないように保護し、(b) 選択された通信トラフィックだけが保護されたネットワーク外にアクセスできることを許可する。

**不均衡 (out of balance).** バッチの収支を合わせようとして失敗したこと。総合計が一致していない。

**復号 (decryption).** コンピューター・セキュリティーにおいて、エンコードされたテキストまたは暗号テキストを非暗号テキストに変換する処理。

**プライマリー・アカウント番号 (PAN) (primary account number (PAN)).** カードの発行者や所有者を識別するために割り当てられている番号。このアカウント番号は、発行者の識別番号、個人のアカウント番号の ID、および ISO 7812-1985 によって定義されているチェック・ディジットから成っている。

**ブランド (brand).** IBM e-commerce では、すべての WebSphere Commerce Payments カセット (例: VisaNet のカセットや Paymentech のカセット) の Cassette オブジェクト。WebSphere Commerce Payments カセットの金銭的取引はそれぞれ、特定のブランド (たとえば、MasterCard または VISA) に関連付けられる。金融機関の各口座は、1 つまたは複数のブランドをサポートするよう構成できる。

**プロトコル (protocol).** ネットワークの管理、データの転送、およびネットワーク・コンポーネントの状態の同期化に使用される、要求と応答の意味、およびそれらの順序付け規則。

**分割払い (installment payments).** マーチャントとカード所有者との間で取り決める決済トランザクションの種類で、複数の与信の処理をマーチャントに認める。

**文書タイプ定義 (DTD) (document type definition (DTD)).** SGML または XML 文書の個々のクラスの構造を指定する規則。DTD は、エレメント、属性、および記法を持つ構造を定義し、特定のクラスの文書で各エレメント、属性、および記法をどのように使用するかについて、制約を設定する。DTD は、特定のマークアップ言語の構造を完全に記述するという点で、データベース・スキーマに類似している。

**ポート (port).** インターネットのプロトコルにおいて、伝送制御プロトコル (TCP) またはユーザー・データグラム・プロトコル (UDP) と、高水準プロトコルまたはアプリケーションの間の特定の論理結合子。「ウェルノウン・ポート」を参照。

**ポート番号 (port number).** インターネットのプロトコルにおいて、アプリケーション・エンティティーとトランスポート・サービスの間の論理結合子の ID。

**ホスト (host).** Web サイトを管理するためのソフトウェアとサービスを提供すること。

**ホスト名 (host name).** インターネットのプロトコルにおいて、コンピューターに付けられた名前。「ホスト名」は、完全修飾ドメイン・ネームを意味することもあり、また完全修飾ドメイン・ネームの最も特定のサブネームを意味することもある。たとえば、

mycomputer.city.company.com が完全修飾ドメイン・ネームである場合、次のどちらもホスト名と考えることができる。

- mycomputer.city.company.com
- mycomputer

## [マ行]

**マーチャント (merchant).** 商品、サービス、または他の情報 (あるいはそれらすべて) の売り手で、これらの品目に対する支払いを電子的に受け取る側のこと。マーチャントは、電子的な販売サービスや、販売する商品の電子的な納入も提供することがある。マーチャントは、ストアの販売状況を追跡するだけでなく、ストアの全体的な目標や管理を監視する。

**マーチャント・サーバー (merchant server).** (1) マーチャント・サーバー・コンポーネントは、決済カードのトランザクションと与信を処理するために、オンライン・マーチャントが実行するプロダクトである。これは、カード所有者のウォレット、Payment Gateway、および認証局のコンポーネントと通信する。(2) e-commerce において、カタログ式販売を行う Web サーバー。

**マーチャント・チャージバック (merchant chargeback).** IBM e-commerce において、不正行為が発生したために代金が支払われなかった場合の責任をマーチャントが取る場合、金融機関はマーチャント・チャージバックを発行して、マーチャント・アカウントに以前に返金した額の返還を要求できる。

**マーチャント・バンク (merchant bank).** 決済金融機関。マーチャント・バンクは、マーチャントに対してクレジット・カード決済のための手段を提供することにより、マーチャント・ビジネスを獲得する。そのような金融機関は、マーチャントに対してサービス提供料金を課す。

