TIBCO iWay® Service Manager

Business Activity Monitor User's Guide

Version 8.0 and Higher
July 2021
DN3502286.0721

Contents

Chapter 1

Introducing TIBCO iWay® Business Activity Monitor

This section provides an overview of iWay® Business Activity Monitor, including key features and components.

In this chapter:

iWay Business Activity Monitor Architecture

iWay Business Activity Monitor is an extension of TIBCO iWay® Service Manager and provides an end-to-end, non-invasive view into transaction life cycles as they span across multiple channels and/or iWay® Service Manager servers. It enables you to capture, analyze, resolve, and act upon business transaction events gathered by iWay Business Activity Monitor. The architecture of iWay Business Activity Monitor is based on iWay Service Manager (iSM) and uses standard iSM components to provide a seamless integration with an existing application life cycle.

iWay Business Activity Monitor consists of two layers, a data capture layer facilitated by the iWay Business Activity Monitor driver facility, and a data presentation layer facilitated by the iWay Business Activity Monitor web application console.

iWay Business Activity Monitor Driver Facility

After iWay Business Activity Monitor is configured on iSM, an internal iSM activity facility is created that enables iWay Business Activity Monitor to capture data about the transactions that flow through the server (iSM). The activity driver runs in parallel to the business application logic providing a non-invasive data capture of transaction events. Even though the iWay Business Activity Monitor activity driver is exposed in the iSM Administration Console, it is not recommended to change this configuration directly, unless it is required for advanced IT operations. All of the essential iWay Business Activity Monitor configuration tasks should be done using the iWay Business Activity Monitor tools, as described in *Configuring and Using iWay Business Activity Monitor* on page 25.

Database Prerequisites

The iWay Business Activity Monitor activity driver uses a database for metric definitions, performance metric data archival, and so on. As a result, a database is also a prerequisite of iWay Business Activity Monitor. Currently, iWay Business Activity Monitor supports:

- Oracle 8i, 9i, 10g, and 11g
- MS SQL Server 2005 and higher
- DB2 non-Mainframe

iWay Business Activity Monitor Web Application Console

iWay Business Activity Monitor also includes a web application console that presents you with all of the information that is captured. This console is developed using the iSM non-blocking HTTP listener with an underlying process for data rendering as provided in the pre-configured channel for iWay Business Activity Monitor. This channel is called BAMChannel. The BAM web application console enables you, among other things, to view and analyze captured transaction data. It also provides a facility to resubmit messages that enables you to repair and reprocess transactions. The iWay Business Activity Monitor web application console can be accessed using the following default URL once the BAMChannel is deployed:

```
http://localhost:8087
```

For more information on using the web application console, see *Configuring and Using iWay Business Activity Monitor* on page 25.

Chapter 2

Installing iWay Business Activity Monitor

This section describes how to install iWay Business Activity Monitor.

In this chapter:

Prerequisites

Before continuing, ensure that the following components are available:

- iSM Version 8.0
- During the iSM installation (using the Windows or Java-based installer), ensure that the *Business Activity Monitor* feature is selected.

  After iSM has been installed, confirm that the *iwcpBAM.jar* file is located in your <*iway_home*>\etc\manager\console directory.

  In the iSM Administration Console, click *Tools* in the top menu and verify that *Business Activity Monitor* displays in the left pane.
- Ensure that one of the following supported database systems is installed:
  - Oracle 8i, 9i, 10g, and 11g
  - MS SQL Server 2005 and higher
  - Db2 non-Mainframe

  These are the only databases that are supported by iWay Business Activity Monitor.
- Ensure that the corresponding .jar file for your database driver is located in your <*iway_home*>\lib directory. If you have to copy the .jar file to this directory, ensure to first stop iSM and then start iSM when finished.
- Web Browsers:
  - Mozilla Firefox Version 31.0 and higher.

    iWay Business Activity Monitor is currently tested with Mozilla Firefox Version 55.0.3 (32 bit) and Version 57.0 (64 bit).

- Google Chrome Version 40 and higher.

  iWay Business Activity Monitor is currently tested with Google Chrome Version 61.0.3163.100 (64 bit).

- Microsoft Internet Explorer Version 10 and higher.

  iWay Business Activity Monitor is currently tested with Microsoft Internet Explorer Version 11.0.9600.18817.

Note: Support for JavaScript and Cookies must be enabled in your browser settings. There are known issues with browser caching where the caching must be cleared periodically to reload data. In addition, several pop-up windows may have their corners cut off slightly, which does not prevent the functionality. This will be resolved in a future release.

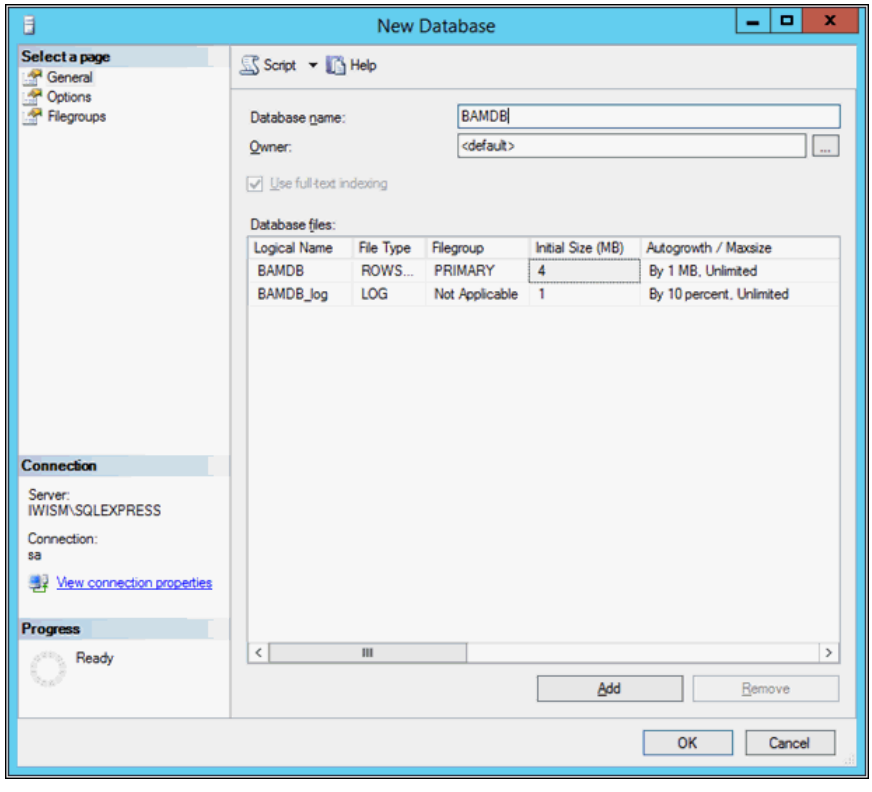
Installing iWay Business Activity Monitor

This section describes how to install iWay Business Activity Monitor.

Procedure: How to Install iWay Business Activity Monitor

1. Open your database management tool. For example, in SQL Server Management Studio, right-click *Databases*, and then select *New Database*.

The New Database dialog box opens, as shown in the following image.

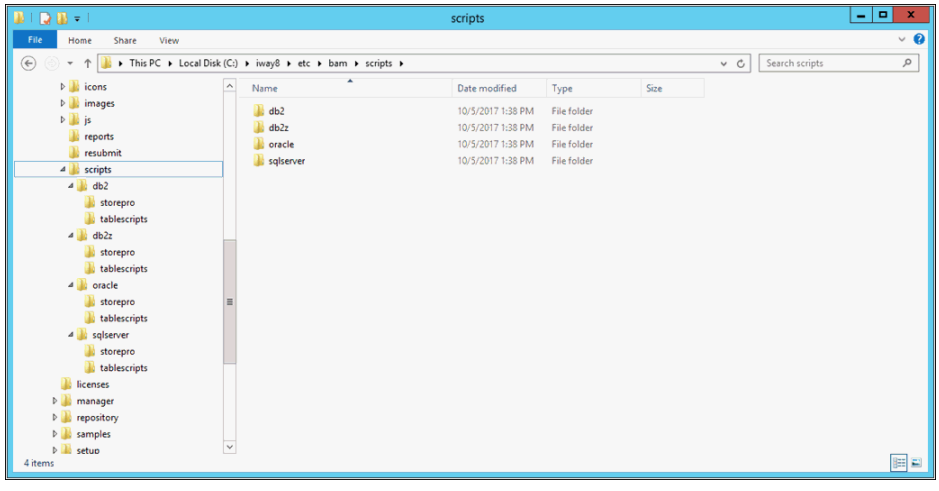


2. In the Database name field, enter *BAMDB* and then click *OK*.

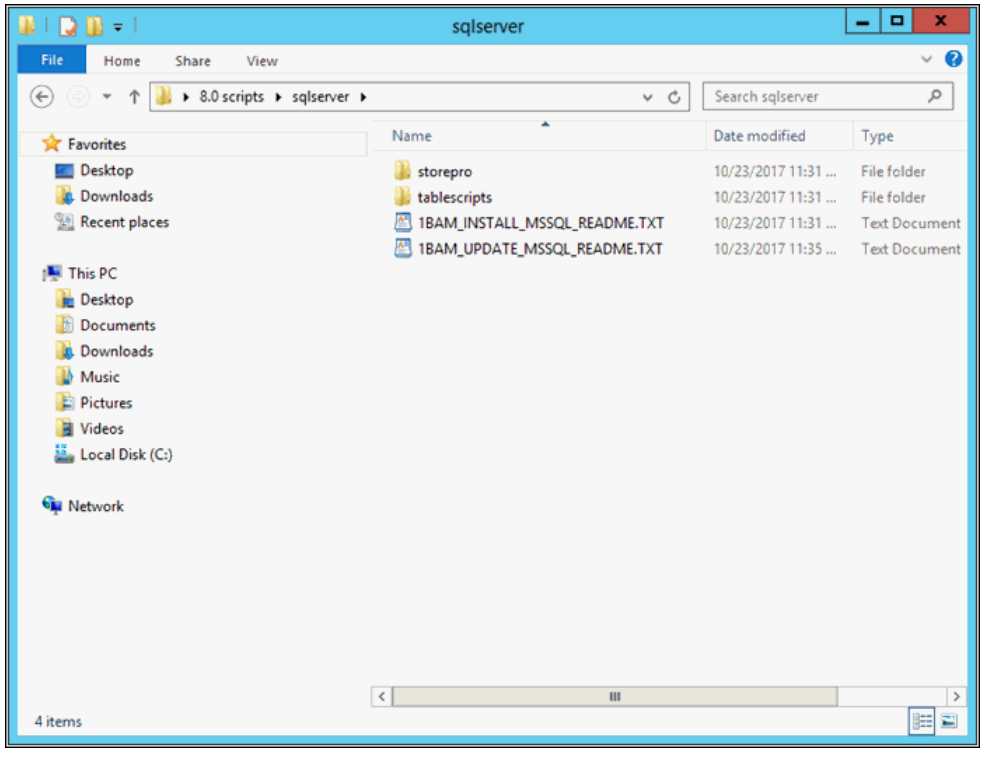
   The Data Definition Language (DDL) scripts are provided to generate database tables. They are located in the following directory:

   `<iway_home>\etc\bam\scripts`

Note that each set of scripts is database specific, as shown in the following image.

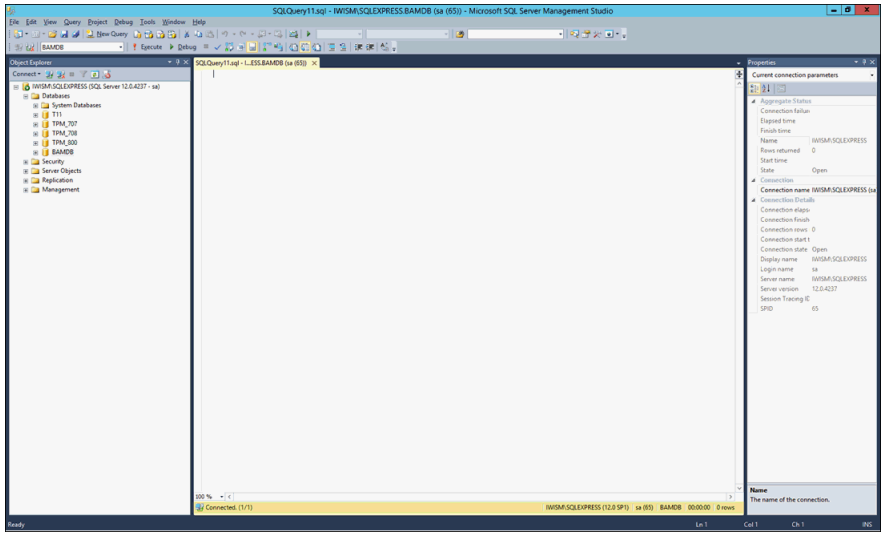


There are also *storepro* and *tablescripts* folders for each database, as well as a *readme* file in the directory, as shown in the following image.

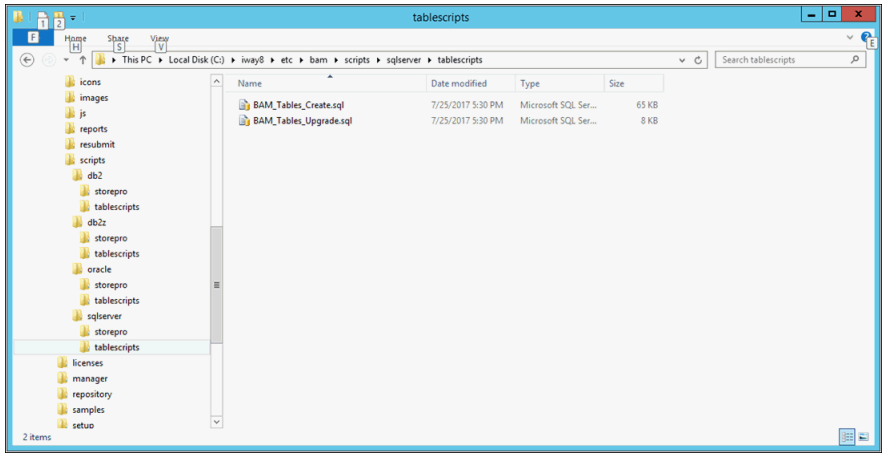


3. In SQL Server Management Studio, highlight the database name (BAMDB) and click *New Query* at the top of the screen.

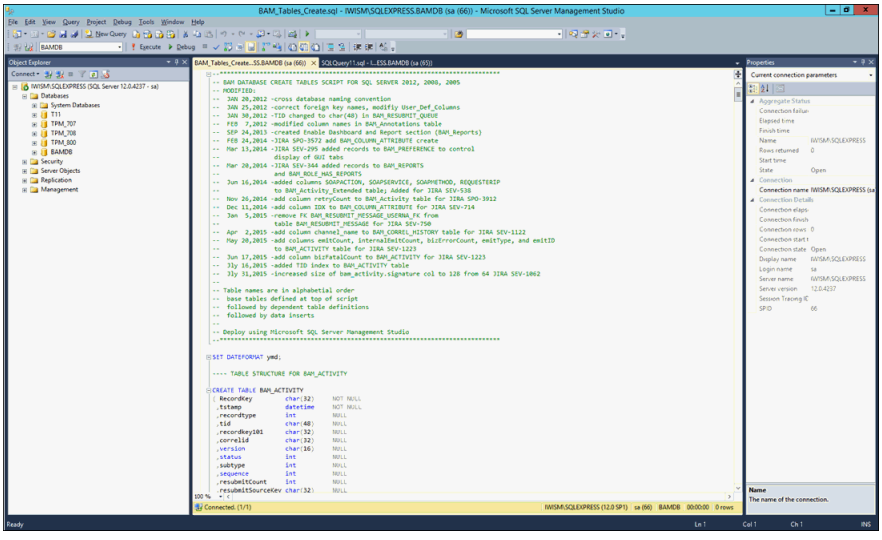
   A query palette opens, as shown in the following image.



4. From the *tablescripts* folder, select *BAM_Tables_Create.sql*, as shown in the following image.



5. Drag the script and drop it into the editor palette. You can also open it in your text editor by pressing *Ctrl+A* to select all, *Ctrl+C* to copy, and *Ctrl+V* to paste into SQL Server Management Studio.

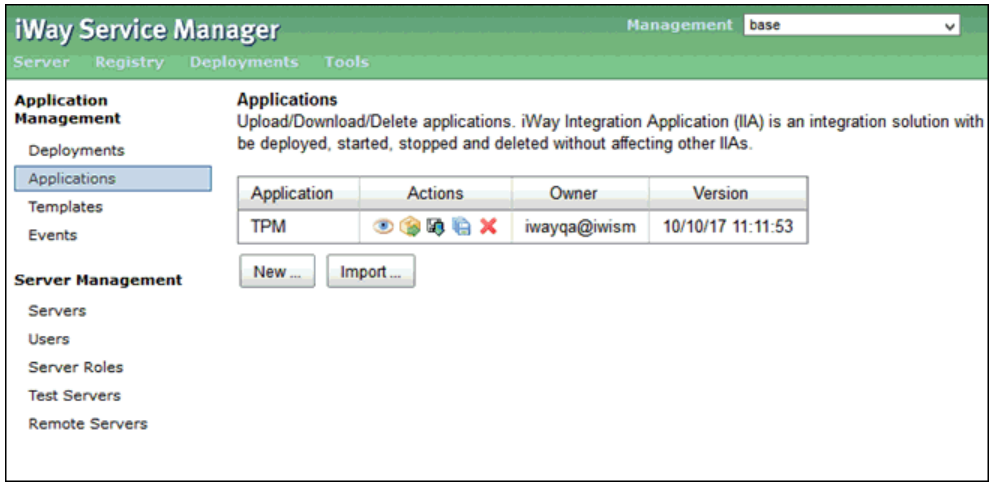


6. Click *Execute*.
7. At completion, note any errors and then close the open palette.
8. Repeat Steps 3 to 7 to run the seven stored procedures from the *storepro* folder in the following order:

   - ❏ 01_f_INITCAP.sql
   - ❏ 02_f_SetStringWildCard.sql
   - ❏ 03_f_Script_DynCol.sql
   - ❏ 04_f_TxChannelStatus.sql
   - ❏ 05_p_BAM_Activity_TransHistory.sql
   - ❏ 06_p_BAM_ActivityMgt.sql
   - ❏ 07_p_BAM_ActivityMgt_TransActivity.sql

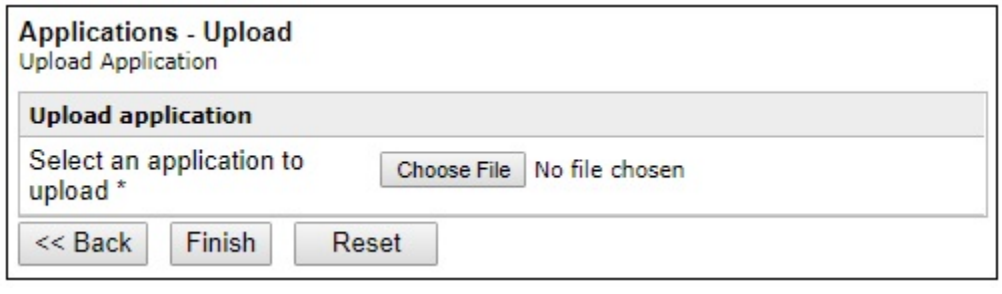
9. After you run the seven stored procedures, minimize or close the SQL Server Management Studio.

10. From the iSM Administration Console, click *Management*, and then on the left pane under Application Management, click *Applications*, as shown in the following image.



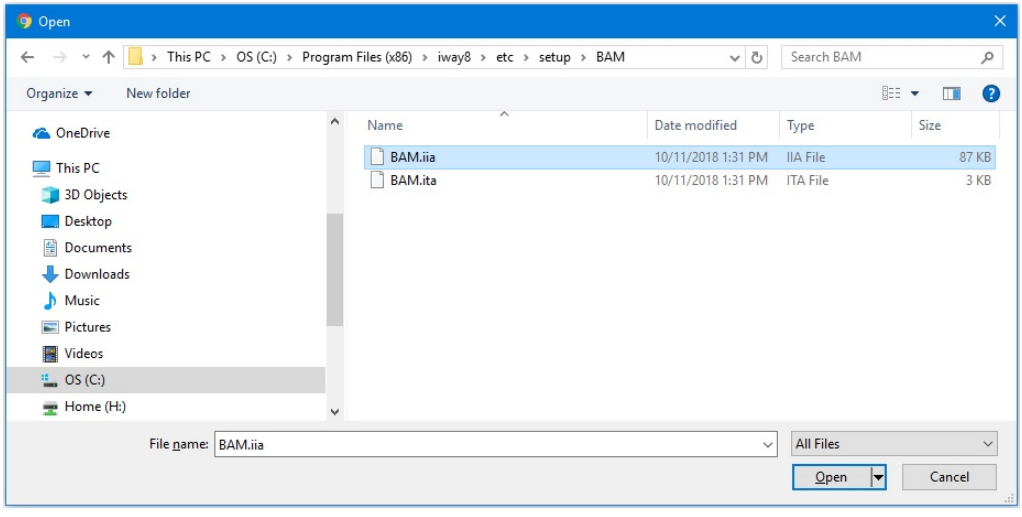
11. Click *Import*.

