



OpenBRM User Guide V2.0

[Overview](#)

[Users](#)

[User Guide](#)

[Installation](#)

[Getting Started](#)

[Custom Business Logic](#)

[Database Guide](#)

[Translating Openbrm](#)

[Payment Processor](#)

[User Guide](#)

[Installation](#)

[Windows](#)

[To Install JRE on Windows](#)

[To Install openbrm on Windows](#)

[To Run openbrm](#)

[Mac / Linux](#)

[Upgrading](#)

[How to find the differences in a file](#)

[Building openbrm from Source](#)

[Building Openbrm from Source](#)

[Useful Links](#)

[Getting Started](#)

[How to Get Started](#)

[Welcome](#)

[Overview](#)

[Creating a Company](#)

[Email Instructions:](#)

[Entering Your Customer's Data](#)

[Product Categories](#)

[To Create a Category:](#)

[Products](#)

[TO CREATE A PRODUCT](#)

[Customers](#)

[TO CREATE AN ACCOUNT TYPE](#)

[TO CREATE AN ACCOUNT INFORMATION TYPE \(AIT\)](#)

[TO CREATE A CUSTOMER](#)

[Purchase Orders](#)

[TO CREATE A ORDER PERIOD](#)

[Order Periods](#)

[TO CREATE A PURCHASE ORDER:](#)

[Scenarios](#)

[System Configuration](#)

[Notifications](#)

[Email Delivery Parameters](#)

[The Collections Process](#)

[TO CONFIGURE THE COLLECTIONS PROCESS](#)

[The Billing Process](#)

[TO CONFIGURE THE BILLING PROCESS](#)

[Invoice Numbering](#)

[TO CHANGE THE INVOICE NUMBER](#)

[Payment Processors](#)

[TO CONFIGURE A PAYMENT METHOD](#)

[My First Invoice](#)

[Generating an Invoice](#)

[Editing Customer Information](#)

[Purchase Order Changes](#)

[Generate the Invoice](#)

[TO MANUALLY GENERATE AN INVOICE:](#)

[Pay the Invoice](#)

[TO PAY THE INVOICE:](#)

[What's Next?](#)

[Custom Business Logic](#)

[Introduction](#)

[Meta Fields](#)

[The Plug-In System](#)

[Database Guide](#)

[Database Guide](#)

[Configuring Hibernate](#)

[OPENBRM-DATASOURCE.GROOVY](#)

[Database Initialization](#)

[Database Compatibility](#)

[Additional Notes](#)

[MySQL](#)

[Oracle](#)

[Allow use of "&" when loading openbrm-data.sql](#)

[Use NUMBER\(1,0\) instead of BIT](#)

[MySQL Database Guide](#)

[Use the MyISAM database engine instead of InnoDB](#)

[Return NULL for null date/time values instead of 0](#)

[Use DATETIME instead of TIMESTAMP](#)

[Translating Openbrm](#)

[Translating Openbrm](#)

[Payment Processor](#)

[Configure Payment Processor](#)

[Merchant Account Request](#)

[Configuring Plug-In Within openbrm](#)

[1\)MERCHANT_ACCOUNT_CODE-VALUE PROVIDED BY INTRANNUITY](#)

[2\) PASSWORD-VALUE PROVIDED BY INTRANNUITY](#)

[3\)SUBMIT_AVS-TRUE/FALSE](#)

[4\)TEST-TRUE/FALSE](#)

[5\)TIMEOUT_SEC](#)

[Fig 1: Plug-in](#)

[Remove Default Payment Processing Plug-In](#)

[Completion](#)

Overview

Users

The information provided in this section applies to openbrm's Community Edition.

Below is a brief description of what each page covers in the User Section.

User Guide

A link to download the pdf User Guide for openbrm's Community Edition is available on this page. The User Guide provides a detailed and concise breakdown of all the functions of the system. It is broken down into fifteen chapters, followed by appendices.

Installation

To install openbrm's Community Edition on Windows, Mac or Linux, both written and video instructions are accessible from this page. As new updates are released frequently to provide new features and bug fixes, this page also includes steps for upgrading openbrm to the latest version.

Instructions on Building openbrm from Source are also available.

Getting Started

This page is highly recommended to all new users of openbrm! The Getting Started Guide uses a fictional company example to help familiarize users with the basics of the system. Three of openbrm's main features are covered here: Entering your Customer's Data, System Configuration, and My First Invoice.

Custom Business Logic

Custom business logic is easily handled through the management of plug-ins. Provided here is a guide to setting up plug-ins to meet the specific custom logic of your business needs.

Database Guide

This guide provides instructions on how to configure openbrm if you are using a database other than Hypersonic (the default bundled database). As a general rule, if you are planning to run a more demanding environment, you will want to configure openbrm to use a different database. This guide also contains a list of databases and their exposure with openbrm, and information on MySQL Database.

Translating Openbrm

As openbrm is used all over the world, one of its best features is the ability to configure different elements based on the needs of a country. Follow the steps to add a new language to the system by translating the text in two files.

Payment Processor

While there are many payment plug-ins bundled in with openbrm, this page provides a concrete use case. Here you will find instructions for configuring the payment plug-in used to submit payments with Intrannuity.

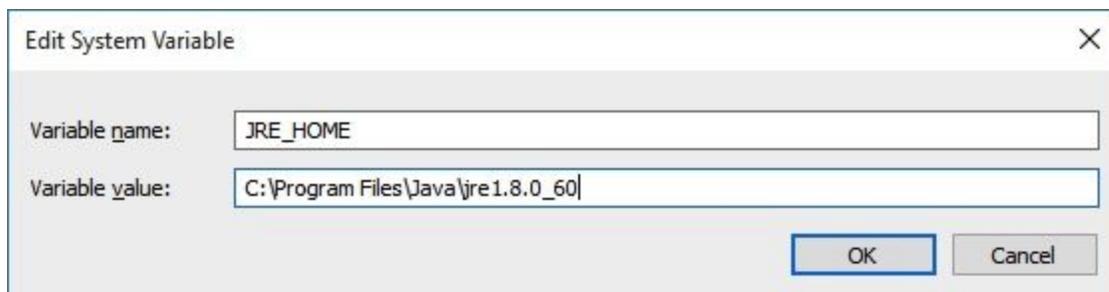
User Guide

Installation

Windows

To Install JRE on Windows

1. Determine if Windows is 32-bit or 64-bit:
 - Go to the Start Menu > Control Panel > System. The System Type properties will indicate 32-bit or 64-bit.
2. Download current Java Runtime Environment (JRE):
 - Go to the Java SE download site - [click HERE](#).
 - Select the current Java Runtime Environment (JRE).
3. Install the Java Runtime Environment (JRE). Double-click on the downloaded JRE file to install. Follow the prompts given by the Java SE Installation Wizard.
4. Add the JRE bin directory to the Windows PATH environment variable:
 - Go to the Start Menu > Control Panel > System > Advanced System Settings > Environment Variables > System Variables.
 - Click New... and add:
 - Variable name: JRE_HOME
 - Variable value: C:\Program Files\Java\jre1.8.0_60 (location and name of JRE on your computer).