## [ヤ行]

**有効期限 (expiry).** (1) 証明書を取得したとき割り当てられた証明書の有効期限。証明書は、決済の種類ごとに固有。(2) カードの有効期限を指定すること。SET プ

ロトコルでは、有効期限の値が必須。この値はストリングとして指定され、決済の開始メッセージで使用される。たとえば、199911 は有効期限の値である。

**与信 (authorization).** (1) 何らかの処理を実行する権限を、適切に指名された人 (複数の場合あり) が、組織を代表して付与するプロセス。このプロセスにより、取引に伴うリスクを見極め、指定した取引でアカウント所有者の借入れがアカウントの信用限度額を超えないよう確認し、指定された金額のクレジットを留保する。(マーチャントが与信を得るときに、そのマーチャントが与信の処理に関連する規則を順守する限り、与信範囲の金額の支払いが保証される。) (2) コンピューター・セキュリティにおいて、コンピューター・システムと通信したりコンピューター・システムを使用したりできるように、ユーザーに付与される権利。(T) (3) アクセス権限。(4) オブジェクト、リソース、または機能に対して、全面的あるいは制限付きのアクセス権限をユーザーに付与するプロセス。

**与信 (authorize).** クレジット・カード業界においては、最初に取引に対して与信 がなされることによって、マーチャントは、カード所有者の資金で取引が行えることが保証される。カードの発行者 (つまり、そのカードを発行した銀行) が支払いを保証する。

**与信取り消し (authorization reversal).** 以前の与信をキャンセルする必要がある場合 (つまり、完全に取り消す場合) または縮小する場合 (つまり、部分的に取り消す場合) に送信されるトランザクション。完全取り消しは、取引を完了できない場合に使用する。たとえば、カード所有者がオーダーをキャンセルする場合や、または製造中止になるなど、商品を販売できないことにマーチャントが気付く場合など。部分取り消しは、オーダー全体の与信を得ているものの、一部の商品を発送できないため、結果的に分割して発送する場合に使用する。

## [ラ行]

**リファンド (refund).** オーダーに使われる特定通貨の最小単位で、クレジット金額を識別すること。

**レルム (realm).** WebSphere ファミリーの製品において、ユーザー、グループ、およびアクセス・コントロール・リストのデータベース。レルムに属するリソースにアクセスするには、ユーザーはそのレルムで定義されていなければなりません。

## C

**CGI プログラム (CGI program).** Web サーバーで実行されるプログラムの一種。共通ゲートウェイ・インターフェース (CGI) を使用して、そのサーバーで通常は



実行されないタスク (データベース・アクセスやフォームの処理など) を実行する。 OS/400® オペレーティング・システムでは、ILE C、ILE RPG、および ILE COBOL の言語で作成されたコンパイル済み CGI プログラムがサポートされている。

## D

**DTD.** 「文書タイプ定義」を参照。

## E

**EAR ファイル (EAR file).** WebSphere Application Server に配置できる J2EE アプリケーションを表すエンタープライズ・アーカイブ・ファイル。 EAR ファイルは標準的な Java アーカイブ・ファイルであり、.ear というファイル拡張子を持つ。

**e-commerce.** (1) 一部または全部の取引が電子通信によってなされる際に、カード所有者とマーチャントが決済対象の商品やサービスをやりとりすること。(2) インターネットなどの電子メディアを介して購入された商品またはサービスの代金を交換する、 e-business のサブセット。

## H

**HTML.** 「Hypertext Markup Language」を参照。

**HTTP.** 「Hypertext Transfer Protocol」を参照。

**Hypertext Markup Language (HTML).** SGML 規格に準じたマークアップ言語。主に、ハイパーテキスト・リンクを含む、テキストおよびグラフィックスの情報のオンライン表示をサポートすることを目的として設計されている。

**Hypertext Transfer Protocol (HTTP).** インターネットのプロトコルにおいて、 Web 上にあるハイパーテキスト文書を転送および表示するために使用されるプロトコル。

## I

**IP アドレス (IP address).** インターネットに接続された各デバイスまたはワークステーションのロケーションを指定する、固有の 32 ビット・アドレス。たとえば、9.67.97.103 は IP アドレスである。

## J

**J2EE アプリケーション (J2EE application).** 展開可能な J2EE 機能の単位。これは単一のモジュールまたはモジュールのグループで、.ear ファイルにパッケージされており、 J2EE アプリケーション・デプロイメント記述子を伴っている。