    The Applications - Upload pane opens, as shown in the following image.



12. Click *Choose File*.

The Open dialog box displays, as shown in the following image.



13. Navigate to the following location on your file system:

    <iway_home>\etc\setup\BAM

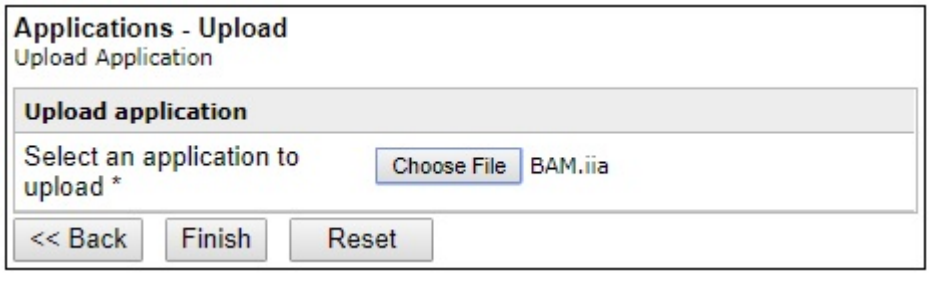
    where:

    <iway_home>

    Is the root installation folder where iSM is installed.

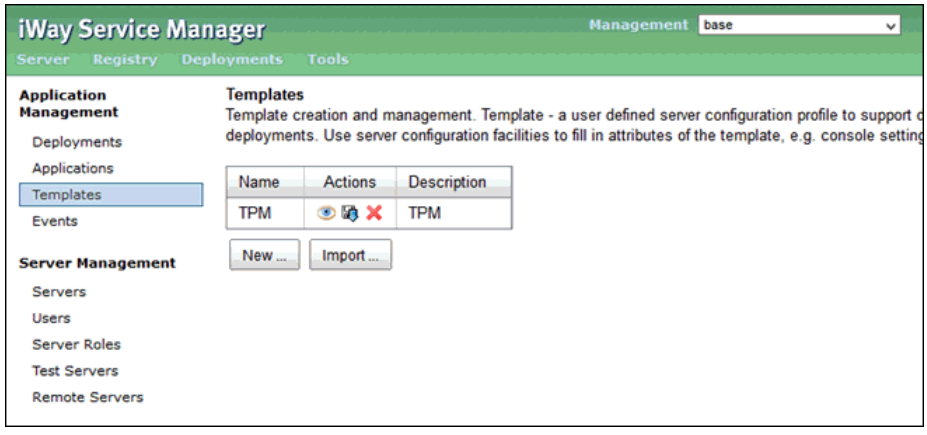
14. Select the *BAM.iia* file and then click *Open*.

    You are returned to the Applications - Upload pane, as shown in the following image.



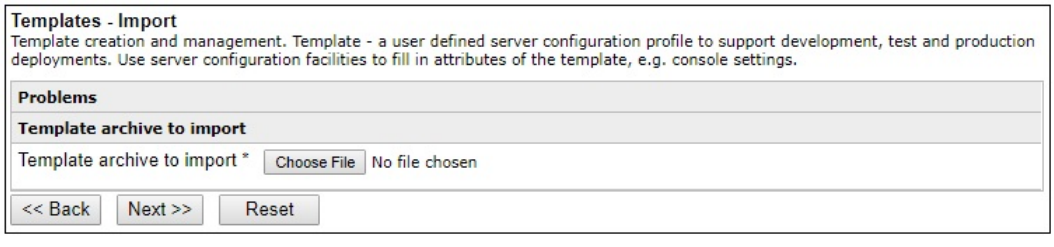
15. Click *Finish*.

16. From the Application Management section on the left pane of the console, click *Templates*, as shown in the following image.



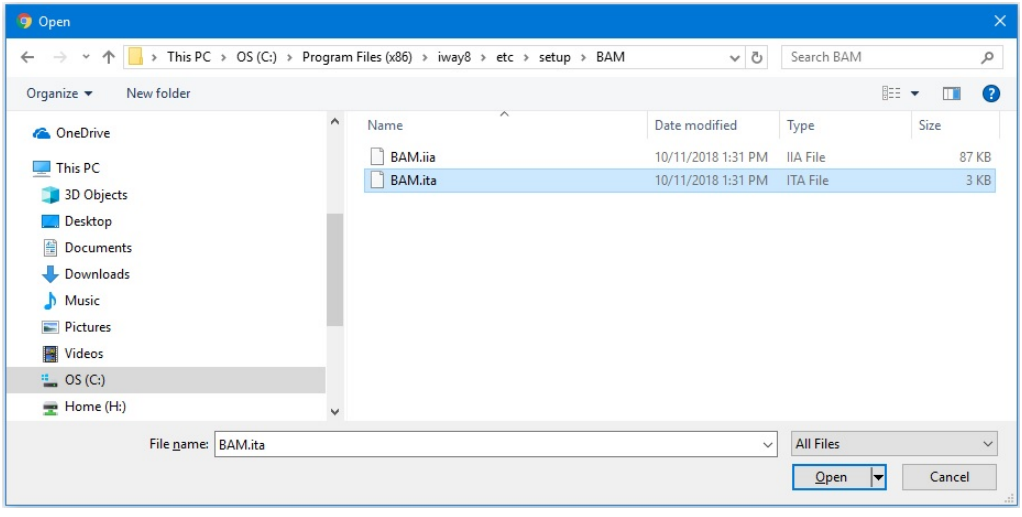
17. Click *Import*.

    The Templates - Import pane opens, as shown in the following image.



18. Click *Choose File*.

The Open dialog box displays, as shown in the following image.



19. Navigate to the following location on your file system:

    <iway_home>\etc\setup\BAM

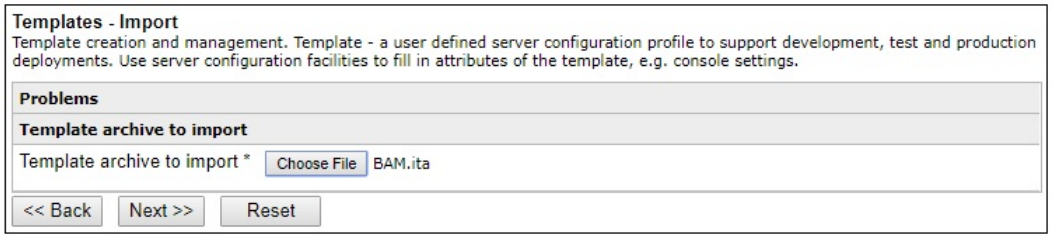
    where:

    <iway_home>

    Is the root installation folder where iSM is installed.

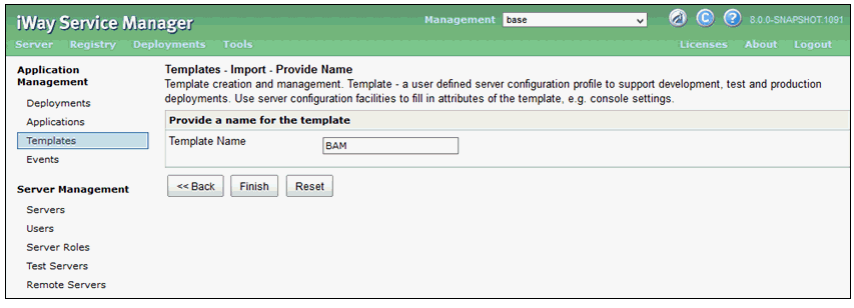
20. Select the *BAM.ita* file and then click *Open*.

    You are returned to the Templates - Import pane, as shown in the following image.

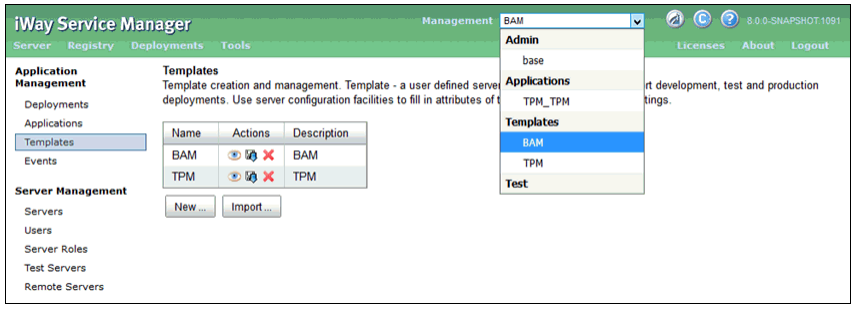


21. Click *Next*.

In the Template Name field, *BAM* is already pre-populated for you, as shown in the following image.

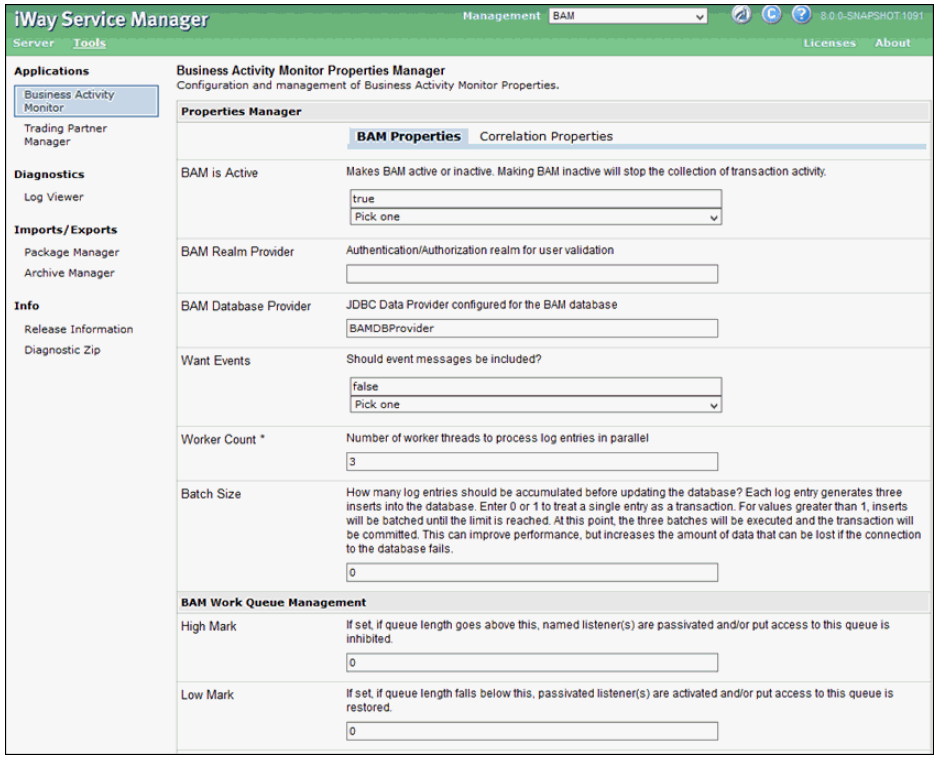


22. Click *Finish*.
23. At the top of the iSM Administration Console, click *Management*, and then in the drop-down list, select *BAM* under the Templates section.

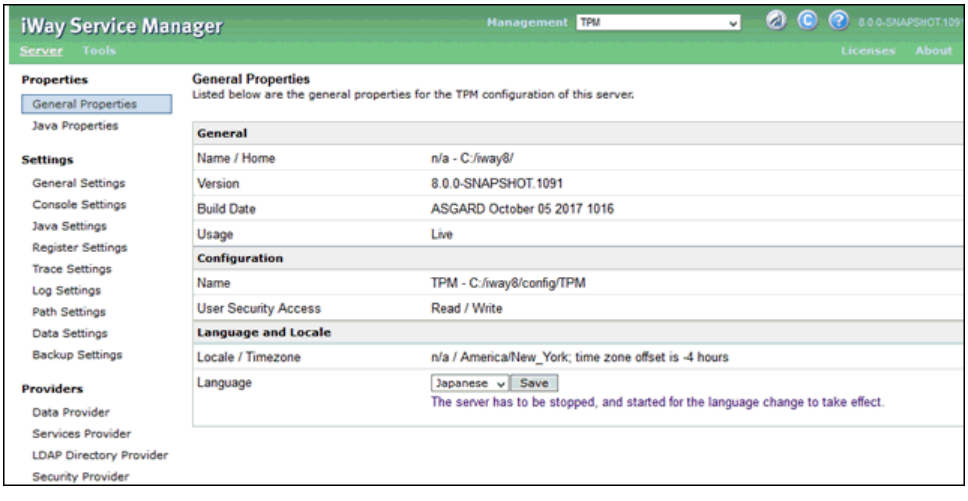


24. In the menu bar of the iSM Administration Console, click *Tools*, then under Applications on the left pane, click *Business Activity Monitor*.

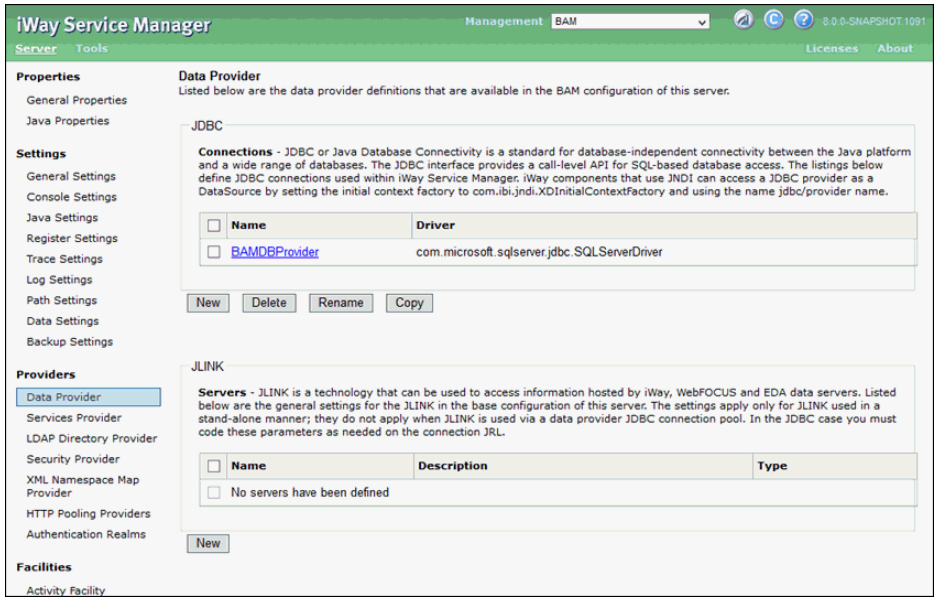
The iWay Business Activity Monitor access handler has been pre-configured. Note the pre-configured DB Provider Name, as shown in the following image.



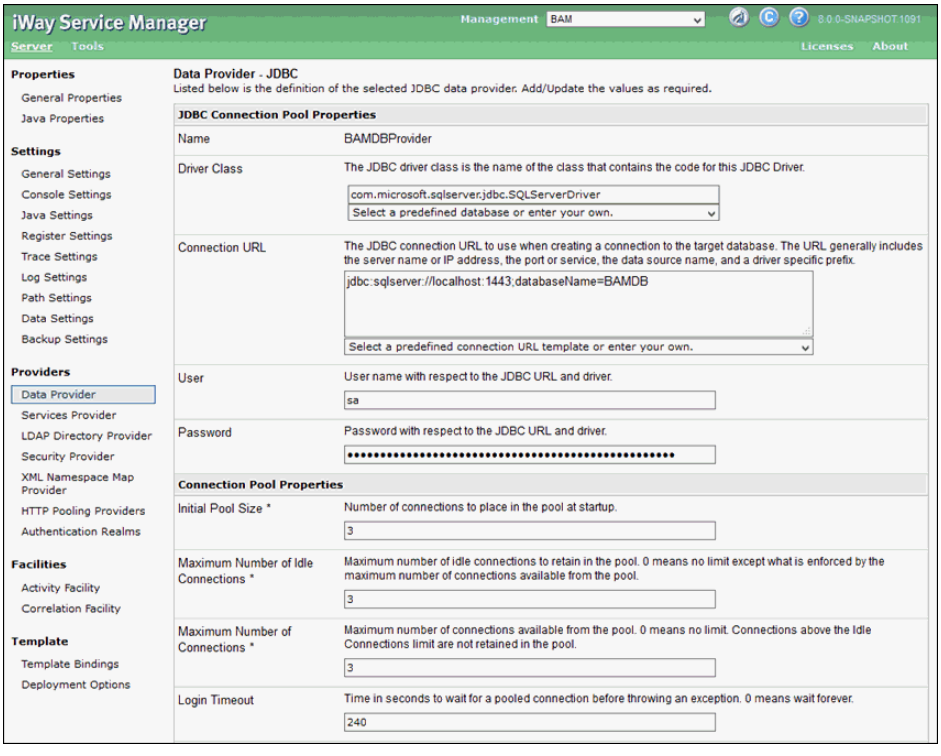
25. In the iSM Administration Console, click *Server*, as shown in the following image.



26. Under the Providers section in the left pane, click *Data Provider*, as shown in the following image.



27. In the JDBC section, click *BAMDBProvider*, as shown in the following image.



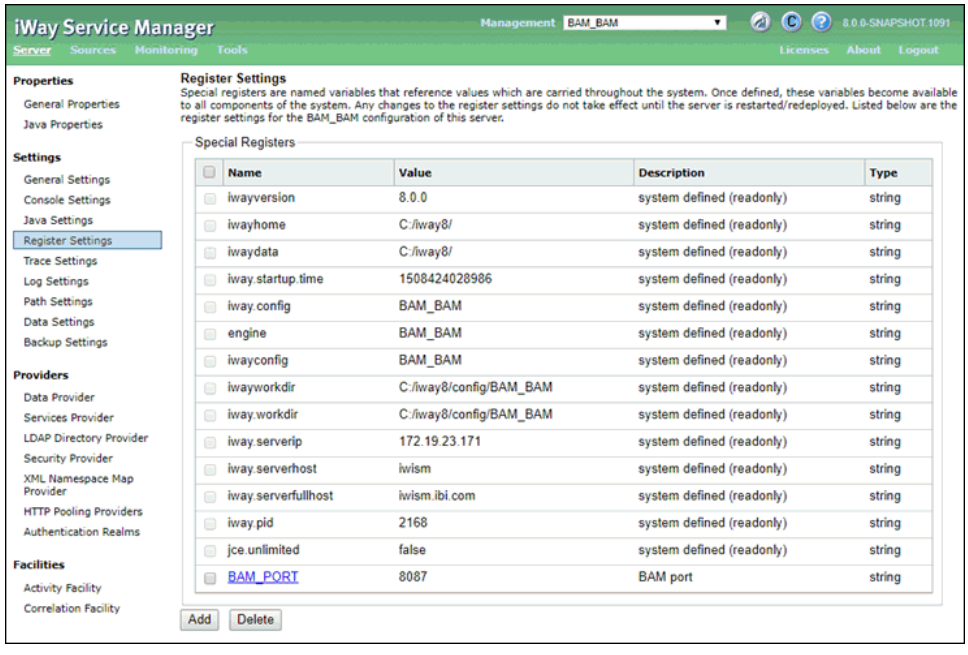
The template has been pre-configured for MSSQL. You can make any required changes to the Connection URL, User, and Password parameters to connect to your own database. You will have to change the Driver Class if you are using ORACLE or Db2.

If you changed the configuration, be sure to click the *Update* button at the bottom of the page.

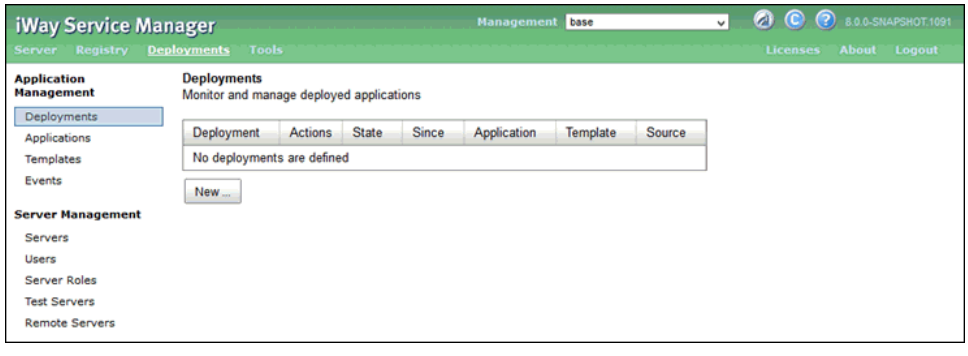
Note: You cannot test a data provider in a template as it is not a running application. You can only test run after the application is deployed.

