Oracle® Retail Xstore Payment (Xpay)

User Guide
Release 2.0
E67429-02

July 2018

ORACLE®

Oracle® Retail Xstore Payment (Xpay), User Guide, Release 2.0

C O N T E N T S

GETTING STARTED

> *The rebranding for the latest version of this documentation set is in development as part of post MICROS acquisition activities. References to former MICROS product names may exist throughout this existing documentation set.*

Overview

Xpay processes Point of Sale (POS) authorization requests. A variety of request messages are processed, including:

- Credit card requests
    - All major credit cards (Visa, MasterCard, Amex, Discover, etc.)
    - Private label cards
- Debit card requests
- Check requests
- Gift card request
    - Gift card activations
    - Gift card redemptions
    - Gift card recharges
    - Gift card balance inquiries
    - Cash out transactions
    - All void transactions (activation void, redemption void, etc.)
- Instant credit applications

Xpay has two components: the Xpay authorization server and the Xpay GUI. The Xpay server is responsible for communicating with the POS and the bank processors to complete

authorizations requests. The Xpay GUI reports authorization data to the user for the purpose of statistical analysis and for troubleshooting.

The Xpay server software runs as a service and routes the information to the bank or processor. Responses follow the same path back. Xpay supports both store-based or centralized authorization.

- ❑ **Store-based** - There are two common implementations of store-based authorizations:
  - ❍ *dialup line -* to the bank used generally when the central server is unreachable or down.
  - ❍ *local Xpay server* - directs Xpay authorizations to the bank (usually over a WAN link).

- ❑ **Centralized** - Xpay server software at a central location communicates directly with a *central payment server*, usually through a secure private line such as a frame relay connection.

The Xpay home page displays a summary of the day's Xpay activity.
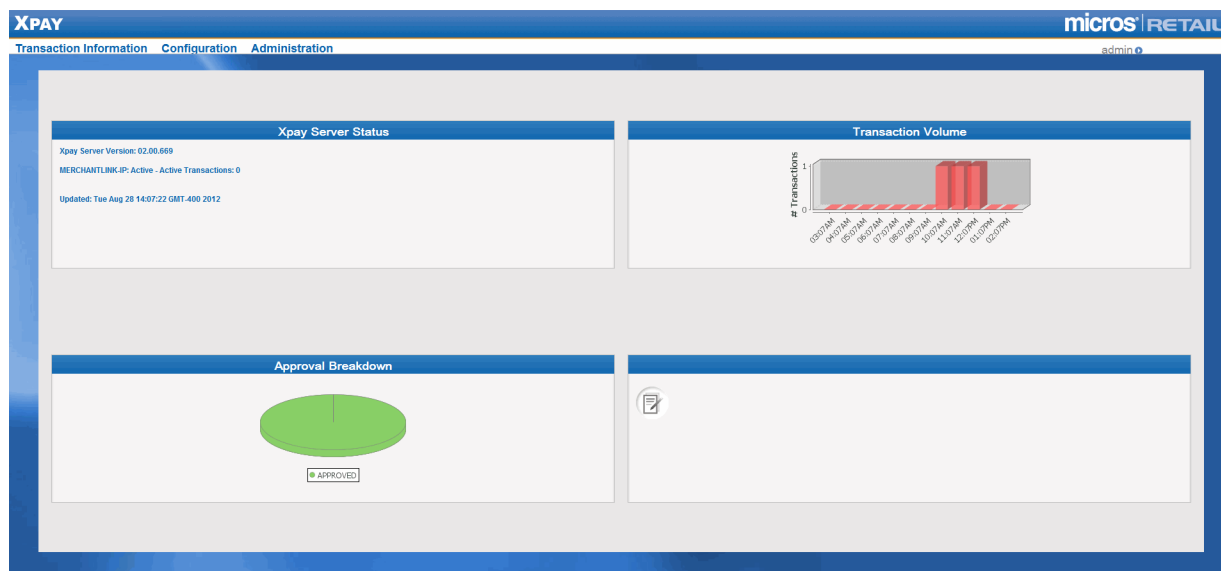
Figure 1-1: Xpay Home Page

About the Xpay Home Page

Xpay Server Status - Provides information about the version of Xpay that is running, as well as all of the enabled processors and the status of connectivity to those processors.

Transaction Volume - Shows the transaction volume for the last eight hours.

Approval Breakdown - Shows the number of transactions for each type of transaction status - DECLINED, APPROVED, or INCOMPLETE.

Transaction Failure Breakdown - Shows a breakdown of the number of transaction failures.

Menu Options

The menu options vary with the user's permissions and whether or not you use Xpay database logging.

Logging Enabled

If an Xpay database is set up and logging is used, users with *Admin* privileges can manage users, configure processors, and view transaction and system logs. All menu options *will be* available. Users with *Non-admin* privileges can view Transaction Information and System Logs (under the Administration menu).
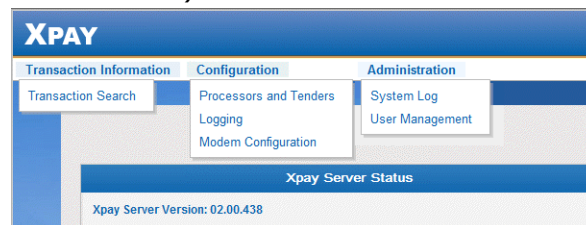
Figure 1-2: Menu Options If Logging is Used

Logging Disabled

If database logging is disabled, an *Admin* user can only see the Configuration menu and the Administration menu. The Transaction Information menu and System Log information (under the Administration menu) *will not* be available without database logging. If database logging is disabled, there are no options available for users with *Non-admin* privileges.
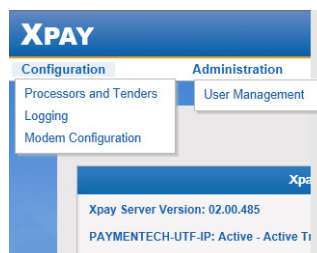
Figure 1-3: Menu Options For Disabled Logging

Xpay

CHAPTER 2

XPAY DATABASE SETUP FOR LOGGING

Overview

	Although Xpay's database logging is optional, MICROS-Retail recommends it. If Xpay is installed **without** database logging enabled, the application will be installed without any issues; however, the Transaction menu and the System Log submenu found under the Administration menu **will not** be available.

If you want to **enable** database logging, use the following instructions to create an empty database and a database user account with complete database access prior to installing Xpay.

If you want to **disable** database logging proceed to "Xpay Installation" on page 11

Creating a Database

There are two options for creating a database in SQL Server:

❑   Create a database in SQL Server: create database **your database name**

<OR>

❑   Create a New Database in the SQL Server Management Studio.

SQL Server Management Studio Instructions

1. Right-click the **Databases** node in the SQL Server Management Studio and select **New Database**.
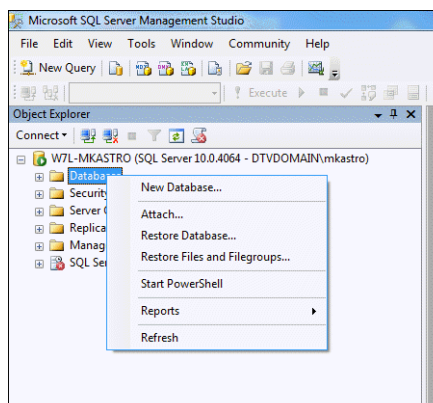
Figure 2-1: Select New Database

2. In the **Database name** field, enter the name of the database (***Xpay is recommended***) and click **OK**.
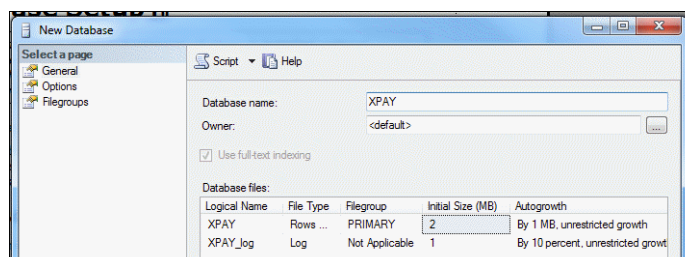
Figure 2-2: New Database Name Window

Create a new security login and user mapping

3. Expand the **Security** node.

4. Right-click **Logins** and select **New Login**.
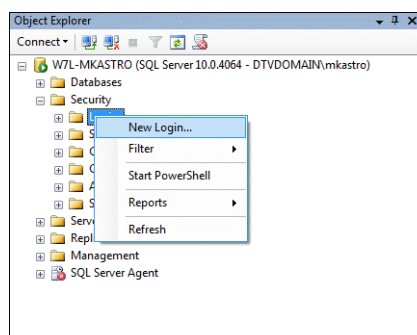
Figure 2-3: Navigation Menu

5. At the security Login-New window, **General** page define the following:

> You may wish to make a note of this information. The Login name and password will be needed during the Installation Process (see *Chapter 3*).

Table 2-1: *General Page Fields*

Field	Setting
Login name	Enter a name such as XpayAdmin01.
SQL Server authentication	Select the radio button.
Password	Enter a password.
Confirm password	Enter the password again to confirm.
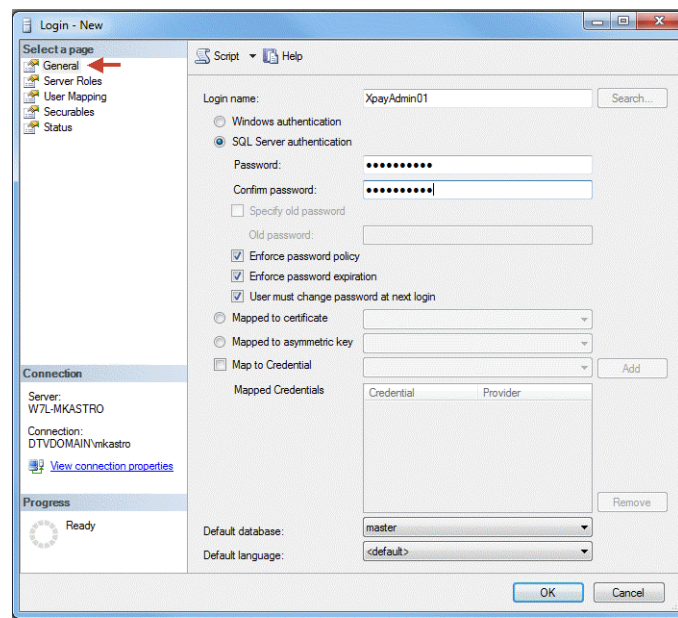
Figure 2-4: Login-New Window, General Page

   a. Select the **User Mapping** page and define the following:

Table 2-2: *User Mapping Fields*

Field	Setting
Database	Check mark the database name you set up in <u>step 2 on page 6</u>.
Database role membership	Check mark the following database roles: ❑ db_datareader ❑ db_datawriter ❑ public (selected by default)
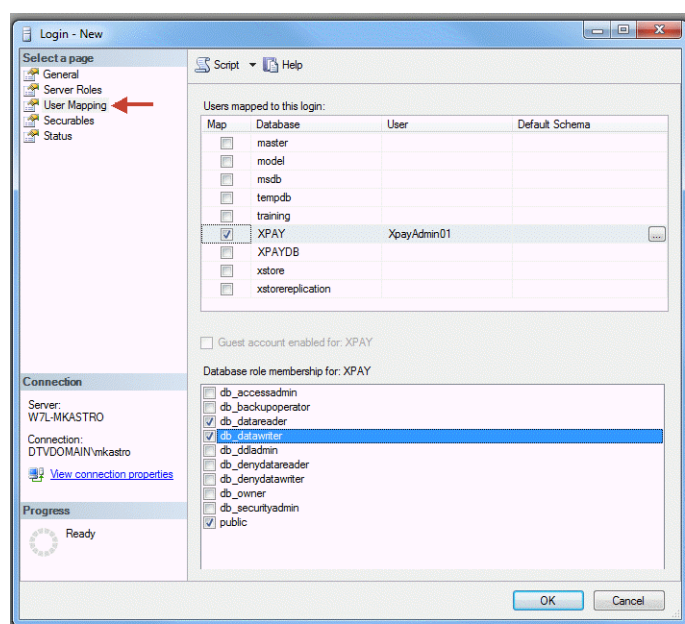
Figure 2-5: Login-New Window, User Mapping Page

6. Click **OK**.

7. Follow the Xpay installation process <u>"Installing Xpay" on page 12</u>.

After Installing Or Upgrading Xpay

1. Open MS SQL Management studio and connect to Xpay's database server.

2. Open a new query window.

3. Apply the following database scripts to the database. The scripts are found in `C:\xpay\dbscripts\mssql` if the default location was used during the install.

Table 2-3: *Apply Database Scripts*

If installing, then apply:	If upgrading, then apply:
ms_xpay_schema.sql	ms_xpay_views.sql
ms_xpay_data.sql	**[IF INCLUDED IN FILE]** bug_fix.sql
ms_xpay_views.sql	

4. Restart both Xpay and Xpay GUI services. See *"Restarting Xpay and Xpay GUI Services - Windows" on page 34* for procedural information.

Xpay

C H A P T E R

3

XPAY INSTALLATION

Overview

Upon completing the installation, Xpay will be running as a Windows service. The services are named Xpay and Xpay GUI. Xpay is the authorization service and Xpay GUI the server process for the GUI.

If you are installing Xpay for MerchantLink, follow instructions in <u>Appendix A: "MerchantLink Configuration" on page 83</u>.

Prerequisites:

- Assumes Windows only at this time.
- Assumes Java is installed.
- Assumes the Keystore file and Trust Store file for Xpay have been created. Refer to the *Xstore™ Implementation Guide* for instructions.
- Assumes if logging is desired, an Xpay database has been created if using database logging See <u>"Creating a Database" on page 5</u> for database setup instructions.