**Java.** リモート・オブジェクト間での対話をサポートする移植可能な解釈コードのためのオブジェクト指向プログラム言語。 Java は、Sun Microsystems, Incorporated によって開発され、使用が制限された。

**Java Database Connectivity (JDBC).** オープン・データベース・コネクティビティ (ODBC) と同じ特性を備え、しかも Java データベース・アプリケーションで使用するために特に設計された、アプリケーション・プログラミング・インターフェース (API)。また、データベースに JDBC ドライバーが組み込まれていない場合に備えて、 JDBC には JDBC から ODBC へのブリッジが含まれている。このブリッジは、 JDBC を ODBC に変換するためのメカニズムであり、 Java データベース・アプリケーションに JDBC API を提供し、これを ODBC に変換する。 JDBC は、Sun Microsystems, Inc. および同社の多くの協力企業ならびに取引先によって開発された。

**Java 仮想計算機 (JVM) (Java Virtual Machine (JVM)).** コンパイル済み Java コード (アプレットおよびアプリケーション) を実行する中央処理装置 (CPU) のソフトウェア実装。

## L

**LUHN 公式 (LUHN formula).** クレジット・カードの不正使用の基本的防止策として、多くのクレジットカード会社が採用している業界標準規格。

## P

**Payment Server.** e-commerce において、キャッシュ・レジスターに相当する電子的機能であって、購入された商品およびサービスに対する決済を編成して受け入れる。 Payment Server は、決済ゲートウェイや決済管理システムなどの他のコンポーネントを使用して、金融取引を完了する。

## S

**Secure Electronic Transaction (SET).** 「SET Secure Electronic Transaction」を参照。

**Secure Sockets Layer (SSL).** クライアントがサーバーを認証したり、あらゆるデータや要求を暗号化したりするためのセキュリティ・プロトコルの 1 つ。SSL によって保護されたセキュア・サーバーの URL は (HTTP ではなく) HTTPS で始まる。

**SET.** 「SET Secure Electronic Transaction」を参照。

**SET Secure Electronic Transaction™.** インターネットなど、オープン・ネットワーク上で、クレジット・カードやデビット・カードによる決済を安全に実行するために開発された業界標準規格。

**socks サーバー (socks server).** 非セキュア・ネットワークで、ファイアウォールを介してサーバー・アプリケーションにセキュアは一方接続を提供するプロキシ・サーバー。セキュア・ネットワーク内のサーバー・アプリケーションは、ソケット・インターフェースとの互換性がなければならない。

**socks プロトコル (socks protocol).** 保護されたネットワーク内のアプリケーションが socks サーバーを経由して、ファイアウォールを介して通信することを可能にするプロトコル。

**socks ポート (socks port).** socks サーバーが listen するときのポート。

**SSL.** 「Secure Sockets Layer」を参照。

## U

**uniform resource locator (URL).** インターネット上のファイルのアドレス。URL には、プロトコル名、完全修飾ドメイン・ネーム、およびパスおよびファイルの位置が含まれる。

**URL.** 「uniform resource locator」を参照。

## W

**WAR ファイル (WAR file).** Web アーカイブ (WAR) は、下記の 1 つ以上を格納するために使用される Java アーカイブ・ファイルである。サーブレット、JavaServer Pages (JSP) ファイル、ユーティリティ・クラス、静的文書 (HTML ファイル、イメージ、サウンドなど)、クライアント・サイドのアプレットと bean とクラス、記述的メタ情報。その標準的なファイル拡張子は .war である。WAR ファイルは Web モジュールをパッケージするのに使用される。

**Web サーバー (Web server).** Web 上にあって、HTTP 文書に関する要求を処理するサーバー。Web サーバーは、WebSphere Commerce との間のトランザク

ションの流れを制御する。これは、顧客のトランザクションの機密性を保護すると共に、ユーザー識別が WebSphere Commerce Server に安全な方法で送信されることを保証する。そのようなレベルのセキュリティを達成するため、Web サーバーには Secure Sockets Layer (SSL) プロトコルが実装されている。