28. In the iSM Administration Console, click *Server*, then in the left pane, click *Register Settings*.

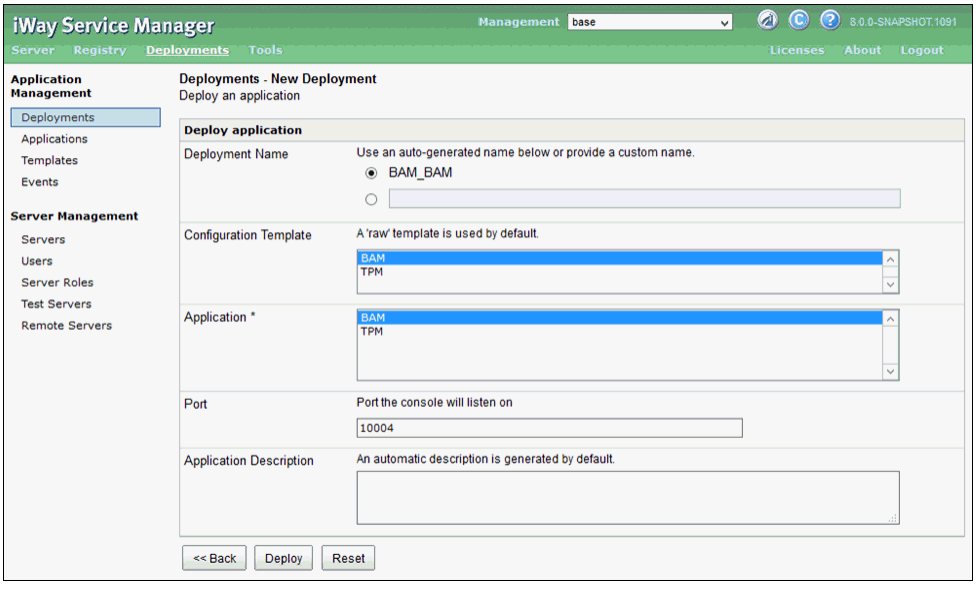
The listener HTTP port number for iWay Business Activity Monitor (the BAM_PORT Special Register) is shown in the following image.



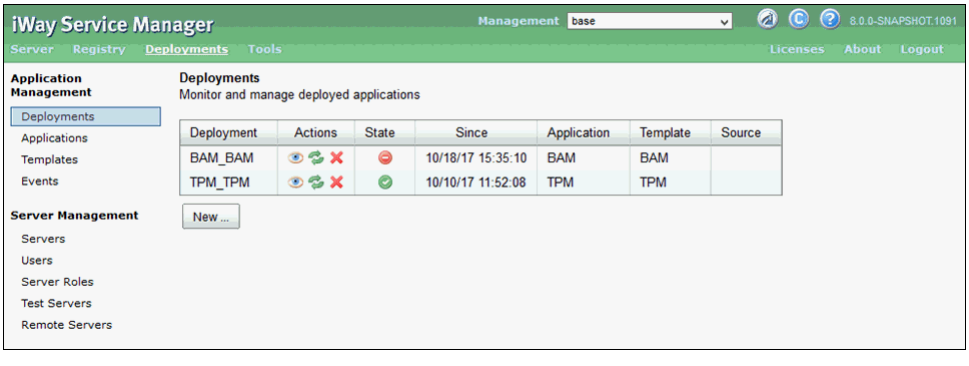
29. Click *Management* at the top of the iSM Administration Console, and ensure that *base* is selected from the drop-down list, as shown in the following image.



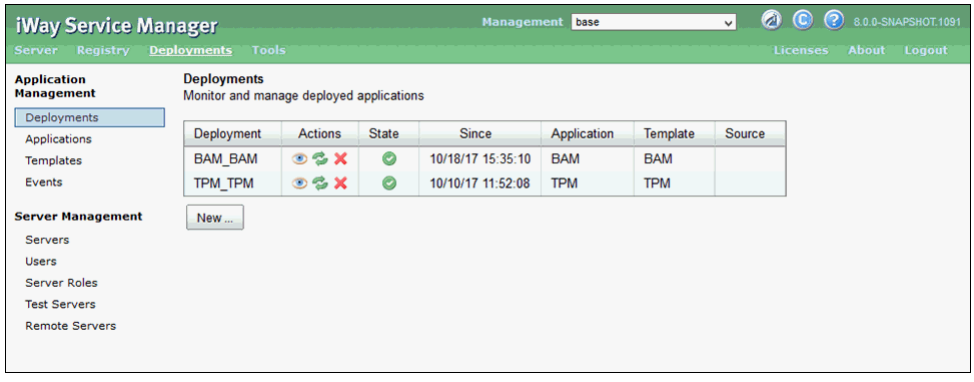
30. Highlight the template and application names, as shown in the following image. Type your own application description and then click *Deploy*.



31. Click the red minus (-) symbol to start the application, as shown in the following image.



When the application has started, the red minus (-) symbol changes to a green check mark, as shown in the following image.



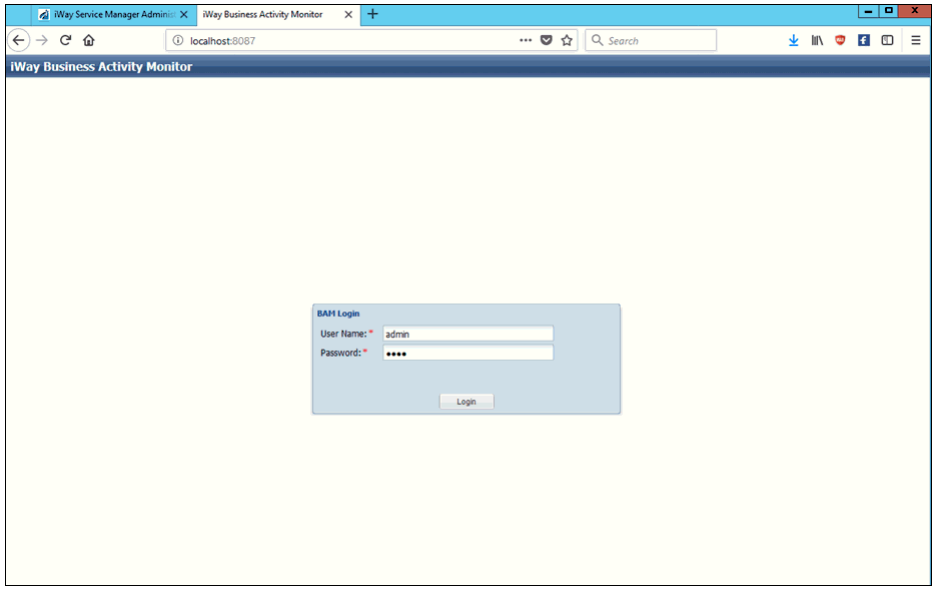
32. Open your browser and enter the following URL:

    `http://localhost:8087`

    Ensure that the port number specified for the BAM_PORT Special Register (shown in Step 28) is used.

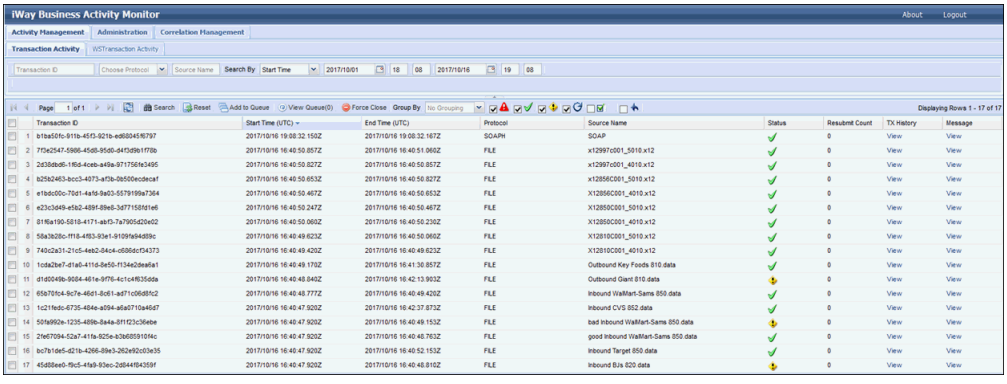
33. Log on to iWay Business Activity Monitor using the following credentials.

    - User Name: *admin*
    - Password: *iway*



34. Click *Login*.

    The iWay Business Activity Monitor console opens, as shown in the following image.



Chapter 3

Configuring and Using iWay Business Activity Monitor

This section describes how to configure and use iWay Business Activity Monitor.

In this chapter:

Accessing the iWay Business Activity Monitor Console

iWay Business Activity Monitor includes a web console that presents you with a view of all of the available transaction data. The console enables you, among other things, to view and analyze captured transaction data. It also provides a facility to resubmit messages that enables you to repair and reprocess transactions.

Procedure: How to Access the iWay Business Activity Monitor Console

To access the iWay Business Activity Monitor console:

1. Enter the following URL in your web browser:

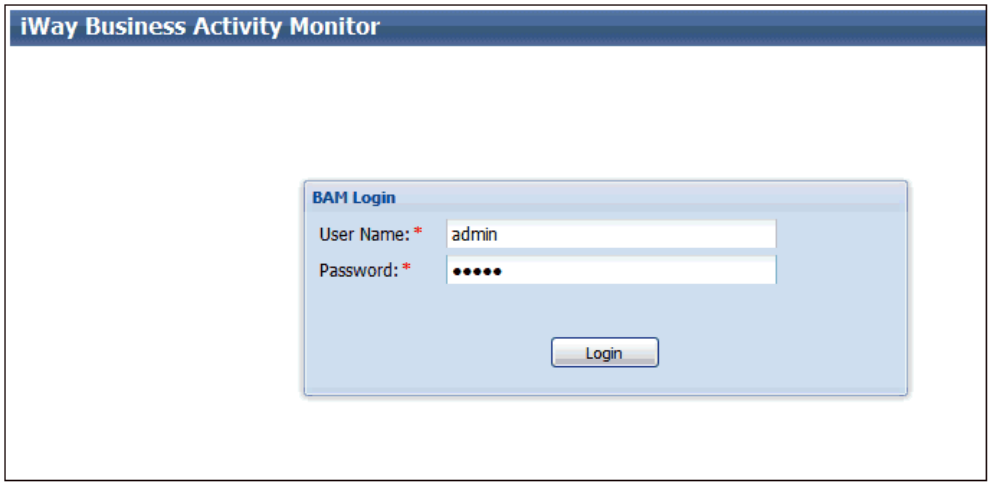
   `http://localhost:port/`

   where:

   port

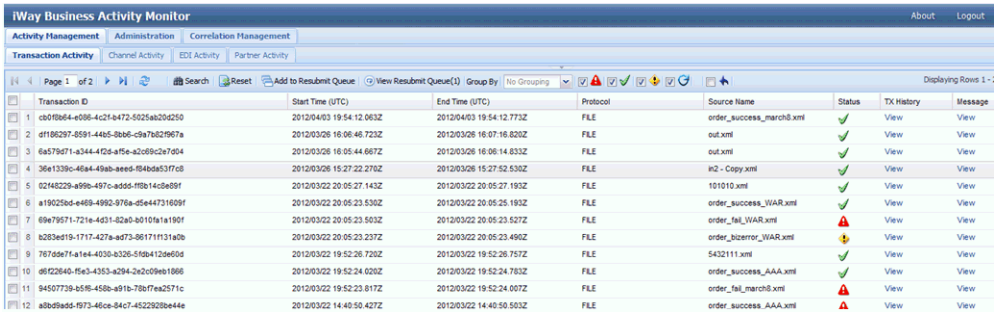
   Is the port that is defined in the BAMChannel. The default port is 8087.

The Login dialog box opens, as shown in the following image.



2. Enter *admin* as the user name (default) and *iway* as the password.
3. Click *Login*.

The iWay Business Activity Monitor console opens, as shown in the following image.



Note: The initial view will have no data displayed, unless there were some transactions processed by the system.

The various iWay Business Activity Monitor facilities can be accessed by clicking the corresponding tabs at the top of the pane. The following sections describe the iWay Business Activity Monitor facilities in more detail.

Activity Management Tab

The Activity Management tab allows you to monitor activities at the iSM channel level. In the iWay Business Activity Monitor console, four tabs are provided in the Activity Management facility for configuration purposes:

- Transaction Activity tab
- Channel Activity tab
- EDI Activity tab
- Partner Activity tab

The following image shows the top pane of the iWay Business Activity Monitor console, where the main tabs are located.

iWay Business Activity Monitor provides several levels of monitoring. The three major types of monitor categories are:

- **Channel Level Monitoring.** This includes Channel Activity and EDI Activity.
- **Transaction Level Monitoring.** This includes Transaction Activity and Partner Activity.
- **Correlated Transaction Monitoring.** This includes Correlation Management.

Channel Level Monitoring provides a view into the transaction life cycle as it passes through a single channel with a unique Transaction ID. The message is tracked within a single channel with a single Transaction ID. This functionality enables backward compatibility with prior 6.0.1 releases of iWay Business Activity Monitor. It is still useful in isolated implementations where the transaction life cycle is of interest only within a single channel and not across multiple channels. This is true for monitoring EDI-based transactions where the entire transaction is processed by a single channel or for transactions which carry single partner information for each channel and can be monitored through Partner Activity.

Transaction Level Monitoring provides an enhanced view into the transaction life cycle as it passes through multiple channels or multiple servers. This is the recommended method of monitoring transactions for most applications. It is important to note that with this enhanced functionality, there is the additional application responsibility to incorporate Transaction ID management into its process. For example, if you have multiple channels through which a transaction is propagated and they are linked by an internal iWay component, such as Internal Queue mechanism, there is no additional work that needs to be performed by an application. The Transaction ID will remain the same through the message life cycle even as it is processed by multiple internal channels. However, if the link between a multi-channel architecture is externalized, it is the responsibility of the application to incorporate Marshaling services to manage and propagate the message across multiple channels or even servers maintaining its Transaction ID. This provides a unified view into its entire message life cycle united by a single Transaction ID. For more information about Marshaling services, see the *TIBCO iWay® Service Manager Component Reference Guide*.

Correlated Transaction Monitoring uses correlation services to link long-running transactions with different Transaction IDs into a single view. An example would be an invoice that is sent for processing, which can take several hours. In this case, the sending of the invoice, which has a unique Transaction ID will be linked with Payment processing, which also has its own Transaction ID through a unified and propagated Correlation ID. This implementation is most applicable when used as an addition to Transaction level monitoring. For more information about correlation services, see the *TIBCO iWay® Service Manager Component Reference Guide*.

The following sections provide more information on each available tab in the Activity Management facility.

Transaction Activity Tab

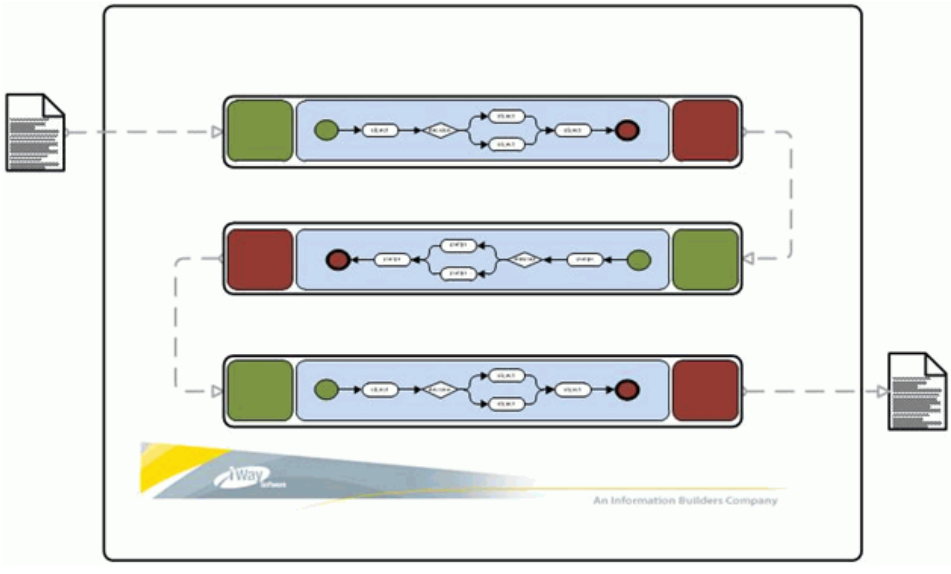
Prior releases of the iWay Business Activity Monitor console relied on a channel-based view of the environment. This allowed operational people to monitor channels and deal with issues from a channel perspective.

However, it has been determined that a transaction-based view has greater business and operational benefits. The transaction screen allows users to watch a transaction through its life within the iWay service bus.

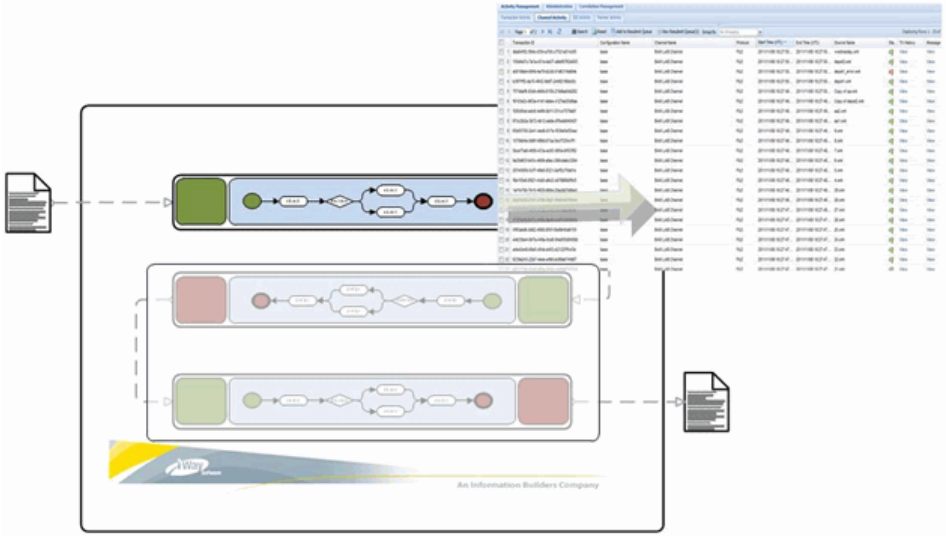
Note that the Transaction History and Message views found in the Transaction Activity tab are the same as in the Channel Activity tab.

Most implementations involve transactions spanning multiple channels. Tracking this activity from a channel perspective forces the user to have to understand the application infrastructure (for example, what channels relate to each other and why). The transaction channel removes the need to understand the application architecture and allows users to logically follow the life cycle of the transaction in realtime.

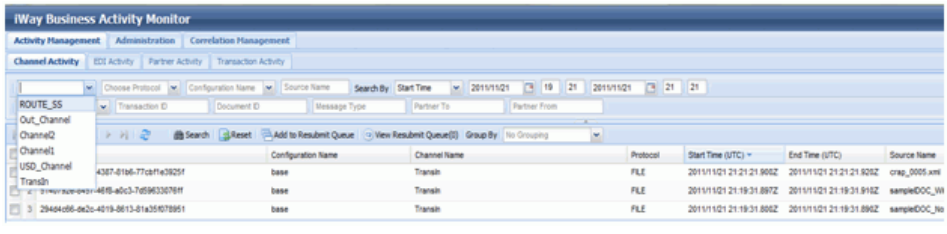
The following image shows an architecture that uses three channels to process a transaction.



The Channel Activity view gives you a view of channel activity, as shown in the following image.

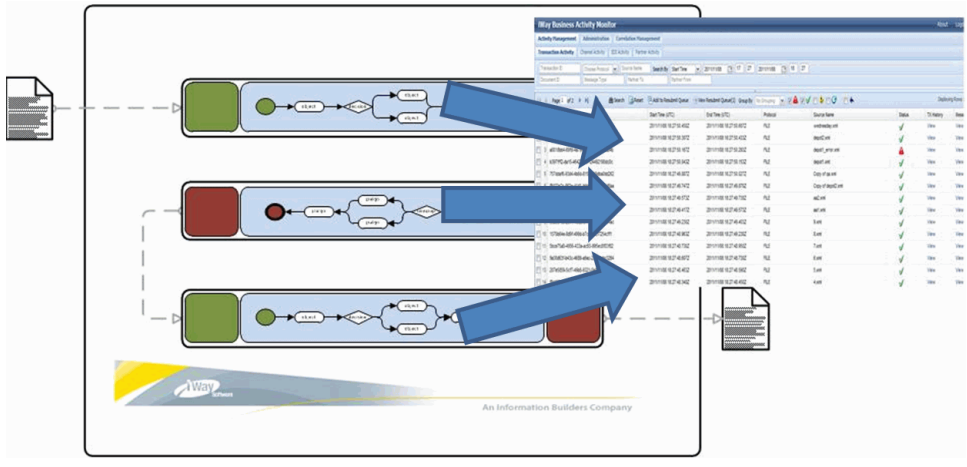


The iWay Business Activity Monitor console provides the option to define which channel to view, as shown in the following image.