To Install openbrm on Windows

1. Download the most recent version of openbrm on [SourceForge](#).
 - Please note: Ensure that you download the binary package, and not the source. The source has the suffix '-src'.
 - Example: Download openbrm.zip,
2. Unzip the downloaded openbrm file:
 - To unzip, right click on the zipped file. A shortcut menu will appear.
 - Select the menu option: Extract All. A new unzipped folder with the same title (openbrm) will appear in your Downloads folder.
3. Copy and paste the unzipped openbrm folder from your Downloads folder into your C:\Applications folder.
 - Please Note: If you do not have an Applications folder in your C: drive, you will need to create one.

To Run openbrm

1. In Windows Explorer / File Explorer, navigate to the openbrm directory:
C:\Applications\openbrm\bin.
2. Double click on the startup.bat file. Tomcat will open in a separate window.
3. Wait for Tomcat to finish starting up.
 - When it is complete, the text: INFO: Server startup in (a number) ms will appear in the Tomcat window.
 - Minimize the Tomcat window.
4. In your browser, login to openbrm and create your own company.
5. To create your own company, [click HERE](#).
6. To login after creating your company, [click HERE](#).
7. Follow the steps provided in the [Getting Started Guide](#) to begin using openbrm.

Mac / Linux

In 11 easy steps. Use the video and read along for best results.

1. Download openbrm here: <http://sourceforge.net/projects/openbrm/>
2. Unzip openbrm by double-clicking the .zip.
3. Make a new folder in your home folder (or wherever you'd like), called 'openbrm_install'
4. Move the contents of the openbrm extraction to this new folder.
5. Your openbrm source files are now located at
/users/YOURHOMEFOLDER/openbrm_install/
6. Open the application Terminal (located in Applications > Utilities).
7. Navigate to /users/YOURHOMEFOLDER/openbrm_install/bin by typing: `cd
~/openbrm_install/bin`
8. Make all current directory files executable by typing: `chmod +x *.sh`
9. At this point, you're ready to start up openbrm. Do so by typing: `./startup.sh`
10. The terminal should return five lines of installation logs, indicating the startup has been completed and openbrm is now set up!
11. Navigate your web browser to <http://localhost:8080/openbrm/signup> to sign up your username. Or try this link to login with the test credentials (Username: admin / Password: 123qwe) so you can try out openbrm 3!

Getting Started

How to Get Started

Welcome

Hello, and welcome to openbrm! openbrm is a mature product that generates invoices and processes payments for companies of all sizes. Thousands of users across the world are thrilled with the freedom and possibilities this billing and invoicing system provides.

By choosing openbrm, you will save countless hours of tedious, manual work. Some of openbrm's time-saving features include: invoice generation, following complex business rules and taxes, customer self-care through its web-based interface, and automatic payment processing.

Getting openbrm set up and ready to run does require some effort. Are you going to accept credit cards? How often are your customers going to get invoices? How are they going to receive these invoices? How much time can a customer take to pay an invoice? What happens if a customer doesn't pay? You know the answers to these questions. Eventually, you will need to let openbrm know so that it can follow your business rules while processing your billing.

The purpose of this guide, however, is for you follow instructions that are designed to help you become familiarized with the basics of openbrm. If you need more information on how to install openbrm, click [here](#).

Get yourself a cup of hot chocolate, be patient, and keep reading! Running openbrm is well worth the effort.

Overview

This guide is made up of the following four sections:

1. Creating a Company
2. Entering Your Customer's Data
3. System Configuration
4. My First Invoice

The steps provided in these sections will leave you with a better understanding of how to set up and run through openbrm's billing and invoicing process. Let's take a look at a high level overview of this process now:

Products represent the goods and services that your company offers. Your customers buy products by placing purchase orders. Periodically, a batch process called the billing process will run and generate invoices based on these purchase orders. Your customers will then pay their invoices by submitting payments, or the system can automatically process these payments through its automatic payment processing feature.

Creating a Company

After you install openbrm you need to create a company before you can use the system.

To configure a Company, navigate to your openbrm home page. The URL should look something like this:

<http://localhost:8080/openbrm/login>

The key here is to remove all text starting from 'login' and replace it with: 'signup'. For example:

<http://localhost:8080/openbrm/signup>

Hitting the 'Enter' key will refresh to the company sign-up screen. Enter your company information in the appropriate fields. Refer to the steps below as an example:

1. In the Login Name field enter, admin.
2. In the First Name field enter, Sarah.
3. In the Last Name field enter, Wilson.
4. In the Phone Number field enter, 123-456-7890.
5. In the Email field enter a valid email address for the Company Super User Administrator. Once the company is saved, the system will send an email with a link for the user to create their password (for example: admin@edgeset.com).
6. In the Language field select, English.
7. In the Currency field select, United States Dollar.
8. In the Organization Name field enter, Edgeset Solutions.
9. In the Address field enter 123 Street.
10. In the State/Province Field enter New York.
11. In the Country drop-down field enter United States.
12. In the Zip/Postal Code field enter 90456.
13. Click on the 'Save Changes' button. openbrm will refresh to the Login screen, and Edgeset Solutions is now available in the Company drop-down menu.

To login to your company:

1. Click on the link provided in the email sent by the system. (Please note: if you are running openbrm locally without an email server configured, refer to the instructions below for details on how to retrieve the Reset Password link.)
2. Create a Password (example: 123qweQWE!). The system will redirect you to the login screen.
3. Select the company from the drop-down menu and enter valid login credentials.
4. In the Login field enter, admin.
5. In the Password field enter your password.
6. Click on the 'Login' button.

Email Instructions:

It is important to note that when openbrm is run locally without an email server configured, all emails are sent to a file in the Resources folder of the application.

Example: C:\Applications\openbrm\openbrm\resources\emails_sent.txt

To retrieve this link and view the Reset Password screen: Copy the applicable link from the email_sent.txt file and paste into the address bar of your browser. In the link, replace the text "www.yourcompany.com" with "localhost:8080", and press Enter. The Reset Password screen will appear.

For more information on configuring email, please refer to the Notifications chapter of the Community Edition User Guide.

You are now ready to enter information into openbrm. Let's get started!