**Web サイト (Web site).** 単一エンティティ (組織または個人) によって管理され、そのユーザーに関するハイパーテキストの情報を含む Web サーバーで、多くの場合、他の Web サイトへのハイパーテキスト・リンクを含んでいる。各 Web サイトにはホーム・ページがある。URL では、Web サイトは完全修飾ドメイン・ネームによって示される。たとえば、<http://www.as400.ibm.com/icswg.html> という URL の場合、IBM AS/400 の Web サイトは、[www.as400.ibm.com](http://www.as400.ibm.com) という完全修飾ドメイン・ネームによって表される。

**Web ブラウザー (Web browser).** (1) IBM e-commerce において、カード所有者の処理システムで実行するソフトウェアで、公衆データ網へのインターフェースを提供する。(2) Web サーバーに対する要求を開始し、サーバーから戻された情報を表示するクライアント・プログラム。

**Web ページ (Web page).** World Wide Web (WWW) の URL によってアクセスできる文書。

**WebSphere.** 基本的な Web 作成のための環境や、トランザクションの多い企業で使用する e-business アプリケーションを開発および展開するための環境を提供する IBM ソフトウェア製品ファミリーに使用される用語。

## X

**XML.** マークアップ言語を定義する標準のメタ言語。SGML から派生し、SGML のサブセットである。XML は、SGML のより複雑でより使用頻度の低い部分を省略し、文書タイプを操作するためのアプリケーションの作成、構造化情報の作成および管理、およびさまざまなコンピューティング・システム間での構造化情報の伝送および共有を、より容易に実行できるようにしたものである。XML の定義は、World Wide Web Consortium (W3C) によってなされている。

# 索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

## [ア行]

- アクセス制御, 役割ベースの 16
- アドレス検証サービス 25
- イベント
  - 内容 49
- イベント通知
  - Event ListenerURL パラメーター 52
- イベント通知サービス 49
  - イベントのタイプ 49
- イベント・リスナー
  - タイプ 51
- イベント・リスナーの定義 3
- インスタンス, 複数 55
- エスケープ・シーケンス 13
- エンコード, URL
  - 規則 13
- オーダー 111
- オーダーの作成
  - 必須キーワード 59
- オーダーの定義 4
- オーダーのライフ・サイクル 111
- 応答クラス 42
- オブジェクト
  - 状態 114
  - 定義する方法 112
- オブジェクト, 決済 111
- オブジェクト, フレームワーク 111
- オブジェクト, Account
  - 属性 126
- オブジェクト, Cassette 124
- オブジェクト, Credit
  - 属性 119
- オブジェクト, Event Listener
  - 属性 127
- オブジェクト, Merchant
  - 属性 125
- オブジェクト, Order
  - 属性 112
- オブジェクト, Payment
  - 属性 115
- オブジェクト, Payment System
  - 属性 125
- オブジェクト, User
  - 属性 128

## [カ行]

- 拡張
  - 作成 36
- カセット固有イベント 49
- カセットの定義 3
- 稼働パラメーター 55
- キーワードと値の対 11
- キー・コレクション 55
- 基準, 検索 55
- キャッシャー
  - エラー 26
  - 概要 21
  - トレース 26
  - 例外 26
- キャッシャーの定義 3
- キャッシャー・オブジェクトの作成 33
- キャッシャー・プロファイルの作成 26
- 許可, 役割に対する 18
- 行えること, 役割で 19
- 金融照会 55
- クライアント API ライブラリー 41
- クライアント API ライブラリー (CAL) 41
- クライアント・クラス 41
- クラス, クライアント 41
- クラス, PaymentServerClient 46
- クラス, PObject 42
- クレジット 111
  - 状態 120
- クレジットの定義 3
- 結果コード, AVS 118
- 決済 111
  - 状態 117
- 決済, 分割 118
- 決済オブジェクト 111
- 決済開始メッセージ 105
- 決済の定義 4
- 検索基準 55
- 検索修飾子 55
- コード, 通貨 59
- コード, 1 次戻り 14, 131
- コード, 2 次戻り 14
  - タイプ 131
- 構築, プロファイルの 31
- 購入ページ情報 (BuyPageInformation) 29
- コマンド
  - 必須値 55
  - About 56
  - AcceptPayment 57
  - Approve 60

コマンド (続き)