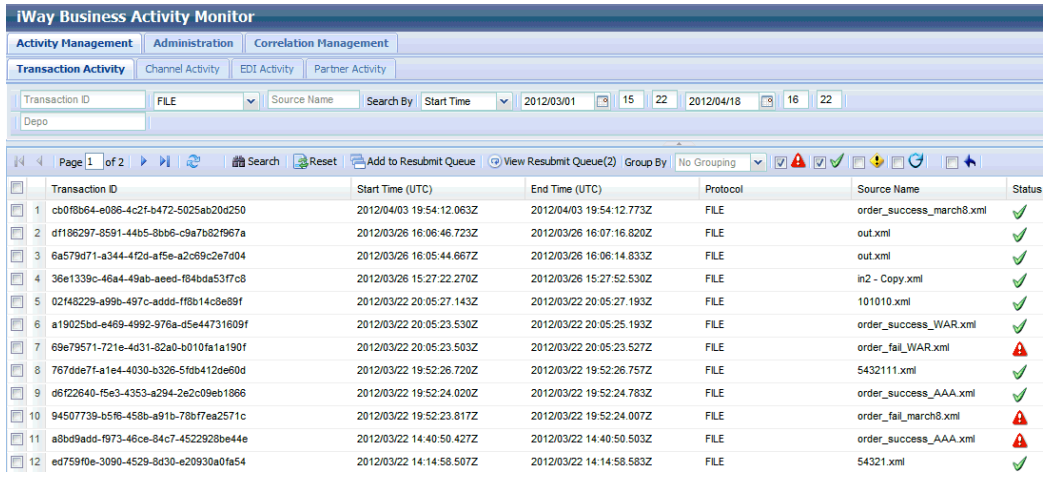


Transaction View

The transaction view allows transactions to be monitored as they pass through multiple channels, as shown in the following image.



The transaction view provides a quick way to see the overall situation. This allows you to easily diagnose a problem and trace it back to the source of the problem, as shown in the following image.



The following table lists and describes all of the columns that are available in the Channel Activity tab of the Activity Management facility.

Column Name	Description
Transaction ID	Unique ID that is automatically generated by the processing channel.
Start Time (UTC)	Start time (shown in GMT) at which iWay Business Activity Monitor logged the start time of the message.
End Time (UTC)	End time (shown in GMT) at which iWay Business Activity Monitor logged the end time of the message.
Protocol	Protocol on which transactions have been received and processed.
Source Name	Name of the transaction, depending on the protocol. For example, File-based and FTP-based protocols generate the name of the file read. The Queuing (MQ) protocol generates the name based on the header values. This value can also be set using the *bam_sourcename* register.

Column Name	Description
Status	Overall transaction status, that is shown as a flag. ❑ Success icon  indicates that the message execution has executed successfully. ❑ Warning icon  indicates that a business level warning has generated. ❑ Error icon  indicates that the message execution has failed. ❑ In Progress icon  indicates that the message execution status is currently in progress. ❑ Resubmit icon  indicates the resubmission status (for example, if the message has been resubmitted after a failure).
TX History	Shows the transaction history and transaction path of the given transaction.
Message	Hyperlink opens the message with which the transaction was started.

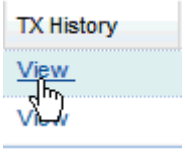
You can also customize the view and clear or select any column for viewing. Some of the additional non-default columns are shown in the following table.

Column Name	Description
Duration	Total time it took to complete the execution process.
Resubmit Count	Count of the number of times the transaction was resubmitted.
Record Key	Internal reference for this specific record in the iWay Business Activity Monitor table. This is used for debugging or specific data retrieval purposes.

Note: Any user-defined register columns can also be added.

Transaction History View

To view a detailed transaction history of the event message, click the *View* hyperlink under the TX History column, as shown in the following image.



The information displayed by the TX History view depends on the iWay Business Activity Monitor options that are configured.

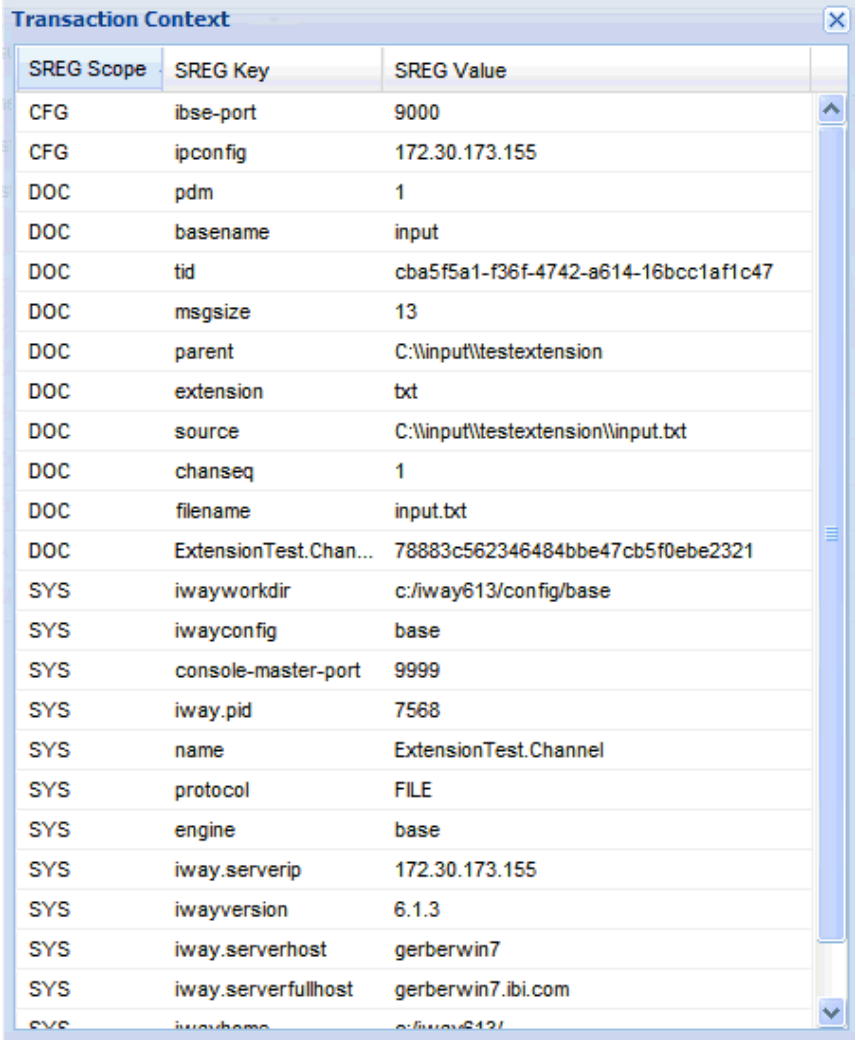
- The Startentry and Endentry options (success/fail) are always displayed.
- Based on the Emit option configuration, the Emit messages may or may not be displayed.
- Based on the Want Events option configuration, the internal steps of the transaction (such as every single step of the process) may or may not be displayed.
- Based on whether the application uses the specialized Log Message services, the Business Error events and Check Point events may or may not be displayed.

Note: The application design process determines the account performance requirements and detailed view requirements. For example, increased logging that is performed for each transaction will result in a loss of performance. It is recommended to disable the Want Events option to maximize performance and to utilize specialized Log Message services to log transaction check points at needed locations. For more information about Log Message services, see the *TIBCO iWay® Service Manager Component Reference Guide*.

An example of the Transaction History view is shown in the following image.

Notice that the transaction has maintained its transaction ID and was executed by multiple channels. It was first received by Demo.Receiver.Channel and then sent for processing to Demo.Processing.Channel. This is one of the key features, which enables a single view into the transaction life cycle as it spans across multiple channels, or even servers, as long as it maintains its transaction ID.

For a more detailed perspective, clicking *View* under the TX Context column enables the view of the transaction contexts (SREGs) at a given step of the process. This view enables the user to identify what has happened to the transaction and view the attributes, as shown in the following image.

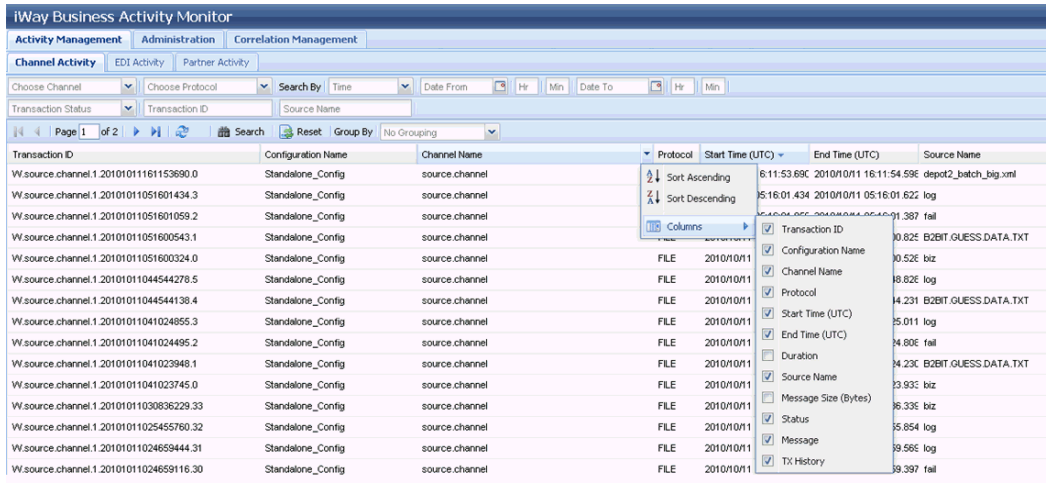


Transaction Context

SREG Scope	SREG Key	SREG Value
CFG	ibse-port	9000
CFG	ipconfig	172.30.173.155
DOC	pdm	1
DOC	basename	input
DOC	tid	cba5f5a1-f36f-4742-a614-16bcc1af1c47
DOC	msgsize	13
DOC	parent	C:\\input\\testextension
DOC	extension	txt
DOC	source	C:\\input\\testextension\\input.txt
DOC	chanseq	1
DOC	filename	input.txt
DOC	ExtensionTest.Chan...	78883c562346484bbe47cb5f0ebe2321
SYS	iwayworkdir	c:/iway613/config/base
SYS	iwayconfig	base
SYS	console-master-port	9999
SYS	iway.pid	7568
SYS	name	ExtensionTest.Channel
SYS	protocol	FILE
SYS	engine	base
SYS	iway.serverip	172.30.173.155
SYS	iwayversion	6.1.3
SYS	iway.serverhost	gerberwin7
SYS	iway.serverfullhost	gerberwin7.ibi.com

Channel Activity Tab

The Channel Activity tab is used to display activities that are occurring within a channel. It provides the channel name, when the activity happened, and when the activity ended. Status information (completed, failed, active) is also provided. You can also view the specific message with which the transaction was executed and the transaction history of that particular transaction. In a multi-channel environment where a single transaction passes through multiple channels, an entry line for each channel executed for the given transaction will appear in the Channel Activity. Each entry line contains its own Transaction ID auto-generated by the processing channel. To track transactions across multiple channels or iSM servers or configurations, it is recommended to use the Correlation view for long running transactions or the Transaction Activity view for short running transactions in which case the same Transaction ID is preserved across multiple channels and servers through an internal mechanism or explicit marshaling of the message.



The following table lists and describes all the columns that are available in the Channel Activity tab of the Activity Management facility.

Column Name	Description
Transaction ID	Unique ID that is auto-generated by the processing channel.
Configuration Name	Name of the configuration where the transaction has been processed. In a multi-node deployment, it can be used to identify multiple client configurations.

Column Name	Description
Channel Name	Name of the processing channel on which the message is being executed.
Protocol	Protocol on which transactions have been received and processed.
Start Time (UTC)	Start time is shown in GMT at which iWay Business Activity Monitor logged the start time of the message.
End Time (UTC)	End time is shown in GMT at which iWay Business Activity Monitor logged the end time of the message.
Source Name	Name of the transaction depending on the protocol. For example, File-based and FTP-based protocols generate the name of the file read. The Queuing (MQ) protocol generates the name based on the header values. This value can also be set using the bam_sourcename register.

Column Name	Description
Status	The overall transaction status that is shown as an icon. ❑ Success icon  indicates that the message execution has executed successfully. ❑ Warning icon  indicates that a business level warning has generated. ❑ Error icon  indicates that the message execution has failed. ❑ In Progress icon  indicates that the message execution status is currently in progress. ❑ Resubmit icon  indicates the resubmission status (for example, if the message has been resubmitted after a failure).
TX History	Shows the transaction history and transaction path of the given transaction.
Message	Hyperlink opens the message with which the transaction was started.

You can also customize the view and clear or select any column for viewing. Some of the additional non-default columns are shown in the following table.

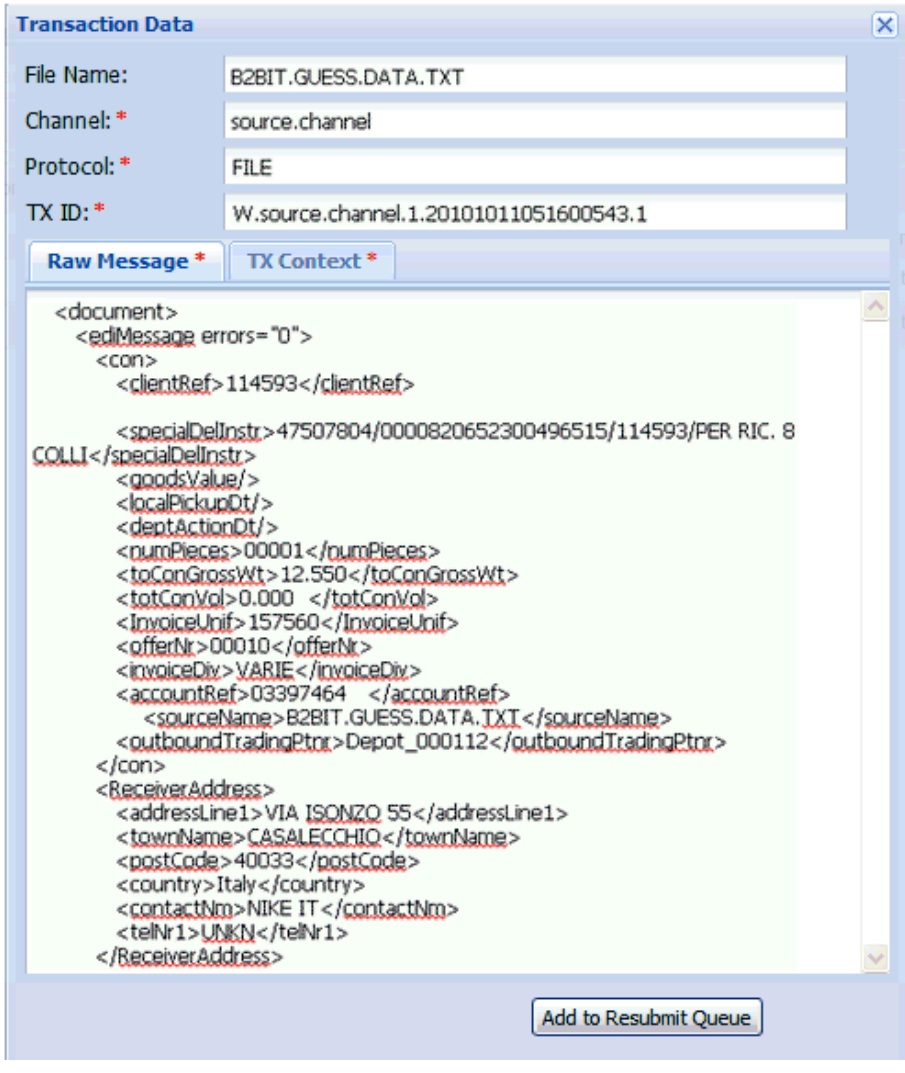
Column Name	Description
Duration	Total time it took to complete the execution process.
Message Size	Size of the processed execution.
Resubmit Key	Internal reference key for the record that has been resubmitted.

Note: Any user-defined register columns can also be added.

Message View

To view the input message received by the channel for processing, click the *View* hyperlink under the Message column, as shown in the following image.

The Transaction Data dialog box opens, as shown in the following image.



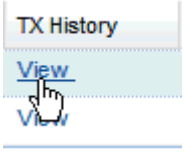
The following table lists and describes the available fields and tabs in the Transaction Data dialog box.

Field/Tab	Description
File Name	Name of the file received for processing (availability depends on protocol).

Field/Tab	Description
Channel	Associated channel for the message.
Protocol	Protocol on which the message is received.
TX ID	Transaction ID for the message.
Raw Message tab	Displays the actual raw data of the message.
TX Context tab	Provides information regarding the Special Registers (SREGs) associated with the processed messages. Based on your permissions, all field values can be edited and resubmitted. For more information, see *Message Resubmission* on page 52.

Transaction History View

To view a detailed transaction history of the event message, click the *View* hyperlink under the TX History column, as shown in the following image.



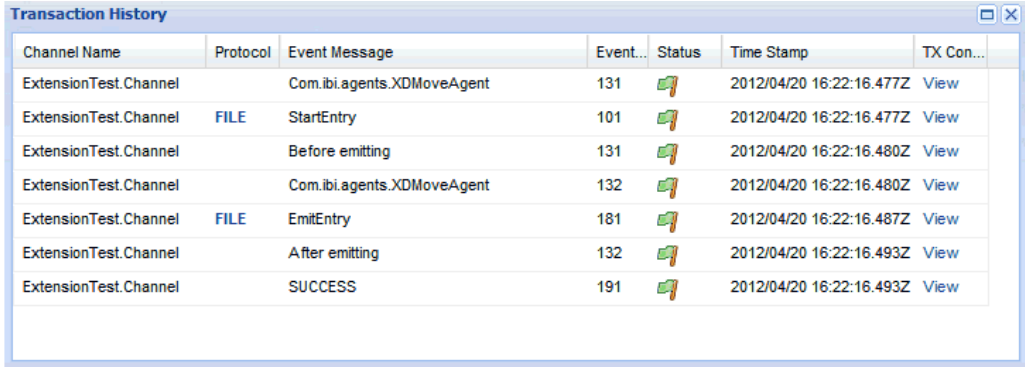
The information displayed by the TX History view depends on the iWay Business Activity Monitor options that are configured.

- The Startentry and Endentry options (success/fail) are always displayed.
- Based on the Emit option configuration, the Emit messages may or may not be displayed.
- Based on the Want Events option configuration, the internal steps of the transaction (such as every single step of the process) may or may not be displayed.
- Based on whether the application uses the specialized Log Message services, the Business Error events and Check Point events may or may not be displayed.

Note: The application design process determines the account performance requirements and detailed view requirements. For example, increased logging that is performed for each transaction will result in a loss of performance. It is recommended to disable the Want Events option to maximize performance and to utilize specialized Log Message services to log transaction check points at needed locations. For more information about Log Message services, see the *TIBCO iWay® Service Manager Component Reference Guide*.

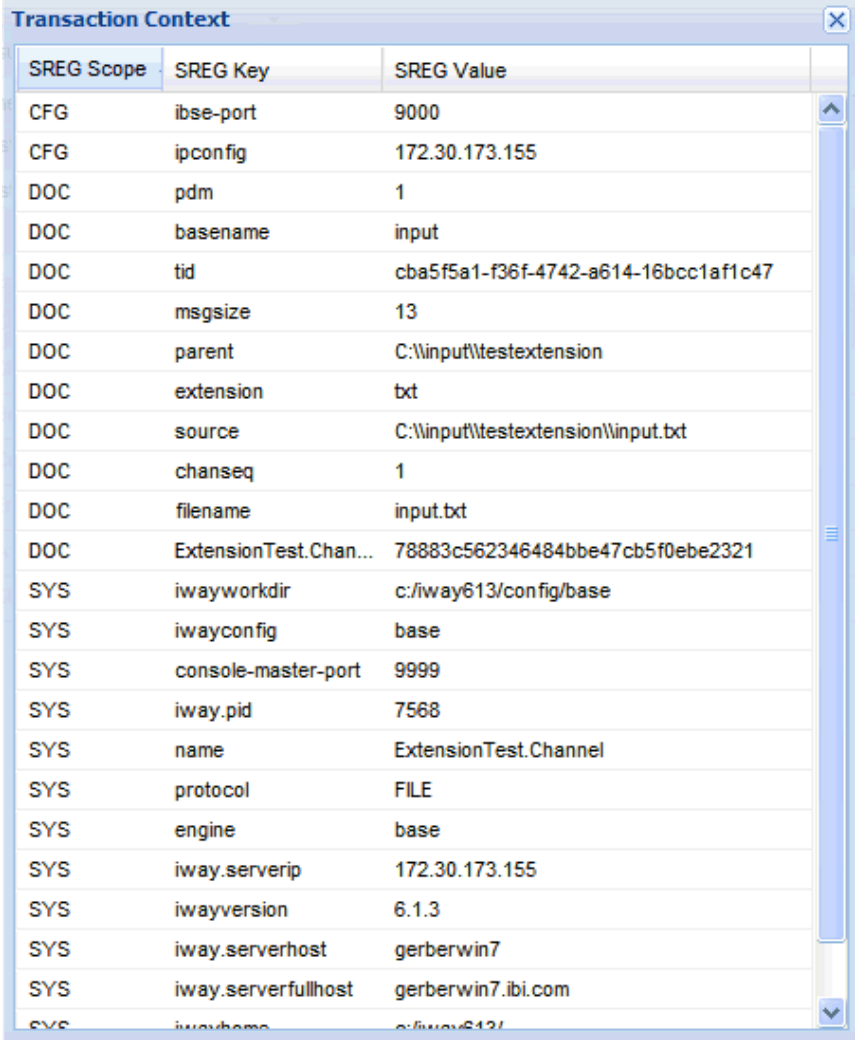
In the example that is shown, all of the events are enabled to demonstrate a full cycle of the transaction.