> *Database logging is optional. Xpay can be installed without MS-SQL Server dependency for the logging. However, without database logging enabled, the transaction information and system log search options will not be available.*
>
> *Since Xpay's database logging is optional, user account information is stored in a realm file (*xpayRealm.properties*) rather than in the database.*

Installing Xpay

1. Launch the Xpay installer: `xpay-02_xx_xxx.exe`

2. At the Xpay Setup Welcome screen, click **Next** to begin the install.
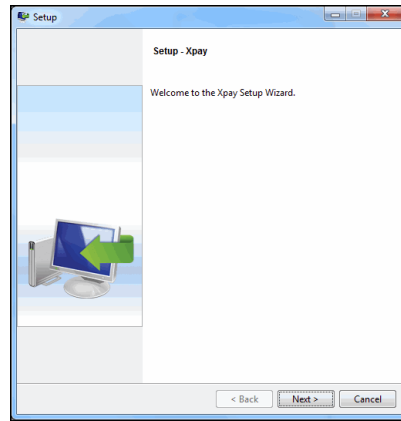
Figure 3-1: Xpay Setup Welcome Screen

> *During the installation, if you need to change the information entered in a previous screen, click the **Back** button to return to a prior screen.*
>
> *If, at any point, you must stop the installation, click the **Cancel** button. You will be prompted to confirm the cancellation. Click the **Yes** button to cancel the installation and exit the GUI.*

3. MICROS-Retail recommends you keep the default value for the **Installation Directory**. Click **Next**.
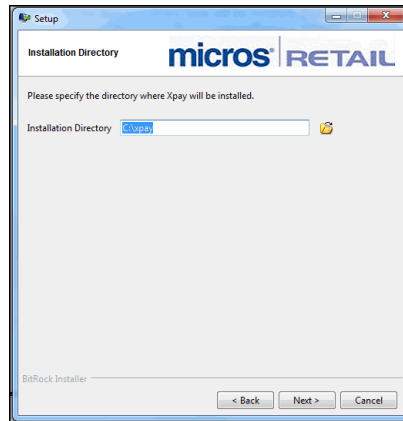
Figure 3-2: Installation Directory Screen

> *Using the default installation directory specified here makes it easier to locate the directory when troubleshooting.*

4. If applicable, specify the configuration zip file provided to you by MICROS-Retail. Click **Next**.
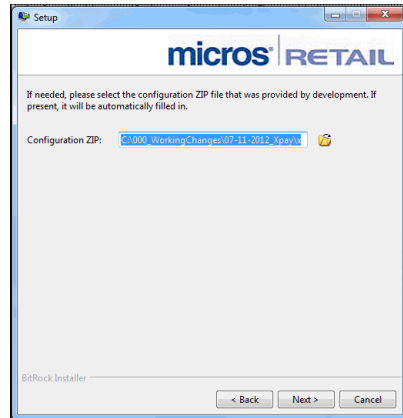
Figure 3-3: Configuration Zip File Screen

> *The Configuration ZIP field will be populated automatically if the file is located in the same folder as the installer. If the field is not auto-populated, use the browse button to locate the file.*

5. Select the type of installation: Fresh Install or Upgrade:

❑ **Fresh installation** - Select this option for a new Xpay installation.

❏   **Upgrade to a new version** - Select this option to upgrade from a previous version of Xpay (see <u>"Upgrading Xpay" on page 21</u>).
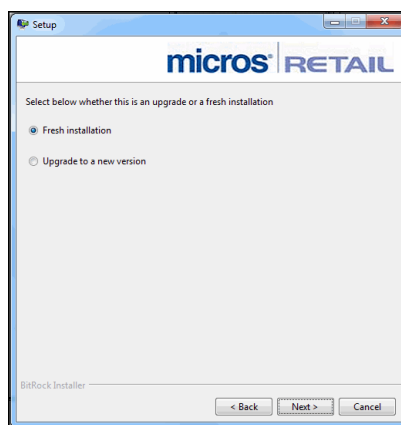
Figure 3-4: Installation Type Screen

6.   Click **Next**.

7.   **[OPTIONAL]** Override the **Hostname** field if you want to use a local host instead of your machine.

> *The **Hostname** must match the **Common Name** fields used to set up your certificates. Refer to the Xstore Implementation Guide for more information about SSL Certificates.*
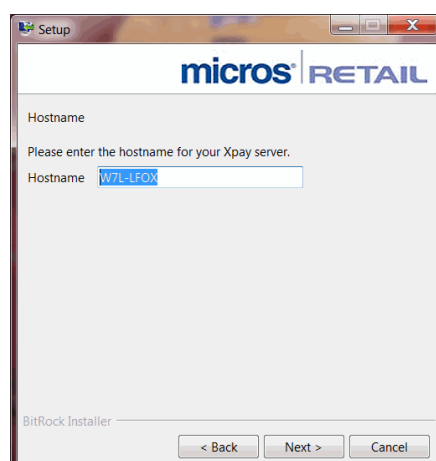
Figure 3-5: Hostname Screen

8.   Click **Next**.

9. Enter the user account information that will be used to log in to the Web Interface.

> This user will have administrative access to the Xpay system.

Password Rules:

- Must be six to eight alphanumeric characters in length
- Must contain at least one digit
- Must contain at least one uppercase letter
- Must contain at least one lowercase letter
- Cannot be the same as the Username

Figure 3-6: Web Interface Login Information

10. Click **Next**.

11. Specify whether or not you want to enable database logging:

> ### *About database logging:*
> *A logging component logs information to a database. This information is used by the Xpay Transaction Viewer to report data. Logging to the database is not required, but is recommended.*

- ❑ [**RECOMMENDED**] Select **Yes** to configure Xpay to *log* to the database and then click **Next**.

- ❑ Select **No** to skip database logging. Click **Next** and then go to .

12. Specify the database logging platform, then click **Next**.
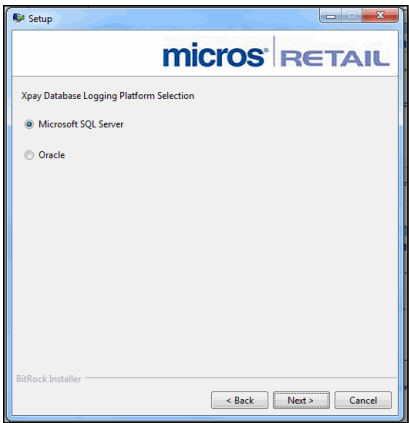
Figure 3-7: Database Logging Platform Screen

13. Enter the database information:

Table 3-1: *Database Information Fields*

Field	Setting
Database Server	Enter the name of the system where the database is being installed. **Note**: This is not necessarily the same as the Xpay server entered at the "Hostname Screen" on page 14.
Database Name	Enter the name of the Xpay database.
Database Username	Enter the username for the Xpay database.
Database password	Enter the password for the Xpay database.
Re-Enter your password	Enter the password again to confirm.
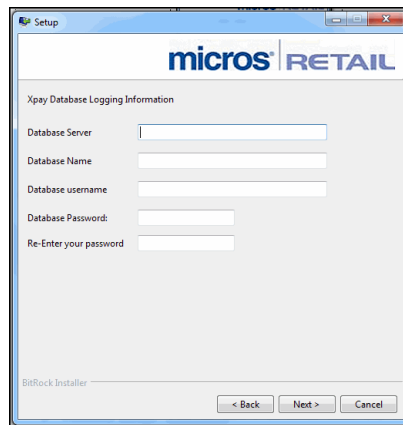
Figure 3-8: Xpay Database Logging Information Screen

14. Click **Next**.

15. Enter the Xpay Keystore configuration information:

Table 3-2: *Keystore Fields*

Field	Setting
Java Keystore	This may default depending on the location of the file, if not browse to the file location where the Xpay Keystore is located.
Keystore password	Enter the password used when the Xpay Keystore was set up.
Re-Enter your password	Enter the password again to confirm.
SSL Certificate Alias	Enter the alias for the Xpay Keystore certificate.
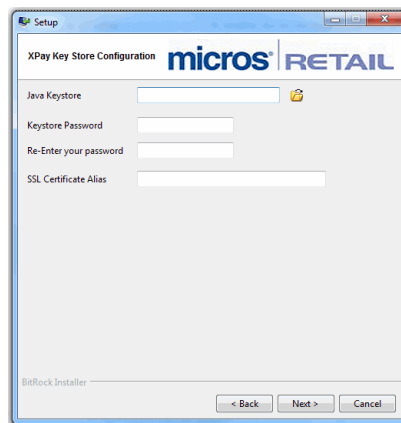
Figure 3-9: Xpay Keystore Configuration Screen

Refer to the Xstore Implementation Guide for more information about SSL Certificates.

16. Click **Next**.

17. Enter the Xpay Trust Store configuration information:

Table 3-3: *Java Trust Store Fields*

Field	Setting
Java Trust Store	This may default depending on the location of the file, if not browse to the file location where the Xpay Trust Store is located.
Trust Store password	Enter the password used when the Xpay Trust Store was set up.
Re-Enter your password	Enter the password again to confirm.
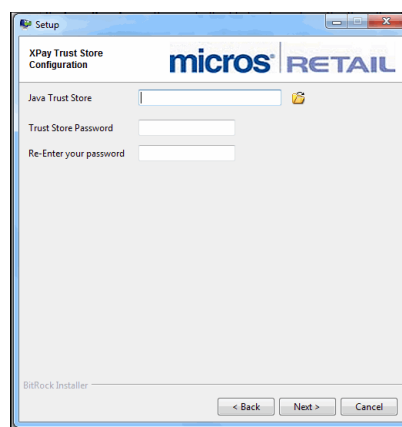
Figure 3-10: Xpay Trust Store Configuration Screen

18. Click **Next**.

19. The Ready to Install screen will display, click **Next** to begin installing Xpay.
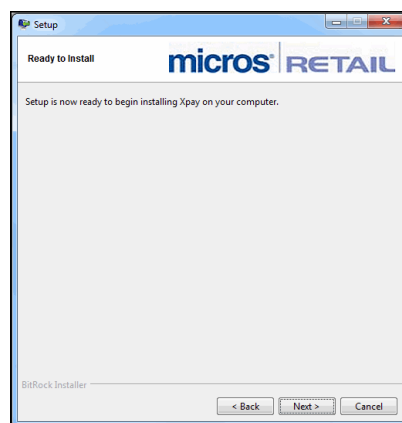
Figure 3-11: Ready to Install Screen

20. Wait until installation is complete.
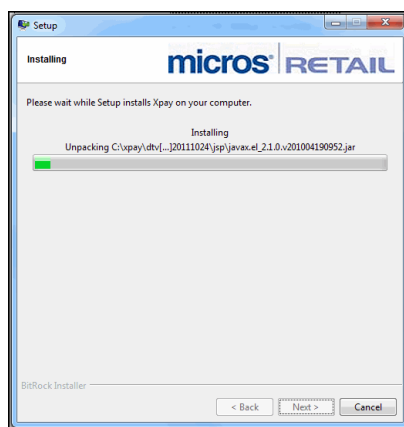
Figure 3-12: Installation Progress Screen

21. When installation is complete, click **Finish** to close the screen.
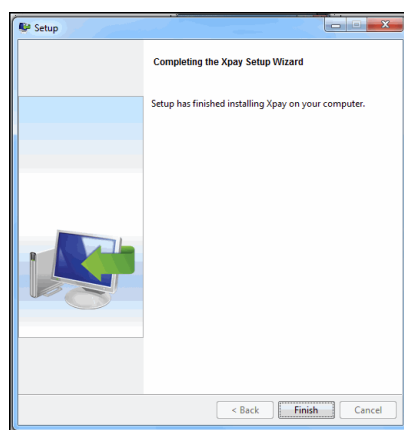
Figure 3-13: Installation Complete Screen

22. If you created a Database for logging, apply the database scripts ("After Installing Or Upgrading Xpay" on page 9).

> *You will also need to apply the database scripts if you upgraded Xpay and have logging set up.*

23. The installation process creates a shortcut on your desktop. Double-click the Xpay icon to access the Xpay GUI. You can also access the Xpay GUI via the Start menu.

24. When prompted, enter the Username and Password you specified during the install process in order to access Xpay.
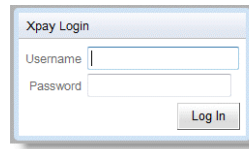
Figure 3-14: Xpay Login Prompt

Upgrading Xpay

It is not necessary to uninstall previous versions of Xpay in order to upgrade the application. The Xpay installer automatically shuts down the service associated with the old Xpay application and run the new service.

User account information is stored in a realm file rather than in the database (`gui/jetty/etc/xpayRealm.properties`). During an upgrade, the `xpayRealm.properties` file is copied automatically and requires no user action.

Also, the DB logging configuration data (database server name, database name, and the database username) automatically populate during the upgrade process. However, the user must enter and confirm the database password.

1. Launch the Xpay installer: `xpay-02_xx_xxx.exe`

2. At the Xpay Setup Welcome screen, click **Next** to begin the install.
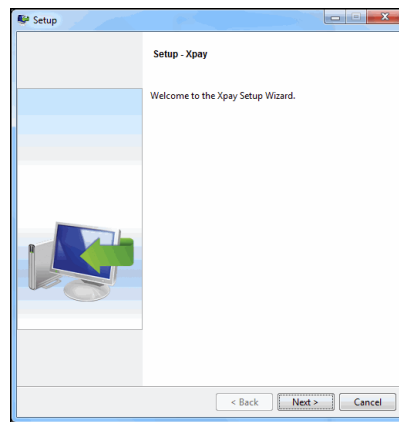
Figure 3-15: Xpay Setup Welcome Screen

> *During the upgrade, if you need to change the information entered in a previous screen, click the **Back** button to return to a prior screen.*
>
> *If, at any point, you must stop the upgrade, click the **Cancel** button. Confirm cancellation by clicking the **Yes** button to cancel the upgrade and exit the GUI.*

3. MICROS-Retail recommends you keep the default value. Click **Next**.
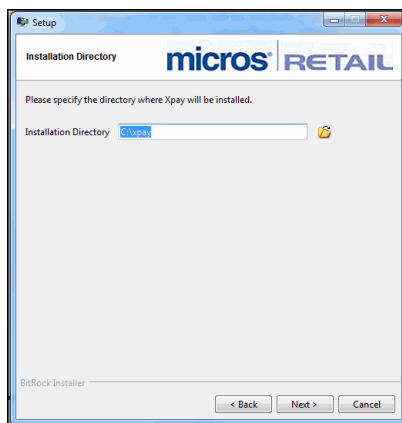
Figure 3-16: Installation Directory Screen

Using the default installation directory specified here makes it easier to locate the directory when troubleshooting.