Entering Your Customer's Data

This section reviews the basic data you will need to enter into openbrm:

- Product Categories: These help to group your products.
- Products: Represent the catalog of goods or services your company sells.
- Customers: These people buy from you, and pay!
- Purchase orders: A customer's subscription or purchase from your company.
- Order periods: Represents how often your customers receive an invoice.

Product Categories

Product categories are the first thing you need to enter into openbrm. It is possible to create as many categories as you want. Categories help to group products. This grouping will be helpful later on when you run a report, or for invoice calculations.

To Create a Category:

1. In the navigation pane, Click on the Products tab. A list of categories will appear in the middle pane on your screen.
2. Click on the '+ Add Category' button. The Add Product Category screen will appear.
3. Ensure that Items is selected in the drop-down pane beside Type. Please note: The category type lets openbrm know how to treat the products that belong to this category. A regular product should belong to the Items category type.
4. Enter the category name into the text box beside Name. (Example: Hardware)
5. Click on the 'Save Changes' button to save the category. You will return to the Products screen. Or,
6. Click on the 'Cancel' button to discard the category. You will return to the Products screen.

Products

Products are the catalog of goods and services your company sells. Keep in mind that there are other factors that affect how much your customers pay you, such as taxes or fees. These too must be represented as products.

TO CREATE A PRODUCT

1. Select the category you created.
2. Click on the '+ Add Product' button in the Products pane. The page will refresh to the Add Product screen.
3. Beside the Add Description field, click on the green plus sign '+'. The English Description field will appear.
4. In the English Description field, enter the product name. (Example: Modem)
5. In the Product Code field, enter the product's code. (Example: 'M-01') Please Note that this field determines how products will be ordered in an invoice.
6. Make sure that your category is selected in the Categories pane.
7. Under Prices, in the text box provided, enter: 100.
8. Click on the 'Save Changes' button. You will return to the Products screen. The newly added product is available in the Products list. Or,

9. Click on the 'Cancel' button to disregard any changes. You will return to the Products screen.

You have probably noticed that several fields in this area have not been addressed. There is also more to learn about the fields described above. For more information, please refer to openbrm's Community Edition User Guide.

Customers

Now that you have something to 'sell,' you need to create an Account Type and Account Information Types (AITs). Then you can create your customers - the individuals or companies that purchase your goods and services.

TO CREATE AN ACCOUNT TYPE

Account Types help you to organize groups of similar customers. The following steps include the minimal information required to configure an account type. For more details, please refer to openbrm's Community Edition User Guide.

1. In the navigation pane at the top of the screen, click on the Configuration tab. Preferences and the configuration menu will appear.
2. From the menu on the left of the screen, select: Account Types. The account types area of the system will appear.
3. Click on the 'Add New' button. A new account type screen will appear.
4. In the name field, enter the name of the account type (example: Government)
5. Click on the 'Save Changes' button. You will return to the Account Type list screen. The newly added account type will appear in the list. Or,
6. Click on the 'Cancel' button to disregard any changes. You will return to the account type list screen.

TO CREATE AN ACCOUNT INFORMATION TYPE (AIT)

Account Information Types define the fields that will appear on the customer screen when the account type is selected. For more information on this feature, please refer to openbrm's Community Edition User Guide.

1. From the Account Type screen, select the Account Type you created. Its details will appear on the right.
2. Click on the 'Add Information Type' button. The Account Information Type area of the system will appear.
3. In the Name field enter: Email Information.
4. In the Display Order field enter: 1.

5. Select the Include in Notifications field.
6. Click on the 'Add New Meta Field' button. The details to configure a meta field will appear on the right.
7. In the Name field enter: Email.
8. Select the Mandatory field.
9. In the MetaField Type drop down select: Email.
10. In the Validation drop down select: Email.
11. Beside the Error Message field, click on the green plus sign '+'. A field called, English Error Message will appear.
12. Enter: Please provide email address.
13. Click on the 'Update ' button.
14. Click on the 'Save Changes' button. The AIT is saved to the Account Type. Or
15. Click on the 'Cancel' button to disregard any changes. You will return to the Account Type list screen.

TO CREATE A CUSTOMER

In the navigation pane at the top of the screen, click on the Customers tab. The system will direct you to the Customers screen.

1. Click on the '+ Add New' button. From here, select which Account Type you want to create the customer under.
2. Click on the 'Select' button. The New Customer screen will appear.
3. The New Customer form contains information about a customer, including any Account Information Type meta fields you configure.
4. In the Login Name field, enter 'bsmith'.
5. In the Email field, enter 'bsmith@acme.com'.
6. Scroll down to the bottom of the screen and click on the 'Save Changes' button to create the customer. Or
7. Click on the 'Cancel' button to disregard any changes. You will return to the Customer screen.

More information about this customer, such as organization name, first name, last name, and address, can also be entered by creating additional meta fields in the AIT area of the system (if desired or required).

Purchase Orders

Now that you have a customer, it is time to define how often he or she will be billed, and therefore, how often he or she should pay for the products on a purchase order.

The most common order period is monthly. openbrm however, can handle any number of days, weeks, months, or years as an order period.

Follow the instructions below to create two order periods: one monthly, and one for a three month period.

TO CREATE A ORDER PERIOD

In the navigation pane at the top of the screen, click on the Configuration tab. The configuration area of the system will appear.

1. From the menu on the left, select, Order Periods. The Order Periods pane will appear.
2. Click the 'Add New' button.
3. In the value field, enter: 3.
4. From the Unit drop down list, select Month.
5. In the Description field, enter: Quarterly.
6. Click the 'Save Changes' button to save the order period. To discard any changes, click the 'Cancel' button. You will return to the Configuration screen.
7. Repeat steps 3 to 4 and enter the following: Value: 1, Unit: Month, and Description: Monthly.

Order Periods

Congratulations! Now that you have a customer and something to sell to them, you are ready to make some sales.

A purchase order in openbrm represents your company's sales. This could be a one-time purchase or a subscription for which the customer pays regularly (recurring). It might be helpful to think of a purchase order like a shopping bag, full of the many different products that a customer buys.

TO CREATE A PURCHASE ORDER:

1. From the Customer tab, ensure the customer you created is selected. Their information will display on the right.
2. Scroll down to the bottom of the page.
3. Select the 'Create Order' button.
4. Select 'Monthly' from the dropdown list beside Period.
5. Select 'pre paid' from the dropdown list beside Type to charge the customer in advance for the service. Select 'post paid' from the dropdown list beside Type to charge the customer for the services after the period is finished. (For more information on this field, please refer to the openbrm Community Edition User Guide).

6. Click on the Products tab, located beside the Sub-Orders tab. You will see a list of all products.
7. Find your product and click on it. The product will appear in the Line Changes pane on the right.
8. Click the 'Update' button. The product is listed in the Review pane on the right. (To remove the product, click the 'Remove' button).
9. Click the 'Save Changes' button to save the purchase order. The order appears under the Orders tab. (To disregard any changes, click the 'Delete' button. You will return to the Customer screen).