The following image shows the steps and transactions that occurred in the process flow.



After the start node, the Source Name node has been executed, followed by the Decision Test node to test, followed by a call to IWFAIL, indicating an error. It is up to the application to put as much detailed information into the node and its message to enable you to track the information and identify each step of the process. You can immediately see what step the transaction failed to process.

For a more detailed perspective, clicking *View* under the TX Context column enables the view of the transaction contexts (SREGs) at a given step of the process. This view enables the user to identify what has happened to the transaction and view the attributes, as shown in the following image.



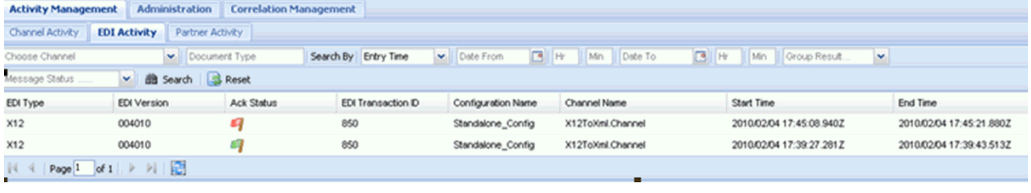
Transaction Context

SREG Scope	SREG Key	SREG Value
CFG	ibse-port	9000
CFG	ipconfig	172.30.173.155
DOC	pdm	1
DOC	basename	input
DOC	tid	cba5f5a1-f36f-4742-a614-16bcc1af1c47
DOC	msgsize	13
DOC	parent	C:\\input\\testextension
DOC	extension	txt
DOC	source	C:\\input\\testextension\\input.txt
DOC	chanseq	1
DOC	filename	input.txt
DOC	ExtensionTest.Chan...	78883c562346484bbe47cb5f0ebe2321
SYS	iwayworkdir	c:/iway613/config/base
SYS	iwayconfig	base
SYS	console-master-port	9999
SYS	iway.pid	7568
SYS	name	ExtensionTest.Channel
SYS	protocol	FILE
SYS	engine	base
SYS	iway.serverip	172.30.173.155
SYS	iwayversion	6.1.3
SYS	iway.serverhost	gerberwin7
SYS	iway.serverfullhost	gerberwin7.ibi.com

EDI Activity Tab

The EDI Activity tab is used to display EDI activities. These activities are automatically identified based on the configured Ebix for the specific channel in the iSM Administration Console. This view is populated only if there is a valid EDI-based channel running on the system. The transaction details are similar to channel activity and are based per transaction.

For EDI-specific transactions, there are two types of statuses: an overall transaction status as it is processed, and an acknowledgment status indicating the state of the acknowledgment for the transaction. Note that the transaction can be successfully processed even though there are warnings or errors in the acknowledgment document. Properly handled errors in the application channel generates a successful state for the transaction as the execution has been able to complete all the steps in the application.



The following table lists and describes all the columns that are available in the EDI Activity tab of the Activity Management facility.

Column Name	Description
EDI Type	Displays the type of EDI document that is received (for example, X12).
EDI Version	Displays the version of the EDI document that is received (for example, 4010).
Ack Status	The acknowledgment status is shown as a flag. ❑ A green flag indicates that the message is accepted. ❑ A red flag indicates that the message is rejected.
EDI Transaction ID	Displays the EDI transaction ID of the EDI document that is received (for example, 850).
Configuration Name	Name of the iWay Business Activity Monitor configuration that was provided during the multi-node configuration.

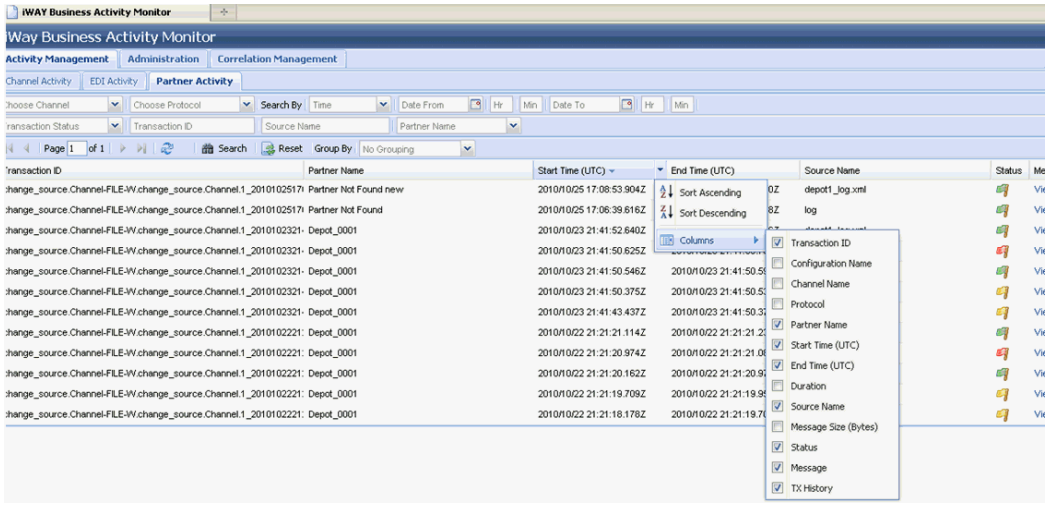
Column Name	Description
Channel Name	Name of the channel on which the message is being executed.
Start Time	Start time is shown in GMT at which iWay Business Activity Monitor logged the start time of the message.
End Time	End time is shown in GMT at which iWay Business Activity Monitor logged the end time of the message.
Status	The overall message status is shown as a flag. ❑ A green flag indicates that the message execution has executed successfully. ❑ A red flag indicates that the message execution has failed. ❑ A yellow flag indicates a business level warning has generated. ❑ A blue flag indicates that the message execution status cannot be determined.
Message	Hyperlink opens the message with which the transaction was started.
TX History	Shows the transaction history and transaction path and steps of the given transaction.

Partner Activity Tab

The Partner Activity tab is used to display the activity that is occurring for partners. This is useful for business users to know how many transactions were executed for a particular partner. Activities are captured only if the bam_tpm_partnerid and bam_tpm_partnername Special Registers (SREGs) are set. If these SREGs are not set, then the Partner Activity tab will be empty. The values of these special registers are set in the process flow dealing with the transaction, either based on the information available from the incoming document or based on the TIBCO iWay® Trading Partner Manager look up. The Partner Activity enables you to group and organize transactions for any given partner and filter them based on various criteria.

Note: In the current release, the monitoring of partner information can also be achieved in the Transaction Activity tab by utilizing monitoring of the user-defined registers for partner information.

The following image shows the contents of the Partner Activity tab displayed in the iWay Business Activity Monitor console.



The following table lists and describes all the columns that are available in the Partner Activity tab of the Activity Management facility.

Column Name	Description
Transaction ID	Unique ID auto-generated by the processing channel.
Partner Name	Name of the partner for whom the transaction is being processed.
Start Time (UTC)	Start time is shown in GMT at which iWay Business Activity Monitor logged the start time of the message.
End Time (UTC)	End time is shown in GMT at which iWay Business Activity Monitor logged the end time of the message.

Column Name	Description
Source Name	Name of the transaction depending on the protocol. For example, File and FTP protocols generate the name of the file read. The Queuing (MQ) protocol generates the name based on the header values. This value can also be set using the bam_sourcename register.
Status	The message status is shown as a flag. ❑ A green flag indicates that the message execution has executed successfully. ❑ A yellow flag indicated a business level warning has generated. ❑ A red flag indicates that the message execution has failed. ❑ A blue flag indicates that the message execution status cannot be determined.
Message	Hyperlink opens the message with which the transaction was started.
TX History	Shows the transaction history and transaction path and steps of the given transaction.

You can also customize the view and clear or select any column for viewing. Some of the additional non-default columns are shown in the following table.

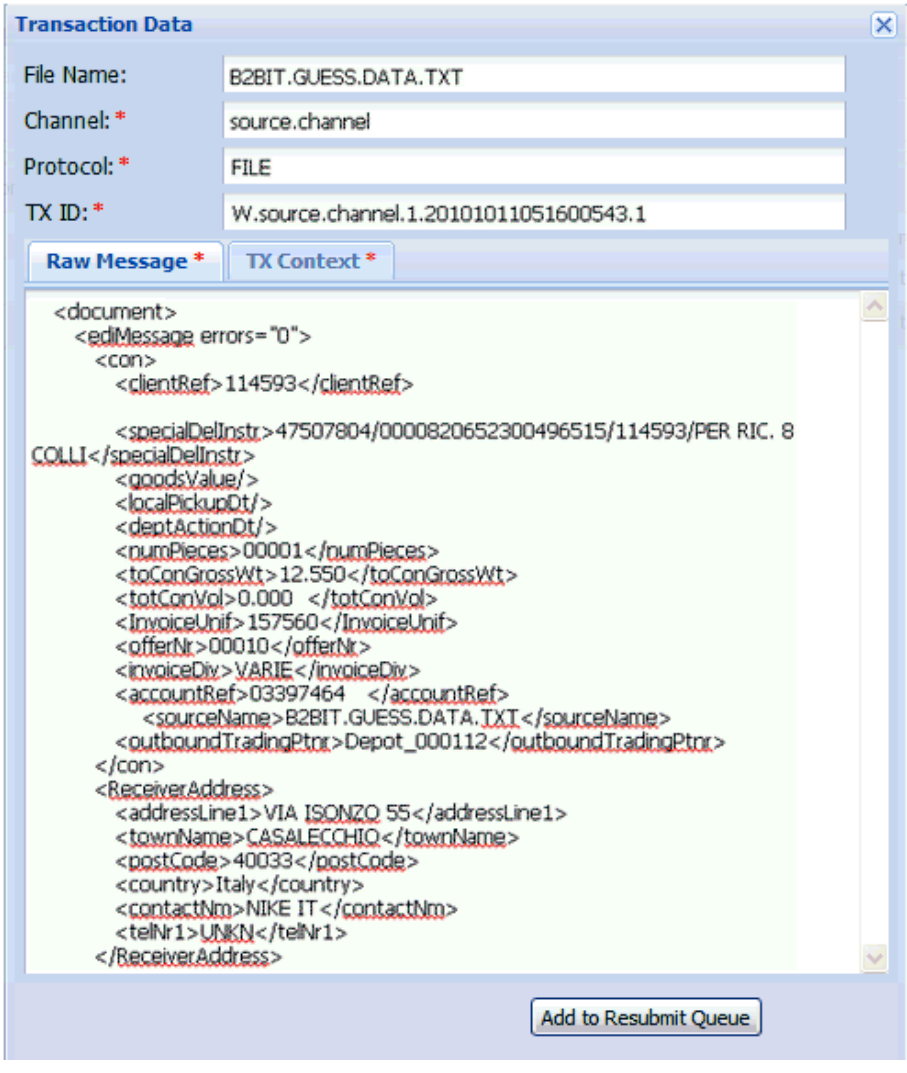
Column Name	Description
Duration	Total time it took to complete the transaction.
Message Size	Size of the processed transaction.
Configuration Name	Name of the configuration where the transaction has been processed. In a multi-node deployment, the name can be used to identify multiple client configurations.
Channel Name	Name of the processing channel.

Column Name	Description
Protocol	Protocol on which a transaction has been received and processed.

Messages View

To view the partner message, click the *View* hyperlink under the Message column, as shown in the following image.

The Transaction Data dialog box opens, as shown in the following image.



The following table lists and describes the available fields and tabs in the Transaction Data dialog box.

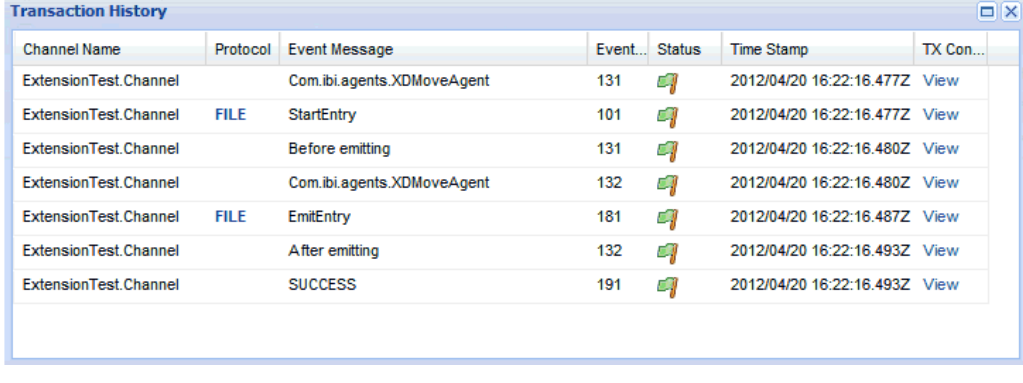
Field/Tab	Description
File Name	Name of the file received for processing (availability depends on protocol).

Field/Tab	Description
Channel	Associated channel for the message.
Protocol	Protocol on which the message is received.
TX ID	Transaction ID for the message.
Raw Message tab	Displays the actual raw data of the message.
TX Context tab	Provides information regarding the Special Registers (SREGs) associated with the processed messages. Based on your permissions, all field values can be edited and resubmitted. For more information, see *Message Resubmission* on page 52.

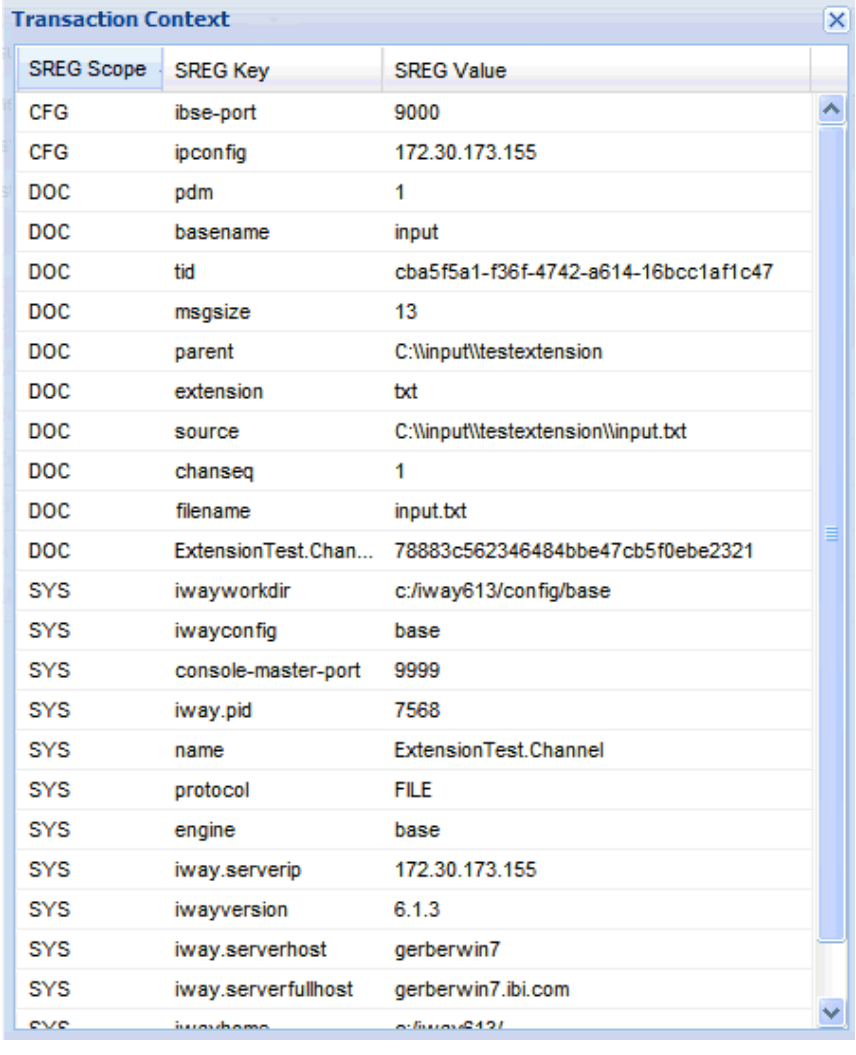
Transaction History View

Clicking the *View* hyperlink under the TX History column enables the detailed view of the transaction history. It displays every step logged for the executed transaction. This helps determine the possible reason for a transaction failure.

For example, in the following image, you can see the transaction steps of the process flow. After the start node, the Source Name node has been executed, followed by the Decision Test node to test, followed by a call to IWFAIL, indicating an error. It is up to the application to put as much detailed information into the node and its message to enable you to track down the information and identify each step of the process. You can then see at what step the transaction failed to process.



For more information, click *View* under the TX Context column. This enables the view of the transaction contexts (SREGs) at a given step of the process. This view provides more details and enables you to identify what happened to the transaction and view the attributes, as shown in the following image.



Transaction Context

SREG Scope	SREG Key	SREG Value
CFG	ibse-port	9000
CFG	ipconfig	172.30.173.155
DOC	pdm	1
DOC	basename	input
DOC	tid	cba5f5a1-f36f-4742-a614-16bcc1af1c47
DOC	msgsize	13
DOC	parent	C:\\input\\testextension
DOC	extension	txt
DOC	source	C:\\input\\testextension\\input.txt
DOC	chanseq	1
DOC	filename	input.txt
DOC	ExtensionTest.Chan...	78883c562346484bbe47cb5f0ebe2321
SYS	iwayworkdir	c:/iway613/config/base
SYS	iwayconfig	base
SYS	console-master-port	9999
SYS	iway.pid	7568
SYS	name	ExtensionTest.Channel
SYS	protocol	FILE
SYS	engine	base
SYS	iway.serverip	172.30.173.155
SYS	iwayversion	6.1.3
SYS	iway.serverhost	gerberwin7
SYS	iway.serverfullhost	gerberwin7.ibi.com

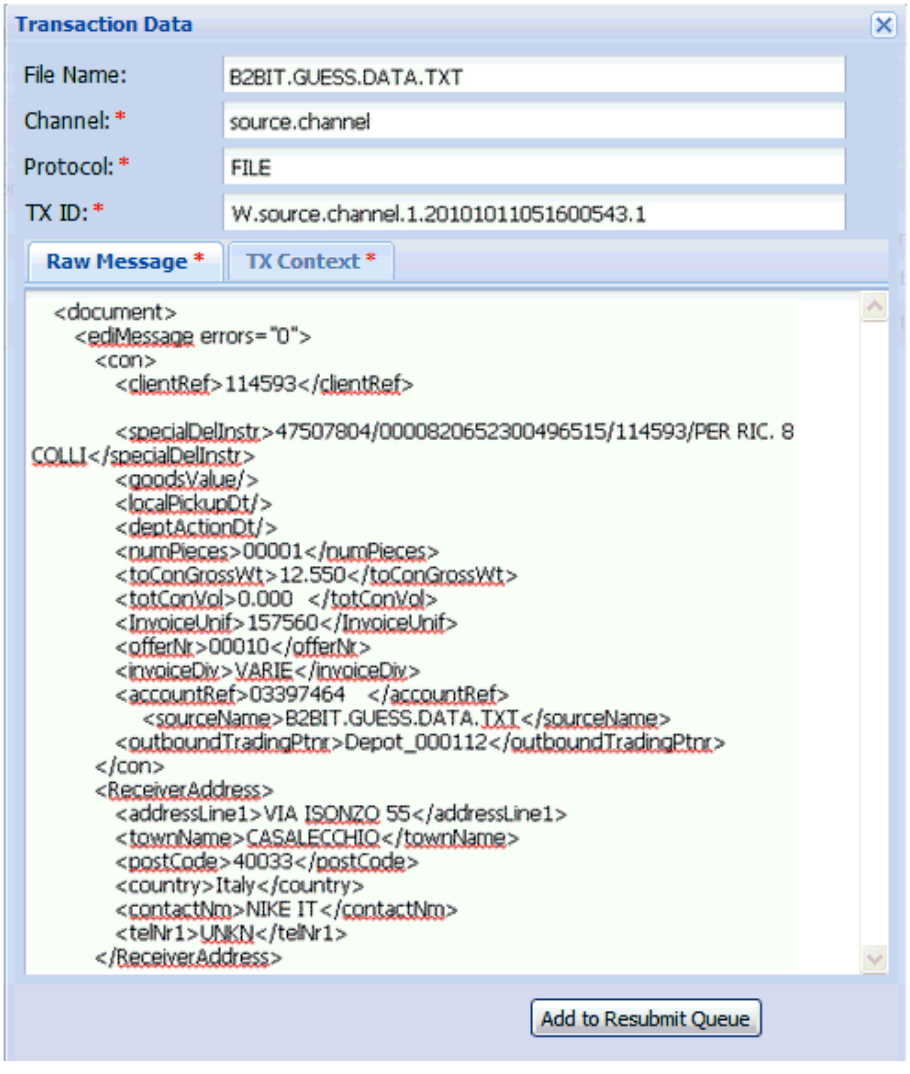
Message Resubmission

Message resubmission allows a message to be retrieved for a particular transaction. It returns the message to be retrieved for a transaction with which it was run.

The View hyperlink under Message in the Activity Management section enables you to view the specific data of the message used for the transaction and resubmit the message either with changes or as is prior to resubmission. The View of the message provides default parameters, which you can modify to assist in the resubmitting.

Click *View* in the Message column for a corresponding transaction, as shown in the following image.