4. If applicable, specify the configuration zip file provided to you by MICROS-Retail. Click **Next**.
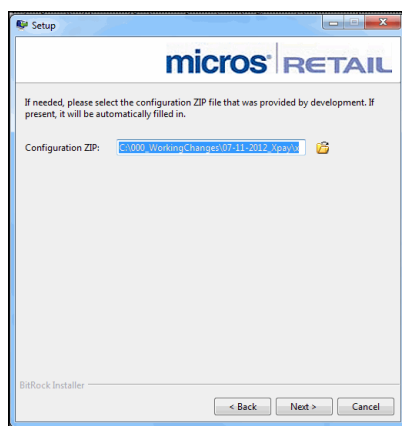
Figure 3-17: Configuration Zip File Screen

The Configuration ZIP field will be populated automatically if the file is located in the same folder as the installer. If the field is not auto-populated, use the browse button to locate the file.

5. Select the Upgrade installation type, then click **Next**.
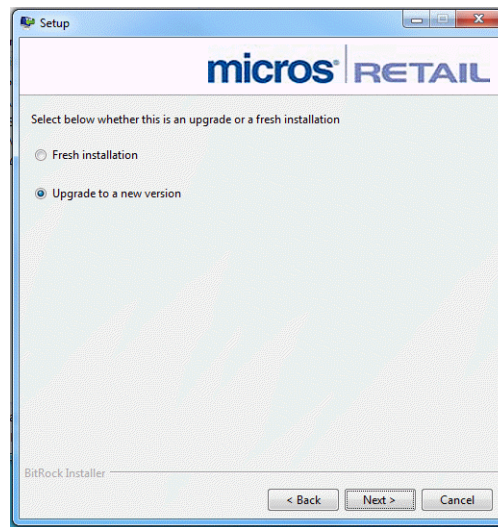
Figure 3-18: Installation Type Screen

6. The next steps are the same as those performed during a fresh install. Continue with .

Xpay

C H A P T E R **4**
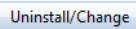
UNINSTALLING XPAY [AS NEEDED]

Overview

In Windows, there are two ways to begin the Xpay uninstall process: the Windows Uninstaller and the Uninstaller Program.

Windows Uninstall

Xpay can also be uninstalled using the Windows® Add/Remove programs functionality. The following steps assume Windows 7 OS.

1. To access Add/Remove programs, click the **Start** menu and select the **Control Panel** option.

2. Click the **Programs and Features** program icon.

3. Scroll down until you find the Xpay application and select it.

4. Right-click **Uninstall/Change** [Uninstall/Change] to begin the Xpay uninstall process.

5. Go to step 2 of the Uninstaller Program below.

Uninstaller Program

The uninstall program for Xpay is located in `C:\xpay\install` (assuming the default directory was used during installation).

> *There can be multiple instances of Xpay installed on any given machine, and each installed version of Xpay will have its own uninstaller within the directory. When uninstalling Xpay, run the uninstaller from the most recent installed release. Any older releases will also be uninstalled.*

1. Run the uninstaller program: `uninstall-02_xx_xxx.exe`

2. Click **Yes** to uninstall Xpay and all its modules.
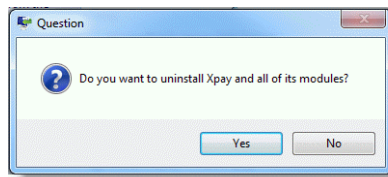
Figure 4-1: Uninstall Xpay Prompt

3. Wait while Xpay and all its modules are uninstalled.
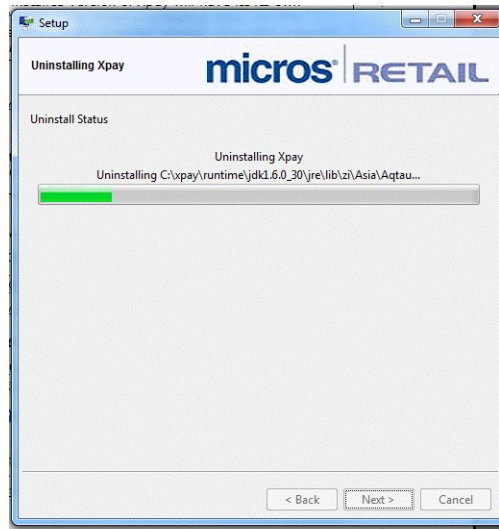
Figure 4-2: Uninstall Progress Window

4. Click **OK** to close the Uninstallation Completed message box.
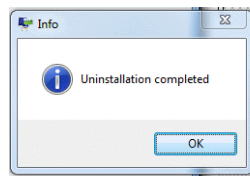
Figure 4-3: Uninstallation Complete Message Box

5. If you are prompted to restart your computer, click **Yes**.
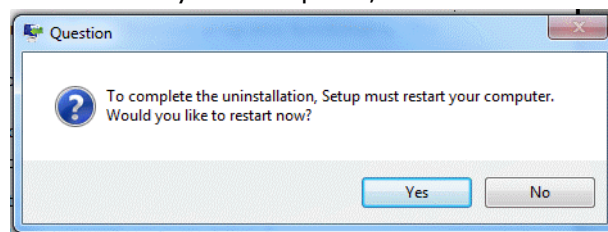
Figure 4-4: Restart Computer Prompt

CHAPTER

5

XPAY CONFIGURATION
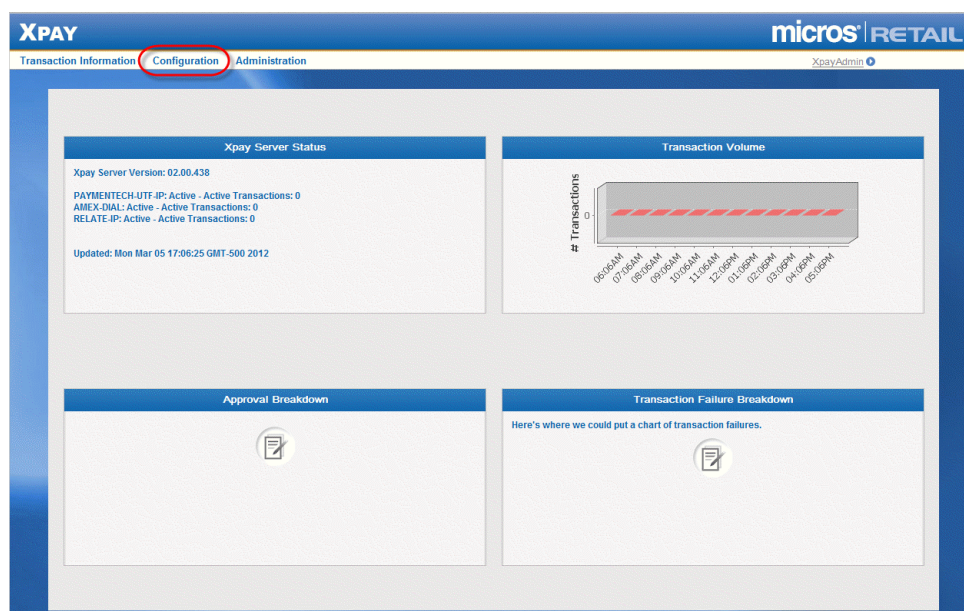
Figure 5-1: Xpay Home Page - Configuration Menu Option

Overview

There are three Xpay configuration options: **Processors and Tenders**, **Logging**, and **Modem**. Refer to the sections below for more information about these options.

All configuration file names are preceded by the processor name and communication type (except for the `modems-config` and `xpay-log-config` files). These files are installed to the Xpay\Config folder on the Xpay server.

Processors and Tenders Configuration

Prerequisites

Before you configure processors and tenders, you should have the following information available:

- A list of all of the processors you plan to use

- Processor connection information

- A list of the tender types that each processor handles

About Processors and Tenders Configuration

Use the Processors and Tenders Configuration feature to select which authorization providers you will be using. Once the providers are selected, define which tender types are going to be sent to each of the selected providers. For example, if you select Vital and FDMS as your processors, you can select check tenders to be authorized through FDMS, and all other credit cards to be authorized through Vital.

You can also configure the communication settings specific to each processor:

- If the provider class indicates that the processor uses TCP/IP, you must enter the IP Address and Port values.

- If the class indicates that Xpay will communicate through a dialup connection, you must enter the communication port settings such as Port Number, Baud Rate, Parity, Stop Bits, Modem Init String, etc.

Once all required information has been entered and saved, an XML file is created. This file will contain all of the configuration information, broken down by each provider.

Accessing Processors and Tenders Configuration

> The Xpay and XpayGui services must be restarted for any processor/
> tender changes to take effect, both adding a new processor and making
> any changes to an existing processor. You can restart the services after
> you have completed processor and tender setup.
>
> See *"Restarting Xpay and Xpay GUI Services - Windows" on page 34*
> for procedural information.
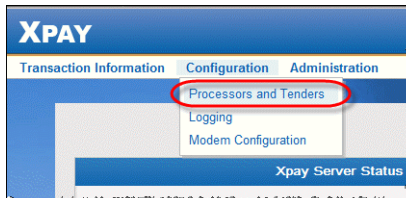
To access the Processors and Tenders page, click
Configuration-->**Processors and Tenders**.
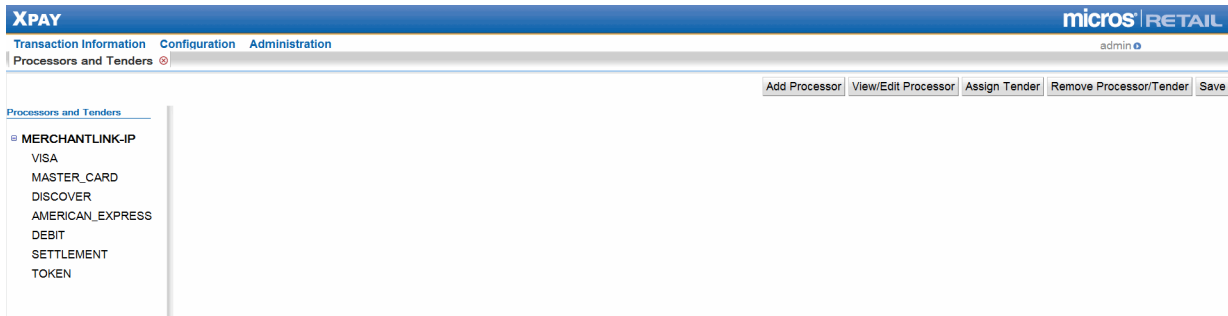
Figure 5-2: Processors and Tender Configuration Page

> The default processor and tenders installed during the Xpay install is
> shown in the Processors and Tenders list on the left side of the page.

Adding a Processor

1. To add a processor, click **Add Processor**. See .

2. At the Add Processor list, select a processor from the processor list and click **Add**. The list shows all of the supported/available processors.
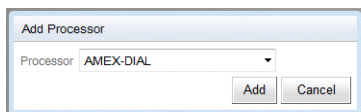
Figure 5-3: Add Processor List

You can select multiple processors from the list. Repeat this step for each processor until all of the processors that will be used appear in the **Processors and Tenders** list on the left side of the page.

3. When prompted, click **OK** to acknowledge and close the message indicating the Xpay and XpayGui services must be restarted for the change to take effect. You can restart the services after you have completed processor and tender setup.
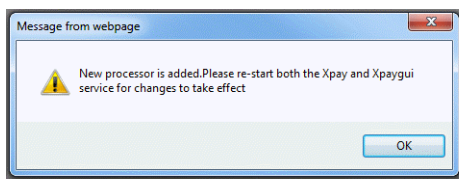
Figure 5-4: Restart Services Message

4. Click **Save** to save the changes, then restart the services for the changes to take effect. See .

5. Next, review and edit the processor connection settings as needed. Continue with .

Viewing/Editing Processor Settings

1. To view and or edit processor connection settings, select a processor from the Processors and Tenders list on the left side of the page, then click **View/Edit Processor**. See Figure 5-2 on page 29.

2. Select Assigned Addresses and edit the information as needed. The information will vary with each processor:

    ❑ If the provider class indicates that the processor uses TCP/IP, you must enter the IP Address and Port values.

    ❑ If the class indicates that Xpay will communicate through a dialup connection, you must enter the communication port settings such as Port Number, Baud Rate, Parity, Stop Bits, Modem Init String, etc.

Figure 5-5: Processor Information, IP Type Example

3. If prompted, click **OK** to acknowledge and close the message indicating the Xpay and Xpay GUI services must be restarted for the change to take effect. You can restart the services after you have completed processor and tender setup.

4. Click **Save**, then restart the *Xpay* and *Xpay GUI* services. See "Restarting Xpay and Xpay GUI Services - Windows" on page 34.

5. Next, add or edit tenders for the processor as needed. Continue with "Assigning Tenders to Processors" on page 32.

Assigning Tenders to Processors

> The tender selection screen will display a list of available tenders based on the authorization provider selected.