Your new order will now appear under the Orders tab, in the Orders list. From here, you can edit or delete the purchase order by selecting it.

Scenarios

So far, you have been working with simple examples. You might be wondering if other scenarios common to your company can be addressed using openbrm. It is actually very hard to come up with a billing scenario that openbrm cannot handle. Setting up your system is all about good management of products and orders. To learn more, please refer to the openbrm Community Edition User Guide.

System Configuration

Data alone is not enough to run your billing. You also need to tell openbrm about the billing process itself, as well as other system-wide parameters. This guide will briefly address the following aspects in order for you to understand how to run billing:

- Notifications
- Email Delivery Parameters
- The Collection Process
- The Billing Process
- Invoice Numbering
- Payment Processing

For more detailed information on any of the above topics, please refer to the openbrm Community Edition User Guide.

Notifications

Openbrm will notify your customers about various events, the most important one being any new invoice.

To access the Notification area of the system:

- Clicking the Configuration tab, and then Notifications from the menu on the left it will reveal four notification categories:
 1. **Invoices:** Remind your customers that an invoice has been sent to them. They can be sent after an invoice has been generated, and until that invoice is paid or reaches its due date (whichever comes first).
 2. **Orders:** Remind your customers that their purchase orders will expire on a certain date.
 3. **Payments:** Let your customers know if their payments have been successful, or if they have failed.
 4. **Users:** Let your customers know if they have overdue payments, if they have a suspended account, if they have been deleted, if they have lost their password, or if they need to provide updated credit card information.

Email Delivery Parameters

Openbrm will send emails to customers on your behalf for multiple reasons. Sending them invoices is one reason, but reminders, failure to receive a payment, balance overdue, and others, can also be sent. You can opt out of sending any emails, but consider that this service is completely free and, once well configured, will save you a considerable amount of time.

For more information, please refer to the Community Edition User Guide.

The Collections Process

What happens if the due date has come and gone and the customer has not yet paid? This is an issue addressed by the collection process. This process notifies your customers about their overdue invoices, triggers payment retries, and changes the status of that customer's account.

For example, you might have several steps in your collection process: A grace period, a few payment retries, some reminder notifications and eventual suspension (if the customer still does not provide payment).

TO CONFIGURE THE COLLECTIONS PROCESS

1. In the navigation pane at the top of the screen, click on the Configuration tab. The configuration area of the system will appear.
2. From the menu on the left, click on Collections. The collections area of the system will appear.
3. In the Steps field, enter: Grace Period
4. In the For Days field, enter: 5
5. Select the Notification field.
6. To add another step, click on the green plus sign '+' located at the end of the row. A blank row will appear.
7. Add the following steps:
8. Step: Payment, For Days: 10, Select: Payment
9. Step: Warning, For Days: 11, Select: Notification
10. Step: First Retry, For Days: 15, Select: Payment
11. Step: Second Retry, For Days: 16, Select: Payment
12. Step: Suspend, For Days: 30, Select: Notification and Suspend.
13. Click the 'Save Changes' button to save your Collections Process. Or,
14. Click the 'Cancel' button to return to the main Configuration screen.

The Billing Process

The billing process is the batch process that will generate invoices. Having a good understanding of what it does is important. For more detailed information, please refer to the Community Edition User Guide.

TO CONFIGURE THE BILLING PROCESS

1. Under the Configuration tab, from the menu on the left, click on Billing Process.
2. Click the calendar icon in the Next Run Date field and select the date for the first day of next month. Please note: This field automatically updates by the billing process when it runs.
3. Select the check box beside Generate Review Report.
4. Enter '3' in the Days to Review Report field.
5. Enter '1' and select Month for the Billing Period field.
6. Click the 'Save Changes' button. Or,
7. Click the 'Cancel' button to cancel the changes and return to the Configuration screen.

Invoice Numbering

Openbrm will apply the number 1 to the first invoice generated. If, however, your company has generated previous invoices, it is necessary to tell the system about this number so it can properly number the next invoices.

TO CHANGE THE INVOICE NUMBER

1. Under the Configuration tab, from the menu on the left, click on Invoice Display.
2. In the Next Invoice Number field, you can change the value to any number you want to start invoicing your customers with.
3. Click on the 'Save Changes' button to save the number. Or,
4. Click on the 'Cancel' button.

Payment Processors

This section only applies if you plan on using automatic payment processing (credit cards and direct debit).

When openbrm runs the Billing or Collections Process, it can automatically pay generated invoices with your customers' credit cards or banking information. This allows for several other automation features: payment retries, notification to your customers about the results of their payments (usually via email), etc.

Each payment processor has its own API. In plain English, each payment processor talks a different language so openbrm needs an 'interface' for each of them. openbrm supports many payment processors like, Authorize.net, Sage Pay, and PayPal (to name a few). Creating new interfaces to support other payment processors is not difficult, but it requires the development of a plug-in which is out of the scope of this guide.

You do however, have the ability to add payment methods from openbrm's user interface. The options are: Payment Card, ACH (Direct Debit) and Cheque.

TO CONFIGURE A PAYMENT METHOD

1. In the navigation pane at the top of the screen, click on the Configuration tab. The configuration area of the system will appear.
2. From the menu on the left, select: Payment Method. The payment method area of the system will appear.
3. Click on the 'Add New' button. The option to select a payment information type will appear.
4. In the Payment Method Template drop down select: Payment Method. The payment method configuration screen will appear.
5. In the Method Name field enter: VISA.

6. Select: Is Recurring?
7. Select: All Account Types.
8. Click the 'Save Changes' button to save your Collections Process. Or,
9. Click the 'Cancel' button to return to the main Configuration screen.

For more information about payments and payment methods, please refer to the Community Edition User Guide

My First Invoice

Congratulations! You have completed learning about some areas of configuration within openbrm. You are now ready to generate invoices.

Generating an Invoice

Invoices are created from purchase orders. This ensures that your customers do not get billed unless they have bought something from you first. Never modify an invoice directly; instead, adjust the purchase order, as they are the templates from which the customer's invoices are generated.. The idea is to have a consistent model where you can trace documents and the reasons for their existence.

Even though you have entered a customer and his or her purchase order, you still do not have any invoices. There are two ways to generate an invoice in openbrm:

1. **Billing Process:** This is the recommended way to generate invoices. It is an automated process that runs periodically, going through all orders and generating invoices only when necessary.
2. **Manually:** It is possible to create an invoice out of the configured order. This is not the way to generate the bulk of your invoices, but it is helpful when you want to create an invoice immediately, instead of waiting for the next billing process to run.

