

Integration Guide

iPlanet BillerXpert 4.6 B2B Edition

May 2002

Copyright © 2002 Sun Microsystems, Inc.

Sun, Sun Microsystems, and the Sun logo are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Identrus and Identrus Trustbase are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation. No part of the product or this document may be reproduced in any form by any means without prior written authorization of the Sun-Netscape Alliance and its licensors, if any.

THIS DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright © 2002 Sun Microsystems, Inc.

Sun, Sun Microsystems, et le logo Sun sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et d'autre pays.

Identrus and Identrus Trustbase are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Le produit décrit dans ce document est distribué selon des conditions de licence qui en restreignent l'utilisation, la copie, la distribution et la décompilation. Aucune partie de ce produit ni de ce document ne peut être reproduite sous quelque forme ou par quelque moyen que ce soit sans l'autorisation écrite préalable de l'Alliance Sun-Netscape et, le cas échéant, de ses bailleurs de licence.

CETTE DOCUMENTATION EST FOURNIE "EN L'ÉTAT", ET TOUTES CONDITIONS EXPRESSES OU IMPLICITES, TOUTES REPRÉSENTATIONS ET TOUTES GARANTIES, Y COMPRIS TOUTE GARANTIE IMPLICITE D'APTITUDE À LA VENTE, OU À UN BUT PARTICULIER OU DE NON CONTREFAÇON SONT EXCLUES, EXCEPTÉ DANS LA MESURE OÙ DE TELLES EXCLUSIONS SERAIENT CONTRAIRES À LA LOI.

Contents

Before You Begin	1
Audience	1
Organization	2
Documentation Conventions	2
The Document Online	2
Related Documentation	3
Product Support	3
Chapter 1 iPlanet BillerXpert B2B Edition Integration	5
Introduction	5
BillerXpert B2B Edition Solution Overview	6
Overview	8
Portals	9
iPlanet Portal Server	9
Payment Services	11
Payflow(SM) by VeriSign Payment Services	12
Overview	12
ClearLink by Clear Commerce	13
Currency Feed	14
Reuters Investor	15
Appendix A Portal Server	1
Integrating an Application with iPlanet Portal Server Software	1
BillerXpert Implementation	1
Storing the HTML Template	1
Dependencies	2
Storing the Class File	2
Dependencies	2
Restarting iAS	2
Portal Server Implementation	3

Contents of the Tar File	3
Extracting the Tar File	3
Importing the XML File	3
Specifying the BillerX Host	3
Specifying the Default Page	4
Adding the BillerX Channel	4
Restart the Portal Server and the Gateway	4
Appendix B Payment Services	1
VeriSign	1
Payment Handlers	2
Payment Creation handler	2
Payment Status Inquiry handler	2
Payment Modification handler	3
Payment Cancellation handler	3
Data Synchronization handler	3
Activating VeriSign	4
Becoming a Vendor	4
Downloading	4
BillerXpert Setup	5
Login to Admin Console	5
Editing The Biller Profile	6
OFX URL	6
Payment Testing	6
Login into BillerXpert	7
Pay Balance	7
Test Payment	7
Payment Completed	7
Clear Commerce Setup	8
Packaging	8
Payee Configuration on the ClearCommerce Hosting Engine	8
Authentication	8
Setting Up A Payment Vendor	9
ClearCommerce As A Payment Server	9
Appendix C Currency Feed	1
Implementation Strategies	1
Server-side method – XML	1
Currency Conversion Process	2
Installation Process for Currency Converter	3
How to Load Currency Converter	3
Configuring CCLoad Properties	4
Configuring getData.csh	5

Installation of Reuters Investor	5
Appendix D Enrollment	1
Realtime Enrollment and Activation (REA)	1
Request - Response Flow in the REA Framework	3
REA Implementation Methods	4
Installation, and Handler Procedures	4
REA Server Installation Procedure	4
BillerXpert Installation	4
Handler Procedure	5
Example of a Request Response	6
Testing REA	13
To enroll a new member	13
Verification	21
Appendix E XML Payment Specification	1
PaymentRequest	1
PaymentRequest DTD	1
Fully Expanded PaymentRequest	2
PaymentCancellationRequest	3
PaymentCancellationRequest DTD	3
Fully Expanded PaymentCancellationRequest	4
ResponseMessage	5
Fully Expanded ResponseMessage	5
Legend of Message Data Requirement Codes	6

Preface

This guide gives instructions on how to integrate iPlanet BillerXpert 4.6 B2B Edition with other applications. It covers an overview of each application, setup, and verification processes.

This preface contains the following sections:

- Before You Begin
- Audience
- Organization
- Documentation Conventions
- The Document Online
- Related Documentation
- Product Support

Before You Begin

You need to use this manual if you are integrating other applications with iPlanet BillerXpert 4.6 B2B Edition. In particular, tasks such as setup and verification support the Integration Guide.

Audience

The intended audience for this guide is the person(s) who will be managing BillerXpert and will be responsible for subsequent product integration efforts for BillerXpert.

Organization

This guide is divided into the following sections:

- Overview
- Portal Server
- Payment Services
- Currency Feed
- Enrollment
- Related appendices A-E

Documentation Conventions

This document uses the following conventions:

- The `monospace` font is used for sample code and code listings, Application Program Interface (API) and language elements (such as method names and property names), file names, path names, directory names, Hypertext Markup Language (HTML) tags, and any text that must be typed on the screen.
- The *italic* font is used in code to represent placeholder parameters (variables) that should be replaced with an actual value.
- Brackets ([]) are used to enclose optional parameters.
- A slash (/) is used to separate directories in a path. (Windows NT supports both the slash and the backslash.)

The Document Online

For an overview of iPlanet BillerXpert 4.6 B2B Edition changes, refer to the iPlanet BillerXpert 4.6 B2B Edition Release Notes.

An electronic version of these notes are available at:

<http://docs.iplanet.com/docs/manuals/>

Related Documentation

The BillerXpert documentation set includes:

- *Release Notes*—Contains important information on the current release of iPlanet BillerXpert 4.6 B2B Edition. Read this document before working with the new iPlanet BillerXpert 4.6 B2B Edition release.
- *Administrator's Guide*—Provides reference information and instructions on administering a fully-installed iPlanet BillerXpert 4.6 B2B Edition system.
- *Installation Guide*—Provides instructions for installing the iPlanet BillerXpert 4.6 B2B Edition product and its enabling software.
- *Customization Guide*—Provides guidelines and instructions for customizing the iPlanet BillerXpert 4.6 B2B Edition system.

Documentation for all iPlanet products can be found at the following web site:

<http://docs.iplanet.com/docs/manuals/>

Product Support

If you have problems with your BillerXpert system, contact iPlanet customer support using one of the following mechanisms:

- iPlanet online support web site at:

<http://www.iplanet.com/support/online/>

From this location, the CaseTracker and CaseView tools are available for logging problems.

- The telephone dispatch number associated with your maintenance contract

So that the technical support staff can best assist you in resolving problems, please have the following information available when you contact support:

- Description of the problem, including the situation where the problem occurs and its impact on your operation
- Machine type, operating system version, and product version, including any patches and other software that might be affecting the problem
- Detailed steps on the methods you have used to reproduce the problem
- Any error logs or core dumps

iPlanet BillerXpert B2B Edition Integration

This chapter contains the following sections:

- Introduction
- Overview
- Portals
- Payment Services
- Currency Feed

Introduction

Business-to-Business invoice presentment is a rapidly growing market. It is expected that the Business-to-Business invoicing transaction value will reach \$14.2 Trillion by 2010, with over 8 billion invoice presented online by 2004. Online invoice presentment not only saves money for the companies involved through less paper work, more defined and thorough processes, but also allows for companies to have more control over and insight into the entire invoice process.

The invoicing process for companies is not just the collection of payment for the goods defined in the invoice, but also the management of company approvals for payment, management of disputes, and the ways companies would like to make payment. iPlanet BillerXpert B2B Edition gives companies the capabilities to manage the entirety of the invoice process including presentment, payment, and dispute management.

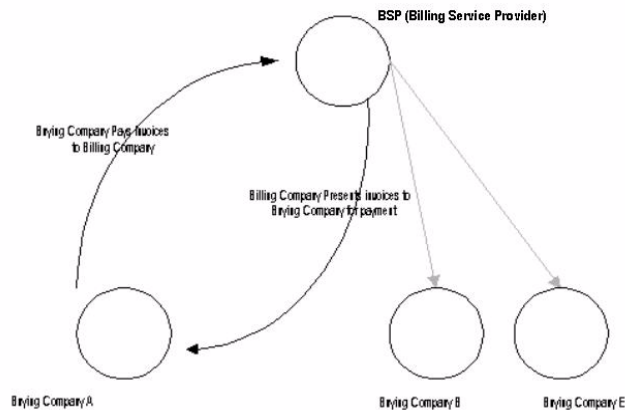
BillerXpert B2B Edition Solution Overview

Businesses today spend millions of dollars in the actual billing of its customers. Current billing system and processes are predominately paper based and mostly conducted through human interaction. BillerXpert B2B allows businesses to eliminate paper and automate the routing of invoices for their customers, giving them more efficiency and control of the entire process.

The Billing Company, using a special XML format, loads invoices for its customers (the Buying Companies) into the system on a regular basis. The system distributes the invoices for approval and payment to the appropriate buying company. Within the buying company, a customized approval process can be constructed to manage the necessary approvals required before a payment is made.

The Billing Company can use the system to monitor the state of any invoice, manage buying company disputes of a particular invoice, and use the recorded data to generate reports on the overall invoicing process.

Figure 1-1 BillerXpert B2B Solution



The Billing Company is the company that will be presenting invoices to its customers (Buying Companies). The Billing Service Company (BSP) is the company that hosts the BillerXpert B2B system. The Buying Companies are the companies that are being billed and will use the system to approve the invoices and make the payment. An overview of this solution can be seen in Figure 1-1.