1. To assign a tender to a processor, click **Assign Tender** (see Figure 5-2 on page 29).

2. Select the processor from the list of available processors.
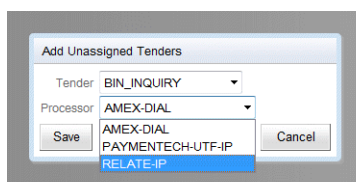
Figure 5-6: Available Processors List

3. Select the tender for this processor.
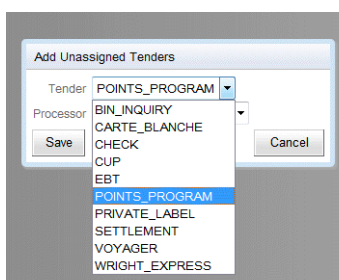
Figure 5-7: Available Tenders List

4. When prompted, click **OK** to acknowledge and close the message indicating the Xpay and Xpay GUI services must be restarted for the change to take effect. You can restart the services after you have completed processor and tender setup.

5. Click **Save** to map the tender to the processor.

6. Repeat this process to map tenders for each processor as needed.

7. When you have completed Processor and Tender configuration, click **Save**, then restart the ***Xpay*** and ***Xpay GUI*** services. See "Restarting Xpay and Xpay GUI Services - Windows" on page 34.

Removing Processors and/or Tenders

To remove a processor

1. To remove a processor and all mapped tenders, select a processor from the Processors and Tenders list on the left side of the page, then click **Remove Processor/Tender**.

2. When prompted, click **OK** to acknowledge and close the message indicating the Xpay and Xpay GUI services must be restarted for the change to take effect.

3. Click **Save** to save the changes, then restart the services for the changes to take effect. See "Restarting Xpay and Xpay GUI Services - Windows" on page 34.

To remove a tender

1. To remove a tender, select a tender associated with the processor from the Processors and Tenders list on the left side of the page, then click **Remove Processor/Tender**.

2. When prompted, click **OK** to acknowledge and close the message indicating the Xpay and Xpay GUI services must be restarted for the change to take effect.

3. Click **Save** to save the changes, then restart the services for the changes to take effect. See "Restarting Xpay and Xpay GUI Services - Windows" on page 34.

Restarting Xpay and Xpay GUI Services - Windows

1. To restart the services, go to **Start**--> **Control Panel**--> **System and Security**--> **Administrative Tools** --> **Services**.

2. With *Xpay* service selected, either click **Restart the service** or right-click the mouse and select **Restart** from the drop-down menu.
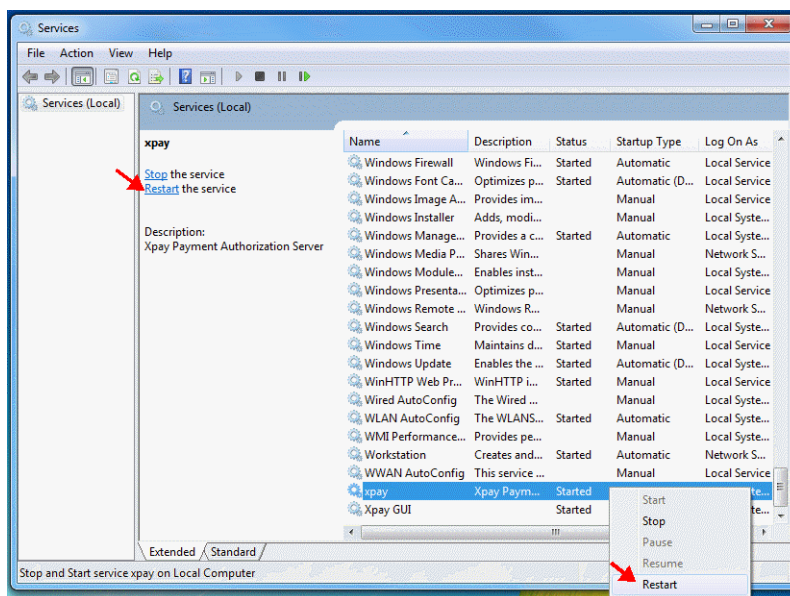
Figure 5-8: Xpay Restart Service

3. Windows will stop, then restart the Xpay Service.

4. Repeat steps 2 and 3 for the *Xpay GUI* service. The configuration changes you made will now be in effect.

5. Sign into Xpay again, navigate to the **Processor and Tender** tab, and validate that the configuration is correct.

Logging Configuration

> *Database logging is set up during installation.*

About Xpay Logging Configuration

Xpay supports three logger types: ***log4j***, ***jdbc***, and ***xml***.

Xpay provides logging information to indicate status, function, and errors that may change or occur as a result of authorization processing.

Seven logging levels for each logger type are provided: Trace, Debug, Info, Warn, Error, Fatal, and Shutdown. These levels, as listed, provide logging data with decreasing frequency. For example, the Trace logging level generates more information on a per-request basis than for a lower debugging level such as Error, which provides only error information.
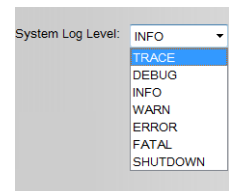
Troubleshooting issues with Xpay can be facilitated by turning up the debugging level. High logging levels produce large log files. Conversely, lower logging levels produce log files of decreasing size.

The Info logging level is usually suitable for everyday use and generates enough information for high-level troubleshooting while creating only moderate sized logging files.

The type of logging to be used by Xpay, the level of logging, and the destination where the logs are written to can be configured. Logger types are predefined by implementing IlogListener interface.

> *You cannot add or remove a logger, or add logger parameters. Only logger configuration can be edited.*

Accessing Logging Configuration
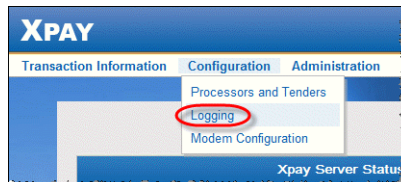
To access the Logging Configuration page, click **Configuration**-->**Logging**.
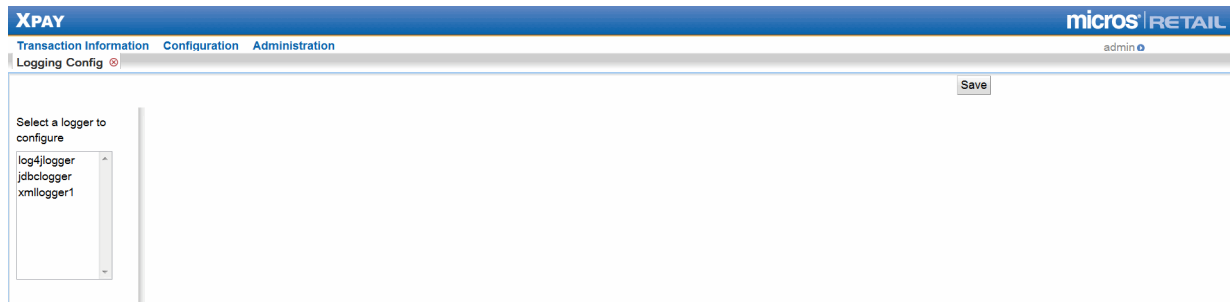
Figure 5-9: Logging Configuration Page

> *If logging levels are modified, the Xpay service must be restarted for changes in the logging configuration to take effect.*

To Configure Logging

1. Select a logger from the list to display the logger values.

2. Edit the values as needed and click **Save** to save your changes.

Logger Information

log4jlogger

Figure 5-10: log4jlogger Page

log4j is designed to handle Java exceptions. The logger is the core component of the logging process. The behavior of loggers is hierarchical. A logger will only output messages that are of a level greater than or equal to it. If the level of a logger is not set, it will inherit the level of the closest ancestor.

In log4j, the following levels of logger are available:

TRACE - This level has the lowest possible rank and is intended to turn on all logging.

DEBUG - This level provides fine-grained informational events that are used to debug an application.

INFO - This level provides informational messages that show the progress of the application at a coarse-grained level.

WARN - This level provides information about potentially harmful situations.

ERROR - This level provides information about error events that might still allow the application to continue running.

FATAL - This level provides information about severe error events that will cause the application to fail.

SHUTDOWN - This level has the highest possible rank and is intended to turn off logging.

jdbclogger

Figure 5-11: jdbclogger Page

Database logging is the most robust logging included with Xpay. The database log is highly useful for reporting, analysis, and troubleshooting.

Database Tables

System information is logged to the System_Log table. System information is available in the Administration component of Xpay. See "System_Log table" on page 73 for table data.

Transaction information is logged to the Transaction_Header table. This table logs high level details of individual transactions, including: date/time, transaction id, store, register, and authorization response. See "Transaction_Header table" on page 76.

Transaction activity information is logged to the Transaction_Log table. This table logs details of individual authorizations. See "Transaction_Log table" on page 78.

xmllogger1

Figure 5-12: xmllogger1 Page

The XML log data can be found under the Xpay directory, in a subdirectory named data. (For example: `C:\xpay\dtv-xpay-02_xx_xxx\data`).

The logging data is broken down into three separate files: `system-log.xml`, `transaction-log.xml`, and `transaction-activity-log.xml`.

- `system-log.xml` - Logs information about Xpay system activity, such as starting Xpay, stopping Xpay, and restarting Xpay

- `transaction-log.xml` - Logs header information for individual transactions, and includes:
    - ❑ Date/time
    - ❑ Transaction ID
    - ❑ Store ID
    - ❑ Register ID
    - ❑ Response

■   `transaction-activity-log.xml` - Logs transaction details and is divided into five sections:

    ❑   NEW_REQUEST_RECEIVED

    ❑   POS_REQUEST_DATA_LOADED (Inquiry functions)

    ❑   PROVIDER_RESPONSE

    ❑   RESPONSE_SENT_TO_POS

    ❑   POS_ACK_RECEIVED

Modem Configuration

Modem configuration is only required for Xpay servers that communicate through a dial-up connection to the processor.

About Modem Setup

Xpay Requires the following:

■   modem must send response codes to AT commands (ATQ1)

■   responses must be in text format, not numeric (ATV1)

■   modem should send reasonably descriptive response code text (ATX4)

■   "normal" carrier detect operation (AT&C1)

■   "normal" DTR operation (AT&D2)

■   no data compression

■   no advanced error control protocols

■   modem must not be allowed to connect at a higher speed than what the serial port is initialized to (due to a limitation of RXTX, Java's serial port driver for Linux)

Optional, but recommended:

■   hardware flow control (RTS/CTS)

Optional configuration options:

■   if modem echoes AT commands (ATE0 /ATE1)

■   if the speaker is on (ATM0 / ATM1) or what the volume level is (ATLn, where n=0..3)

Accessing Modem Configuration
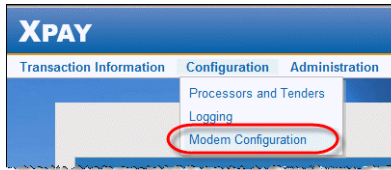
To access the Modem Configuration page, click
Configuration-->**Modem Configuration**.

Figure 5-13: Modem Configuration Page

Name: `<modem-id>` The modem Id that matches a modem Id defined in the `serial-port-settings.xml` file.

Serial Port: `<serial-port>` It is important to use the underlying operating system's naming convention for specifying serial ports; i.e. DOS (windows) calls "COM1" what Linux calls "/dev/ttyS0"

Dial Prefix: `<dial-prefix>` The modem's dial command in the dial-prefix node.

Hangup Timeout: `<hangup-timeout>` If there is inactivity for the time specified here, the connection is shut down.

InitString: `<modem-init-string>` The modem initialization string should first try to restore factory default settings from ROM (commonly done with AT&Fn), then make alterations as needed to meet Xpay's requirements.

Use the **Add**, **Delete**, and **Edit** options as needed and **Save** your changes.

modems-config.xml Example

The `modems-config.xml` file will only be used for dial backup servers and contains the configuration information.

```
-->
    <modem-info>
        <modem-id>Modem-1</modem-id>
        <serial-port>/dev/ttyS0</serial-port>
        <dial-prefix>ATDT</dial-prefix>
        <hangup-timeout>15</hangup-timeout>
        <!-- USR 56K Sportster -->
        <modem-init-string>AT&amp;F1L0&amp;K0&amp;M0</modem-init-string>
        <!-- Motorola UDS V.3400 -->
        <!--
    <modem-init-string>AT&amp;F2L0&amp;C1&amp;D2\Q3</modem-init-string>
-->
        <!-- Hayes -->
        <!--
    <modem-init-string>AT&amp;F0L0\N0%C0S11=50</modem-init-string>
-->
    </modem-info>
</modems-config>-->
```

> *Many modem commands contain an ampersand (&) character. Since this is an xml file, the ampersand must be encoded as &amp. For example, AT&C1 would have to be entered as AT&C1*

[PROCESSOR]_serial-port-settings.xml Example*

The `serial-port-settings.xml` file will only be used for dial backup servers. This file contains serial port settings such as the modem baud rate, port, parity, etc. It also contains a modem-id node that must match a modem-id value defined in the `modems-config.xml` file.

```
<serial-port-settings xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="schemas/modserial-port-settings.xsd">
    <serial-port-info>
        <modem-id>Modem-1</modem-id>
        <baud-rate>1200</baud-rate>
        <data-bits>7</data-bits>
        <stop-bits>1</stop-bits>
        <parity>EVEN</parity>
        <flow-control>RTS/CTS</flow-control>
        <primary-number>1-800-416-1282</primary-number>
        <!--        <primary-number>1-800-237-2626</primary-number>
    <secondary-number>1-800-351-4143</secondary-number>
    -->
        <!-- TODO: these last 3 items are not yet observed by Xpay -->
        <init-before-each-tran-fg>FALSE</init-before-each-tran-fg>
        <max-request-attempts>1</max-request-attempts>
        <max-response-attempts>1</max-response-attempts>
    </serial-port-info>
</serial-port-settings>
```

Xpay Configuration Files

All configuration files names are preceded by the processor name and communication type (except for the `modems-config` and `xpay-log-config` files). These files are installed to the xpay\dtv-xpay-02_xx_xxx\config on the Xpay server.