Ideally, you would have already completed entering all your data before you move to generating the invoice. However, this example requires you to edit information for your customer and purchase order first.

Editing Customer Information

Let's edit some of Brian Smith's information:

1. Access the Customers tab.
2. Click on your customer, Brian Smith. Information will appear on the right.

3. Scroll down and click on the 'Edit' button. Your screen will refresh to the Edit Customer page.
4. Change the Invoice and Delivery Method to: email and paper.
5. In the Payment Method pane, Payment Method Type drop down, select: VISA. The associated fields will appear.
6. In the Processing Order field enter: 1.
7. In cc.cardholder.name enter: Brian Smith.
8. In the cc.number field enter credit card number: 4111111111111152.
9. In the cc.expiry.date field enter: 11/2020.
10. Click the 'Save Changes' button to save the changes you have made. OR,
11. Click the 'Cancel' button to disregard any changes and return to the Customer screen.

Purchase Order Changes

Click on the Orders tab, located at the top of your screen. Find and click on the purchase order you created for Brian Smith. You will be changing or editing this purchase order so that it is only active for six months.

1. Select the purchase order. A summary of its details will appear on the right.
2. Scroll down and click on the 'Edit this Order' button. The purchase order screen will appear.
3. Click on the calendar icon beside Active Until and select the date that is six months from today's date. For example, if today's date is June 1st, 2012, you would select, December 1st, 2012.
4. Click on the 'Save Changes' button. You will be returned to the Orders screen. Or,
5. Click on the 'Cancel' button to disregard any changes. You will be returned to the Orders screen.

Generate the Invoice

The order is ready! It is now possible for you to create the first invoice. For this example, you will be manually triggering the invoice generation.

Please note: This is not the recommended way to generate an invoice. If you leave the order the way it is, the next billing process will automatically generate the invoice for you. However, in this example, you want the invoice right now and can't wait.

TO MANUALLY GENERATE AN INVOICE:

1. Ensure that the purchase order you have just edited is selected.

2. Scroll down and click on the 'Generate Invoice' button. You will be redirected to the Invoices screen.

Pay the Invoice

Last but not least, you want this invoice paid.

TO PAY THE INVOICE:

1. On the invoice you selected in the above step, scroll down and click on the 'Pay Invoice' button. You will be redirected to the Payments and Refunds screen.
2. In the Payable Invoices pane, ensure that the button beside your invoice is selected. Please note: It is also possible to review the invoice on this screen by clicking on the View Invoice link, located beside the Due Date column.
3. Scroll down. You will notice that the total amount of the invoice appears in the Payment Amount text box.
4. Click on the check box beside Process Payment in Real-Time so that the payment is sent to the payment processor for immediate approval.
5. Confirm that the credit card screen appears with the customer credit card information already populated in the appropriate fields.
6. Click on the 'Make Payment' button, located at the bottom of the New Payment screen. You will be redirected to the Confirm Payment screen. Please note: This is your last chance to ensure that all the data is correct before it is applied.
7. Click on the 'Make Payment' button located at the bottom of the screen to complete the payment. Your invoice will appear in the Payments and Refunds screen, with a message indicating that you have entered a new payment and it is being processed. Or,
8. Click on the 'Cancel' button. Your changes will not be saved and you will be returned to the Payments and Refunds screen.

What's Next?

Well done! You have just completed the Getting Started guide. As stated throughout, if you require more information or further clarification on openbrm, please refer to the Community Edition User Guide.

Your next stop is the Getting Started - BRMS guide. It covers the usage of rules for rating, complex pricing and product relationship management through a Business Rules Management System (BRMS).

Custom Business Logic

Introduction

Starting with version 3.1, openbrm provides you with it's own simple and accessible method for handling custom business logic through the use of easy to manage plug-in extensions. Earlier versions required the use of separate BRMS systems with their own GUI, in particular JBoss/Drools. This required a good deal more technical expertise and while still available to use, it is no longer the default for openbrm. If you wish to read our old BRMS guide which includes examples and information about Drools please click [here](#).

It is important to note that this guide is mostly applicable to the Community Edition. The Enterprise Edition of openbrm already comes with built-in support for complex pricing, plans, and bundles. In the Enterprise Edition, complex pricing is done with pricing models, which are extensible components that encapsulate pricing logic. Plans, and bundles, group and price products in a way that greatly simplifies product management.

This is not to say that plug-ins with custom logic are not used in the Enterprise Edition; they are, but mostly for areas other than pricing and bundling.

There are numerous benefits for openbrm's decision to run custom logic from plug-ins.

- With the business logic contained in the plug-ins, you do not need to change the core code. This means upgrading your logic is a snap and, conversely, you can just as easily upgrade your application without any merges: your custom logic is safely separated in plug-ins.
- openbrm stands for 'Java Billing'. We love Java. Writing custom logic in Java just makes sense.

Your billing system should always reflect your business rules. In many cases, this will lead to ever-increasing complexity for the billing system. For example, it is very common that a product needs more than just a simple flat price. Many factors might need to be considered to determine that price, such as a volume discount, a limited-time promotion, or pricing based on the location of the customer. The possible options are endless.

Another area of a billing system that requires great flexibility is the relationship among products. If you could clearly establish 'commands' to how products relate to each other, then you could implement business rules such as: "if you buy 10 books, you get a free calendar," or "bundle A includes a book, a calendar, and a poster."

Those are just two simple examples of what could be a source of great complexity in a billing system. How can your billing system provide you with the necessary flexibility you require to have all your business rules successfully implemented? Using plug-ins allow you to apply operational decisions based on company policy. It also gives you the ability to maintain or modify these rules without needing to change the core code of the application, providing you with increased control over your billing system for compliance and better business management.

This guide is an overview of how to implement these plug ins to easily manage your business logic with openbrm. For more detailed documentation, read the [openbrm Extension Guide](#). If you have not done so, please read the Getting Started/How to Get Started guide that will walk you through the basics of customer and product creation, and how to create your first invoice. This will provide you with an order for B.Smith, once that is in place we can begin.

Meta Fields

Before you begin to implement and make changes to the openbrm plug-in system. It is helpful to know a little bit about meta fields. You will need to implement a meta field before you work with the example plug-in.

Meta fields were introduced first in openbrm 3.1.0 Community Edition. A meta field is simply an extra field you add to a particular form within your billing system. openbrm gives you valuable flexibility by providing you with the ability to create fields in one of five billing areas called “Entity Types”.

- Customer
- Product
- Order
- Invoice
- Payment

Creating a meta field is easy. For the plug-in example that follows you will need to create the following meta field.

1. Click on the configuration tab and then in the sidebar on the left click on the meta field option. This should bring up the five entity types we listed above to choose from.