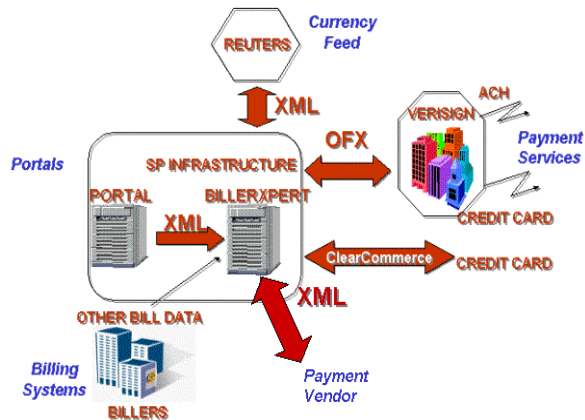
iPlanet BillerXpert B2B Edition handles approvals of invoices down to the line item level. This means that for complex invoices that represent orders from multiple departments, BillerXpert B2B has the capability to manage the departmental approvals of each line number across the entire invoice.

Overview

The evolutionary growth of the BillerXpert product market has presented the opportunity to integrate additional functionality with other business applications. By foreseeing the demands of our new net economy, the BillerXpert product, as seen in Figure 1-2, has been enhanced to accommodate the following opportunities for product integration:

- Portals
- Payment Services
- Currency Feed

Figure 1-2 BillerXpert Integration

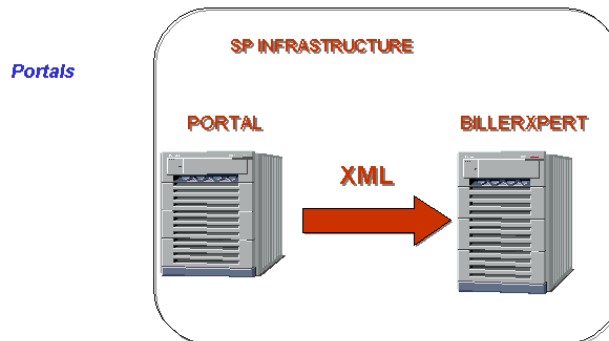


Portals

In just a short time, the Internet has become the standard by which corporate networking organizations increase productivity and enhance competitiveness.

There are many types of speciality portals, including knowledge management, content management, and retail portals. However, portals are typically custom built or purchased individually for a specific purpose. Individually built or purchased portals are costly to integrate, if integration is even possible, and maintain.

Figure 1-3 iPlanet Portal Server



iPlanet Portal Server

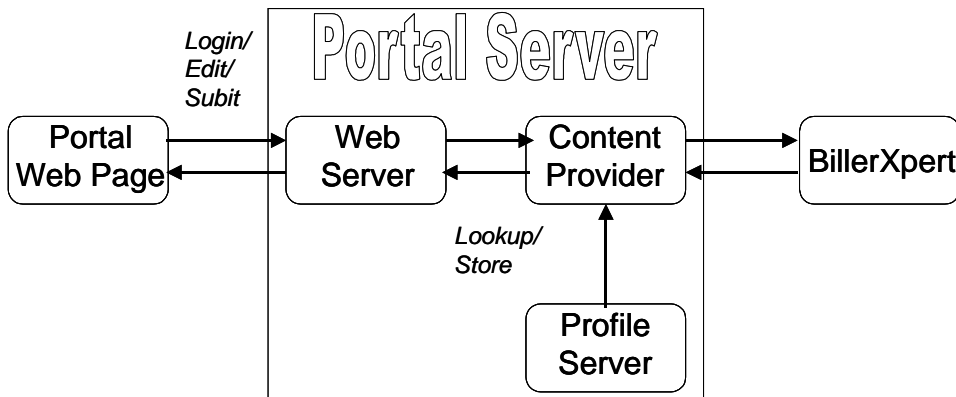
Figure 1-3 illustrates the communication between iPlanet Portal Server and BillXpert.

A major benefit of iPlanet Portal Server software—compared to other portal offerings—is the simplicity of its architecture: a server software-based solution that does not require any client software installation, management, or configuration. The client application is either pushed to the client’s browser as

HTML web pages, or is a Java™ applet that is downloaded on demand. With iPlanet Portal Server software, users are only required to have a web browser and an Internet connection. In this way, iPlanet Portal Server can support a complete, heterogeneous computing environment. Private networks are not affected by the use of iPlanet Portal Server; server-side plug-ins are not required on intranet Web servers; protocol conversion modules are not required on file servers; and existing authentication schemes can be used. The portals are displayed in the form of channels on the desktop. Integrating iPlanet Portal Server with BillerXpert provides you the ability to see a billing summary as a channel on the portal server.

The diagram seen in Figure 1-4 depicts a flow diagram of the various components involved in integrating BillerXpert as a channel on the Portal desktop.

Figure 1-4 Portal Server flow diagram



See also

For more technical information regarding the iPlanet Portal Server integration, including setup and verification processes, please refer to *Appendix A* of this document.

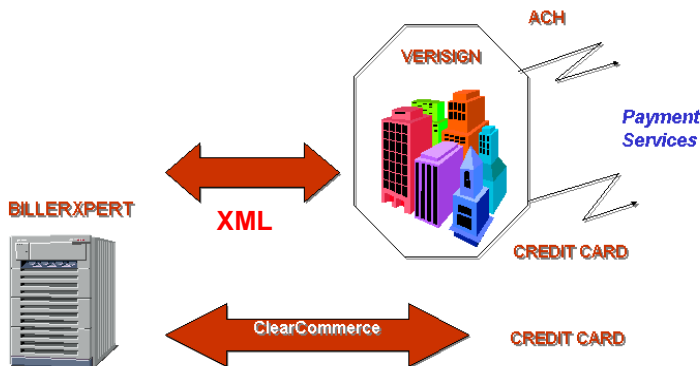
Payment Services

BillerXpert has introduced an Open Financial Exchange (OFX) client to support consumer OFX payments and to generate applicable OFX payment requests. OFX Payments consist of a set of functions for creating, modifying, cancelling, and for synchronizing payment information.

BillerXpert is building functionality for XML payments. This payment method will be used with future editions of BillerXpert B2B, with Identrus, the European banking consortium.

BillerXpert’s OFX client allows users to send payment transactions to an OFX compliant Payment Service under the “Pay-some” model. “Pay-some” systems restrict users to only make payments to payees that appear on an approved list. The relationship between BillerXpert and Payment Services is depicted in Figure 1-5.

Figure 1-5 Payment Services Integration with BillerXpert



BillerXpert includes support for both VeriSign and ClearCommerce payment services. Details for these payment services are included in the body of this integration guide, as well as in *Appendix B*. For information regarding the new XML Payment specification, please refer to *Appendix D*.

Payflow^(SM) by VeriSign Payment Services

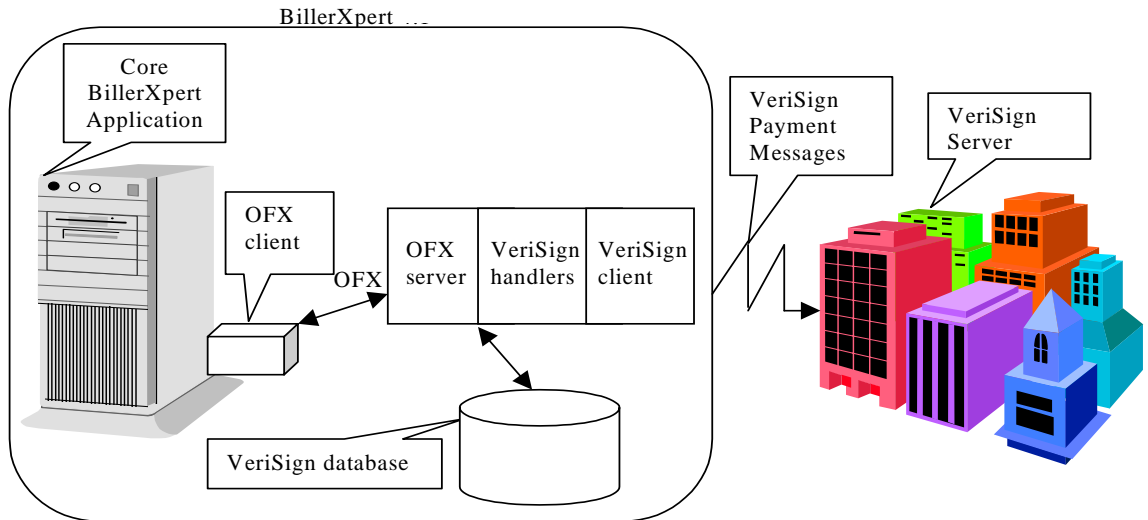
The integration of BillerXpert and VeriSign offers a customer a high volume IBPP solution that integrates bill presentment with a convenient online payment service.

VeriSign is an e-commerce service company which provides multiple tender options in a single interface, with 6 payment solutions. By using VeriSign's Payflow payment services, iPlanet BillerXpert businesses can automate the validation and acceptance of all major consumer credit cards, debit cards and automated clearing house (ACH) transactions. Further, the average VeriSign transaction times are less than 3 seconds for real-time credit card and Internet check authorizations.

Overview

The communication between BillerXpert and VeriSign Payment Services is transparent as OFX Payment functions are used to interface BillerXpert and VeriSign Payment Services.

When a user submits a payment transaction, BillerXpert's OFX client generates the appropriate OFX request which is routed to BillerXpert's OFX Server. The applicable Payment Handler is invoked and the request is translated and submitted to the VeriSign Client. The VeriSign client then sends the request to the VeriSign Payment Service's Server. BillerXpert's OFX Server sends the OFX Response to BillerXpert's OFX client as acknowledgement of the transaction. Figure 1-6 describes the payment flow for BillerXpert.

Figure 1-6 BillerXpert VeriSign Payment Services model*See also*

For more technical information regarding VeriSign Payment Services, including Payment Handlers, Setup and Verification processes, please refer to *Appendix B* of this document.