- ApproveReversal 61
- BatchClose 62
- BatchOpen 62
- BatchPurge 63
- CancelOrder 64
- CassetteControl 65
- CloseOrder 66
- CreateAccount 67
- CreateMerchant 69
- CreateMerchantCassetteObject 69
- CreateMerEventListener 70
- CreatePaySystem 71
- CreateSNMEEventListener 72
- CreateSystemCassetteObject 72
- DeleteAccount 73
- DeleteBatch 74
- DeleteMerchant 75
- DeleteMerchantCassetteObject 75
- DeleteMerEventListener 76
- DeletePaySystem 77
- DeleteSNMEEventListener 77
- DeleteSystemCassetteObject 78
- Deposit 78
- DepositReversal 79
- ModifyAccount 80
- ModifyCassette 82
- ModifyMerchant 82
- ModifyMerchantCassetteObject 83
- ModifyMerEventListener 84
- ModifyPayServer 85
- ModifyPaySystem 85
- ModifySNMEEventListener 86
- ModifySystemCassetteObject 87
- ModifyUserStatus 88
- QueryAccounts 88
- QueryBatches 89
- QueryCassette 91
- QueryCredits 92
- QueryEventListeners 94
- QueryMerchants 95
- QueryOrders 96
- QueryPayment 98
- QueryPaymentServer 100
- QueryPaySystems 101
- QueryUsers 102
- ReceivePayment 105
- Refund 107
- RefundReversal 108
- SetUserAccessRights 109

コマンド, 照会 14

コレクション 55

## [サ行]

作成、オーダーの  
必須キーワード 59  
作成、拡張の 36  
作成、キャッシャー・プロファイルの 26  
作成、統合システムの 30  
修飾子、検索 55  
照会、金融 55  
照会コマンド 14  
規則 55  
状態、相対オブジェクトの 13  
状態、バッチ 122  
状態変更イベント 49  
商標 166  
情報、認証 16  
スプリット・ペイメント 118  
ゼロ、先行 55  
先行ゼロ 55  
相対オブジェクトの状態 13

## [タ行]

対、キーワードと値の 11  
通貨、ISO 59  
通貨コード 59  
通貨コード、ISO 151  
統合システム 21  
作成 30  
設計 23  
特記事項 165  
トレース、キャッシャー 26

## [ナ行]

名前と値のペア  
指針 55  
認証情報 16  
ネットワーク管理イベント 49

## [ハ行]

バイヤーの定義 3  
バッチ 111  
バッチの状態 122  
バッチの定義 3  
パラメーター、稼働 55  
パラメーター、RETURNATMOST 56  
非マーチャント・リスナー 51  
複数インスタンス 55  
フレームワークの定義 4  
フレームワーク・オブジェクト 111

プログラム、マーチャント  
CAL 用に作成された 47  
プログラム、CAL  
形式 47  
プロファイルの構築 31  
プロファイルの作成、キャッシャー 26  
文書、XML 14  
文書タイプ定義、DTD 14  
ペア、名前と値の  
指針 55  
ポーリング・ループ 49

## [マ行]

マーチャントの定義 4  
マーチャント・ソフトウェアの定義 4  
マーチャント・プログラム  
CAL 用に作成された 47  
マーチャント・リスナー 51  
メッセージ、HTTP POST 11  
文字、Unicode 13  
文字セット 13  
戻りコード  
概要 131  
バージョン 1.2 用の新しい構造 131  
場所 131  
1次 131  
2次 134  
戻りコード、1次 14, 131  
戻りコード、2次 14, 131

## [ヤ行]

役割、ユーザーの 16  
役割で行えること 19  
役割に対する許可 18  
ユーザー ID の作成 102  
ユーザーの役割 16  
ユーザー・オブジェクト  
属性 128  
要求、HTTP POST 55  
用語 vi

## [ラ行]

ロケール 12

## [数字]

1次戻りコード 14, 131  
2次戻りコード 14, 131

## A

About コマンド 56  
AcceptPayment コマンド 57  
Account オブジェクト  
属性 126  
AmountExp10 キーワード 59  
AMOUNTEXP10 パラメーター 59  
Approve コマンド 60  
ApproveReversal コマンド 61  
AVS 25  
AVS 共通コード 118  
AVS 結果コード  
共通 AVS コードへのマッピング 118  
CyberCash カセットへのマッピング  
118  
SET カセットへのマッピング 118