Number **New** <- An ID # will be assigned automatically

Name

Data Type

Mandatory

Disabled

Display order

Default value

2. Choose the entity type, "Product". At this point there will be no meta fields entered so click on the "+ Add New" button on the right to take you to the meta field form.
3. Notice the first field "Number" is listed as "New". This will be the meta field id # and it will not be assigned until after the meta field is created. You will need this id # for the plug-in example. Please make a note of it.
4. Fill out the following fields; Name = "Discount", Data type = "Decimal", and Display Order = 1. Leave Default Value blank.

All	Entity Type
Aging	CUSTOMER
Billing Process	
Blacklist	PRODUCT
Company	
Currencies	ORDER
Rate Cards	
Email	INVOICE
Invoice Display	
Mediation	PAYMENT
Notification	
Order Periods	
Plug-ins	
Users	
Contact Types	
Enumerations	
Meta Fields	

5. Ensure the check boxes Mandatory and Disabled are not selected and click Save Changes.

You should be back at meta field configuration page. If you like, you can go back to the product creation page to see that your newly created field "Discount" is now part of the product creation form. You will be using the information in this field in our next example.

The Plug-In System

To make your billing system as easy to use as possible openbrm has implemented a system of plug-ins based on Strategy and Chain of Responsibility designs. This allows you to create the steps needed, (using the plug-ins provided or plug-ins you create yourself), that your business rules require and excluding those that do not apply. For example, you might normally add all products ordered to your invoice but then have a series of taxes such as federal, state, local and maybe a VAT, (Value Added Tax), at the end. You would simply select and configure the plug-ins needed to create the tax rules and openbrm will take care of the rest. Let's walk through a particular example to show you how simple business rules can be established, managed, and maintained.

The example we will use is a basic product discount rule. You will create and maintain a category of products you wish to sell at a discount. You should have already created a customer called B.Smith along with a product and category. You will now add a new category for discount products, modify it, and apply your discount rule by implementing a supplied plug-in.

1. Create a new category of products called "Discount products". Within that category create a product called "Discount Banners". You will see your new Discount Meta Field; For now continue to keep the meta field "Discount" blank. Keep track of the category id # as you will be needing it soon.

Unique ID number

24

Category

Product pricing

Type

com.sapienter.jbilling.server.item.tasks.DiscountPricingTask



Plug-In drop down list

Order

1



Rule Precedence Order

Discount Category Id

201

Discount Percentage Meta

1

Field Id

Default Discount Percentage

10

Values to determine where you want to apply the rule and what you want to apply.

2. Click on the configuration tab at the top of the screen and then click on plug-ins on the left side bar. This brings up a full list of all installed plug-ins. Changing any of the pre-installed plug-ins without a full understanding of what they do may result in unwanted results so please be careful.



3. Take a moment to click on at least one of these plug-in categories. You will see some pre-installed plug-ins displayed to the right. Click on any of these displayed plug-ins to bring up a detailed description of what that plug-in does.
4. In the list of plug-ins categories you will see a category called "Product Pricing". Click on it to highlight this category. There is no default plug-in implemented yet but we will install one now by clicking the "+ Add New Button" that will be on the right. This should take you to the plug-in page.

All	Plug-ins categories
Aging	Item management and order line total calculation <i>com.sapienter.jbilling.server.pluggableTask.OrderProcessingTask</i>
Billing Process	Billing process: order filters <i>com.sapienter.jbilling.server.pluggableTask.OrderFilterTask</i>
Blacklist	Billing process: invoice filters <i>com.sapienter.jbilling.server.pluggableTask.InvoiceFilterTask</i>
Company	Invoice presentation <i>com.sapienter.jbilling.server.pluggableTask.InvoiceCompositionTask</i>
Currencies	Billing process: order periods calculation <i>com.sapienter.jbilling.server.pluggableTask.OrderPeriodTask</i>
Rate Cards	Payment gateway integration <i>com.sapienter.jbilling.server.pluggableTask.PaymentTask</i>
Email	Notifications <i>com.sapienter.jbilling.server.pluggableTask.NotificationTask</i>
Invoice Display	Payment instrument selection <i>com.sapienter.jbilling.server.pluggableTask.PaymentInfoTask</i>
Mediation	Penalties for overdue invoices <i>com.sapienter.jbilling.server.pluggableTask.PenaltyTask</i>
Notification	Alarms when a payment gateway is down <i>com.sapienter.jbilling.server.pluggableTask.ProcessorAlarm</i>
Order Periods	Subscription status manager <i>com.sapienter.jbilling.server.user.tasks.ISubscriptionStatusManager</i>
Plug-ins	Parameters for asynchronous payment processing <i>com.sapienter.jbilling.server.payment.tasks.IAsyncPaymentParameters</i>
Users	Add one product to order <i>com.sapienter.jbilling.server.item.tasks.IItemPurchaseManager</i>
Contact Types	Product pricing <i>com.sapienter.jbilling.server.item.tasks.IPricing</i>
Enumerations	Mediation Reader <i>com.sapienter.jbilling.server.mediation.task.IMediationReader</i>
Meta Fields	
Shortcuts	
Recent Items	
△ Product 201	
③ Order 300	
☑ Customer 41	
③ Order 300	
③ Order 201	

- You will see a drop down list with several similar sounding plug-ins. Select the one ending with “.DiscountPricingTask”.
- Once selected you will see a list of windows to fill in. These are the parameters needed to provide openbrm with the right information to implement your new rule. Different Plug-Ins will require different fields.
- The first box is labeled “Order” and this allows you to determine the order in which the rules will be applied. Please enter “1” here.
- The next parameter is the Discount Category ID box. Please enter the category id number you have for the “Discount products” category. This will apply the discount to that category of products
- You can now define the discount percentage at the product level using a meta field by entering the meta field id and enter 10 in the “Default Discount Percentage” box. Whole numbers here represent the percentage amount so 10 = 10%. Your fields should look like this.

Order #203		Monthly, Pre Paid
bsmith		01/07/2013 To 04/01/2013
Banners	1 x US\$100.00	US\$100.00
Discounted Banners	1 x US\$90.00	US\$90.00
		Total = US\$190.00

10. Now, return to use that category and you will see the prices in the product category have not changed, but when applied to an order, the price has been reduced by 10 percent.

Product ID 201

Add Description

English Description

Line Percentage

Allow decimal quantity

Discount

Also, you may now go and add new products or edit pre-existing ones and, in the discount meta-field, determine individual discount prices for those particular products. Try this out by entering 15 in the “Discount” meta field, and you will see it overrides the default value of the plug-in.