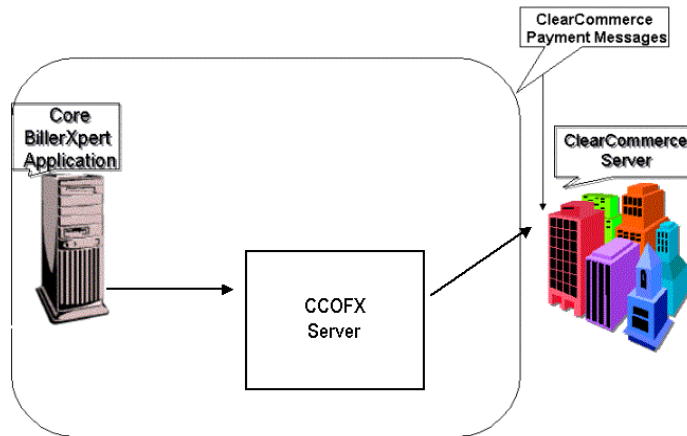
ClearLink by Clear Commerce

ClearLink for BillerXpert allows an enterprise to take full advantage of both powerful Internet applications; BillerXpert's comprehensive Internet Bill Presentment and Payment solutions and the ClearCommerce Engine's fast, reliable, and scalable credit card fraud detection and transaction processing environment for their customers.

By working with ClearCommerce, BillerXpert is extending its payment system offerings for customers requiring choice in their payment solution deployments. Payment options allow payment by credit card, or even with scheduled bill payments for a future date. Additionally, customers can create automatic payments for recurring bills, or make partial payments against bills when appropriate.

ClearCommerce and BillerXpert communicate using a Clear Commerce OFX (CCOFX) Server as seen in Figure 1-7.

Figure 1-7 ClearLink by ClearCommerce



See also

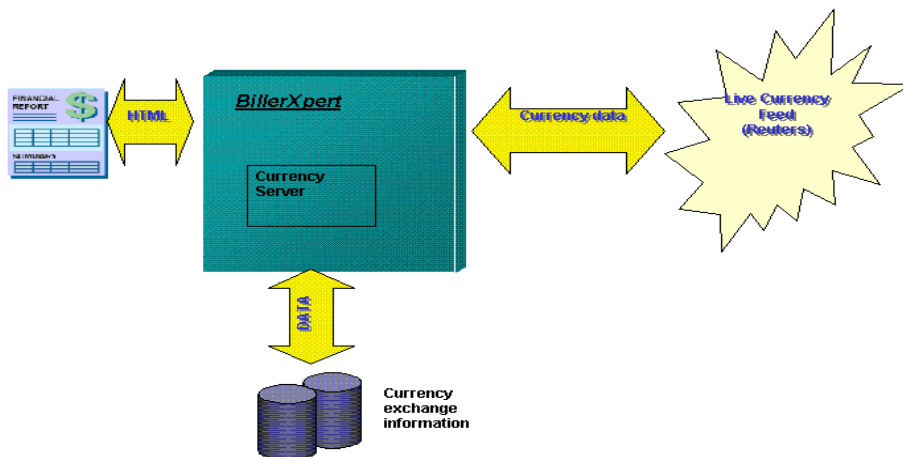
For more technical information regarding ClearCommerce, including setup and verification processes, please refer to *Appendix B* of this document

Currency Feed

BillerXpert B2B Edition introduces a Currency Converter module to support currency conversion capabilities for both banks and billers. The Currency Converter module is a pluggable component that provides host applications APIs to help convert currencies. It is EURO compliant.

Within the Currency Conversion module, there is also a loader program, which is used to load the currency conversion entries. The loader program takes in an Extensible Markup Language (XML) file of conversion data and loads it into the conversion table. Updates to the currency conversion table are performed by the administrator either manually, or via an external scheduling software program.

Figure 1-8 Reuters live Currency Feed to BillerXpert enables accurate financial reporting in response to customer inquiries.



In Figure 1-8, a user submits an HTML request for financial information to BillerXpert, which calls its stored currency data, which is refreshed regularly from Reuters XML feeds.

Reuters Investor

Few categories of Internet based information are more highly used than personal financial information. And no company's financial information is as comprehensive, accurate, and well respected as Reuters. Quotes, market and company news, price histories, charts and portfolio information are the essential financial components everyone needs to see all day, every day - from serious investment managers to stock market debutantes.

Reuters Investor enables you to include reuters news and market data in your website on the public internet, subject to the correct data licenses, agreements, and exchange fees. You can provide your users with the latest news headlines and stories, as well as price information and charts from a range of markets.

The Reuters Investor service is hosted at a Reuters Data Centre and delivers all data via the Internet. You do not need to install any Reuters hardware or software applications.

Reuters Investor is built on top of the standard HTTP and HTTPS protocols. It provides a set of functions that let you access the relevant Reuters Investor server, which retrieve the information your requested.

See also

For more technical information regarding Currency Feed, including setup and verification processes, please refer to *Appendix C* of this document.

Portal Server

This appendix contains the following sections:

- Integrating an Application with iPlanet Portal Server Software
- BillerXpert Implementation
- Portal Server Implementation

Integrating an Application with iPlanet Portal Server Software

A major benefit of iPlanet Portal Server software—compared to other portal offerings— is the simplicity of its architecture: a server software-based solution that does not require any client software installation, management, or configuration. The client application is either pushed to the client's browser as HTML web pages, or is a Java™ applet that is downloaded on demand. With iPlanet Portal Server software, users are only required to have a web browser and an Internet connection.

BillerXpert Implementation

This sections briefly explains the various activities involved in implementing BillerXpert as a channel on the Portal desktop.

Storing the HTML Template

The HTML template for displaying the modified summary is stored in the templates directory.

The templates directory is `$WEB_SERVER/./docs/ProjDir`

Dependencies

If the HTML template (`CH_billing_center.htm`) is modified in any manner then it will impact the functionality of the code written specifically BillerX-Portal Server project.

Storing the Class File

For implementing the new functionality there are two extra pieces of AppLogic code. These files must be stored in the same directory as the existing AppLogics. All files must be compiled together using a make file to satisfy the dependencies. Therefore if any changes are made in the AppLogic code, a make must be done to generate the new class file in the zipped format.

The zipped class file for the new functionality must then be stored in the java directory.

The class file directory is `$BX_HOME/java`.

NOTE To use the environment variables you should have set the environment to the BillerX.

Dependencies

If the AppLogics (`CH_BXSignOnAL.java`, `CH_BXBillingCenterAL.java`) are modified in any manner then it will impact the functionality of the code written specifically BillerX-Portal Server project.

Restarting iAS

After storing the files in the appropriate directories, iAS must be restarted. This is done by a series of three steps:

1. `cd $IAS_HOME`

2. `KIVAes.sh stop`
3. `KIVAes.sh start`

Portal Server Implementation

The files related to the BillerX integration are packaged into a tar file called `billerxportal.tar`.

Contents of the Tar File

```
billerx/iwtBillerXpertProvider.xml
billerx/iwtBillerXpertProvider.class
billerx/BXHOST
billerx/BXSTATIC
```

Extracting the Tar File

1. Copy the tar file to the directory
`/opt/SUNWips/lib/com/iplanet/portalserver/providers`
2. Extract the tar file using the command

```
tar xvf billerxportal.tar
```

A directory called `billerx` is created with the required files.

Importing the XML File

1. Change to `billerx` directory
2. As Unix user root, import the XML file `iwtBillerXpertProvider.xml` using the `ipsadmin` command:

```
ipsadmin -import iwtBillerXpertProvider.xml
```

Specifying the BillerX Host

The BillerX application will be hosted in a particular machine. The host name and the port number has to be specified in the format *http://hostname:portnumber/*. For example:

```
http://celery:8081/
```

This entry is made in the file `BXHOST`.

Specifying the Default Page

The page to be displayed when the BillerX username is not known is stored in the file `BXSTATIC`. The complete URL of the html page has to be specified. For example:

```
http://celery:8081/telco/portal.html
```

Adding the BillerX Channel

Once the class files are put in the correct directory and the XML file is imported we have to now add the channel into the desktop. The procedure to add a channel is as follows:

1. Login into administration console
2. Click on Manage Domains
3. Click on the domain where the channel has to be added
4. Expand the Applications option
5. Click on the Desktop option
6. Fill in the following details:
 - Channel Name : `iwtBillerXpertProvider`
 - Provider Class Name:
`com.iplanet.portalserver.providers.billerx.BillerXpertProvider`
7. Click on the Add button and then on the Submit button.

Now the channel gets added into the Available Channels list. If you want this channel to be available as a default choice to a new user, include this channel into the Selected Channels list by using the arrow keys.

Restart the Portal Server and the Gateway

This is done by a series of four steps:

1. `ipsgateway stop`
2. `ipsserver stop`
3. `ipsserver start`
4. `ipsgateway start`

Payment Services

BillerXpert provides, through the Administration console, the ability to route OFX payment requests to an OFX compliant Payment Service. The setup is done by the Billing Service Provider (BSP) for the hosted Biller.

This appendix contains the following sections:

- VeriSign
- BillerXpert Setup
- Payment Testing
- Clear Commerce Setup
- Setting Up A Payment Vendor

NOTE Testing payment verification is subject and dependent upon customization of BillerXpert and the Payment Services provider. For reference as an example of testing Payment Services with VeriSign, see the following section *VeriSign Setup* in this appendix.

VeriSign

To activate VeriSign payment processing in BillerXpert, a Biller simply needs to set up a VeriSign account, create a VeriSign Oracle user and install the BillerXpert support modules during installation.

In order to support VeriSign's Payment Services, BillerXpert has five (5) Payment handlers. Details of these Payment handlers are included in the following section.