The Transaction Data dialog box opens, with the message that the selected transaction used to execute, as shown in the following image.



You can provide a file name to save the document and also change the document content as displayed. Click *Resubmit Message* once you are ready.

The document will be made available under the following directory:

iway_home\config*current_config*\bam\resubmit*channel_name**file_name**TX_ID*

where:

iway_home

Is the directory where iSM is installed.

current_config

Is the configuration where the server is running.

channel_name

Is the channel name provided in the iWay Business Activity Monitor Resubmit window. This enables you to resubmit the documents to a specified directory. The directory is auto-created if it does not exist.

file_name

Is the file name provided in the iWay Business Activity Monitor Resubmit window. This enables you to rename the file for resubmission or keep the default, which is the source name for the transaction.

TX_ID

TX ID provided in the iWay Business Activity Monitor Resubmit window. This should be left as the default value as it enables you to look up further information from the iWay Business Activity Monitor database for the specific Transaction ID. However, the value can be changed to anything else as mentioned in the application requirements.

The document is now available (for example, to run in a process flow). Additional logic to use the document must be designed accordingly, but in most cases, the logic can involve moving the document from this directory to another.

Once the transaction is stored in the resubmit directory, the application will implement a resubmit channel to process the transaction. In most cases, the resubmit channel should validate that the transaction is allowed to be resubmitted and to which channel it should be routed to based on the transaction information.

Resubmit Facility Overview

Resubmitting a transaction is an application-supported service. iWay Business Activity Monitor provides tools to assist in developing a resubmission facility, but does not, alone, affect a resubmission.

The main tools needed to develop a resubmission capability are:

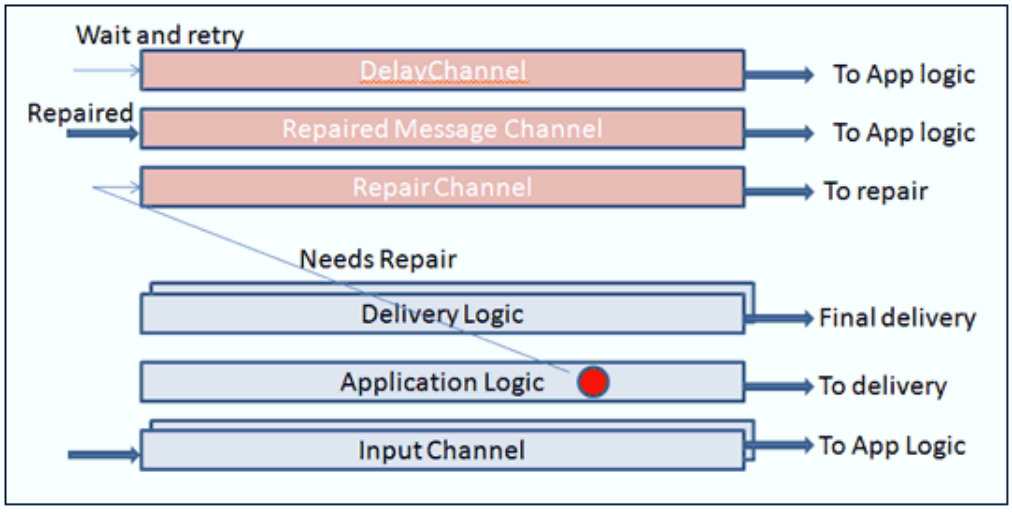
- An Emit Agent that moves the message or message reference to be resubmitted to a queue or other location in which a channel can find and work with the message.

- The XDMarshallAgent (currently packaged with the iwgateway extension) that associates the current context with the message to be resubmitted.
- One or more application-defined special registers holding information, stored when the message is selected to be repaired and resubmitted, pertaining to the cause of the resubmission. This can be used during the resubmission present error information to a user, to the hold name of a channel to which the message is to be resubmitted, and so on.

Resubmission works most effectively with a multiple-channel architecture in which the application logic is separated from the initial acquisition of the message. In this case, the repaired and resubmitted message can be passed to the application logic at an appropriate stage of the execution.

Once a message is selected for resubmit, it needs to be written, by application logic, to an appropriate resubmit queue. From there, it might be passed to a repair station or simply set up for a later retry. Selection of a message is either by application logic (perhaps a communication failed or iWay Data Quality Center detected a rule violation in the original message) or manually in a form in which the message is selected and as a consequence written to the appropriate queue for resubmit handling.

A sample system is shown in the following image.



The application acquires the message through an input channel, which in the example elects to pass it on to the application logic. At some point(s) in the application logic, the message is tested. If the test passes, the message moves on, eventually passing to the delivery channel. If, however, the message fails the test (shown by the circle with the arrow) the message might be, depending on the test, written to the repair channel. Other errors might send it to a delay channel to simply await some triggering event. The scheduler facility can help with this. In our diagram, the repair channel sends the message to a repair station, where it is sent to a user to be modified, and from there the result arrives at the repaired message channel. From there, the message is passed back to the application logic.

The process of writing to the target repair or delay channel would encompass steps on the resubmit (failure) branch of the application logic flow, such as the following.

1. Load agreed-upon special registers with information needed to understand and repair the cause of the failure. Examples might include:
   a. What test failed
   b. What part of the message needs to be examined (if any)
   c. Error message describing the failure
2. Use the XDMarshallAgent to associate the registers with the message to be passed. The record identifier (key) needed to obtain the actual message is previously stored by the iWay Business Activity Monitor driver in the resubmitSourceKey special register. An SQLAgent can load the message as it arrived on the Application Logic channel that issued the resubmit using this key. Alternatively, the message as it stood when it entered the Emit Agent to the resubmit queue is the message that reaches the resubmit queue.
3. The marshaled message is sent through an emit agent (probably internal emit, but perhaps another protocol) to the appropriate channel (repair channel or delay channel in our example diagram).

The Transaction ID will follow the message from channel to channel, enabling later analysis to associate the message with all of the steps (including the resubmission(s)) taken toward its final disposition.

The action on the repair channel itself or the delay is part of the application logic and is not a part of the resubmit facility.

Selecting Messages to be Resubmitted

A message can be resubmitted from various points of the process to provide extended flexibility to the application. Based on the user privileges, the message and its context (SREGs) can be modified for a resubmit process.

Two types of resubmit options are available:

- Message at hand resubmit
- Selectable resubmit

Message At Hand Resubmit

The message at hand resubmit option enables you to add the current message being viewed to the resubmit queue for processing. This type of resubmit takes the message with its context and adds it to the iWay Business Activity Monitor resubmit table directly without any pre-processing. This would include adding a message from the Message or Transaction History views.

- Resubmit from the Message view

  In the Message view, there is an option to add the selected message to the resubmit queue. This will add the original message to the resubmit queue.

- Resubmit from the Transaction History view

  Any Emit or Check point available in the Transaction History enables you to view the message under the Protocols column. The message can be added to the resubmit queue at any point, enabling you to add a partially processed message to the resubmit queue. This avoids duplicate work of the original application logic.

Selectable Resubmit

The selectable resubmit option enables you to add multiple messages to the resubmit queue at the same time. In this case, the transaction ID for each selected message is propagated to the pre-configured channel (BAM_TID_RESUBMIT_Channel), which, based on the configuration, selects the proper record to resubmit. The channel should be modified to meet your specific application requirements. By default, the original (101 event) message is added to the resubmit queue for the given transaction ID. For more information on the BAM_TID_RESUBMIT channel, see *Configuring the BAM_TID_RESUBMIT_CHANNEL* on page 59.

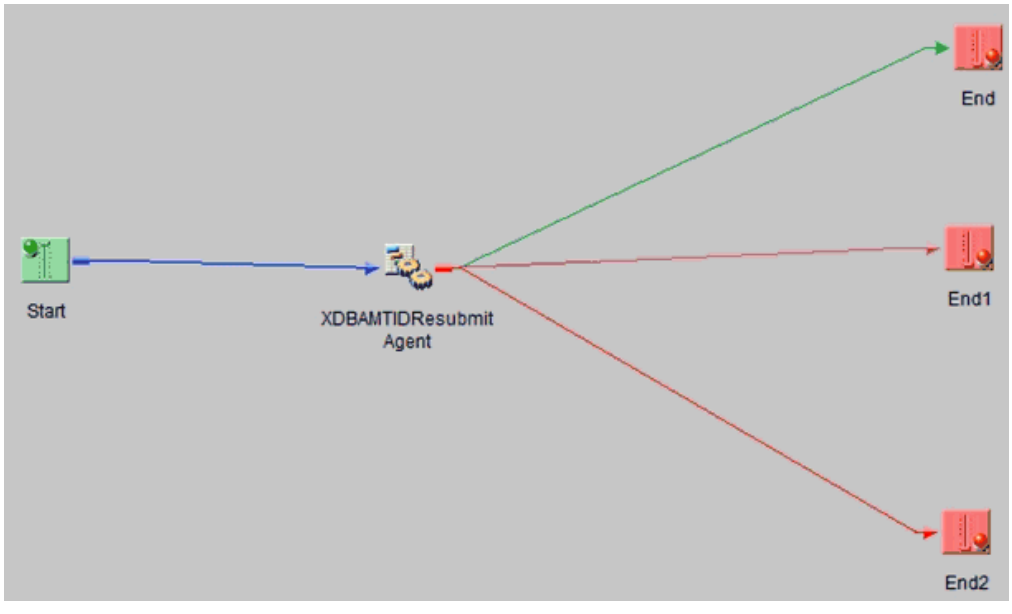
Mutli-message Resubmit

The Transaction Activity tab enables you to select multiple messages to be added to the resubmit queue by selecting multiple check boxes next to the messages. This will add the original message for each transaction to the resubmit queue.

Configuring the BAM_TID_RESUBMIT_CHANNEL

iWay Business Activity Monitor includes a pre-configured channel called BAM_TID_RESUBMIT_CHANNEL. This channel enables you to handle selective resubmits for multi-message resubmission. You must import, build, and deploy the channel to make it available to the application. If this channel is not available, then the resubmit facility will not work and will generate an error message indicating that the channel is not available.

The BAM_TID_RESUBMIT_CHANNEL is based on an internal listener and has a corresponding process flow, as shown in the following image.



It is recommended that you modify this channel to meet your specific application requirements, as described in the following sections.

iWay Business Activity Monitor Web Console Perspective

Using the iWay Business Activity Monitor web console, when you select transactions and click *Add to Resubmit Queue*, the iWay Business Activity Monitor logic creates an XML document for each selected transaction, which includes the following information about the selected transaction:

- Transaction ID for the selected transaction.
- Record key 101 for the first record.

- Name of the channel to which the record key applies.
- User login ID.
- Host (IP address) from which the form was submitted.
- Date and time.

The generated XML document is written to an internal listener queue, called BAMTIDResubmit, which is hosted by the BAM_TID_RESUBMIT_CHANNEL. When a document for each selected transaction ID is created, the operation is considered complete and a message indicating *Done* is displayed in the console. If the internal queue name is not available for any reason, an error message is displayed in the console.

User Process Flow

The user process flow associated with the internal queue channel receiving the resubmit record can analyze the input request in any necessary manner. As a final step in the process flow, it must call the service called BAMTIDResubmitAgent to enable the resubmit action. No output is expected or required from the user process flow. However, since the process flow operates under normal rules, it can take any additional channel action desired by the application designer.

The actual resubmit record key that has been determined by the process flow from the resubmit document is the data that will be resubmitted.

The process flow must handle the result edges from the service that are listed in the following table.

Edge	Purpose
success	Request has been added to the resubmit release queue.
fail_notfound	Record key given to the service does not represent a record in the iWay Business Activity Monitor database.
fail_security	User does not have authority to submit this record.
fail_operation	Resubmit release queue could not be updated.

The original terminal user will not see these errors, as the internal channel is operating asynchronously. Therefore, the process flow must handle the error, perhaps by sending an email or adding a message to a queue to be reviewed by the appropriate authority.

Processing Resubmitted Messages

After the messages have been added to the resubmit queue, a user who has authorized credentials for the iWay Business Activity Monitor web console can view, modify, and release resubmission from this queue.

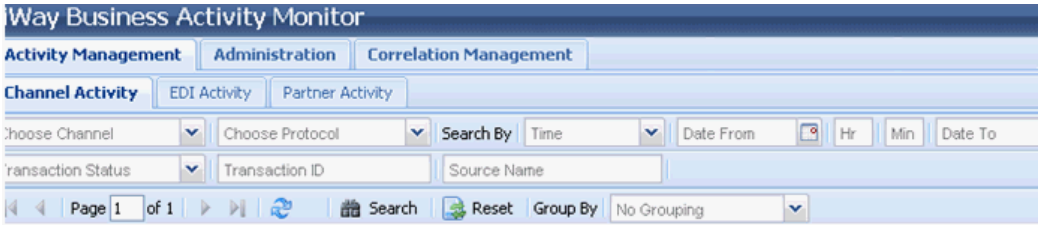
When a message is resubmitted, it is written to the BAMResubmit internal queue. An internal listener should be configured by the user to process this queue. If this internal queue does not exist, then the message will be written under the default location configured in the Preference tab.

On entry to the process flow of the channel, the message and context is configured according to the resubmitted message. The process flow can then take any desired action based on your application requirements. Common actions include:

- Sending the message through the Internal Emit service (XDInternalEmitAgent) to the queue associated with the application process for the message. For example, this might be a step in the actual application flow that can now deal with the message after it has been modified or repaired.
- Serializing the message and sending it to another appropriate protocol. The Marshal service (XDMarshalAgent) is available to assist in formatting such a message. iSM features such as Asynchronous Forward Transfer Interface (AFTI) can assist with this activity.
- Sending the data to another table for additional, application-assisted modification and handling.

Using the Search Function

Search functionality, as shown in the following image, is provided in all Activity Management facility tabs to filter the desired results.

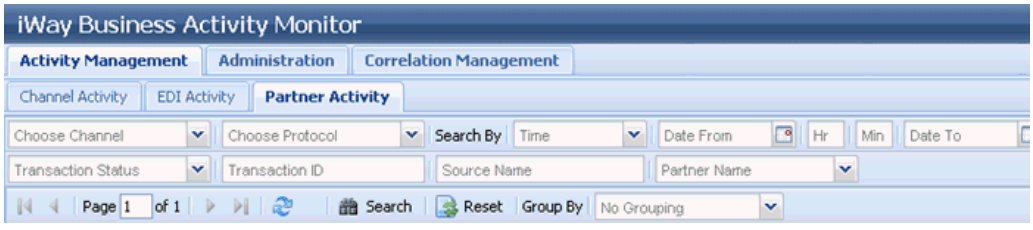


Provide search criteria in the corresponding fields and click *Search* to display only the filtered records based on the search criteria. Click *Reset* to remove the selected search criteria and show all the records.

The following table lists and describes all the search fields that are available.

Field Name	Description
Choose Channel	Allows data filtering based on a specific channel name. Channel data is displayed for which iWay Business Activity Monitor has collected the data.
Protocol	Allows searching based on the protocol type that is used in the channel (for example, Internal, File, MQ Series).
Search By	Start and End time indicating the filtering which should be applied to the processed transactions based on time stamp initialization or completion.
Time	Time at which the message was first monitored on iSM.
Date From	Date since the message was processed.
Hr	Hour from which the message starts to be monitored.
Min	Minutes from which the message starts to be monitored.
Date To	Date range up to which the message needs to be filtered.
Hr	Hour up to which the message needs to be filtered.
Min	Minute up to which the message needs to be filtered.
Transaction Status	Allows transaction search based on the transaction status of Success or Fail.
Transaction ID	Allows filtering based on the Transaction ID.
Source Name	Allows filtering based on the Source Name of the transaction.

The Partner Activity Facility provides an additional filtering capability to search by partner name, as shown in the following image.

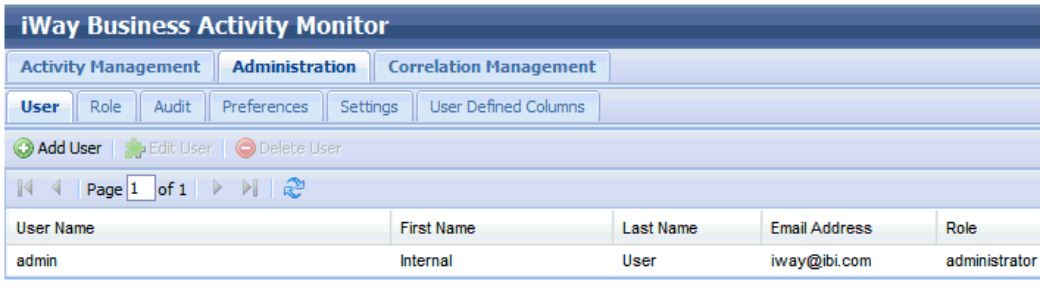


The following table lists and describes the additional search field in the Partner Activity tab.

Field Name	Description
Partner Name	Allows filtering based on the Partner Name for the transaction.

Administration Tab

The Administration tab, shown in the following image, allows users with administrator privileges to create users who can access an iWay Business Activity Monitor application.

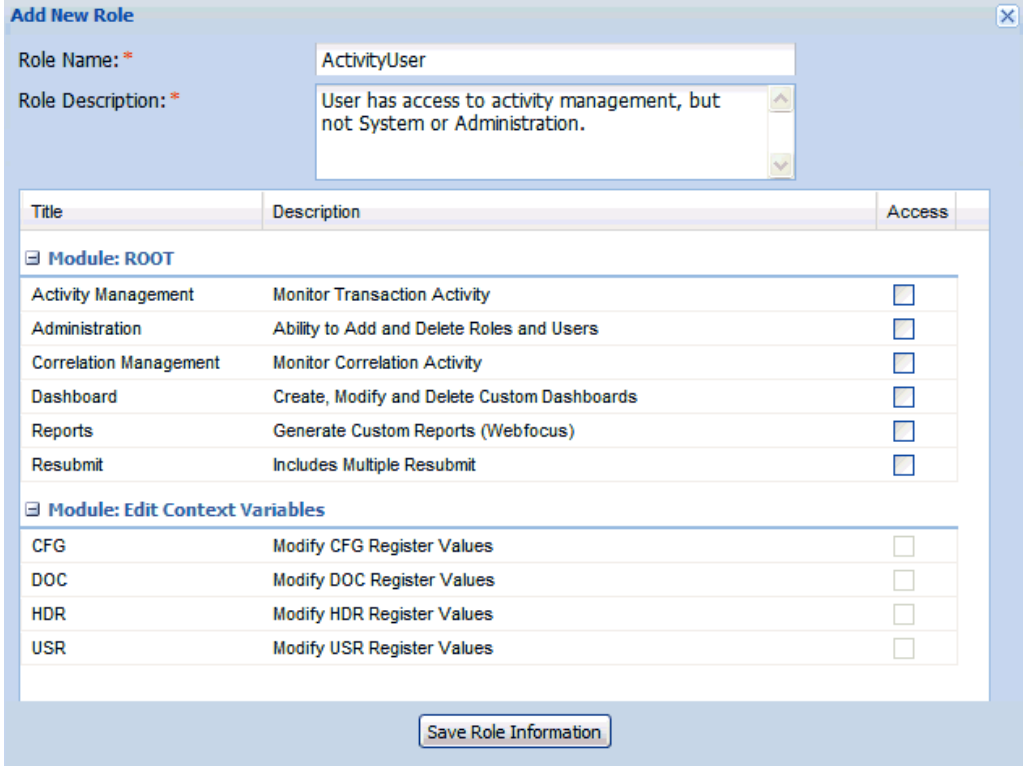


Role Management

The Role tab allows administrators to manage existing roles or create new roles for users. Roles define specific access to given modules of the iWay Business Activity Monitor application, such as enabling fine control of grouping users into assigned roles for specific access.

A new role can be added by clicking *Add Role* under the Role tab, as shown in the following image.

Once a role is configured for a user, this user will only have access to the specific modules that have been assigned after logging into the iWay Business Activity Monitor console, as shown in the following image.

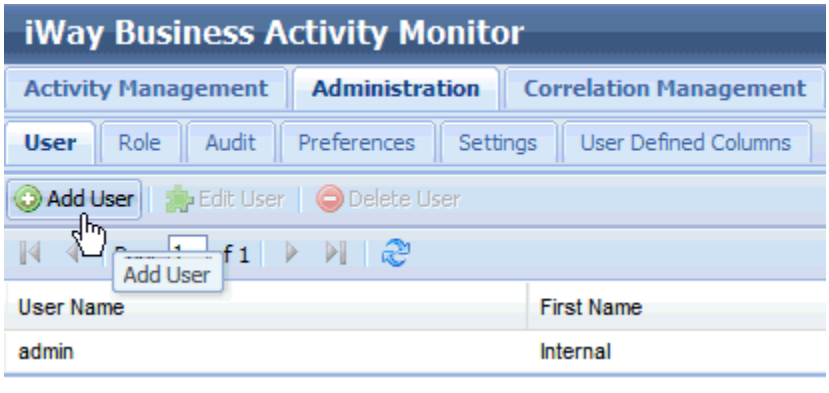


User Management

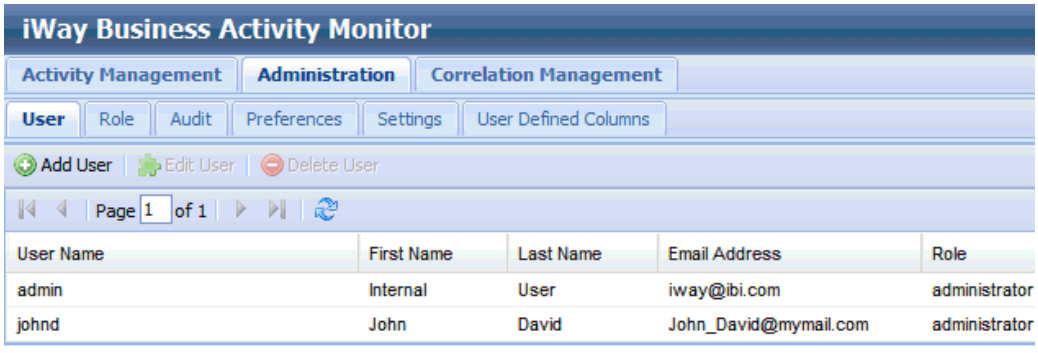
The User tab allows administrators to manage existing users or add new users, as shown in the following image.