Even simpler you can skip the meta field entirely. Just use the plug-in directly as a default value for your entire discount line of products. This takes away your ability to modify individual products, but it does help you see just how flexible and scalable openbrm’s plug-in system can be. Including your own plug-ins you build in Java can help your billing system grow even further providing you with ultimate customization.

As you can see, there is a lot you can do with this simple plug-in that is provided mostly just as an example. The best part really is that you have the source of the plug-in. You can change it, enhance it, extend it, or create a new one entirely, all in Java. This is open source enterprise billing software at its best.

Openbrm provides you with the tools you need to implement all your business rules quickly and easily, without needing to change core code. You simply chain together the list of plug-ins to generate the invoices you need.

Database Guide

Database Guide

By default, the openbrm binary distribution runs on Hypersonic, a simple database written entirely in Java. Hypersonic provides a good example of a fully functional openbrm installation, but if you're planning to run a more demanding environment, you'll want to configure openbrm to use a different database vendor.

openbrm has been designed and built to be database vendor agnostic; it doesn't matter what database engine you run it on. This is done by using standard ANSI SQL statements combined with Hibernate's SQL dialect feature.

To configure openbrm to work with a different vendor, you'll need to do two things:

1. Tell Hibernate what kind of database it has to work with.
2. Initialize a new database with the openbrm schema (tables, indexes, keys, and base data).

Configuring Hibernate

In your openbrm-community-3.x/openbrm/ directory you'll find the openbrm-DataSource.groovy configuration file. Edit this file and enter in your database connection information. Typically you'll need the user name, password, server host name, and database instance name.

The exact syntax of the URL property depends on the JDBC driver and the vendor you obtained it from. Please refer to your vendors JDBC driver's documentation for information on how to specify the JDBC connection URL to your database.

Also in the openbrm-DataSource.groovy file, you'll find the Hibernate dialect property. The dialect tells Hibernate how to talk to your database, as well as the supported statements and syntax that are allowed. By default, openbrm is configured to use the Hypersonic dialect, 'org.hibernate.dialect.HSQLDialect.' You'll need to change the dialect to a value that matches your database vendor (see the [Hibernate Core documentation](#) for a complete list).

OPENBRM-DATASOURCE.GROOVY

```
dialect = "org.hibernate.dialect.PostgreSQLDialect"  
driverClassName = "org.postgresql.Driver"  
username = "openbrm"  
password = ""  
url = "jdbc:postgresql://localhost:5432/openbrm_test"
```

Last but not least, you'll need to provide the JDBC driver .jar file to the application server (Tomcat 6). This is done by copying the jar file to the lib/ folder. openbrm 3.x already comes packaged with drivers for Hypersonic and PostgreSQL; for other database vendors, you'll need to get the driver file from the particular vendor you choose.

Database Initialization

To initialize your database, you must run one of the scripts in the openbrm-community-3.x/openbrm/resources/db-samples/ folder. This of course, assumes that you have your database up and running with a user and target database instance created.

In the db-samples/ folder, you'll find a set of files named openbrm-schema*.sql and another file named openbrm-data.sql. There is one schema creation script per supported database engine. If you open any of these files in a text editor you'll see a series of DDL (data definition language) commands that will create the openbrm schema, as well as all the indexes, foreign keys, and constraints.

openbrm's base data resides in the openbrm-data.sql file and must be inserted after the schema has been created, but before all the foreign keys and constraints are created. To get this to work we'll need to separate the schema creation from the constraint creation statements and run them separately.

Open up the file openbrm-schema-postgresql.sql and look for the first foreign key constraint statement for the ACH table. This statement should appear about two thirds of the way into the schema creation script. Take all of the statements below (and including) the ACH constraints and cut-and-paste them into a new file named openbrm-schema-postgresql-constraints.sql.

Now that you have separated the schema creation statements from the constraints, you can execute the scripts using your databases client utility. To initialize your database, run the scripts in the following order:

```
run openbrm-schema-postgresql.sql  
run openbrm-data.sql  
run openbrm-schema-postgresql-constraints.sql
```

When you're done initializing your database, start up openbrm and browse to <http://localhost:8080/openbrm/signup> to create your first company and administration user.

Database Compatibility

There are many factors to consider when selecting which database engine to use for a critical application such as a billing system. You'll need performance, scalability, and security. Although openbrm will run on a wide variety of engines, it doesn't mean you can switch from one to another easily. This is mostly a one-time decision you have to make before you install openbrm, but remember that once you're up and running on your chosen database, switching to another can be very difficult.

Please note that openbrm has not been equally tested on all supported database vendors and engines. We “support” all of the following vendors, but that only means that we provide initialization files to get openbrm up and running, not that we have fully tested the systems, or that we give any kind of guarantee of performance or stability.

We need your help to keep this chart updated. Let us know about your experience with any of these databases, whether good or bad. For additional notes on a particular engine, see the bottom of this page.

Database Vendor	Compatibility	Comments
DB2	Red	openbrm's table names may exceed the maximum length supported by DB2.
Hypersonic	Red	This is the default engine, but is meant only as an example. Hypersonic does not support “read-committed” transaction isolation and therefore reliably cannot handle concurrent access.
Microsoft SQL Server	Yellow	There are some companies using MS SQL Server with success. We are not very familiar with it here at openbrm, which makes it

		harder for us to evaluate its compatibility.
MySQL	Yellow	<p>There are large deployments of openbrm powered by this database, however MySQL does not support nested transactions. Failure of MySQL to fully rollback a nested transaction can leave openbrm in an inconsistent state after a crash or run-time exception.</p> <p>For more information, please read the MySQL Database Guide.</p>
Oracle	Green	A great choice if budget is not an issue. Companies can easily scale up to meet growing needs.
PostgreSQL	Green	Free, open-source and very stable. openbrm has been implemented using PostgreSQL with companies of all sizes without any issues.
Sybase	Red	

Red = We are not aware of any production installations using this database.

Yellow = Some instances of openbrm are known to be running using this database, but not for long enough or in a big enough context for us to be fully confident of its compatibility. May also indicate some known issues with this database vendor (see comments above).

Green = Thoroughly tested, known to perform well in demanding scenarios.

Additional Notes

MySQL

For more information, please read the [MySQL Database Guide](#).

Oracle

Allow use of "&" when loading openbrm-data.sql

The data to initialize the database includes '&' characters, which is a reserved symbol for sqlplus. Use 'set define off' to skip prompting for '&' values.

```
sqlplus> set define off;
```

Use NUMBER(1,0) instead of BIT

Edit the openbrm-schema-oracle.sql schema script and change all instances of BIT to NUMBER(1,0)

MySQL Database Guide

Use the MyISAM database engine instead of InnoDB

The InnoDB engine does not support nested transactions and will cause the scheduled billing process to fail. The problem is that data created in the transaction in the beginning of the process is unavailable until the billing process completes and the transaction ends. This means that statistics, billing process records, and other data is missing when the billing process goes to look for it. The only solution is to use MyISAM which has no transaction support whatsoever - or to use a database that fully supports nested transactions.