Payment Handlers

The five (5) Payment handlers that convert OFX messages-responses to VeriSign requests-returns to handle basic consumer payments. These handlers provide the following:

- Payment Creation
- Payment Status Inquiry
- Payment Modification Status Values
- Payment Cancellation
- Payment Synchronization

The details of each Payment handler follow:

Payment Creation handler

A Payment Request <PMTRQ> is sent from the OFX client within BillerXpert to request a payment to schedule an electronic payment. Payment Response <PMTRS> is sent from the VeriSign Payment Services server in response to the Payment Request.

- A.C.H. (Automated Clearing House)
- Credit card

Reference <http://www.ofx.net/ofx/downloads/spec16/payments.html#9033> for specific assigned parameters for <PMTRQ> and <PMTRS>.

Payment Status Inquiry handler

The Payment Status Inquiry Request, <PMTINQRQ> is sent from the OFX client within BillerXpert to obtain the current processing status of a payment from the VeriSign Payment Service's server. Payment Status Inquiry Response, <PMTINQRS> is sent from the VeriSign Payment Service's server to a Payment Status Inquiry Request.

- Credit card
- A.C.H. (Automated Clearing House)

Reference <http://www.ofx.net/ofx/downloads/spec16/payments.html#9033> for specific assigned parameters for <PMTINQRQ> and <PMTINQRS>.

Payment Modification handler

The Payment Modification Request <PMTMODRQ> is sent from the OFX client within BillerXpert to request modification of a payment from the VeriSign Payment Service's server. Payment Modification Response <PMTMODRS> is sent from the VeriSign Payment Service's server in response to a Payment Modification Request.

- Credit card

NOTE ACH is NOT supported for Payment Modification handler.

Reference <http://www.ofx.net/ofx/downloads/spec16/payments.html#9033> for specific assigned parameters for <PMTMODRQ> and <PMTMODRS>.

Payment Cancellation handler

A Payment Cancellation Request <PMTCANCRQ> is sent from the OFX client within BillerXpert to request a cancellation of a scheduled payment request from the VeriSign Payment Service's server. Payment Cancellation Response <PMTCANCRS> is sent from the VeriSign Payment Service's server in response to a Payment Cancellation Request.

- Credit card

NOTE ACH is NOT supported for Payment Modification handler.

Reference <http://www.ofx.net/ofx/downloads/spec16/payments.html#9033> for specific assigned parameters for <PMTCANCRQ> and <PMTCANCRS>.

Data Synchronization handler

A synchronization scheme insures that an accurate copy of the server data is maintained.

The interface between BillerXpert and VeriSign Payment Service's uses an "instance" temporarily stored database to expedite data retrieval. The database maintains a module capability and stores the last transaction ID.

Payment Synchronization Request <PMTSYNCRQ> is sent from the OFX client within BillerXpert to request a synchronization of data from the VeriSign Payment Services database. Payment Synchronization Response <PMTSYNCRS> is sent from the VeriSign Payment Services database in response to a Data Synchronization Request.

Reference <http://www.ofx.net/ofx/downloads/spec16/payments.html#9033> for <PMTSYNCRQ> and <PMTSYNCRS>.

Activating VeriSign

To activate VeriSign payment processing in BillerXpert, the biller simply needs to set up a VeriSign account, create a VeriSign oracle user, and install the BillerXpert support modules during the installation process.

Before downloading VeriSign Payment Services software, you have to register with VeriSign Payment Services as a vendor. If you are already a vendor, skip the following setup and go to downloading instructions.

Once you are a registered Biller you will be given a userid and password.

Becoming a Vendor

1. VeriSign Payment Services can be accessed at <http://www.verisign.com/payment/>
2. Choose the sign up tab and then choose either Try it! (if you are testing) or Buy it! (if you are buying) VeriSign Payment Services.
3. Clicking on Try it brings the software selection screen. Select PayFlow Pro.
4. This will bring you to the VeriSign Registration screen.
5. Enter the Vendor ID that you want and enter the password twice.
6. Enter the required information in the fields provided.
7. Select FDMS Nashville or Nova for Credit Card processing.

For ACG processing, contact VeriSign sales (sales@signio.com) to sign up.

Downloading

1. Connect to <https://manager.signio.com/>
2. Enter your Vendor ID and Password.
3. Select Downloads.
4. Follow the instructions for downloading software. You might have to send email to downloads@signio.com before you will be able to download.

5. Once you are authorized, go back to `manager.signio.com` and select the appropriate software to download (depending on your architecture and operating system).
6. For SunSparc, the download file is `pfpro_sunsparc.tar`.
7. Extract the software using the following command:


```
>tar - xvf pfpro_sunsparc.tar
```
8. This will create a directory structure under VeriSign Payment Services.
9. The VeriSign Payment Services directory contains the *PayFlow Pro guide*. It also contains the directory for the architecture. Under this directory, you will find the software.
10. The bin directory contains the *PayFlow Pro client* which allows you to test payment transactions from the command line.
11. Read the `readme.txt` file in the directory for test instructions. You will have to put the VeriSign lib directory in the library path in order for the tests to work properly (environment variable `LD_LIBRARY_PATH`). It is recommended that you copy the `libpfpro.so` file to `/usr/lib` as to avoid having to change your library path setup or change your `LD_LIBRARY_PATH` variable.

BillerXpert Setup

In order to route Payment transactions under the adopted model, BillerXpert's OFX client needs to be pointed to BillerXpert's internal OFX Server. The setup is done through the Billing Services provider (BSP) administration screen. The procedure consists of:

- Login into BillerXpert's Admin console
- Edit the Biller profile
- Add the internal OFX URL

Login to Admin Console

1. Enter the URL of the Administrative Interface Website, *http://URL:port#/admin*
For example: `http://plantation.mcom.com:8080/admin`
2. The Login window appears.

3. Enter `BSP` in the **Biller** field.
4. Enter the `user name` (default name - `bxadmin`) in the **user name** field.
5. Enter the `password` (default value - `bxadmin`) in the **Password** field.
6. Click the **Login** button.

Editing The Biller Profile

1. Click the **Billers** button from the main **Admin** menu to display the **Biller** window.
2. Select the **Biller** from the **Biller** selection box.
For example: *California Telecom*
3. Select the **Retrieve Billers** button.
4. Select the **Biller** to edit from the *Billers that meet the above search criteria* window.
For example: *California Telecom*
5. The **Biller Profile** window should appear.

OFX URL

1. In the **Biller Profile** window, select the **Update Profile** button.
2. The **Biller Profile** edit window appears. Scroll down to the *URL to send OFX messages* selection box.
3. Enter the URL as *http://URL:port#/NASApp/billxpert/OFXServlet*
For example:
`http://plantation:8080/NASApp/billxpert/OFXServlet`
4. Scroll down to the bottom of the **Biller Profile** edit window and select **Update Biller Profile**.
5. The **Biller Name** was successfully updated message should be displayed.
For example: *California Telecom was successfully updated*
6. Select the **Logout** button to exit the **Biller Profile** window.

Payment Testing

Once BillerXpert has been installed with the VeriSign support modules and configured, the simplest method to check functionality is to make a test payment from BillerXpert's sample user (*bambam*) account. The procedure is:

- Login to BillerXpert
- Select the Pay Balance link
- Make a test payment and verify payment acceptance
- Verify payment is shown in the Payment link under completed payments

Login into BillerXpert

1. Type the Biller URL address (`http://biller_name:port#/telco`)
For example: `http://plantation.mcom.com:8080/telco`
2. Type the test account name and password which is *bambam*

Pay Balance

1. Select the Sign On button to log in. Soon after, the Billing Center screen will be displayed.
2. Select the Pay Balance link and the Pay screen will be displayed.

Test Payment

Select the Make Payment button and the payment request will be sent out to VeriSign Payment services.

After the payment request is received, the VeriSign payment services responds with an acknowledgement message and BillerXpert displays the Payment confirmation.

Payment Completed

Once the payment transaction is completed, the *Pay Balance* screen is updated and the *Payments* page is updated reflecting the payment has been submitted.

See Also

For further details regarding VeriSign Payment Services, please visit www.verisign.com.

Clear Commerce Setup

Packaging

The ClearCommerce Engine for BillerXpert integration consists of 3 pieces. The Jlink Integration API, packaged in the `ccc_ssl.jar` file. The Biller Expert API Integration code, packaged in the `.jar` file. and a configuration file to specify the customization and installation specific parameters to the integration code. All three files must be installed into the system somewhere in the CLASSPATH of the JVM that is running the BillerXpert code.

Payee Configuration on the ClearCommerce Hosting Engine

In order for a credit card transaction to be processed on behalf of a payee on the BillerXpert system, an account for that merchant must be created and configured on the ClearCommerce Hosting Engine. For details of account creation and configuration see the manual *ClearCommerce Engines for Unix 3.8.2*.

BillerXpert supports a 'pay-some' model. 'Pay-Some' systems restrict users to only make payments to payees that appear on an approved list. A unique Payee ID exists for each approved payee on the BillerXpert system. This ID is used to identify the merchant on the ClearCommerce engine and must correspond to the store name or config file name on the ClearCommerce engine. BillerXpert also allows the payee to be specified by name. In that case the store name is mapped to the payee ID by means of properties in the `ClearCommerce.properties` file.

Authentication

Each payee supported by the ClearCommerce Engine requires a digital certificate pair for authentication by the ClearCommerce engine. Refer to the *ClearLink (Jlink) manual* and release notes for information on converting the digital certificate from .PEM format to .DER format. The certificate can be obtained from ClearCommerce. These certificates are stored on the BillerXpert system under a directory identified by the ClearCommerce property files.

In order to support the queries to the database required to implement the Payment Inquiry and Payment Synchronization messages, direct access to the payment-processing database is required. A single database userID/password pair grants this access. This Id, which will be created by the database administrator, should only have read access to the transaction and batch_transaction tables of the ClearCommerce engine database. This integration relies on the Iplanet OFX layer to insure that entities making requests are authorized to view those particular transactions (based on server transaction id).