The following configuration files are used by Xpay to determine how to process authorization requests:

- serial-port-settings.xml (for example: `amex-dial_serial-port-settings.xml`).

- processor-config-ex.xml (for example: `fdms-north-ip_processor-config-ex.xml`).

- modems-config.xml (for example: `modems-config.xml`). This configuration file is not preceded by the processor name and communication type.

- xpay-log-config.xml (for example: `xpay-log-config.xml`). This configuration file is not preceded by the processor name and communication type.

serial-port-settings.xml File

The `serial-port-settings.xml` file is only used for dial backup servers. This file contains serial port settings such as the modem baud rate, port, parity, etc. It also contains a modem-id node that must match a modem-id value defined in the `modems-config.xml` file. See for a sample XML file.

processor-config-ex.xml File

The `processor-config-ex.xml` file contains miscellaneous information that is specific to each authorization provider depending on the communication type. The node names and values will vary between each processor. For example, the FDMS North processor uses a persistent socket connection for communication. They require "heartbeat" messages to be sent to them during a period of inactivity. This configuration information is defined in the<heartbeat-interval> node of the `FDMS-North-IP_Processor-Config-Ex.xml` file.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<processor-config-ex xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="schemas/processor-config.xsd">
    <ack-request-timeout>60</ack-request-timeout>
    <request-thread-count>5</request-thread-count>
    <communication-type>SOCKET-PERSISTENT-INTERLEAVED</communication-type>
    <persistent-socket-connection-timeout>15</persistent-socket-connection-timeout>
    <heartbeat-interval>300</heartbeat-interval>
    <logon-required>true</logon-required>
    <overall-tran-timeout>40</overall-tran-timeout>
</processor-config-ex>
```

modems-config.xml File

The `modems-config.xml` file is only used by Xpay servers that communicate through a dial-up connection to the processor. This file contains the serial port, init string, the modem's dial command in the dial-prefix node and the modem-id that matches a modem-id defined in the `serial-port-settings.xml` file. See for a sample XML file.

xpay-log-config.xml File

The `xpay-log-config.xml` file is used to define what type of data the Xpay server will log, and where the data will be written. Currently, the data can be logged to a database, an xml file or written to the console. There are seven log levels that may be defined in the <loglevel> node of this configuration file. These levels are: TRACE, DEBUG, INFO, WARN, ERROR, FATAL, and SHUTDOWN.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xpay-log-config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="schemas/xpay-log-config.xsd">
    <backup-log-file>./xpay.log</backup-log-file>
    <log classname="dtv.xpay.logger.Log4jLogListener" enabled="true" name="log4jlogger">
        <param name="configPath" value="./config/log4j.xml"/>
        <param name="output" value="true"/>
        <log-level>INFO</log-level>
        <log-mask dataSource="log4j" enabled="true" id="SYSTEM"/>
        <log-mask dataSource="log4j" enabled="true" id="TRANSACTION"/>
        <log-mask dataSource="log4j" enabled="true" id="TRANSACTION_ACTIVITY"/>
    </log>
    <log classname="dtv.xpay.logger.JdbcLogListener" enabled="true" name="jdbclogger">
        <log-level>INFO</log-level>
        <log-mask dataSource="System_Log" enabled="true" id="SYSTEM"/>
        <log-mask dataSource="Transaction_Header" enabled="true" id="TRANSACTION"/>
        <log-mask dataSource="Transaction_Log" enabled="true" id="TRANSACTION_ACTIVITY"/>
        <param name="url" value="jdbc:sqlserver://W7L-MKASTRO;databaseName=XPAY"/>
        <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
        <param name="username" value="4pjsk2zUA3vsgHB1hygaFw=="/>
        <param name="password" value="h9e0AzkF0qPMFzW+qprDAQ=="/>
    </log>
    <log classname="dtv.xpay.logger.XmlLogListener" enabled="false" name="xmllogger1">
        <log-level>WARN</log-level>
        <log-mask dataSource="./data/system-log.xml" enabled="true" id="SYSTEM"/>
        <log-mask dataSource="./data/transaction-log.xml" enabled="true" id="TRANSACTION"/>
        <log-mask dataSource="./data/transaction-activity-log.xml" enabled="true" id="TRANSACTION_ACTIVITY"/>
    </log>
    <use-async-logging>false</use-async-logging>
</xpay-log-config>
```

The following configuration files are also installed to the xpay\dtv-xpay-02_xx_xxx\config folder but should not be modified:

op-chain.xml

state-codes.xml

check-auth-format-codes.xml

response-codes.xml

supported-transactions.xml

C H A P T E R

6

XPAY ADMINISTRATION
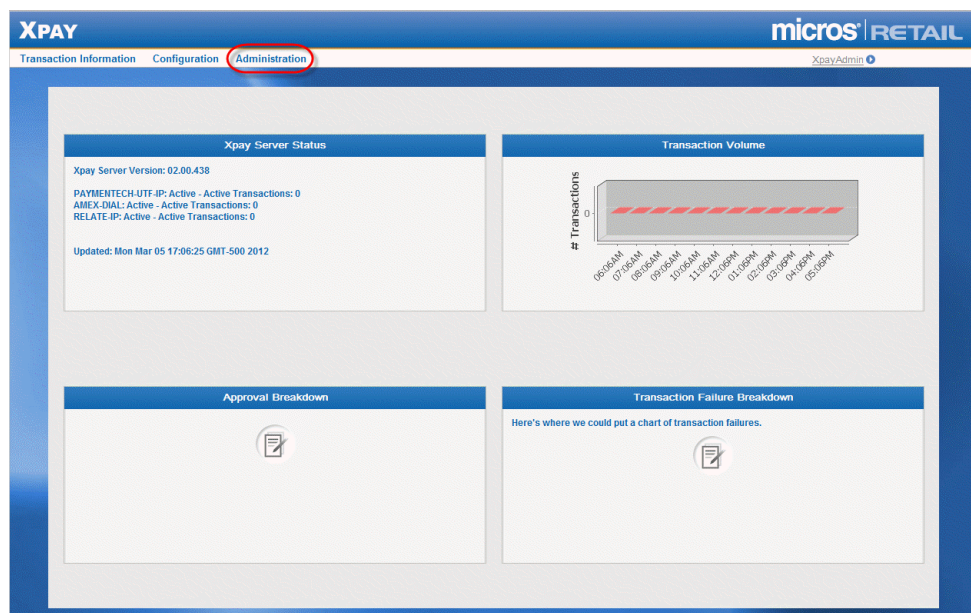
Figure 6-1: Xpay Home Page - Administration Menu Option

Overview

Xpay Administration provides the ability to manage user accounts and to view the System Log.

Managing User Accounts

Xpay supports two types of users: Admin Users and Non-Admin Users.

> *User account details are stored in gui/jetty/etc/xpayRealm.properties file.*

An **Admin** user can manage users, configure processors, and view transaction and system logs. However, if database logging is disabled, an Admin user can only see the Configuration menu and the Administration menu. The Transaction Information menu and System Log information (under the Administration menu) are not available without database logging.

A **Non-admin** user can view transaction logs and system logs, assuming database logging is enabled. If database logging is not used, transaction information and system log information are not available.

Accessing Administration: User Management
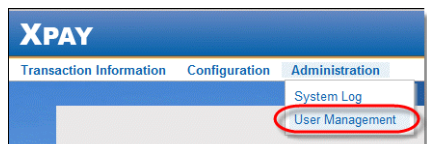
To access the User Management page, click **Administration**-->**User Management**
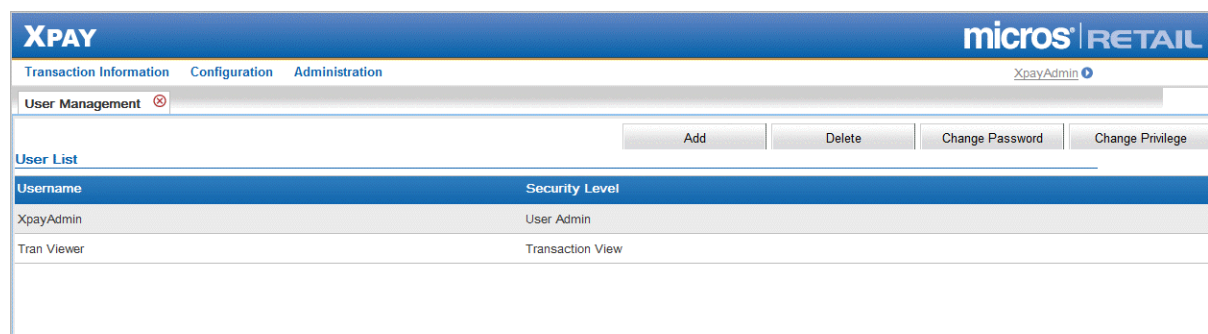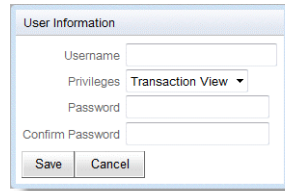
Figure 6-2: Administration - User Management Page

Adding a User

1. To add a new user, click **Add** ().

2. Enter the **Username** (case sensitive). This name will be shown on the page when the user is logged in.

Figure 6-3: User Information Prompt.

3. Select a **Privilege** from the list. The list will vary depending if database logging is enabled:

❑ ***Transaction View*** privilege:

❍ If database logging is ***enabled*** — a non-admin user will be able to view transaction logs and system logs

❍ If database logging is ***disabled*** — no visibility (the Transaction View privilege is not listed if database logging is disabled.)

❑ ***User Admin*** privilege:

❍ If database logging is ***enabled*** — an admin user will be able to manage user accounts, configure processors, and view transaction and system logs.

❍ If database logging is ***disabled*** — an admin user will be able to manage user accounts and configure processors.

4. Enter a password for the new user.

Password Rules

❑ Must be six to eight alphanumeric characters in length

❑ Must contain at least one digit

❑ Must contain at least one uppercase letter

❑ Must contain at least one lowercase letter

❑ Cannot be the same as the Username

5. Enter the password again for confirmation.

6. Click **Save**.

Changing a Password

1. To change a user's password, select the row you want to change, then click **Change Password** (Figure 6-2 on page 48).

2. When prompted, enter the old password, enter the new password, and confirm the new password for the selected user. See Password Rules above for a list of password rules.
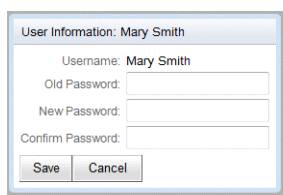
Figure 6-4: Change Password Prompt

3. Click **Save**.

Changing a Privilege

Always make sure at least one user has Admin privileges!

1. To change a user's privilege, select the row you want to change, then click **Change Privilege** (Figure 6-2 on page 48).

2. When prompted, select a new privilege from the New Privilege list.
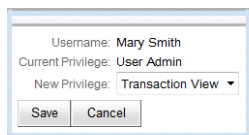
Figure 6-5: New Privilege Prompt

3. Click **Save**.

Deleting a User

> ! *Always make sure at least one user has Admin privileges!*

1. To delete a user, select the row you want to delete, then click **Delete** (Figure 6-2 on page 48).

2. The user is immediately removed from the User List.

System Log

Throughout the lifetime of the transaction, details of the transaction are written to the System Log. The System Log logs activity as it happens.

Accessing Administration: System Log
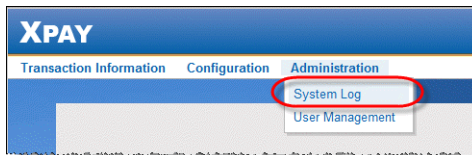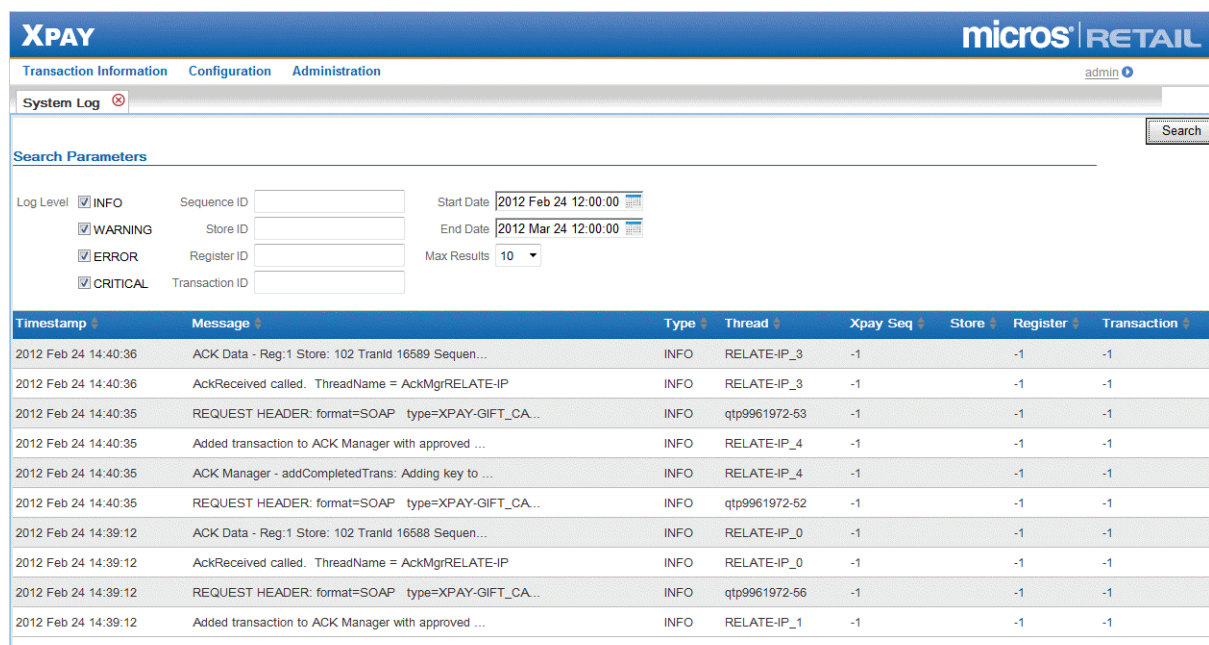
To access the System Log page, click **Administration-->System Log**

Figure 6-6: Administration - System Log Page

Viewing the System Log

1. To view the system log, enter the search criteria and click **Search**.

2.   Xpay returns a list of system messages based on your search criteria.

Figure 6-7: System Log Page - Messages

3.   To view additional detail about a message, click the row.

4.   Xpay displays detailed information about the selected message.

Figure 6-8: System Log Page - Message Detail

5. To view detail for other messages listed on the System Log page, click the up and down

arrows .