## B

Batch  
アカウントの関連 120  
属性 120  
BatchClose コマンド 62  
BatchOpen コマンド 62  
BatchPurge コマンド 63

## C

CAL 41  
必須ファイル 47  
CAL プログラム  
形式 47  
CAL、Java クライアント API ライブラリ  
ー 41  
CancelOrder コマンド 64  
Cassette オブジェクト 124  
CassetteControl コマンド 65  
checkPayment 34  
close メソッド 46  
CloseOrder コマンド 66  
CollectPayment 28, 33  
Command 28  
CreateAccount コマンド 67  
CreateMerchant コマンド 69  
CreateMerchantCassetteObject コマンド 69  
CreateMerEventListener 70  
CreatePaySystem コマンド 71  
CreateSNMEventListener コマンド 72  
CreateSystemCassetteObject コマンド 72  
Credit オブジェクト  
属性 119

## D

DeleteAccount コマンド 73  
DeleteBatch コマンド 74  
DeleteMerchant コマンド 75  
DeleteMerchantCassetteObject コマンド 75  
DeleteMerEventListener コマンド 76  
DeletePaySystem コマンド 77  
DeleteSNMEventListener コマンド 77  
DeleteSystemCassetteObject コマンド 78  
Deposit コマンド 78  
DepositReversal コマンド 79  
DTD (文書タイプ定義) 14

## E

Event Listener オブジェクト  
属性 127  
Event ListenerURL 52  
EventType 49

## H

HTTP POST メッセージ 11  
HTTP POST 要求 55  
HTTP ヘッダー  
計算値 11  
追加ヘッダー・フィールド 12  
必須フィールド値 11  
HTTP 本体  
エンコード 12  
形式の規則 12

## I

ISO 通貨 59  
ISO 通貨コード 151  
issueCommand 34  
issueCommand メソッド 45

## J

Java クライアント API ライブラリー 41  
Java クライアント API ライブラリー、  
(CAL) 41

## M

Merchant オブジェクト  
属性 125  
ModifyAccount コマンド 80  
ModifyCassette コマンド 82  
ModifyMerchant コマンド 82  
ModifyMerchantCassetteObject コマンド  
83

ModifyMerEventListener コマンド 84  
ModifyPayServer コマンド 85  
ModifyPaySystem コマンド 85  
ModifySNMEventListener コマンド 86  
ModifySystemCassetteObject コマンド 87  
ModifyUserStatus コマンド 88

## O

ObjectID 50  
Order オブジェクト  
属性 112

## P

Payment System オブジェクト  
属性 125  
Payment オブジェクト  
属性 115  
PaymentServerClient  
サブクラス 43  
引き数 42  
PaymentServerClient クラス 46  
PaymentServerResponse 46  
PaymentServerSSLClient 43  
POST メッセージ、HTTP 11  
PRC 131  
PSObject クラス 42

## Q

QueryAccounts コマンド 88  
QueryBatches コマンド 89  
QueryCassette コマンド 91  
QueryCredits コマンド 92  
QueryEventListeners コマンド 94  
QueryMerchants コマンド 95  
QueryOrders コマンド 96  
QueryPayment コマンド 98  
QueryPaymentServer コマンド 100  
QueryPaySystems コマンド 101  
QueryUsers コマンド 102

## R

ReceivePayment コマンド 105  
Refund コマンド 107  
RefundReversal コマンド 108  
Request For Comments, RFC  
URL アクセス 163  
RETURNATMOST パラメーター 56  
RFC, Request For Comments  
URL アクセス 163

## S

SET  
取引の開始 105  
SetUserAccessRights コマンド 109  
socksHost 42  
socksPort 42  
SRC 134  
SSL 接続 13

## T

Timestamp 49

## U

Unicode 文字 13  
URL エンコード  
規則 13

## W

WebSphere Commerce Payments  
用語 3  
WebSphere Commerce Payments About オ  
ブジェクト 122  
WebSphere Commerce Payments 管理オブ  
ジェクト 123  
WebSphere Commerce Payments コマンド  
11  
例 11  
WebSphere Commerce Payments の用語 3  
WebSphere Commerce Payments 要求 11

## X

XML 文書 14







Printed in Japan