Setting Up A Payment Vendor

BillerXpert B2B Edition provides the ability to establish a specific payment vendor. In this section, you will learn how to establish ClearCommerce as a payment server.

ClearCommerce As A Payment Server

The following section outlines the steps to follow when setting up ClearCommerce as a payment server.

NOTE These settings are for the DEMO only.

Login to the application as the *Biller Admin*.

3. Click on the Profiles tab and select Update Biller Profile.
4. In the *Payment Vendor Attributes* section, select Payment Vendor as ClearCommerce

In the service URL type:

`http://your-server:port/NASApp/billxb2b/CCOFXServer`

5. The userid is `noel`
The password is `netscape`
6. Save the settings and exit.
ClearCommerce is not setup as a payment server.

See Also

For further information regarding ClearLink by ClearCommerce, please refer to the *ClearLink (JLink) documentation*. For information regarding the new XML Payment specification for future use with Identrus, please refer to *Appendix D* of this document.

Currency Feed

This appendix discusses the following topics:

- Implementation Strategies
- Currency Conversion Process
- Installation Process for Currency Converter
- Installation of Reuters Investor

Implementation Strategies

With BillerXpert, you can integrate Reuters Investor using the Server-side XML method as described in the following section.

Server-side method – XML

Here, your webserver requests each item of data from the Reuters Investor servers in XML format, which your scripts must parse and format for display in pages or applications (for example, a ticker or charting applet). The user has no direct contact with the Reuters Investor servers.

This method gives you much more flexibility in how you format and display the information that you retrieve, and consequently requires greater programming effort. Note that any pages or applications that include Reuters information must adhere to the Reuters online branding guidelines (see the section on Reuters online branding guidelines within the *Reuters Investor Programmer's Guide* for details).

All your requests for Reuters data must be built dynamically (using a server extension technology). This enables you to include a valid Session ID, so that each request can be authenticated.

Note that if you are caching any Reuters Investor data in your servers, your company is obliged to measure and report the number of Reuters Investor data items delivered to your users. (For more details, see the section on Data charges and caching within the *Reuters Investor Programmer's Guide*).

See Also

The following related documents from Reuters:

- Reuters Investor Access Details
This provides the relevant details that enable your website to connect to the Reuters Investor servers. It includes your user ID and password, the Reuters Investor server addresses, and the function names (which are different from the names in the *Functions* section of this guide).
- Reuters Investor Data Guides
These contain detailed information about the data that is provided by Reuters Investor. This includes the key RICs (Reuters Instrument Codes) and news codes for each market, exchange lists, field explanations, and lists of exchange and currency identifiers.
- Reuters Online Branding Guidelines
This describes the Reuters branding guidelines that you must implement in your website if you are not only using the default HTML format pages.

Currency Conversion Process

The currency conversion process is as follows:

1. Currency codes and related information are loaded at the time of installation. These codes are International Organization for Standardization (ISO) standard and should not change. The Euro currencies are identified through the use of a special flag.
2. The latest currency rates are loaded into the database using a loader program. This program reads the data in XML format from a file and loads the rate table. Each load is tracked on the basis of conversion date. Each load is tracked in the database by a sequentially generated id, fileid. Please refer to the database schema for more details.
3. When the conversion method is called, it retrieves the appropriate fileid by using the date specified to find the latest track for that date. This search is on the basis of conversion date (not the load date).

4. Once the appropriate load is determined, the required rate is searched from the currency conversion table, using the fileid retrieved in step 3.
5. If the currency is a Euro currency, the rate between the other currency and Euro is retrieved. This rate is then used to convert the amount from one currency to another. In case of Euro currencies, the conversion always passes through Euro, i.e. Euro currency -> Euro -> other currency. Please refer to the official site
http://europa.eu.int/eur-lex/en/lif/dat/1997/en_397R1103.html for conversion rules regarding the EURO.

Installation Process for Currency Converter

The currency conversion module is installed automatically as part of the iPlanet BillerXpert installation. The following tasks are performed as part of the BillerXpert installation, which are specific to the currency converter module:

1. Three tables are created in the BSP schema. They are:
 - currency
 - currency conversion
 - currencyloadtrack.

Please refer to the database schema for more details.
2. The files `currconv.zip` and `xml.jar` are located under `$BX_HOME/java`.
3. The files `billxpert_env.csh` and `billxpert_env.sh` include `currconv.zip` and `xml.jar` in the CLASSPATH.

How to Load Currency Converter

1. Contact Reuters to get the licence
2. Configure `CCLoad.properties` file in `$BX_HOME/currconv` dir.
3. Configure variables in `$BX_HOME/getData.csh`
4. Run the script `$BX_HOME/getData.csh`
5. Check the logs. The Error logs are kept in `/tmp`. The file name is `CCLoadErr<date>_<time>.log`
 (e.g. `CCLoadErr20001128_235527.log`)

Configuring CCLoad Properties

`CCLoad.properties` file is read while pulling data from reuters. The following properties need to be configured:

1. `currConv.load.PullData.servername=LOCAL_FULLY_QUALIFIED_HOSTNAME_USED_TO_PULL_REUTERS_DATA`
 - sets the local server from where connection to reuters will be made. Typically this server gets data from external sites, if the application is hosted within a firewall, this would be the proxy server. (host.domain)
2. `currConv.load.PullData.serverPort=LOCAL_PORT`
 - The port number of the proxy server to connect to external sites (typically 80 or 8080)
3. `currConv.load.PullData.reutersHostname=REUTERS_HOSTNAME`
 - This is Reuters host name e.g. "ri2.rois.com"
4. `currConv.load.PullData.reutersPort=REUTERS_PORT`
 - http port on Reuters host (80)
5. `currConv.load.PullData.reutersSession=SESSION_ID_FROM_REUTERS`
 - session id string as given by Reuters
6. `currConv.load.PullData.reutersFormat=XML`
 - reuters data format (XML or HTML). This should remain as XML
7. `currConv.load.PullData.reutersRICs=REUTER_RICs_SPACE_SEPERATED`
 - Reuters RICS that need to be pulled (e.g. FX JPYX)
8. `currConv.load.PullData.outDir=DIR_TO_PULL_THE_RICs`
 - Output dir where the data will be pulled to. The pull program creates a directory.Pulldata under this directory
9. `currConv.load.CCLoad.CCFFile=${BX_HOME}/currconv/euro_fixed_rates.xml`
 - Allows you to specify an XML file that will be loaded with each load. The purpose is to load certain fixed currency rates (e.g. the European Union rates). It is recommended to load this file if conversion requires EURO currencies. Since these are fixed rates, Reuters feed will not have these rates.

Configuring getData.csh

getData.csh defines the following properties. It is recommended to use these values.

1. `setenv DATABASEPROFFILE $BX_HOME/currconv/database.properties`
The file contains the database id and password. This file is created at time of installation.
2. `setenv REUTERS_PROP $BX_HOME/currconv/CCLoad.properties`
Reuters properties file. See section on “*Configuring CCLoad.properties*” for more details.

Installation of Reuters Investor

To use Reuters Investor, your website and server software that access Reuters Investor must have access to the public internet. You can choose your own hardware and operating system, and you can choose which programming language to use. You do not need to install any Reuters hardware or software applications.

If you are using default HTML pages only, then Microsoft Internet Explorer v4.0 (or later) or Netscape v4.05 (or later) are recommended for users' browsers for displaying PNG format charts.

Customers interested in purchasing the Reuters Investor Product to enable Live currency updates with the iPlanet currency converter should contact their local Reuters sales representative at <http://www.reuters.com/onlinemedia>. Click on CONTACT US.

Enrollment

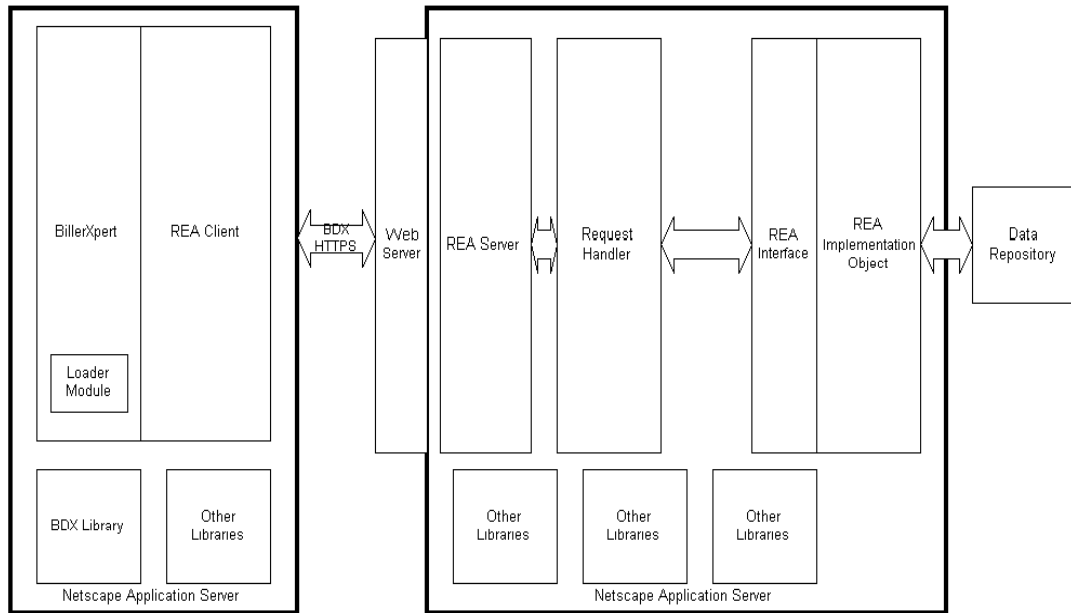
This appendix contains the following sections:

- Realtime Enrollment and Activation (REA)
- Installation, and Handler Procedures
- Testing REA

Realtime Enrollment and Activation (REA)

The Realtime Enrollment and Activation feature of BillerXpert makes it possible to immediately verify and activate a user's online request for membership and personal billing information. Figure D-1 illustrates the components of the REA process.