Added users are associated with specific roles that have been assigned to them. After logging into the iWay Business Activity Monitor console, each user is granted with the privileges that their role allows.

A new user can be added by clicking the *Add User* button under the User tab, as shown in the following image.



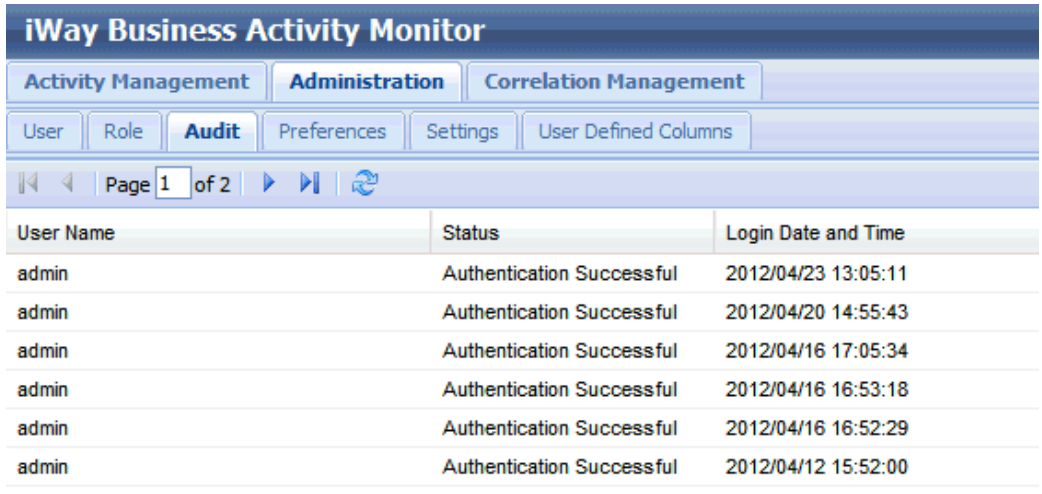
The user configuration pane, shown in the following image, enables the administrator to create a new user account and associate it with the specific role available in the iWay Business Activity Monitor application.



The newly created user appears in the list of existing users with its associated data. The administrator also has the authority to temporarily block user access by editing the user account. Blocking a user enables a temporary lock on the user account without actually deleting the user from the system.

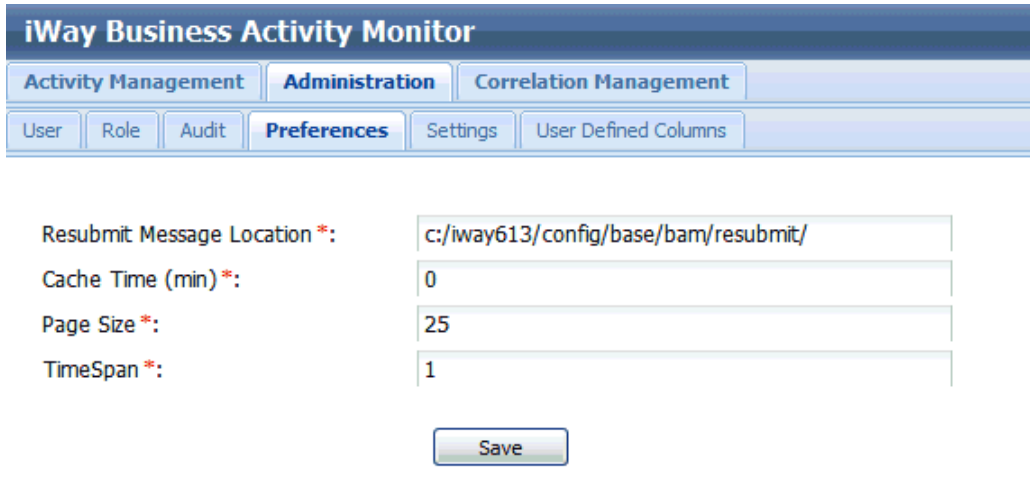
User Audit

The Audit tab, shown in the following image, enables the administrator to monitor all the logon activity. The administrator can monitor all the successful and failed logon attempts with their corresponding date, time, and IP address. It is useful to monitor the overall user access status to the iWay Business Activity Monitor application.



User Preferences

The Preferences tab provides additional configuration options, as shown in the following image.



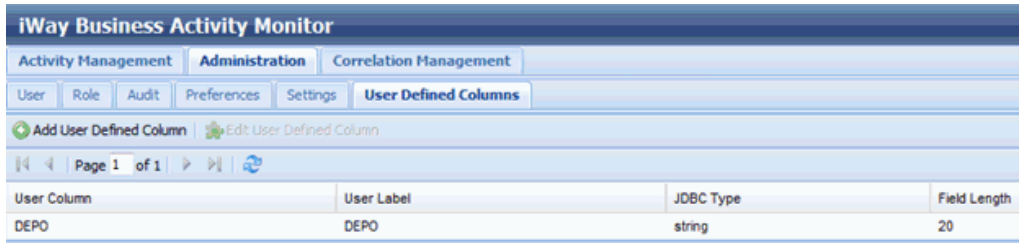
The following table lists and describes the available fields in the Preferences tab.

Field	Description
Resubmit Message Location	Default message resubmit location if a file-based resubmit is used.
Cache Time (min)	Cache setting used to improve the performance of the iWay Business Activity Monitor console. This value indicates how often the data in the Activity screens is refreshed for the following selections: ❑ Choose Channel ❑ Choose Protocol ❑ Configuration Name ❑ Transaction Status Setting a value of 0 will refresh the drop-down list selections immediately.
Page Size	Number of records to be displayed per page.
TimeSpan	Default time interval in hours for which data is displayed.

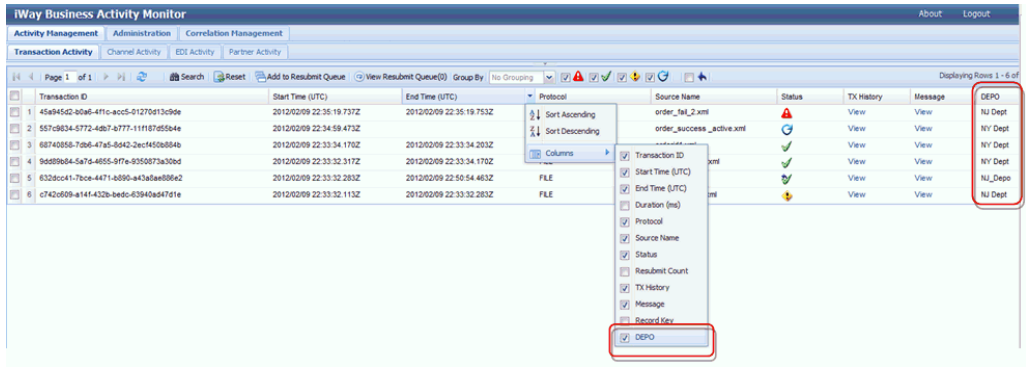
The following example shows the result in the iWay Business Activity Monitor console when TimeSpan is set to one hour.

Monitoring User-Defined Registers

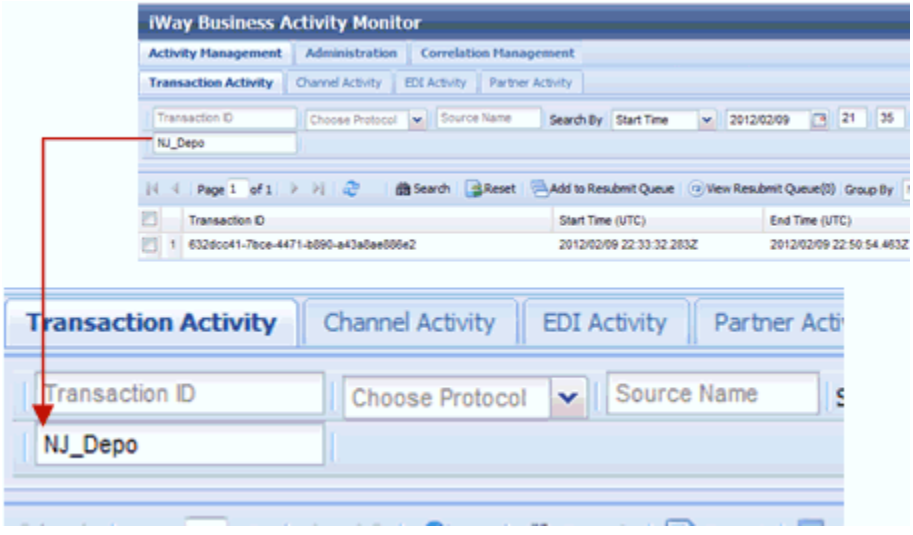
Monitoring user-defined registers enables you to define additional columns in iWay Business Activity Monitor corresponding to the application-defined special registers for data monitoring. As a result, you can store additional information (for example, DepartmentCode, Country, MessageID, and so on) as a special register in the application and then display this information as an additional column in the iWay Business Activity Monitor console. To create user-defined columns, click the *Administration* tab and then the User Defined Columns tab, as shown in the following image.



Once it is added and the server is restarted, the column and its data are included as part of the column list available for display in the iWay Business Activity Monitor Transaction screen, as shown in the following image.



The user-defined column is also added into a filter category so you can search for values in the user-defined field, as shown in the following image.



Correlation Management Tab

The following sections describe Correlation Management in iWay Business Activity Monitor, and its requirements.

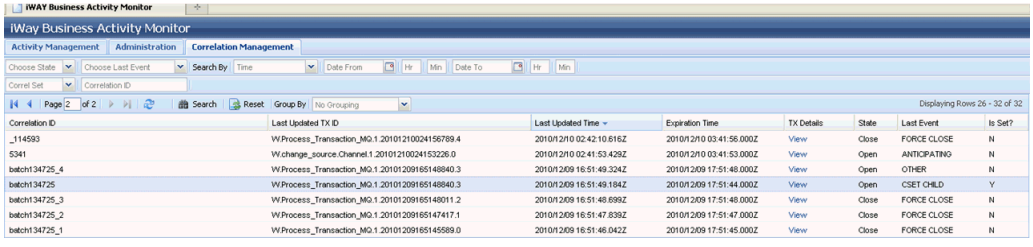
Correlation Management

Correlation Management enables the tracking of long-running transactions across multiple channels or servers. These transactions are correlated as standalone transactions or as part of a correlation set for batch processing. Transactions are correlated using the user-defined correlation ID as single message processing can produce multiple sub-transactions with different Transactions IDs that need to be correlated together.

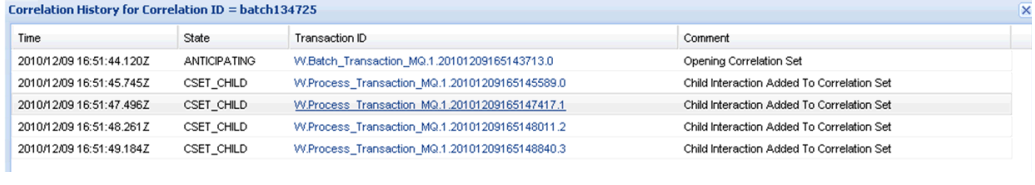
Set Correlation plays an important role during batch processing, where each transaction is correlated to a specific batch correlation set. The transaction correlation state can be updated at any point of the process. This allows transactions to be processed by multiple channels, where each channel can update the transaction state of a single transaction providing a unified view through the entire processing cycle.

Transaction overall state is denoted as open or closed, indicating the final transaction state. Transaction Details provide the transaction correlation history from the initial Open state to various Functional, Business, or Customized transaction updates. This is a key feature enabling the user to track the transaction through all correlation steps and identify actions performed by any given channel on a transaction.

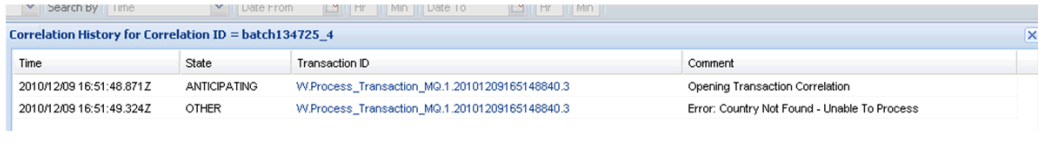
The Correlation Management tab, shown in the following image, enables the user to see the overall state of the transaction. It provides important information, such as Correlation ID, which is used throughout the transaction life cycle, as well as the overall state of the transaction and the Last Event which occurred for the given transaction.



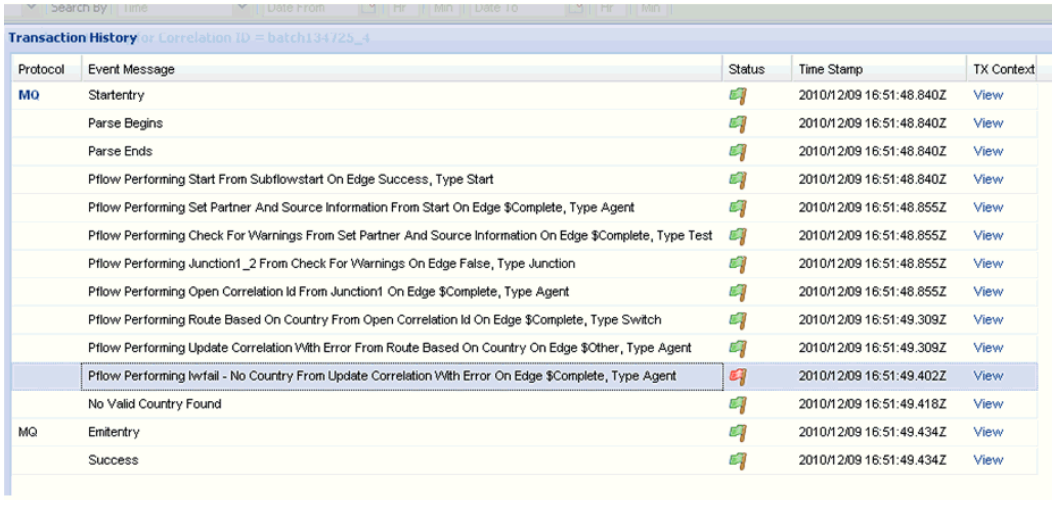
The user can view additional information for the transaction by accessing TX Details (Transaction Details) which provides a detailed view into the transaction history. For example, the following image shows the correlation set being open for the batch transaction followed by linking of the four child transactions.



When looking at a singular transaction from the correlation management view, the user can see various updates to the transaction state. For example, the following image shows the transaction in the Open state since the routing country cannot be found for the transaction.



The user can drill down even further to investigate the error by clicking the associated transaction ID to see the details. The error is marked with a red flag, indicating in the transaction history where the country for the transaction cannot be set, as shown in the following image.



Transaction History for Correlation ID = batch134725_4

Protocol	Event Message	Status	Time Stamp	TX Context
MQ	Startentry		2010/12/09 16:51:48.840Z	View
	Parse Begins		2010/12/09 16:51:48.840Z	View
	Parse Ends		2010/12/09 16:51:48.840Z	View
	Pflow Performing Start From Subflowstart On Edge Success, Type Start		2010/12/09 16:51:48.840Z	View
	Pflow Performing Set Partner And Source Information From Start On Edge $Complete, Type Agent		2010/12/09 16:51:48.855Z	View
	Pflow Performing Check For Warnings From Set Partner And Source Information On Edge $Complete, Type Test		2010/12/09 16:51:48.855Z	View
	Pflow Performing Junction1_2 From Check For Warnings On Edge False, Type Junction		2010/12/09 16:51:48.855Z	View
	Pflow Performing Open Correlation Id From Junction1 On Edge $Complete, Type Agent		2010/12/09 16:51:48.855Z	View
	Pflow Performing Route Based On Country From Open Correlation Id On Edge $Complete, Type Switch		2010/12/09 16:51:49.309Z	View
	Pflow Performing Update Correlation With Error From Route Based On Country On Edge $Other, Type Agent		2010/12/09 16:51:49.309Z	View
	Pflow Performing Iwfail - No Country From Update Correlation With Error On Edge $Complete, Type Agent		2010/12/09 16:51:49.402Z	View
	No Valid Country Found		2010/12/09 16:51:49.418Z	View
MQ	Emitentry		2010/12/09 16:51:49.434Z	View
	Success		2010/12/09 16:51:49.434Z	View

Application Requirements

To monitor correlation activity in iWay Business Activity Monitor, the application has to incorporate the use of correlation agents to manage the state of the transactions. Unlike the automated activity monitoring available in Activity Management of iWay Business Activity Monitor, Correlation Monitoring requires a set of correlation agents to be used as part of the application. For more information on Agents, see the *TIBCO iWay® Service Manager Component Reference Guide* and the *TIBCO iWay® Service Manager Functional Language Reference Guide*.

The following agents need to be used in supplication to achieve the proper correlation process:

- **XDAddCorrelEntryAgent.** Enables you to add a correlation entry for the transaction either as a stand-alone correlation ID, a correlation set, or a correlation ID as part of the correlation set. This is the initiation state of the correlation.
- **XDUpdateCorrelEntryAgent.** Enables you to update any given correlation transaction (stand-alone or a correlation set) with any state change such as Business Ack and Close, as well as the customized comment for the correlation update.

The following is a sample flow of processing a batched transaction with correlation:

- Receive a batch of transactions.

- Create and open the correlation set for the received batch of transactions.
- Iterate through each transaction using an iterator.
- For each successful iteration, create and open a correlation ID for subtransaction within the correlation set. For example, CorrelSetName_TransactionCount (myset_1, myset_2, ...).
- Output the singular transactions to a staging area for another channel to pick up and process, such as a queuing system.
- Second channel picks up singular transactions and processes them. If the process is successful, it will close the correlation ID associated with that transaction. If the processing fails, it will update the correlation ID for the given transaction with an associated error message and leave it in Open state.

iWay Business Activity Monitor Driver Recovery

iWay Business Activity Monitor records information into a database during execution of the application. When properly configured, applications and configurations can share this database. The system relies upon the ability to properly insert records into the database with no errors. Most applications using iWay Business Activity Monitor depend on the iWay Business Activity Monitor driver being able to update its database. The server allows the application to determine the action to be taken in the event of failure.

The iWay Business Activity Monitor driver is designed to impose the least possible performance penalty on the application. The driver database update is asynchronous to the application itself. There is no mechanism for the application to check the status of the driver. For this reason, the driver and server must be in control of the recovery.

When the asynchronous updater detects a loss of connection to the database (SQL State 08xxx) it will attempt to reconnect based on the value specified for the retryCount parameter. It will wait between attempts based on the value specified for the retryInterval parameter. If a reconnect cannot be accomplished, the asynchronous updater shifts to recovery mode and reports the error to the driver itself. When all updaters have reported this condition, the driver begins recovery action.

The driver will attempt to run a process flow as configured in the Loss Connection Flow Name parameter of the driver configuration. You must create and publish this process flow to the system area of the configuration.

The process flow can take actions, such as notifying the user of the problem (for example, by sending an email). The post message (XDControlAgent) service might be used to stop accepting messages on other channels, effectively pausing the application. The process flow should end on one of three End nodes, with specific names. This is the standard method by which a subflow reports its status back to a calling process flow. The designated action takes effect following the return from the process flow. For an application that does not depend upon iWay Business Activity Monitor (for example, using it only for statistical and analysis purposes), Force or Finish can allow the application to continue without such statistics.

End Name	Queue Action	Driver Action
Force	Stop accepting and delete all in queue.	Shutdown
Persist (default)	Serialize any pending entries to a file to be retried on startup.	Shutdown
Continue	Continue accepting entries.	Continue the operation. The driver continues in recovery mode.
Offline	Continue accepting entries and store them in the file system.	Continue the operation. The driver continues in recovery mode.

If a serialized queue exists, the driver will deserialize the entries during startup. This will cause the serialized entries to be written to the iWay Business Activity Monitor log.

The xalog start <*drivername*> command can be used to restart the driver when the database condition is corrected. This command can be scheduled, such as in a run script, as needed.

Important: In order to fully recover from a lost RDBMS connection, it is necessary to add a validation query to the JDBC provider configuration. For MS SQL Server, Select 1 is fine. This adds a small performance cost, but makes it possible for the pool to replace bad connections with good connections.