6. To close the detail window, click the **Close** icon .

C H A P T E R

XPAY TRANSACTION INFORMATION
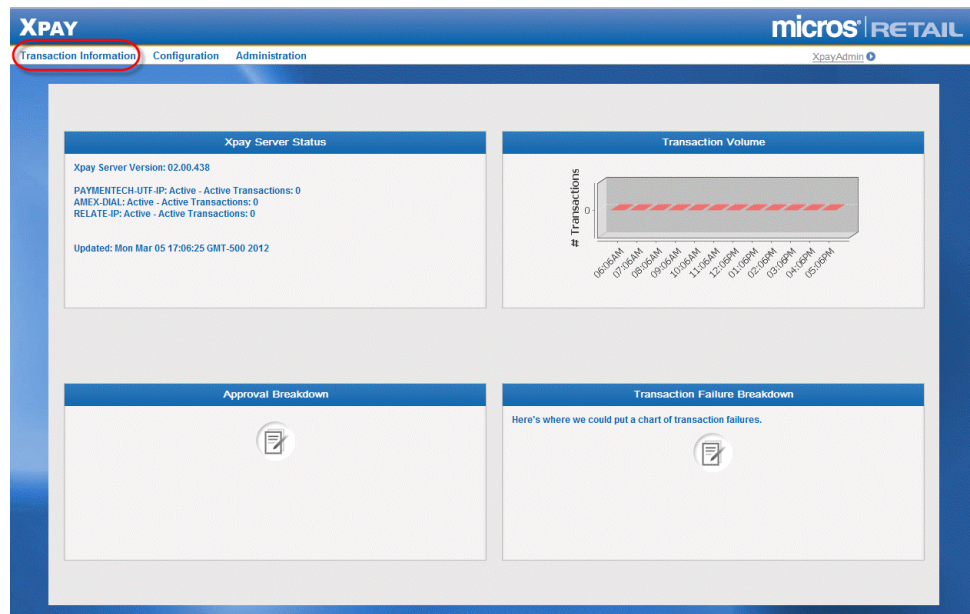
Figure 7-1: Xpay Home Page - Transaction Information Menu Option

Overview

> ! *An Xpay database is required to view transaction information.*

The Transaction Information page provides a list of transactions that have been processed through Xpay. The information shown here includes data that has been transmitted between the POS, Xpay, and the authorization provider.

The following general log records are written:

- ❑ POS request received.
- ❑ Request passed to processor thread.
- ❑ XML Request parsed into RequestObject.
- ❑ Request object passed to communication manager.
- ❑ Request sent to processor.
- ❑ Response message received.
- ❑ Response message parsed into ResponseObject.
- ❑ Response XML string created.
- ❑ Response sent back to POS application.

The main class in the application is the Xpay class. This class processes all request messages from the POS application. It reads the configuration settings from XML files and starts the necessary processor and communication threads. It passes all messages off to the appropriate processor class through the Thread Manager. The processor classes format the request message based on the credit processor specification, and pass the message off to the communication manager. The communication manager uses communication objects to send the request to the credit processor, waits for a response, and sends the response to the appropriate processor thread. When the processor thread receives the response message, the necessary data elements are then parsed from the response message and added to a standard XML response message that the POS application understands.

Accessing Xpay Transaction Information
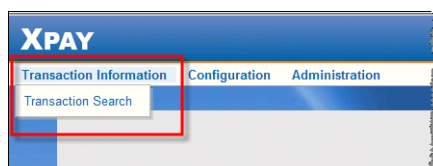
To access the Transaction Information page, click **Transaction Information**-->**Transaction Search** from the Home Page.

1. At the Transaction Search page, enter criteria to find a transaction.

Figure 7-2: Transaction Search Page

> See *"About the Transaction Log Section" on page 61* for the values expected in the search criteria fields.

2. Select a value from the Max Results drop-down field to limit the number of transactions returned in the search.

3. Click **Search**.

4. Xpay returns a list of transactions matching your search criteria.

Figure 7-3: Transaction List

5. To view detailed information about a transaction in the list, click the row in the table. Xpay returns detailed information about the selected transaction.

Figure 7-4: Transaction Page - Detailed Transaction Information

6. To view additional Log Entry detail, click within the Transaction Log pane. Xpay returns additional XML detail about the selected activity
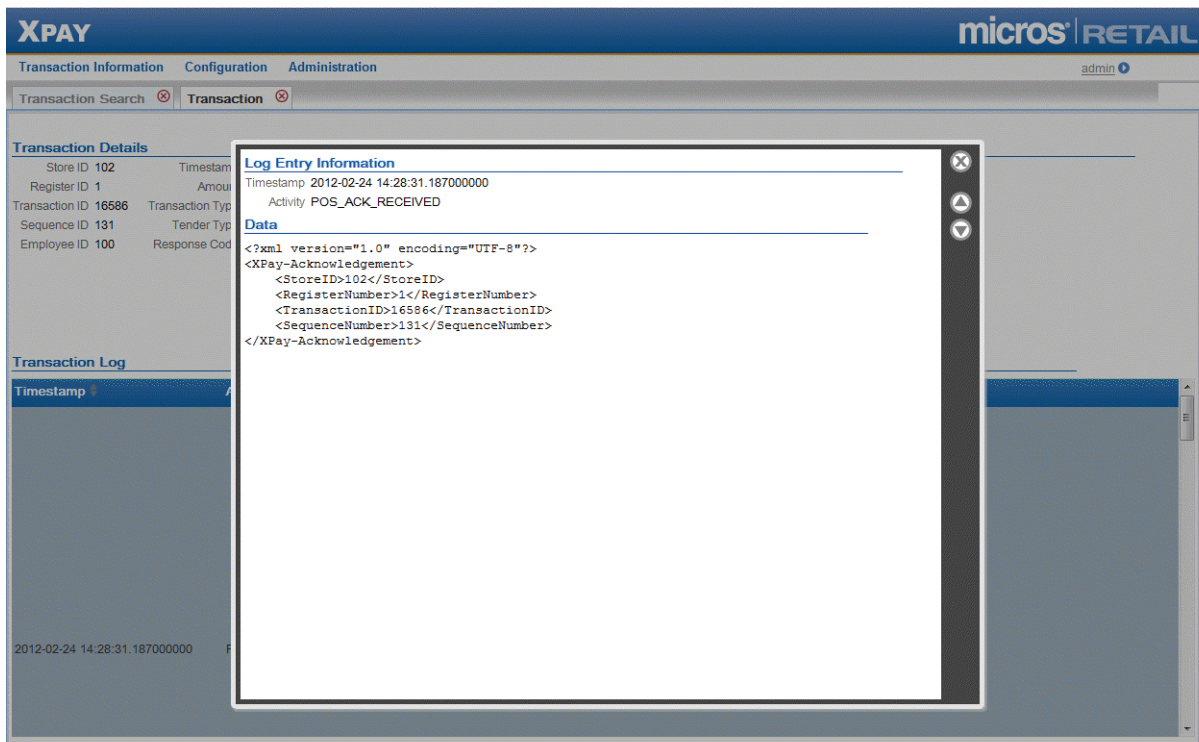
Figure 7-5: Transaction Page - Transaction Log Entry XML Information

About the Transaction Details Section
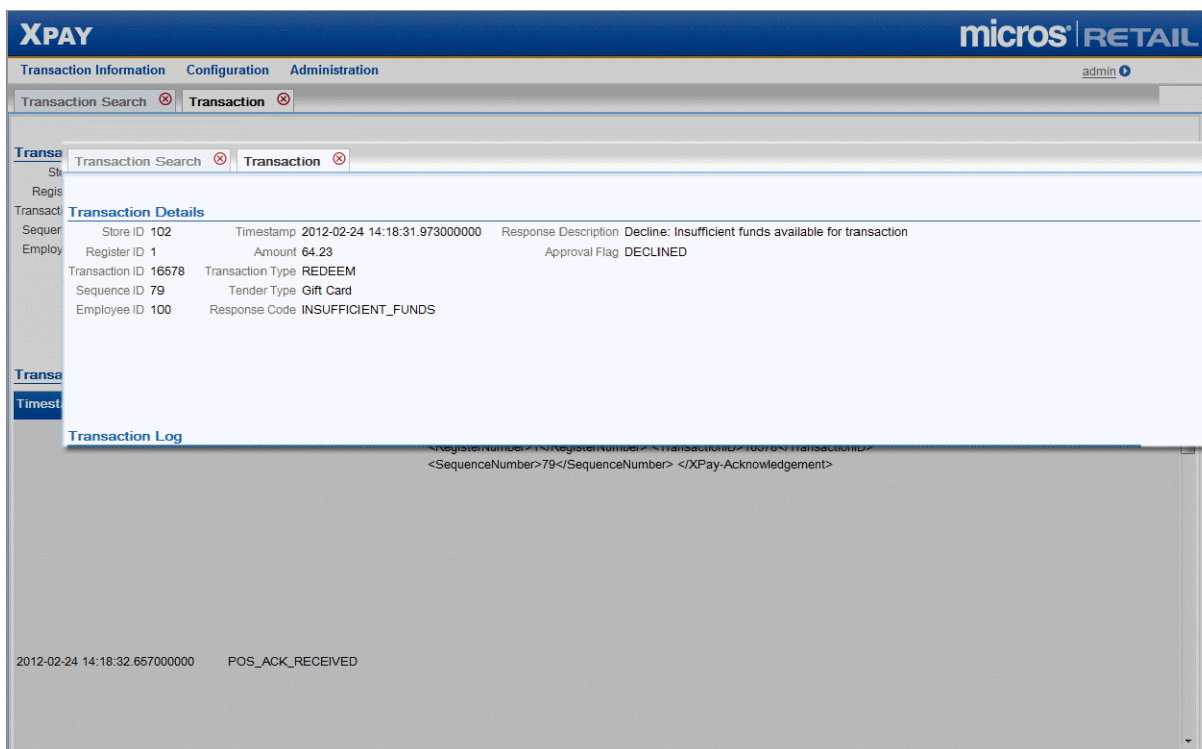
The Transaction Details section of the page shows the following information for the transaction:

- **Store ID** - The store identifier
- **Register ID** - The register identifier
- **Transaction ID** - The transaction identifier
- **Sequence ID** - The sequence number assigned to the transaction by Xpay
- **Employee ID** - The employee identifier
- **Timestamp** - The date/time of the transaction
- **Amount** - The transaction amount
- **Transaction Type** - The transaction type (for example, ACTIVATE or CASHOUT)
- **Tender Type** - The tender type (for example, Gift Card or VISA)
- **Response Code** - The code returned from the processor (for example, INSUFFICIENT_FUNDS)
- **Response Description** - The description of the response code returned from the processor (for example, Decline: Insufficient funds available for transaction). This data is from the `ResponseText` field of the RESPONSE_SENT_TO_POS log.
- **Approval Flag** - The approval flag code: APPROVED, DECLINED, or INCOMPLETE

About the Transaction Log Section

The Transaction Log section of the page shows the raw data that has been transmitted between the POS, Xpay, and the authorization provider. All sensitive account information is masked with asterisks for security reasons.

- **Timestamp** - The date/time of the transaction
- **Activity** - The transaction activity type
- **Data** - The raw data

Transaction Log details has five sections:

- NEW_REQUEST_RECEIVED
- POS_REQUEST_DATA_LOADED (Inquiry functions)
- PROVIDER_RESPONSE
- RESPONSE_SENT_TO_POS
- POS_ACK_RECEIVED

See for sample data for each type of activity.