Figure D-1 Realtime Enrollment and Activation (REA) component diagram



The details of the REA component diagram are as follows:

- *REA Client* is closely integrated with the BillerXpert product. It is responsible for sending REA request to the proper REA Server and receiving responses from that server.
- *REA Server* is implemented as a server running on the iPlanet Application Server. The server accepts the REA requests from the client through the Web server. It is responsible for: accepting, validating, and verifying a request, handing over the request object to the request handler, and returning the response to the REA client.
- *Bill Data Exchange (BDX)* provides communication between the REA Client and Server. BDX protocol is similar to IFX (Internet Financial Exchange) and uses Biller XML as the data format for the response.
- *BDX Library* implements the necessary classes required to invoke the BDX protocol.
- *REA Interface* is used to plug objects into the database for each biller.

- *REA Implementation Objects* are responsible for verification of customer data. Objects can be in a different format of data and multiple objects can be active in the system, for the same biller.
- *Data Repository* can be any form of storing data, including, a database, LDAP, or a file system. BillerXpert maintains this data repository.

Request - Response Flow in the REA Framework

1. REA Client constructs the request using the input parameters provided by the customer during online registration. The input parameters are stored in `/tmp/BDX_*.xml` after enrollment.
2. REA Client then sends the request to the REA Server.
3. REA Server validates the request using the SignOn object. Validation comes from initializing the logging information (request type, source of the request, time or arrival, and so on) and hands over the request to the Request Handler.
4. The Request Handler, depending upon the BillerID in the request, calls the specific REA Implementation Object for this request.
5. The REA Implementation Objects (implementing the IG Interface) validates the user's credential against the data stored in the Data Repository.

In the case of a valid user, the data from the repository is retrieved, and the Data Transformation Library is used to transform the data from the legacy format to the Biller XML format. The data is then returned to the Request Handler.

In case of validation failure, the failure code and failure message are returned to the Request Handler.

6. The Request Handler passes the returned data from the REA Implementation Object over to the REA Server.
7. The REA Server updates the bookkeeping variables, logs, and then returns the response back to the REA Client.
8. The REA Client passes the response to the Response Handler.
9. The Response Handler processes the response using the Data Loader Module that stores the data in the Biller Database or LDAP.

REA Implementation Methods

The REA server is installed at the biller site. An Interface is exposed, whose methods are called internally by the server to send the response back. The Biller or integrator has to implement these methods to get the required data from their legacy systems. The functions are:

1. `isValidUser()` which should validate the user and return the result.
2. `getCustomerProfile()` gets the customer profile of the user in XML format.
3. `get Customer Bill()` gets the latest bill of the specified customer in XML format.

Installation, and Handler Procedures

REA Server Installation Procedure

1. Put `$BX_HOME/java/bds.zip` in their server CLASSPATH
2. Register `netscape.bds.server.BDXServlet` with the Web Server
3. Copy `DBParam.conf` data in the ClassPath Directory

BillerXpert Installation

To install REA, use BillerXpert's installation script (`setup`). Once installed, access the REA Server files from `$BX_HOME/java` for the REA Server installation. After this data is gathered, the following scenarios could occur:

- *User information needs to be downloaded.* Due to the large number of customers, some Billers could be downloading data of only the enrolled users in the BillerXpert database. In such a scenario, the information is verified against the database and then relevant information is downloaded to the Biller database.
- *User information is incorrect.* The user is asked to re-enter the information and the process is repeated.

Handler Procedure

1. REA Client executes with the required parameters:

```
> java netscape.bds.handlers.BDXPresCustomerDataHandler
```

The Parameter details and values are shown below.

Parameter	Directory	Typical Value
Biller URL	URL of BDX Servlet part of bds.zip	http://<hostname>:<port>/NASA pp/billingdata/BDXServlet
First Name		John
Middle Name		Can be null
Last Name		Doe
Address 1		4210 Network Circle
Address 2		Santa Clara
Address 3		Can be null
City		Santa Clara
State		CA
Zip Code		94086
Country		USA
Day Phone		(408) 276-3400
Evening Phone		(408) 276-3400
Email		johndoe@netscape.com
Tax ID	Number used for validation purposes	123-45-6789
Social Security Number	Number used for validation purposes	123-45-6789
Date of Birth		00-00-0000
Account Number	Account number of user with Biller	650 789 000 ABC
Transaction ID	Unique 12 digit number	123456789012
Date	Transaction date	Today's date

NOTE Null parameters should be represented by “.”.

Example of a Request Response

The REA Client sends the request. After the REA Server processes the request, the REA Server responds with the customer and billing information as noted in the following:

```
Given URL in BDHandler is :
http://plantation:8080/NASApp/billingdata/BDXServlet

The Supplied URL Is :
http://plantation:8080/NASApp/billingdata/BDXServlet

After creating URL...

BDXClient SUCCESS

Writing success

SUCCESS BDHANDLER

SGML DATA: <PRESCUSTOMERDATATRNRQ>
    <TRNUID>100000000001</TRNUID>
    <DATE>07262000</DATE>
<PRESCUSTOMERDATARQ>
<CUSTOMERINFO>
    <FIRSTNAME>John</FIRSTNAME>
    <LASTNAME>Doe</LASTNAME>
    <MIDDLENAME></MIDDLENAME>
    <ADDR1>21, Network Circle</ADDR1>
    <ADDR2></ADDR2>
    <ADDR3></ADDR3>
    <CITY>Santa Clara</CITY>
    <STATE>CA</STATE>
    <POSTALCODE>94085</POSTALCODE>
    <COUNTRY>USA</COUNTRY>
    <DAYPHONE>408-276-4300</DAYPHONE>
```

```

<EVEPHONE>408-276-4300</EVEPHONE>
<EMAIL>john.doe@netscape.com</EMAIL>
<TAXID>123-12-1234</TAXID>
<SECURITYNAME>who</SECURITYNAME>
<DATEOFBIRTH>01/01/1970</DATEOFBIRTH>
<ACCOUNTNUMBER>650 937 6181 778 N20</ACCOUNTNUMBER>
</CUSTOMERINFO>
</PRESCUSTOMERDATARQ>
</PRESCUSTOMERDATATRNRQ>

```

The following response is displayed:

```

FileInputStream is not null
ARIF ACNTVParser obj has created
Request Sent
Status Line :HTTP/1.1 200 OK
Header1 Line :Netscape-Enterprise/4.0
Content-Type Line :text/html
Status Code : 200
Response File Name: BDX9646484233900
OUT Element Name : PRESCUSTOMERDATATRNRS
INSIDE PRESCUSTDATARS Element Name : PRESCUSTOMERDATATRNRS
OUT Element Name : PRESCUSTOMERDATATRNRS
INSIDE PRESCUSTDATARS Element Name : PRESCUSTOMERDATATRNRS
OUT Element Name : TRNUID
INSIDE PRESCUSTDATARS Element Name : TRNUID
OUT Element Name : STATUS
INSIDE PRESCUSTDATARS Element Name : STATUS
OUT Element Name : CODE
OUT Element Name : SEVERITY
OUT Element Name : PRESCUSTOMERDATARS

```

INSIDE PRESCUSTDATARS Element Name : PRESCUSTOMERDATARS
OUT Element Name : CUSTOMERBILLS
OUT Element Name : CUSTOMERBILL
OUT Element Name : BILLINXML
OUT Element Name : BillXpertDoc
OUT Element Name : Operation
OUT Element Name : BillerId
OUT Element Name : FileId
OUT Element Name : BXBillerCustomerAccount
OUT Element Name : BcaUserId
OUT Element Name : BcaAcctType
OUT Element Name : BcaCusType
OUT Element Name : BcaAcctNum
OUT Element Name : BcaDesc
OUT Element Name : BcaTaxId
OUT Element Name : BcaSince
OUT Element Name : BcaAcctStat
OUT Element Name : BcaUnPaidBill
OUT Element Name : BcaUnPaidBal
OUT Element Name : BcaAcctBal
OUT Element Name : BcaAcctBalDate
OUT Element Name : BcaDisStat
OUT Element Name : BcaBillAddr
OUT Element Name : BcaServAddr
OUT Element Name : BXContainer
OUT Element Name : BXDocument
OUT Element Name : BillType
OUT Element Name : DocType
OUT Element Name : DocCDate
OUT Element Name : DocBDate
OUT Element Name : DocStatus