Set the engine to MyISAM for your mysql client session before loading [openbrm-schema-mysql.sql](#):

```
mysql> SET storage_engine=MyISAM
```

Warning: When using MyISAM the database will be left in an inconsistent state if the billing process fails. Any transaction that fails will not be rolled back.

Return NULL for null date/time values instead of 0

```
zeroDateTimeBehavior=convertToNull
```

Add the above parameter to your JDBC connection URL. MySQL accepts parameters from the JDBC driver by parsing them out of the connection URL in the format:

```
jdbc:mysql://[host:port]/[database][?propertyName1]=[propertyValue1]&propertyName2]=[propertyValue2]...
```

Use DATETIME instead of TIMESTAMP

According to the MySQL documentation, if a schema is created while in the "MaxDB" mode all TIMESTAPS will be considered DATETIME types.

To enable MaxDB mode, set the MySQL mode at startup using the "--sql-mode=MAXDB" command line option, or by setting the global variable "sql_mode" to MAXDB at runtime:

```
mysql> SET GLOBAL sql_mode=MAXDB;
```

Edit the [openbrm-schema-mysql.sql](#) schema script and change all instances of TIMESTAMP to DATETIME.

Translating Openbrm

Translating Openbrm

One of openbrm's best features is that it can be configured to fit the needs of your country. All it takes to translate openbrm is to get the text in two files translated. In this guide, we'll go through the steps to add a new language to openbrm. You do not need to know any programming to get this done; you just need a text editor and, of course, knowledge of English and the new language you wish to add. Here are the steps:

1. Download [this file](#)
2. Open the file with a text editor, and translate it. Do not translate the keys (what is to the left of the '='), but the content (what is to the right of the '=').
3. Save the file with a new name. The name has to be ApplicationResources_XX.properties. Replace XX for the language code you have translated to. See [here](#) a list of language codes (under the column ISO 639-1).
4. Download [this file](#).
5. Open it with a text editor. Search for the line that starts with '<Language.' Copy and paste that line, replacing the code with the code you found in step 3, the Description with the name of the language, and the ID with a new ID. Please note that there can only be one language with the same ID, so you need to add 1 to the latest language. For example, if Spanish is the last one and has id=3, your language will be id=4.
6. Search for the first line starting with '<InternationalDescription.' There is a large block of these lines (about 800 of them) for English (language=1). You need to copy and paste them. Replace the 1 in LanguageId="1" with the ID of the new language that you entered in the previous step. Translate all of the text in the 'Content' field.
7. Save the file.
8. Send both files to openbrm-developers@lists.sourceforge.net

Now, you can wait for the next openbrm release to see all of the text at work. You can also build openbrm from source to apply the new language (there is documentation for this), or you can ask the developers (same email address where you sent the file) to do this for you and send you a new version of openbrm.

Thank you for your language contribution!

Payment Processor

Configure Payment Processor

openbrm can submit payments to virtually any payment gateway in the world. For this to happen, however, you need a payment processor plug-in that handles the payment requests to the gateway.

The following are configuration instructions for the plug-in that integrates with Intrannuity. The configuration for any other gateway is very similar, so you can use these instructions as an example

Payment Processor

Organization: Intrannuity, LLC

URL: <http://intrannuity.com/>

The Intrannuity payment plug-in can be used to securely process credit card transactions. To enable the plug-in, follow the steps below:

Merchant Account Request

Before you can process any transactions via plug-in, you must setup a merchant account with Intrannuity. Apply for the merchant account using this link: <http://intrannuity.com/apply.html>

Once your request is submitted, the Intrannuity staff will contact you and will guide you through the merchant signup process. Upon completion, you will be given a Merchant Account Code and Password necessary to configure the plug-in and to process credit cards.

Configuring Plug-In Within openbrm

The next step is to configure the plug-in. Log into openbrm:

- Select 'System' and then 'Plug-ins' in the main menu.
- Once the form appears, click on the link at the bottom that says 'Add a new Plug-in.' This will create a new row in addition to any already existing rows for plug-ins.

- In the new row created, select as type 'com.sapienter.openbrm.server.payment.tasks.PaymentAtlasTask.'
- Click on 'Add parameter' below the new task five times. Refer to Figure 1 below.

The following parameters must be setup:

1)MERCHANT_ACCOUNT_CODE-VALUE PROVIDED BY INTRANNUITY

The merchant account code is used to track transactions associated with a specific merchant.

This is a required field. Replace 'parameter_name' with 'merchant_account_code' and 'default' with the value provided by Intrannuity.

2) PASSWORD-VALUE PROVIDED BY INTRANNUITY

The password is used in combination with the merchant account code to facilitate the authorization process. This is a required field. Replace 'parameter_name' with 'password' and default with the value provided by Intrannuity.

3)SUBMIT_AVS-TRUE/FALSE

The parameter indicates whether you want track data and cvv2 values to be sent during the transaction authorization. These values are not required, but they are highly recommended to reduce the possibility of fraud and to lower your transaction processing fees. Replace 'parameter_name' with 'submit_avs' and 'default' with either 'true' (recommended) or 'false'.

4)TEST-TRUE/FALSE

The test parameter indicates whether the transactions are processed via a real production server or via a staging server, which duplicates all functionality of the production server but doesn't perform any charges. The test mode should be used for testing purposes only to ensure that everything is configured correctly before going live. For live transactions, this value should always be false. This field is required. Replace 'parameter_name' with test and 'default' with 'true' (test server) or 'false' (production server).

5)TIMEOUT_SEC

The 'timeout_sec' parameter indicates the time in millisecond before a transaction is timed out. Replace 'parameter_name' with 'timeout_sec' and 'default' with a value in milliseconds. This field is required. The recommended value is 180000 milliseconds (3 minutes).

Below is a sample test configuration:

ADD NEW PLUG-IN

Category
Payment gateway integration

Type
com.sapienter.jbilling.server.payment.tasks.PaymentAtlasTask ▼

Order

PARAMETERS

timeout_sec

submit_avs

merchant_account_code*

password*

test

Fig 1: Plug-in

Remove Default Payment Processing Plug-In

Be sure to remove the task 'com.sapienter.openbrm.server.pluggableTask.PaymentFakeTask.' This is supplied as a default transaction processing plug-in which doesn't actually process any charges.

Completion

You are now ready to use openbrm to process transactions with Intrannuity's Payment Gateway. Additional fraud protection and transaction-validation filters can be configured for your account using Intrannuity's gateway management portal.