To view XML information about the activity, click within the activity window to view the Log Entry Information.
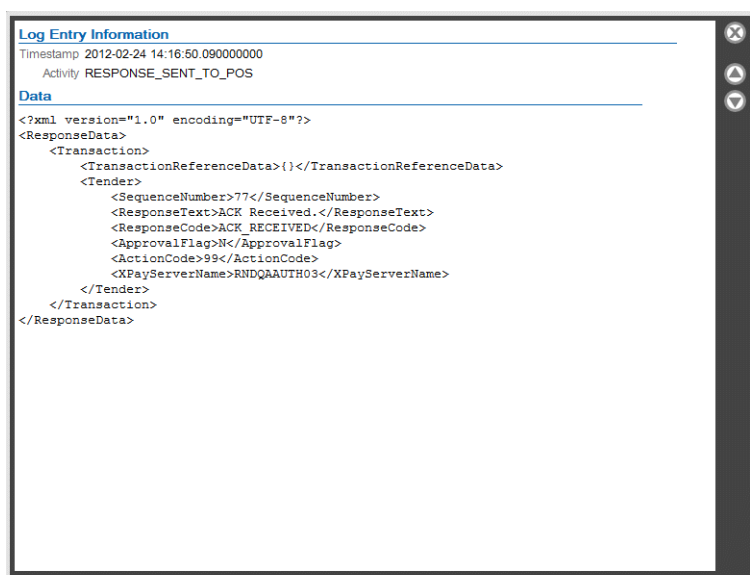
Figure 7-6: Log Entry Information Window

Activity/Data Samples

"Credit Purchase Example" on page 63

"Activate Gift Card (Already activated) Example" on page 66

"Inquiry Example" on page 69

Credit Purchase Example

Transaction Type: CREDIT_PURCHASE **Tender Type**: VISA

Activity: NEW_REQUEST_RECEIVED

Log Entry Information	
Log Entry Information	<?xml version="1.0" encoding="UTF-8"?> <RequestData> <Store> <StoreName>Seaside Outlet</StoreName> <StoreCity>Baltimore</StoreCity> <StoreID>102</StoreID> <StoreState>MD</StoreState> <StoreZipCode>21204</StoreZipCode> <StoreCurrency>840</StoreCurrency> <RegisterNumber>1</RegisterNumber> <MerchantID>6N4G6-ZNBTA-CY6M1-6HZPH-8C438</MerchantID> <SICCode>5946</SICCode> <DaylightSavingTimeFlag>Y</DaylightSavingTimeFlag> <TimeZone>-0600</TimeZone> <OrganizationName>MicrosQA</OrganizationName> <SiteID>74E8RVX2</SiteID> </Store> <Transaction> <AssociateID>100</AssociateID> <TransactionReferenceData>{}</TransactionReferenceData> <Tender> <Amount>4.59</Amount> <CardSwipedFlag>Y</CardSwipedFlag> <EntryMethodCode>MAIN_MSR</EntryMethodCode> <AcceptPinFlag>Y</AcceptPinFlag> <AcceptRfidFlag>N</AcceptRfidFlag> <TransactionTime>131527</TransactionTime> <TransactionDate>20120224</TransactionDate> <TransactionType>CREDIT_PURCHASE</TransactionType> <TenderID>VISA</TenderID> <TransactionID>16582</TransactionID> <ExpiredDate>****</ExpiredDate> <AccountNumber>************1020</AccountNumber> <Track1>*********************************</Track1> <Track2>****************************</Track2> <LineItemSequence>5</LineItemSequence> <AcceptPartialAuth>Y</AcceptPartialAuth> <AcceptBalanceReturn>Y</AcceptBalanceReturn> <BusinessDate>20120224</BusinessDate> </Tender> </Transaction> </RequestData>

Transaction Type: CREDIT_PURCHASE **Tender Type**: VISA

Activity: PROVIDER_RESPONSE

Log Entry Information	
	``<?xml version="1.0" encoding="UTF-8"?>
<Response>
<ReferenceID>56158831</ReferenceID>
<OrderNumber>16582</OrderNumber>
<TXDate>2/24/2012 7:22:36 PM</TXDate>
<ApprovalStatus>APPROVED</ApprovalStatus>
<AuthCode>VI0459</AuthCode>
<CardHolder/>
<Amount>4.59</Amount>
<Type>1</Type>
<CardNumber>************1020</CardNumber>
<CardType>4</CardType>
<AVSResponse/>
<CVResponse/>
<POSEntryType>2</POSEntryType>
</Response>```

Transaction Type: CREDIT_PURCHASE **Tender Type**: VISA

Activity: RESPONSE_SENT_TO_POS

Log Entry Information	
	``<?xml version="1.0" encoding="UTF-8"?>
<ResponseData>
  <Transaction>
    <ProcessorToken>56158831</ProcessorToken>

<TransactionReferenceData>{"ref":"56158831"}</TransactionReferenceData>
    <Tender>
      <SequenceNumber>119</SequenceNumber>
      <AVSResultCode/>
      <AccountNumber>************1020</AccountNumber>
      <ResponseText>APPROVED</ResponseText>
      <ApprovalFlag>Y</ApprovalFlag>
      <AllowManualAuth>N</AllowManualAuth>
      <ActionCode>0</ActionCode>
      <ApprovalCode>VI0459</ApprovalCode>
      <TotalAuthorizationAmount>4.59</TotalAuthorizationAmount>
      <XPayServerName>RNDQAAUTH03</XPayServerName>
      <CIDResultCode/>
      <CardType>VISA</CardType>
    </Tender>
  </Transaction>
</ResponseData>``` |

Transaction Type: CREDIT_PURCHASE **Tender Type**: VISA

Activity: POS_ACK_RECEIVED

Log Entry Information	`<?xml version="1.0" encoding="UTF-8"?>`
	`<XPay-Acknowledgement>`
	`  <StoreID>102</StoreID>`
	`  <RegisterNumber>1</RegisterNumber>`
	`  <TransactionID>16582</TransactionID>`
	`  <SequenceNumber>119</SequenceNumber>`
	`</XPay-Acknowledgement>`

Activate Gift Card (Already activated) Example

Transaction Type: ACTIVATE **Tender Type**: GIFT CARD

Activity: NEW_REQUEST_RECEIVED

Log Entry Information	\<?xml version="1.0" encoding="UTF-8"?\>
	\<RequestData\>
	  \<Store\>
	    \<StoreID\>102\</StoreID\>
	    \<StoreState\>MD\</StoreState\>
	    \<StoreName\>Seaside Outlet\</StoreName\>
	    \<StoreCity\>Baltimore\</StoreCity\>
	    \<StoreZipCode\>21204\</StoreZipCode\>
	    \<StoreCurrency\>840\</StoreCurrency\>
	    \<TerminalID\>-DEFAULT-TerminalID-IN-XPAY_GIFT_CARD_RELATE_STORE_SETTINGS\</TerminalID\>
	    \<RegisterNumber\>1\</RegisterNumber\>
	    \<MerchantID\>-DEFAULT-merchantNumber-IN-XPAY_GIFT_CARD_RELATE_STORE_SETTINGS\</MerchantID\>
	    \<DaylightSavingTimeFlag\>Y\</DaylightSavingTimeFlag\>
	    \<TimeZone\>-0600\</TimeZone\>
	    \<ProcessorVersionID\>40\</ProcessorVersionID\>
	  \</Store\>
	  \<Transaction\>
	    \<AssociateID\>100\</AssociateID\>
	    \<TransactionReferenceData\>{}\</TransactionReferenceData\>
	    \<Tender\>
	      \<Amount\>50.00\</Amount\>
	      \<TransactionType\>ACTIVATE\</TransactionType\>
	      \<TenderID\>GIFT_CARD\</TenderID\>
	      \<TransactionTime\>095138\</TransactionTime\>
	      \<TransactionDate\>20120224\</TransactionDate\>
	      \<TransactionID\>16575\</TransactionID\>
	      \<CardSwipedFlag\>N\</CardSwipedFlag\>
	      \<AccountNumber\>************8646\</AccountNumber\>
	      \<TransactionEscheatable\>Y\</TransactionEscheatable\>
	      \<LineItemSequence\>1\</LineItemSequence\>
	    \</Tender\>
	  \</Transaction\>
	\</RequestData\>

Transaction Type: ACTIVATE **Tender Type**: GIFT CARD

Activity: PROVIDER_RESPONSE

Log Entry Information	
| **Log Entry Information** | ```
<?xml version="1.0" encoding="UTF-8"?>
<ErrorResponse xmlns="http://www.nrf-arts.org/IXRetail/namespace/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xml.datavantagecorp.com/namespace/crm/
DTVStoredValue.xsd">
    <SVATransaction>
        <SVATransactionID>29950</SVATransactionID>
        <Reason>
            <Description>ActivateInstrumentResponse</Description>
            <Code>2</Code>
        </Reason>
        <Error>
            <Code>ACCOUNT_ALREADY_ACTIVE</Code>
            <Description>The account is already active.</Description>
        </Error>
        <SVAAmount>
            <CurrencyID>USD</CurrencyID>
            <Amount>50.00</Amount>
        </SVAAmount>
        <TransactionDateTime>2012-02-24T10:58:39-0500</TransactionDateTime>
    </SVATransaction>
</ErrorResponse>
``` |

Transaction Type: ACTIVATE **Tender Type**: GIFT CARD

Activity: RESPONSE_SENT_TO_POS

Log Entry Information	
| **Log Entry Information** | ```
<?xml version="1.0" encoding="UTF-8"?>
<ResponseData>
  <Transaction>
    <TransactionReferenceData>{}</TransactionReferenceData>
    <Tender>
      <TenderID>USD</TenderID>
      <SequenceNumber>71</SequenceNumber>
      <TransactionDate>2012-02-24</TransactionDate>
      <TransactionTime>10:58:39</TransactionTime>
      <BankReferenceNumber>29950</BankReferenceNumber>
      <ResponseText>Decline: Account already active</ResponseText>
      <ResponseCode>ACCOUNT_ALREADY_ACTIVE</ResponseCode>
      <ApprovalFlag>N</ApprovalFlag>
      <ErrorText>Decline: Account already active</ErrorText>
      <ActionCode>1</ActionCode>
      <TotalAuthorizationAmount>50.00</TotalAuthorizationAmount>
      <XPayServerName>RNDQAAUTH03</XPayServerName>
    </Tender>
  </Transaction>
</ResponseData>
``` |

Transaction Type: ACTIVATE **Tender Type**: GIFT CARD

Activity: POS_ACK_RECEIVED

Log Entry Information	`<?xml version="1.0" encoding="UTF-8"?>`
	`<XPay-Acknowledgement>`
	`  <StoreID>102</StoreID>`
	`  <RegisterNumber>1</RegisterNumber>`
	`  <TransactionID>16575</TransactionID>`
	`  <SequenceNumber>71</SequenceNumber>`
	`</XPay-Acknowledgement>`

Inquiry Example

Transaction Type: BIN_INQUIRY **Tender Type**: UNKNOWN

Activity: NEW_REQUEST_RECEIVED

Log Entry Information	<?xml version="1.0" encoding="UTF-8"?> <RequestData> <Store> <StoreName>Seaside Outlet</StoreName> <StoreCity>Baltimore</StoreCity> <StoreID>0</StoreID> <StoreState>MD</StoreState> <StoreZipCode>21204</StoreZipCode> <StoreCurrency>840</StoreCurrency> <RegisterNumber>0</RegisterNumber> <MerchantID>6N4G6-ZNBTA-CY6M1-6HZPH-8C438</MerchantID> <SICCode>5946</SICCode> <DaylightSavingTimeFlag>Y</DaylightSavingTimeFlag> <TimeZone>-0600</TimeZone> <OrganizationName>MicrosQA</OrganizationName> <SiteID>74E8RVX2</SiteID> </Store> <Transaction> <AssociateID>100</AssociateID> <TransactionReferenceData>{}</TransactionReferenceData> <Tender> <Amount>27.51</Amount> <EntryMethodCode>MAIN_MSR</EntryMethodCode> <TransactionTime>132146</TransactionTime> <TransactionDate>20120224</TransactionDate> <TransactionType>BIN_INQUIRY</TransactionType> <TransactionID>0</TransactionID> <AccountNumber>**8775</AccountNumber> </Tender> </Transaction> </RequestData>

Transaction Type: BIN_INQUIRY **Tender Type**: UNKNOWN

Activity: POS_REQUEST_DATA_LOADED

Log Entry Information	dtv.xpay.messaging.auth_messages.request.CreditAuthRequest@15dbaab

Transaction Type: BIN_INQUIRY **Tender Type**: UNKNOWN

Activity: RESPONSE_SENT_TO_POS

Log Entry Information	<?xml version="1.0" encoding="UTF-8"?> <ResponseData> <Transaction> <TransactionReferenceData>{}</TransactionReferenceData> <Tender> <SequenceNumber>129</SequenceNumber> <ApprovalFlag>Y</ApprovalFlag> <AllowManualAuth>N</AllowManualAuth> <ActionCode>0</ActionCode> <XPayServerName>RNDQAAUTH03</XPayServerName> <LuhnCheck>N</LuhnCheck> <IsCredit>Y</IsCredit> <IsDebit>N</IsDebit> <IsCorporate>N</IsCorporate> <CardType>VISA</CardType> <CsvPaymentRecommendation>CREDIT</CsvPaymentRecommendation> </Tender> </Transaction> </ResponseData>

Transaction Type: BIN_INQUIRY **Tender Type**: UNKNOWN

Activity: PROVIDER_RESPONSE

Log Entry Information	<?xml version="1.0" encoding="UTF-8"?> <Response> <CardType>4</CardType> <LuhnCheck>0</LuhnCheck> <IsCredit>1</IsCredit> <IsDebit>0</IsDebit> <IsCorporate>0</IsCorporate> <PaymentRecommendation>2</PaymentRecommendation> <ErrorCode>0</ErrorCode> <ErrorDescription/> </Response>

Transaction Type: BIN_INQUIRY **Tender Type**: UNKNOWN

Activity: POS_ACK_RECEIVED

Log Entry Information	<?xml version="1.0" encoding="UTF-8"?> <XPay-Acknowledgement> <StoreID>0</StoreID> <RegisterNumber>0</RegisterNumber> <TransactionID>0</TransactionID> <SequenceNumber>129</SequenceNumber> </XPay-Acknowledgement>

C H A P T E R

XPAY DATABASE TABLES

Overview

Xpay reads all configuration data from tables in the Xpay database and all transaction details are saved in this database.

> The following tables are no longer used since user account information is now stored in `gui/jetty/etc/xpayRealm.proeprties` rather than the database:
>
> ❑ Role_Table table
>
> ❑ Role_User_Map table
>
> ❑ User table

Approval_Type_Descr table

The Approval_Type_Descr table is a mapping table used to map the Transaction_Header.approved_flag field to a textual description indicating whether a transaction was approved or not. The value in the Approved_Descr field is displayed in the transaction browser.