OUT Element Name : DocCpType
OUT Element Name : BXSummary
OUT Element Name : StDate
OUT Element Name : EndDate
OUT Element Name : OrigAmt
OUT Element Name : CurBal
OUT Element Name : CurChg
OUT Element Name : PrevBal
OUT Element Name : PrevPmt
OUT Element Name : BalDate
OUT Element Name : DueDate
OUT Element Name : LateAfterDate
OUT Element Name : PenaltyType
OUT Element Name : PenaltyRate
OUT Element Name : MinPayAmt
OUT Element Name : PayAmt
OUT Element Name : LastPayDate
OUT Element Name : PayStatus
OUT Element Name : DisputeAmt
OUT Element Name : DisputeStatus
OUT Element Name : SUMStatus
OUT Element Name : DiscntType
OUT Element Name : DiscntRate
OUT Element Name : BXLineItem
OUT Element Name : BXLineItemType
OUT Element Name : TotalAmount
OUT Element Name : Description
OUT Element Name : CUSTOMERPROFILE
OUT Element Name : BILLINXML
OUT Element Name : BillXpertDoc
OUT Element Name : Operation

```
OUT Element Name : BillerId
OUT Element Name : OrgName
OUT Element Name : FileId
OUT Element Name : BXCUSTOMERPROFILE
OUT Element Name : bcpLogin
OUT Element Name : bcpAcct
OUT Element Name : bcpFname
OUT Element Name : bcpMname
OUT Element Name : bcpLname
OUT Element Name : bcpDphone
OUT Element Name : bcpNphone
OUT Element Name : bcpEmail
OUT Element Name : bcpSWord
OUT Element Name : bcpPwd
OUT Element Name : bcpAddr1
OUT Element Name : bcpAddr2
OUT Element Name : bcpAddr3
OUT Element Name : bcpCity
OUT Element Name : bcpState
OUT Element Name : bcpCountry
OUT Element Name : bcpPostal
OUT Element Name : bcpAcctType
Response Object: <PRESCUSTOMERDATATRNS>
    <TRNUID>10000000001</TRNUID>
<STATUS>    <CODE>0</CODE>
    <SEVERITY>INFO</SEVERITY>
</STATUS><PRESCUSTOMERDATARS>
<CUSTOMERBILLS>
<CUSTOMERBILL>
    <BILLINXML><BillXpertDoc>
    <Operation> UPDATE</Operation>
```

```

<BillerId> California Telecom</BillerId>
<FileId> 80016</FileId>
<BXBillerCustomerAccount>
  <BcaUserId> bambam</BcaUserId>
  <BcaAcctType> Regular Account</BcaAcctType>
  <BcaCusType> REGULAR</BcaCusType>
  <BcaAcctNum> 650 937 6181 778 N20</BcaAcctNum>
  <BcaDesc> bambam</BcaDesc>
  <BcaTaxId> 123-12-1234</BcaTaxId>
  <BcaSince> 1990</BcaSince>
  <BcaAcctStat> ACTIVE</BcaAcctStat>
  <BcaUnPaidBill> 1</BcaUnPaidBill>
  <BcaUnPaidBal> 80.11</BcaUnPaidBal>
  <BcaAcctBal> 80.11</BcaAcctBal>
  <BcaAcctBalDate> 2000-07-02</BcaAcctBalDate>
  <BcaDisStat> NODISPUTE</BcaDisStat>
  <BcaBillAddr> 19980728225960-BXADDRESS</BcaBillAddr>
  <BcaServAddr> 19980728225957-BXADDRESS</BcaServAddr>
  <BXContainer>
    <BXDocument>
      <BillType> TELCO</BillType>
      <DocType> BILL</DocType>
      <DocCDate> 2000-07-02</DocCDate>
      <DocBDate> 2000-01-24</DocBDate>
      <DocStatus> ACTIVE</DocStatus>
      <DocCpType> ORIGINAL</DocCpType>
      <BXSummary>
        <StDate> 1999-12-26</StDate>
        <EndDate> 2000-01-24</EndDate>
        <OrigAmt> 55.24</OrigAmt>
        <CurBal> 0.0</CurBal>
      </BXSummary>
    </BXDocument>
  </BXContainer>
</BXBillerCustomerAccount>

```

```

    <CurChg> 55.24</CurChg>
    <PrevBal> 45.34</PrevBal>
    <PrevPmt> 45.24</PrevPmt>
    <BalDate> 2000-01-24</BalDate>
    <DueDate> 2000-02-18</DueDate>
    <LateAfterDate> 2000-02-23</LateAfterDate>
    <PenaltyType> FIXED</PenaltyType>
    <PenaltyRate> 5.0</PenaltyRate>
    <MinPayAmt> 15.0</MinPayAmt>
    <PayAmt> 25.24</PayAmt>
    <LastPayDate> 2000-02-23</LastPayDate>
    <PayStatus> FULLY_PAID</PayStatus>
    <DisputeAmt> 0.0</DisputeAmt>
    <DisputeStatus> NODISPUTE</DisputeStatus>
    <SUMStatus> ACTIVE</SUMStatus>
    <DiscntType> FIXED</DiscntType>
    <DiscntRate> 0.0</DiscntRate></BXSummary>
  <BXLineItem>
    <BXLineItemType> BASIC</BXLineItemType>
    <TotalAmount> 13.5</TotalAmount>
    <Description> Basic
Service</Description></BXLineItem></BXDocument></BXContainer></BXBi
llerCustomerAccount></BillXpertDoc></BILLINXML>
</CUSTOMERBILL>
</CUSTOMERBILLS>
<CUSTOMERPROFILE>
  <BILLINXML><BillXpertDoc>
    <Operation> UPDATE</Operation>
    <BillerId> California Telecom</BillerId>
    <OrgName> iPlanet</OrgName>
    <FileId> 87845</FileId>
  </BILLINXML>
</CUSTOMERPROFILE>

```

```

<BXCustomerProfile>
  <bcpLogin> bambam</bcpLogin>
  <bcpAcct> 1001</bcpAcct>
  <bcpFname> John</bcpFname>
  <bcpMname></bcpMname>
  <bcpLname> Doe</bcpLname>
  <bcpDphone> 408-276-4300</bcpDphone>
  <bcpNphone> 408-276-4300</bcpNphone>
  <bcpEmail> john.doe@netscape.com</bcpEmail>
  <bcpSWord> who</bcpSWord>
  <bcpPwd> bambam</bcpPwd>
  <bcpAddr1> 21, Network Circle</bcpAddr1>
  <bcpAddr2></bcpAddr2>
  <bcpAddr3></bcpAddr3>
  <bcpCity> Santa Clara</bcpCity>
  <bcpState> CA</bcpState>
  <bcpCountry> USA</bcpCountry>
  <bcpPostal> 94085</bcpPostal>
  <bcpAcctType> Regular
Account</bcpAcctType></BXCustomerProfile></BillXpertDoc></BILLINXML
>
</CUSTOMERPROFILE>
</PRESCUSTOMERDATARS>
</PRESCUSTOMERDATATRNRS>

```

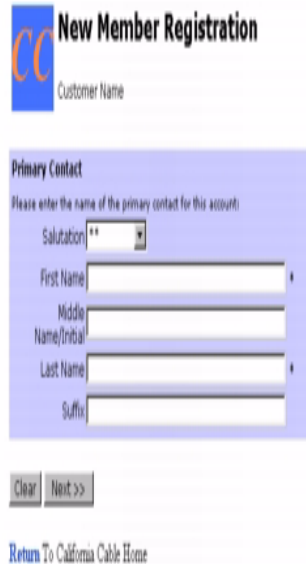
Testing REA

Enrolling a new member is an easy process and is virtually instantaneous, using the new Realtime Enrollment and Activation (REA) feature of BillerXpert.

To enroll a new member

1. Go to your company home page.

Figure D-2 Example of a company home page



The screenshot shows a web form titled "New Member Registration" with a logo on the left. Below the title is a "Customer Name" input field. A section titled "Primary Contact" contains a prompt: "Please enter the name of the primary contact for this account:". This section includes a "Salutation" dropdown menu, and four text input fields for "First Name", "Middle Name/Initial", "Last Name", and "Suffix". At the bottom of the form are "Clear" and "Next >>" buttons, and a link that says "Return To California Cable Home".

2. At the New Online Member area of your home page, click the Sign Up Button.
The New Member Registration page is displayed as illustrated in the following figure.

Figure D-3 New Member Registration

New Member Registration
Payment Information

Primary Payment Method
If you do not plan to pay bills online or would not like to save your payment info, return and pay this way.

Otherwise please enter the primary method to pay your bills:

Bank Account Checking Saving

Bank Routing ID *

Account Number *

Nickname

Credit Card How you would like to refer to this card on the site.

Cardholder Name * As it appears on the card

Type AMEX *

Number

Expiration Date January 2009

Card Address Check "Same as Billing Address" if the address for this card is the same as your billing address otherwise please enter the address for the card below.

Same as Billing Address

Address *

City *

State AK * Zip Code *

Card Nickname How you would like to refer to this card on the site.

[Return To California Cable Home](#)

Primary Contact

Name of field/object	Explanation of field/object
Salutation	Title of primary contact
First Name	First name of primary contact
Middle Name/Initial	Middle name or initial of primary contact
Last Name	Last name of primary contact
Suffix	Suffix or additional title of primary contact
Clear	Clears the input fields for new member registration
Next	Go to the next window
Return	Return to the previous window

3. Enter appropriate data into the input fields to complete your registration.

NOTE: All fields with an asterisk * on them are required entry fields.

4. Click Clear to clear the data incorrectly entered in the input fields or Next to proceed to the next window, as displayed in the following figure.

Figure D-4 New Member Registration - Page 2

New Member Registration
Name and Address

Billing Address
Please enter the address that bills should be sent to:

Address

City

State * Zip Code *

[Return To California Cable Home](#)

Line Account Numbers

Name of field/object	Explanation of field/object
Account Number	Main telephone number and area code
Last Month Total	Total amount of last month's bill (for security purposes)
Account Number	If you have other telephone numbers for which you are billed.
Last Month Total	See previous field, if other telephone numbers apply
Account Number	If second telephone number, enter that phone
Last Month Total	If second telephone number, enter that total
Clear	Clears the input fields for new member registration
Next	Go to the next window
Return	Return to the previous window

5. Enter appropriate data into the input fields to complete your registration.

NOTE: All fields with an asterisk * on them are required entry fields.

- Click Clear to clear the data incorrectly entered in the input fields or Next to proceed to the next window, as displayed in the following figure.

Figure D-5 New Member Registration - Page 3

New Member Registration
Name and Address

Billing Address
Please enter the address that bills should be sent to:

Address *

City *

State * Zip Code *

[Return To California Cable Home](#)

Primary Contact

Name of field/object	Explanation of field/object
Address	Enter address where bills should be sent.
Address	
City	Enter the city where bills should be sent.
State	Enter the state where bills should be sent.
Zip Code	Enter the zip code of the state where bills are sent.
Clear	Clears the input fields for new member registration
Next	Go to the next window
Return	Return to the previous window

- Enter appropriate data into the input fields to complete your registration.

NOTE: All fields with an asterisk * on them are required entry fields.

- Click Clear to clear the data incorrectly entered in the input fields or Next to proceed to the next window, as displayed in the following figure.

Figure D-6 New Member Registration - Page 4

New Member Registration
Payment Information

Primary Payment Method

If you do not plan to pay bills online or would not like to save your payment info online you can skip this step.

Otherwise please enter the primary method to pay your bills:

Bank Account

Checking Saving

Bank Routing ID *

Account Number *

Nickname

How you would like to refer to this card on the site.

Credit Card

Cardholder Name *

As it appears on the card

Type *

Number *

Expiration Date * *

Card Address Check "Same as Billing Address" if the address for this card is the same as your billing address otherwise please enter the address for the card below.

Same as Billing Address

Address *

City *

State * Zip Code *

Card Nickname

How you would like to refer to this card on the site.

[Return](#) To California Cable Home

Primary Payment Method

Name of field/object	Explanation of field/object
Bank Routing ID	Identification number of the bank account used
Account Number	Your bank account number
Nickname	Name (you would prefer to use) for this card
Cardholder Name	Name of cardholder
Type	Type of card
Number	Number of card
Expiration Date	Expiration date of card
Card Address	Address of cardholder
City	City of cardholder

Name of field/object	Explanation of field/object
State	State of cardholder
Zip Code	Zip code of state of cardholder
Card Nickname	Account reference for this card
Clear	Clears the input fields for new member registration
Next	Go to the next window
Return	Return to the previous window

9. Enter appropriate data into the input fields to complete your registration.

NOTE: All fields with an asterisk * on them are required entry fields.

10. Click Clear to clear the data incorrectly entered in the input fields or Next to proceed to the next window, as displayed in the following figure.

Figure D-7 New Member Registration - Page 5

Billing Address

Name of field/object	Explanation of field/object
Address	Address where bills are sent
Address	See above address field
City	City where bills are sent.
State	State where bills are sent

Name of field/object	Explanation of field/object
Zip Code	Zip code of state, where bills are sent
Clear	Clears the input fields for new member registration
Next	Go to the next window
Return	Return to the previous window

- Enter appropriate data into the input fields to complete your registration.
NOTE: All fields with an asterisk * on them are required entry fields.
- Click Clear to clear the data incorrectly entered in the input fields or Next to proceed to the next window, as displayed in the following figure.

Figure D-8 New Members Registration - Page 6

New Member Registration
Customer Profile

Additional Information
The following is optional information so that we may better serve you:

Gender: Female
Age: <18
Profession: HiTech
Salary: \$25K - \$35K
Interests: Outdoor Sport
Vehicle Preference: Coupe

Clear Submit

[Return](#) To California Telecom Home

Additional Information

Name of field/object	Explanation of field/object
Gender	Select from dropdown list box
Age	Select from dropdown list box
Profession	Select from dropdown list box
Interests	Select from dropdown listbox
Vehicle Preference	Select from dropdown listbox

Name of field/object	Explanation of field/object
Clear	Clears the input fields for new member registration
Next	Go to the next window
Submit	Submits data to the IG database

13. Enter appropriate data into the input fields to complete your registration.

NOTE: All fields with an asterisk * on them are required entry fields.

14. Click Clear to clear the data incorrectly entered in the input fields or Submit to proceed and submit the data to the IG database.

Verification

Once REA processes the request and response, the user will be able to login to BillerXpert and see their account and billing information

Testing REA

XML Payment Specification

BillerXpert B2B Edition includes a payment XML message support structure for future vendors. This new payment specification will be a necessary protocol when interfacing with any financial institution.

The original banking language used for transactions among financial institutions is Open Financial Exchange (OFX), which is being phased out. Currently, Interactive Financial Exchange (IFX) is being used, which is an XML-based OFX transaction. In the near future, the banking language of choice will be XML.

There are three message types included in the XML Payment Specification that will be detailed in this appendix:

- PaymentRequest
- PaymentCancellationRequest
- ResponseMessage

This appendix also describes the following:

- Legend of Message Data Requirement Codes

PaymentRequest

The PaymentRequest message is the first message type in the XML Payment specification.

PaymentRequest DTD

Table E-1 outlines the components of the PaymentRequest.

Table E-1 PaymentRequest DTD

Contains	Relationship Description
Header	A standard header which contains the Product and Message Type.
CustomerSignedData	The data signed by the Buyer is contained in a single DTD in the PayRequest block.
CustomerSignatures	The CustomerSignatures block carries one or a number of CustomerSignature from the Buyer to the Buyer's Bank. For ECIPS only a single signature is permitted by the DTD.
SellerBankData	SellerBankData block contains data elements provided by the Seller's Bank to the Buyer's Bank. For ECIPS this only includes the SellerBankReference.
Header	A standard header which contains the Product and Message Type.

Fully Expanded PaymentRequest

Code Example E-1 is a sample of a fully expanded PaymentRequest.

Code Example E-1 Fully Expanded PaymentRequest

```

PaymentRequest
  L2PaymentRequest
    Header
      Product Enum (xPx , xPC)
      MessageType Enum (PaymentRequest,PaymentCancellationRequest,
ResponseMessage)
      Version Decimal
      CustomerSignedData
      NegotiatedData
      Amt Decimal
      CurCode NC-3
      RequestedValueDate Date
      BuyerPublicData
      Contact
        OrganisationName C-96
        OrganisationUnitName C-96
        OrganisationAddress
          Addr1 C-64

```

```

        Addr2 ? C-64
        Addr3 ? C-64
        Addr4 ? C-64
        City ? C-32
        StateProv ? C-32
        PostalCode ? C-11
        Country ? NC-3
    BuyerReference ? NC-36
    BuyerBankCode ? NC-35
    BuyerAccount ? NC-35
    Priority ? Fixed ("URGENT")
SellerPublicData
    Contact
    OrganisationName C-96
    OrganisationUnitName C-96
    OrganisationAddress
        Addr1 C-64
        Addr2 ? C-64
        Addr3 ? C-64
        Addr4 ? C-64
        City ? C-32
        StateProv ? C-32
        PostalCode ? C-11
        Country ? NC-3
    TransactionReference NC-36
    SellerReference ? NC-36
    SellerName C-96
    SellerBankCode NC-35
    SellerAccount NC-35
    PaymentDetails ? NC-35X4
    Obligation Enum("BANKCERTIFIED" , "NONE")

```

PaymentCancellationRequest

The PaymentCancellationRequest message is the second message type in the XML Payment specification.

PaymentCancellationRequest DTD

Table E-2 outlines the components of the PaymentCancellationRequest.

Table E-2 PaymentCancellationRequest DTD

Component	Description
Header	A standard header which contains the Product and Message Type.
CancCustomerSignedData	The CancCustomerSignedData contains the data signed by the Buyer to authorize cancellation.
CustomerSignatures	The CustomerSignatures block carries one or a number of CustomerSignature from the Buyer to the Buyer's Bank. For ECIPS only a single signature is permitted by the DTD.
SellerBankData	SellerBankData block contains the signature authorizing cancellation of the payment. For ECIPS only a single signature is permitted by the DTD.

Fully Expanded PaymentCancellationRequest

Code Example E-2 is a fully expanded PaymentCancellationRequest.

Code Example E-2 Fully Expanded PaymentCancellationRequest

```

PaymentCancellationRequest
  L2PaymentCancellationRequest
    Header
      Product Enum (xPx , xPC)
      MessageType Enum
      (PaymentRequest, PaymentCancellationRequest, ResponseMessage)
      Version Decimal
      CancCustomerSignedData
      References
        TransactionReference NC-36
        (SellerBankReference | NC-36
        BuyerBankReference) NC-36
        BuyerReference ? NC-36
        SellerReference ? NC-36

```

ResponseMessage

The ResponseMessage is the third message type of the XML Payment specification. This section defines a generic response message used to respond to all ECIPS Request and Notification messages as described in this chapter. ResponseMessage DTD

Table E-3 outlines the components of the ResponseMessage.

Table E-3 Response Message DTD

Contains	Relationship Description
Header	A Header which contains the product and message type for the document
References	The reference block contains the reference being used by the various parties in the transaction. The reference block includes the TransactionReference.
Response Type	ResponseType defines the type of response that is being provided. Response type can be: <ul style="list-style-type: none"> • PayInst - for responding to a non-assured Model A payment. • Obligation - for responding to an assured Model A payment. • Cancellation - for responding to a PaymentCancellationRequest. • Notification - for responding to a PaymentCancellationRequestNotification or a PaymentRequestNotification.
Status	The status of the response. The status is either SUCCESS or FAILURE.
Reason Code	Reason Code, indicating either the reason for failure, or indicating success.
Reason Text	Text associated with the reason code above.

Fully Expanded ResponseMessage

Code Example E-3 is a fully expanded ResponseMessage.

Code Example E-3 Fully Expanded ResponseMessage

```

ResponseMessage
  L2ResponseMessage
    Header
      Product Enum (xPx , xPC)
      MessageType Enum (PaymentRequest,
PaymentCancellationRequest, ResponseMessage)
      Version Decimal
    References
      TransactionReference NC-36
      (SellerBankReference | NC-36
BuyerBankReference) NC-36
      BuyerReference ? NC-36
      SellerReference ? NC-36
    ResponseType Enum (PayInst Obligation Notification
Cancellation)
      Status Enum (SUCCESS , FAILURE)
      ReasonCode NC-6
      ReasonText NC-100
    
```

Legend of Message Data Requirement Codes

Table E-4 describes each of the codes used to identify the data requirements of each of the three message types described in this section.

Table E-4 Legend of Message Data Requirement Codes

Code	Description
NC-3	Any alphanumeric string with maximum length 3.
NC-6	Any alphanumeric string with maximum length 6.
NC-35	Any alphanumeric string with maximum length 35.
NC-35X4	Any alphanumeric string with maximum length 35x4.
NC-36	Any alphanumeric string with maximum length 36.
NC-100	Any alphanumeric string with maximum length 100.
C-11	Any string with a maximum length of 11.
C-32	Any string with a maximum length of 32.
C-64	Any string with a maximum length of 64.
C-96	Any string with a maximum length of 96.