Approval_Type_Descr			
Attribute	**Datatype**	**Null?**	**Notes**
APPROVED_FLAG (PK)	TINYINT	No	The numeric value that Xpay writes to the Transaction_Header.approved_flag field to indicate whether a transaction was approved or not. There are three possible values: 0-DECLINED 1-APPROVED 2-INCOMPLETE
APPROVED_DESCR	VARCHAR(30)	Yes	The text that will be displayed in the transaction browser to indicate whether the transaction was approved or not.

System_Log table

The System_Log table is used to log certain events that occur in the Xpay application that may or may not pertain to a specific transaction. The data in this table is mainly used for debugging purposes. Most of the records contain system information such as the operating system being used, the user account name, etc. It also contains debugging information such as tracking socket and modem connections, communication errors, etc.

System_Log			
Attribute	**Datatype**	**Null?**	**Notes**
LOG_ID_MSB (PK)	bigint	No	The unique UUID (Universal Unique Identifier) used as part of the primary key.
LOG_ID_LSB (PK)	bigint	No	UUID used as part of the primary key.
TIME_STAMP	datetime	NO	Time stamp of the log entry.
MESSAGE	varchar(5000)	No	The log message.
TYPE	varchar(50)	No	The type of message. Valid values include TRACE, DEBUG, INFO, WARN, ERROR, FATAL
THREAD_NAME	varchar(100)	No	The name of the thread processing the request.
XPAY_SEQUENCE_ID	integer	Yes	Unique sequence number assigned to the transaction by Xpay.
STORE_ID	varchar(10)	Yes	The store number where the request was generated.
REGISTER_ID	integer	Yes	The point of sale register number where the request was generated.
TRAN_ID	integer	Yes	The point of sale transaction number of the request.
CLASS_NAME	varchar(255)	No	The name of the Java class that generated the system_log record.

System_Log (continued)

Attribute	Datatype	Null?	Notes
METHOD_NAME	varchar(255)	Yes	The method name inside the class where the system_log record was generated.
LINE_NUMBER	numeric(18, 0)	Yes	The line number in the code where the log record was generated.
FILE_NAME	varchar(100)	Yes	The name of the Java file that generated the log record.
STACK_TRACE	varchar(5000)	Yes	The code stack trace.
LOG_SOURCE	integer	Yes	The source of the system_log record.
LOG_DATE	datetime	Yes	The date that the system_log record was created.

Tender_Type_Descr table

The Tender_Type_Descr table is a mapping table which is used to convert Xpay tender codes to textual descriptions that are displayed in the transaction history browser.

Tender_Type_Descr			
Attribute	**Datatype**	**Null?**	**Definition**
TENDER_TYPE (PK)	integer	No	The tender type code used by Xpay.
TENDER_DESCR	varchar(30)	Yes	The tender type description that will be displayed in the transaction browser.
ENABLED_FLAG	tinyint	No	Flag used to either display or hide certain tenders on the transaction browser.

Tran_Type_Descr table

The Tran_Type_Descr table is a mapping table which is used to convert Xpay transaction type codes to textual descriptions that are displayed in the transaction history browser.

Tran_Type_Descr			
Attribute	**Datatype**	**Null?**	**Definition**
TRAN_TYPE (PK)	integer	No	The transaction type code used by Xpay.
TRAN_DESCR	varchar(30)	Yes	The transaction type code description that will be displayed in the transaction browser.

Transaction_Header table

The Transaction_Header table contains one record for each transaction that is processed through the Xpay application.

Transaction_Header			
Attribute	**Datatype**	**Null?**	**Notes**
STORE_ID (PK)	varchar(10)	No	Store number.
REGISTER_ID (PK)	integer	No	Register number.
TRAN_ID (PK)	integer	No	Transaction number.
XPAY_SEQUENCE_ID (PK)	integer	No	Sequence number assigned to the transaction by Xpay.
EMPLOYEE_ID	varchar(16)	Yes	Employee Id
TRAN_START_TIME	datetime	Yes	Transaction's start date and time.
TRAN_AMOUNT	decimal(15, 2)	Yes	Transaction amount.
TRAN_TYPE (FK)	integer	Yes	Transaction type (purchase, void, recharge, etc.) From Tran_Type_Descr.TRAN_TYPE
TENDER_TYPE (FK)	integer	Yes	Tender type (Visa, MasterCard, Amex, etc.) From Tender_Type_Descr. TENDER_TYPE
RESPONSE_CODE	varchar(25)	Yes	Provider response code.
RESPONSE_DESCR	varchar(255)	Yes	Response description.
APPROVED_FLAG (FK)	tinyint	No	Flag indicating whether the transaction was approved or not. From Approval_Type_Descr.APPROVED _FLAG
RETURNED_ACI	varchar(10)	Yes	Authorization Control Indicator returned in the authorization response message.
BANK_REF_DATA	varchar(20)	Yes	Bank reference data returned in the authorization response message.

Transaction_Header (continued)

Attribute	Datatype	Null?	Notes
AVS_RESULT_CODE	varchar(10)	Yes	Address Verification Services result code returned in the authorization response message.
MC_TRAN_ID	varchar(10)	Yes	Possible MC transaction Id that may be returned in the authorization response message.
APPROVAL_CODE	varchar(10)	Yes	Approval code returned in the authorization response message for authorized transactions.
ERROR_CODE	integer	Yes	Error code sent back to the POS application.
TRAN_END_TIME	datetime	Yes	Date and time of transaction's end (when response is sent to the POS).
TRAN_DURATION	integer	Yes	The duration of transaction from start to finish in milliseconds.
SERVICE_ID	varchar(50)	Yes	The name of the processor class that handled the authorization request.
ACTION_CODE	integer	Yes	The transaction activity type code (for example, CREDIT PURCHASE).
ACTION_TEXT	varchar(255)	Yes	The text description for transaction activity type code.
POS_ACK_TIMESTAMP	datetime	Yes	The date and time for the POS Acknowledgement activity.
LOG_SOURCE	integer	Yes	The source of the transaction_header record.
LOG_DATE	datetime	Yes	The date that the transaction_header record was created.
COMMUNICATION_TYPE	integer	Yes	The communication type used by the processor (for example, tcp/ip, dial).
XPAY_SERVER_NAME	varchar(40)	Yes	The machine name of the Xpay server that processed the request.

Transaction_Log table

The Transaction_Log table stores the step-by-step details about each transaction that was submitted to Xpay. Records will be written for steps such as when Xpay received the POS request, when the request was passed to a processor thread, when the request message was created and sent to the processor, etc. The table will also contain a record for all exceptions that are thrown inside the application, whether they're associated with a transaction or not.

Transaction_Log			
Attribute	**Datatype**	**Null?**	**Notes**
LOG_ID_MSB (PK)	bigint	No	The unique UUID (Universal Unique Identifier) used as part of the primary key.
LOG_ID_LSB (PK)	bigint	No	UUID used as part of the primary key.
SERVICE_ID	varchar(50)	No	The name of the processor class that handled the authorization request.
XPAY_SEQUENCE_ID (FK)	integer	Yes	From Transaction_Header. XPAY_SEQUENCE_ID
STORE_ID (FK)	varchar(10)	Yes	From Transaction_Header.STORE_ID
REGISTER_ID (FK)	integer	Yes	From Transaction_Header. REGISTER_ID
TRAN_ID (FK)	integer	Yes	From Transaction_Header.TRAN_ID
LOG_DATE_TIME	datetime	No	The log data and time.
LOG_TEXT	varchar(5000)	No	The log text.
LOG_ACTIVITY	varchar(50)	Yes	The log activity code (for example, POS_ACK_RECEIVED).
LOG_SOURCE	integer	Yes	The source of the transaction_log record.
LOG_DATE	datetime	Yes	The date that the transaction_log record was created.

C H A P T E R 9

XPAY TROUBLESHOOTING

Overview

When troubleshooting Xpay, most issues are specific to a particular processor or configuration.

Xpay will save general log information to the `xpay\dtv-xpay-02_xx_xxx/xpay-output.log` file. This log file can be useful for troubleshooting any errors that may occur.

If it is an issue with authorizations, the first question to ask is: What types of authorizations are being affected?

- If it is all authorizations, there may be a general connectivity issue or a global configuration issue.

- If certain types of authorizations are having issues and others are not, it is likely to be an issue with a particular processor.

If you are unable to view transactions in the transaction viewer, it is possible that the database server has become unavailable.

Xpay Error Codes

If an error inside the Xpay application prevents a transaction authorization, a specific error code will be sent back to the POS application. The specific error code will be added to the <ErrorCode> node and the description will be added to the <ErrorText> node in the XML response message. The following is a list of error codes.

Socket Communication Error Codes 1000-1999

Table 9-1: *Socket Communication Error Codes*

Error Code	Description
1000	Cannot open socket connection to the processor.
1001	Processor socket connection broken.
1002	Socket binding exception (typically returned when the port is in use by another process).
1003	No route to host exception (typically returned due to firewall or router configuration errors).

Serial Communication Error Codes 2000-2999

Table 9-2: *Serial Communication Error Codes*

Error Code	Description
2000	Specified communication port doesn't exist.
2001	Serial port in use by another application.
2002	Cannot initialize modem.
2003	Modem not ready due to carrier detect holding.
2004	Line busy.
2005	No carrier.
2006	Carrier lost.
2007	No dial tone.
2008	Serial IO error. Concatenated with the exception description.

Xpay Processing Error Codes 3000-3999

Table 9-3: *Xpay Processing Error Codes*

Error Code	Description
3000	Error parsing XML request string. Invalid XML message.
3001	Required field missing in XML request. Concatenated with the missing node name.
3002	Error building request string.
3003	Error parsing processor response string.
3004	Transaction type not supported.
3005	Tender type not supported.
3006	Transaction timed out.
3007	Unhandled Java runtime exception thrown.

Database Access Error Codes 4000-4999

Table 9-4: *Database Access Error Codes*

Error Code	Description
4000	Cannot connect to database.
4001	Database connection lost.
4002	Exception thrown executing SQL statement. The actual statement will be contained in the log record.

A P P E N D I X

A

MERCHANTLINK CONFIGURATION

Overview

If you are using MerchantLink for credit authorization, follow configuration instructions in this section. The Xpay server connects to MerchantLink's new transaction vault gateway (TV2G).

Installing Xpay

To install and set up Xpay for MerchantLink:

1. Install Xpay using the `xpay-config.zip` file for MerchantLink.

> *If you don't have this config file, you can proceed with the installation but you need to configure Xpay through the GUI for MerchantLink after the installation. We recommend that you install Xpay at `C:\xpay` directory.*

2. MerchantLink provides two certificate files:

    a. `ca-cert.crt` (CA public cert)

    b. `.p12` file (client certificate)

> *MerchantLink also provides the password of the .p12 file. In the following instructions, this password is referred to as CLIENT_KEY_PASSWORD. Be sure to replace this text with the password you were given.*

3. Copy the two certificate files to `C:\xpay\dtv-xpay-02_00_`**`XXX`**`\config\merchantlink-ip`.

> 👉 ***XXX** is Xpay's build number.*

4. Modify the `C:\xpay\dtv-xpay-02_00_XXX\config\merchantlink-ip\ merchantlink-ip_processor-config-ex.xml` file:

    a. Check the **keyStoreFileName** property. Make sure that its value is the same as the .p12 file.

    b. Update the **keyStorePassword** property. Its value is the encrypted version of the CLIENT_KEY_PASSWORD provided by MerchantLink in .

5. To encrypt the password, open a command line and go to `C:\xpay\runtime\jdk1.6.0_33\jre\bin`.

> 👉 *The Java runtime may be different.*

    a. Execute the command below.

```
java -cp C:\xpay\dtv-xpay-02_00_XXX\lib\dtv-upgrader.jar
dtv.installer.util.StringEncrypter -e "CLIENT_KEY_PASSWORD"
carldrewjeffdanjuliaoliverxpay
```

> 👉
> - Replace the **CLIENT_KEY_PASSWORD** to the password provided.
> - Replace the ***XXX*** to Xpay's build number.

> ❗ *Note the resulting encrypted password. Use this as the new value of the **keyStorePassword** property.*

Installing MerchantLink's Certificates

To install MerchantLink's certificates to Xpay's server:

1. Execute the `.p12` file.

2. From the Certificate Install Wizard, click **Next**.

3. When prompted, type in the **CLIENT_KEY_PASSWORD** provided in and then click **Next**.

> ⚠️ *Do **not** select any check boxes.*

---

4. Click **Next** and then click **Finish**.

Import the CA Public Certificate

In order for Xpay's Java runtime to trust the installed certificate, we need to import the CA public certificate (`ca-cert.crt`) to Xpay's Java truststore:

1. Open a command prompt. Go to `C:\xpay\runtime\jdk1.6.0_33\jre\bin`.

2. Execute the keytool command as follows:

```
keytool –import –trustcacerts –alias "Server name" –file C:\xpay\dtv-xpay-
02_00_XXX\config\merchantlink-ip\ca-cert.crt –keystore
C:\xpay\runtime\jdk1.6.0_33\jre\lib\security\cacerts
```

> 👉 *Change the alias (Server name) as you desire. We recommend using the machine name of the server. Replace the **XXX** with the actual Xpay's build number (ex. 678).*

---

3. When prompted for a password, type **changeit**. This is the password of Java's truststore.

4. Restart the Xpay services. See *"Restarting Xpay and Xpay GUI Services - Windows" on page 34* for procedural information.

A P P E N D I X

B

REVISION HISTORY

Revision History 2.0, Doc Ver 02

Table B-1: *Revision History*

Xpay Version 2.0, Doc Ver 02 Description of Change	
Appendix A	■ Added MerchantLink Configuration Appendix
	■ Added double quotations around CLIENT_KEY_PASSWORD in the Installing Xpay section.