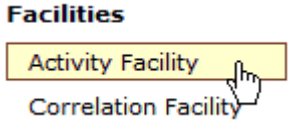
The XDXalogControlAgent is not appropriate for use in this process flow. Use the Continue return to retry the lost connection. An appropriate use of the XDXalogControlAgent service might be to run in a scheduled situation. For example, the lost connection process flow might alert a DBA to problems and then schedule a recovery process flow to test the database connection and restart the driver or reschedule. It would then return Persist, which instructs the server to continue caching events to disk.

Procedure: How to Modify the iWay Business Activity Monitor Driver Recovery Properties

The iWay Business Activity Monitor driver recovery properties are an advanced IT operation and require direct driver configuration.

To modify the iWay Business Activity Monitor driver recovery properties:

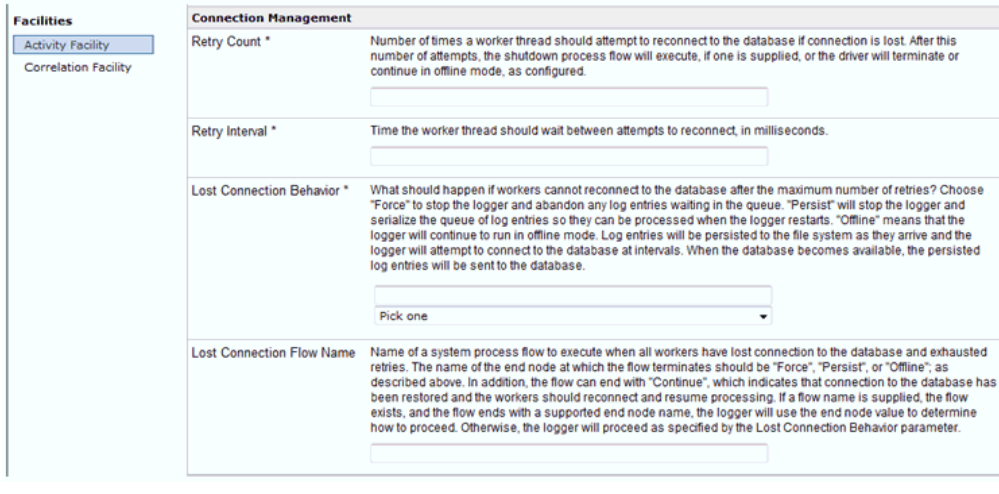
1. In the iSM Administration Console, click *Activity Facility* in the left pane, as shown in the following image.



   The Activity Facility pane opens.

2. Select *BAMSenderDriver*.

   The Activity Facility pane opens for the preconfigured BAMSenderDriver, which lists the configuration parameters for the driver, as shown in the following image.



3. Modify values for the required parameters in the Connection Management section, which are listed and described in the following table.

Parameter	Description
Retry Count	Number of attempts to reconnect to the underlying iWay Business Activity Monitor database.
Retry Interval	Wait interval between reconnect attempts.
Lost Connection Behavior	Select one of the following values from the drop-down list: ❑ **Force.** Terminate any logging activity and abandon any entries in flight. ❑ **Persist.** Terminate any logging activity and serialize any entries in flight. ❑ **Offline.** Continue to run logger in offline mode, which will log the activity to a file store for later processing when the connection is recovered to the database.
Lost Connection Flow Name	Process flow to execute upon connectivity loss and failure to reestablish a connection. The process flow must terminate with End node that corresponds to the action, (for example, Force, Persist, or Offline) or an additional option (Continue) to continue denoting that the connection has been restored. The process flow takes priority over the preconfigured Lost Connection Behavior parameter.

4. Click *Update* when you have finished modifying the parameters for the preconfigured BAMSenderDriver.

Understanding Recovery Mode

When an updater thread shifts to recovery mode, it returns the message that it was processing to the database to the work queue. It then sets itself to write all messages to the local database rather then to the RDBMS. It then resumes accepting messages.

Depending on the settings and the return from the process flow, the driver starts an analysis thread. This thread periodically pools the RDBMS, awaiting its availability. When the RDBMS becomes available, the recovery thread sets the updater threads to resume standard mode. In this mode, the updaters accept messages from the queue and send them to the RDBMS. The recovery thread then begins reading, in order, the persisted messages written to the local disk. Each message is added to the work queue, so that an updater thread will send it to the RDBMS. The persisted messages are deleted as they are added to the queue.

If the system is terminated, on the next startup, the persisted messages will be loaded from disk to the work queue, so that normal operations can continue.

Using iWay Business Activity Monitor Commands

iWay Business Activity Monitor commands provide you with assistance when performing iWay Business Activity Monitor operations. The commands also support subcommands to specify specific actions, as listed and described in the following table.

Action	Description
archive	Given a specific date and time, this action copies all records for transactions that were completed before the given time to a specified set of tables. When successful, these transactions are removed from the active database.
copy	Given a time period or a transaction ID, this action copies all completed transactions identified to a named set of tables.
refresh	Clears the internal support tables.
list	Lists the contents of the database.
info	Displays database configuration information.

iWay Business Activity Monitor commands operate on iWay Business Activity Monitor databases. However, once the generated tables are populated, normal database operations can be used to copy, delete, or manipulate the tables as required. iWay Business Activity Monitor commands are not required in these instances.

iWay Business Activity Monitor commands have one or more operands in common, which are listed and described in the following table.

Operand	Description	Example
<tid>	Single transaction ID, enclosed in quotation marks (").	"80f98d02-2c94-46fc-9a59-72541f05f548"
<pattern>	Distinguishing name of a set of iWay Business Activity Monitor tables. The command will create tables of this pattern if they do not exist in the iWay Business Activity Monitor database.	july You might use the _fmtdate() function to generate names, such as the name of a month.
<timespec>	Date and time using the YYYY/MM/DDtHH:MM format. The default time is 00:00 if not specified. Abbreviations are available, as described in this section.	2012/6/25t03:15 2012/4/15

iWay Functional Language (iFL) can be used to assist in generating a <timespec>. For example, to specify a date four days in the past, specify the _fmtdate() function as follows:

```
"_fmtdate('yyyy/MM/dd\\'t\\'hh:mm',_imul(_isub(_now(u),_imul(86400,4)),
1000))"
```

This function subtracts from the current moment the number of seconds in a day, times the number of days, times the milliseconds in a second.

The enclosing double quotation marks (") tell the command processor that this is a single operand. The inner escapes pass a single escape-surrounded pattern character as a constant. To work from midnight of seven days ago, the <timespec> operand might be:

```
"_fmtdate('yyyy/MM/dd,_imul(_isub(_now(u),_imul(86400,7)),1000))"
```

Use of iFL to generate a <timespec> is useful for scheduled operations. For example, you might set up a schedule to run the command on a weekly basis, archiving information from the prior week.

As a convenience, abbreviations have been created for common specifications. Abbreviations must be spelled out in full.

Abbreviation	Purpose
1week	One week ago, midnight
1month	One month ago, midnight

The <pattern> is also evaluated for iFL. This helps create appropriately named archive databases in a recurring schedule. Users are cautioned that the pattern is used to name the generated databases, and must meet any naming restrictions imposed by the database.

iWay Business Activity Monitor Archive Command

The iWay Business Activity Monitor Archive command archives the identified transactions to a named set of tables. This reduces the size of the active tables and improves performance. The command does not copy partial transactions. That is, if a transaction was in execution prior to the -before specification but continued to execute after that time, it will not be archived. Only completed transactions are archived.

Usage:

```
bam archive <pattern> -before <timespec> [-timeout <seconds>]
```

The specified information is archived to the tables identified by the pattern.

Note: As only one date applies to the archive command, either -before or -after can be used.

Example:

```
Enter command:>bam archive july -before 2013/08/01
Archived 3267
```

All completed transactions will be archived until August 1, 2012.

iWay Business Activity Monitor Copy Command

The iWay Business Activity Monitor Copy command duplicates the information but does not purge it from the source data tables. This is useful if you need to save specific information for other purposes. The command does not copy partial transactions. That is, if a transaction was in execution prior to the -before specification but continued to execute after that time, it will not be archived. Only completed transactions are archived.

Usage:

```
bam copy <pattern> [-tid <tid> | -after <timespec> [-before <timespec>]] [-
timeout <seconds>]
```

The transaction information for the specified transaction ID, or the transactions within the time period, are copied. The default -before <timespec> is the current time. The iWay Business Activity Monitor Copy command will swap the before and after <timespec> if necessary to determine the appropriate time period.

Example:

```
Enter command:>Bam copy saved -tid 06c811fa-ba93-4e9b-9975-97b1cb3b84c8
Saved 1
```

iWay Business Activity Monitor Info Command

The iWay Business Activity Monitor Info command provides information about the specified database that is displayed. You can use the * pattern for the current database. This command is provided for troubleshooting purposes, if required. The contents and format of the returned information is not guaranteed to be consistent between releases.

Usage:

```
bam info [<pattern>]
```

Example:

```
Enter command:>bam info *
DataSource    |UniqueTIDs |TotalTransRecs|StartTime              |
EndTime                |Duration
BAM_ACTIVITY  |38         |563           |2012-01-18 15:43:26.393|
2012-01-27 15:43:29.253|9d 00:00:02.860
```

iWay Business Activity Monitor List Command

The iWay Business Activity Monitor List command returns information contained in the specified database. You can use the * pattern for the current database if you want to use filtering parameters. This command is provided for troubleshooting purposes, if required. The contents and format of the returned information is not guaranteed to be consistent between releases.

Usage:

```
bam info [<pattern> | *] [-tid <tid> | -after <timespec> [-before
<timespec>]] [-timeout <seconds>]
```

Example:

```
Enter command:>bam list * -after 12/1/1 -prov sql2k8
DataSource     |TID                                   |
StartTime              |EndTime                |Duration                  |
TranSteps
BAM_ACTIVITY   |06c811fa-ba93-4e9b-9975-97b1cb3b84c8 |2012-01-20
15:43:06.19 |2012-01-27 15:43:06.453|7d 00:00:00.263           |14
BAM_ACTIVITY   |072cbbea-9791-4a17-a775-75a73de907f8 |2012-01-23
15:43:07.627|2012-01-27 15:43:07.69 |4d 00:00:00.630           |14
BAM_ACTIVITY   |153c8188-b5f1-4860-9698-3f72881ae6c5 |2012-01-26
15:43:07.47 |2012-01-27 15:43:07.533|1d 00:00:00.630           |14
```

Note: This example shows three BAM_ACTIVITY lines, each of which is its own line in the actual system.

iWay Business Activity Monitor Refresh Command

The iWay Business Activity Monitor Refresh command clears internal reporting tables but does not modify the database itself. Therefore, no pattern or other operands are required and will be ignored. This command also clears stale computations and may improve performance.

Chapter 4

Structure of the iWay Business Activity Monitor Database

This section describes the functionality and structure of the following tables that are used by the iWay Business Activity Monitor database.

In this chapter:

BAM_ACTIVITY Table

The BAM_ACTIVITY table contains one record for each entry that is recorded by the iWay Business Activity Monitor activity log driver.

Field	Datatype	Description
RecordKey	char(32)	Primary key for this table and foreign key for RecordKey in BAM_ACTIVITY_EXTENDED and BAM_ACTIVITY_CONTEXT tables. This value is a GUID, with no intrinsic meaning except to uniquely identify an activity record.
Tstamp	Datetime	Time at which this activity record was logged, with precision down to the millisecond if the system supports it.

Field	Datatype	Description
RecordType	Int	Type of activity record, using the following codes: ❑ **100 (Enqueue).** Receipt of message on an iWay internal queue. ❑ **101 (Start).** Start of processing for a message. ❑ **191 (End).** End of processing for a message. ❑ **181 (Emit).** Message is emitted from a channel. This may indicate a message leaving the iWay application or a message being transferred between iSM channels. ❑ **131 (Internal Event).** Indicates the beginning or end of a certain event while processing the message. The RecordType for these entries is actually 131 plus an action code, the meaning of which can vary depending on the event type, but is typically 0, indicating the beginning of a certain processing stage. 1 indicates the successful completion of a stage. 2 indicates a failure of that state. For more information about other action codes, see the subtype field. ❑ **151 (Message).** Indicates a BIZERROR message generated while processing the message.

Field	Datatype	Description
TID	Char(48)	Transaction ID that identifies all activity for this message while being processed by the iWay application. The transaction ID remains constant when the message is transferred between iSM channels. **Note:** An application can generate multiple messages from a single origin. All of these would share the same TID.
Recordkey101	Char(32)	RecordKey of the start or enqueue entry that recorded the beginning of processing of this message on this channel. For example, it can be used to correlate start and end entries.
Correlid	Char(32)	Not used.
Version	Char(16)	iSM version.
Status	Int	Final status for processing of this message on this channel for end entry records (recordtype = 191). A value of 1 indicates success. Any value greater than 1 indicates failure.

Field	Datatype	Description
Subtype	Int	Specific type of event for an event record. This can include, but is not limited to the following: ❑ iSM life cycle events ❑ Inbound decryption ❑ Preparse ❑ XML Parse ❑ Inbound review ❑ Inbound validation ❑ Inbound Transform ❑ Agent or Service ❑ Outbound Transform ❑ Outbound validation ❑ Outbound review ❑ Preemit ❑ Outbound encryption ❑ Emit

Field	Datatype	Description
Subtype (continued)	Int (continued)	❑ Emit Events ❑ **301.** Records data that is being emitted from iSM. ❑ **302.** Records the response from the external system when data is emitted from iSM using a synchronous protocol. ❑ Security Events ❑ **601.** Events relating to a certain general security operation. ❑ **602.** Events relating to digital signatures. This type of event can use the following action codes: 3 (Sign), 6 (Verify), and 2 (Fail). ❑ **603.** Events relating to encryption and decryption. This type of event can use the following action codes: 4 (Encrypt), 5 (Decrypt), and 2 (Fail).
Subtype (continued)	Int (continued)	❑ User Events. ❑ **1000.** Users can generate event entries in the activity log. User-created event codes must be greater than 1000.

Field	Datatype	Description
Sequence	Int	In an iWay application in which a message is processed by multiple channels, the sequence indicates the order in which these channels received the message. This value is only stored with the start entry record (RecordType = 101). **Note:** The sequence represents a logical, rather than chronological, ordering of channels. For example, if the channel with sequence 1 emits two messages to an internal queue for further processing, the start entry records on the internal channel for both of these messages will have a sequence of 2.
ResubmitCount	Int	Number of times this message has been resubmitted.
ResubmitSourceKey	Char(32)	Record key of the activity record that indicates the point in processing from which this message was extracted for resubmission.
Encoding	Varchar(16)	Character encoding of this message.
Signature	Varchar(64)	Name of the channel where this activity occurred.
MAC	Varchar(80)	Not used.
ConfigName	Varchar(128)	iSM managed server or iWay Integration Application (iIA) on which this activity occurred.
HostName	Varchar(128)	Host name of the system where this activity occurred.
Address	Varchar(128)	Not used.
Partner_to	Varchar(128)	Value of the partner_to special register if set to true, by certain EDI components.
Partner_from	Varchar(128)	Value of the partner_from special register if set to true, by certain EDI components.

Field	Datatype	Description
Protocol	Varchar(150)	Protocol specific information. It is often the protocol name of the channel.

BAM_ACTIVITY_CONTEXT Table

The BAM_ACTIVITY_CONTEXT table can be used as a continuation of the BAM_ACTIVITY table, containing one record for each entry that is logged by the iWay Business Activity Monitor activity driver. The following table contains the message itself and its context, stored in large object type fields.

Field	Datatype	Description
RecordKey	Char(32)	Primary key for this table and foreign key for RecordKey in BAM_ACTIVITY_EXTENDED and BAM_ACTIVITY tables. This value is a GUID, solely used to uniquely identify an activity record.
Tstamp	Datetime	Time at which this activity record was logged, with precision down to the millisecond if the system supports it. This is identical to the tstamp value in the corresponding BAM_ACTIVITY record.
Msgkey	Char(32)	Not used.
Msgmac	Char(32)	Not used.
Msgtype	Varchar(32)	Not used.
Msg	BLOB	Message as it was at this point in processing.

Field	Datatype	Description
Context	BLOB	All special registers associated with the message at this point of processing. Each register is stored as type/name=value and registers are delimited by the 0x07 character. Note that registers important to the application can be stored in the BAM_ACTIVITY_EXTENDED table.

BAM_ACTIVITY_EXTENDED Table

The BAM_ACTIVITY_EXTENDED table contains context values that are important to the application. Several fields are automatically included, but the table can be extended with additional fields mentioned previously. The following table contains one record for each entry that is recorded by the iWay Business Activity Monitor activity log driver and can be used as a continuation of the BAM_ACTIVITY table.

Field	Datatype	Description
RecordKey	Char(32)	Primary key for this table and foreign key for RecordKey in BAM_ACTIVITY_CONTEXT and BAM_ACTIVITY tables. This value is a GUID, solely used to uniquely identify an activity record.
Tstamp	Datetime	Time at which this activity record was logged, with precision down to the millisecond if the system supports it. This is identical to the tstamp value in the corresponding BAM_ACTIVITY record.
BAM_tpm_partnername	Varchar(128)	Value of the bam_tpm_partnername special register, when available.
BAM_edi_ackstatus	Varchar(5)	Value of the edi_ackstatus register, when available.

Field	Datatype	Description
BAM_edi_transactionID	Varchar(128)	Value of the edi_transactionID register, when available.
BAM_edi_version	Varchar(128)	Value of the edi_version register, when available.
BAM_edi_type	Varchar(128)	Value of the edi_type register, when available.
Msgsize	Varchar(128)	Size of the message in bytes at this point in processing, when applicable.
Protocol	Varchar(128)	Protocol specific information. This is often the name of the protocol of the channel.
Extension	Varchar(128)	Value of the extension register, when available.
BAM_tpm_partnerid	int	Value of the bam_tpm_partnerid special register, when available.
BAM_tpm_countryname	Varchar(2)	Value of the BAM_TPM_COUNTRYNAME register, when available.
Text	Varchar(512)	Text of the BIZERROR message, for message entry records (recordtype = 151),

Chapter 5

Configuring User Authentication for iWay Business Activity Monitor

This section describes how to configure authentication realms for iWay Business Activity Monitor.

In this chapter:

User Authentication Overview

iWay Business Activity Monitor is hosted on the non-blocking HTTP (nHTTP) listener. As a result, iWay Business Activity Monitor can take full advantage of authentication realms configured through iSM. Authentication realms enable remote user authentication against an external system such as Active Directory (AD) and Lightweight Directory Access Protocol (LDAP).

This section describes how to configure user authentication for iWay Business Activity Monitor using LDAP. The following steps are required in this configuration process:

1. Creating an LDAP Directory Provider to establish a link to the LDAP system.

   For more information, see *Creating an LDAP Directory Provider* on page 94.

2. Creating an LDAP authentication realm to enable authentication against LDAP.

   For more information, see *Creating an LDAP Authentication Realm* on page 97.

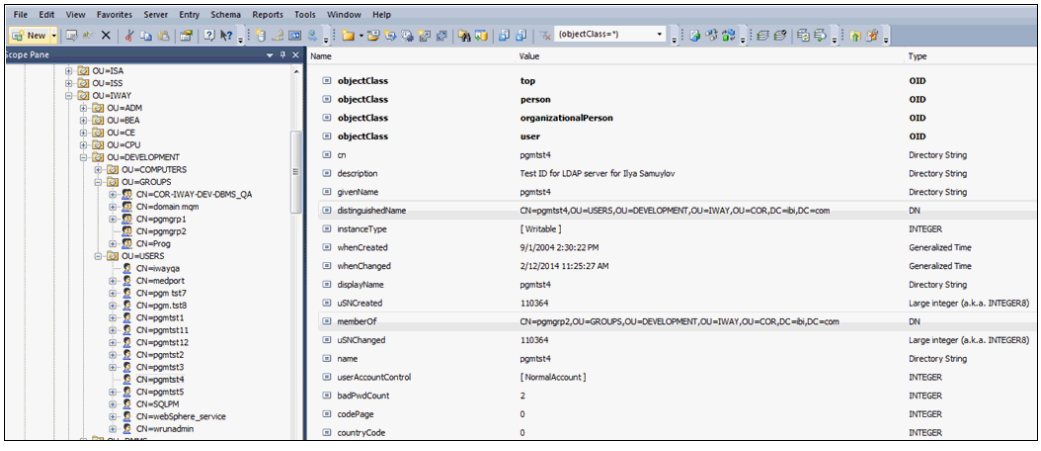
3. Mapping LDAP roles to iWay Business Activity Monitor roles.

   For more information, see *Mapping LDAP Roles to iWay Business Activity Monitor Roles* on page 101.

4. Enabling iWay Business Activity Monitor authentication through the authentication realm.

   For more information, see *Enabling iWay Business Activity Monitor Authentication Through an Authentication Realm* on page 102.

The following image shows a sample user type defined in an LDAP system.



Creating an LDAP Directory Provider

This section describes how to create an LDAP Directory Provider using the iSM Administration Console. An LDAP Directory Provider enables you to establish a link to an existing LDAP system.

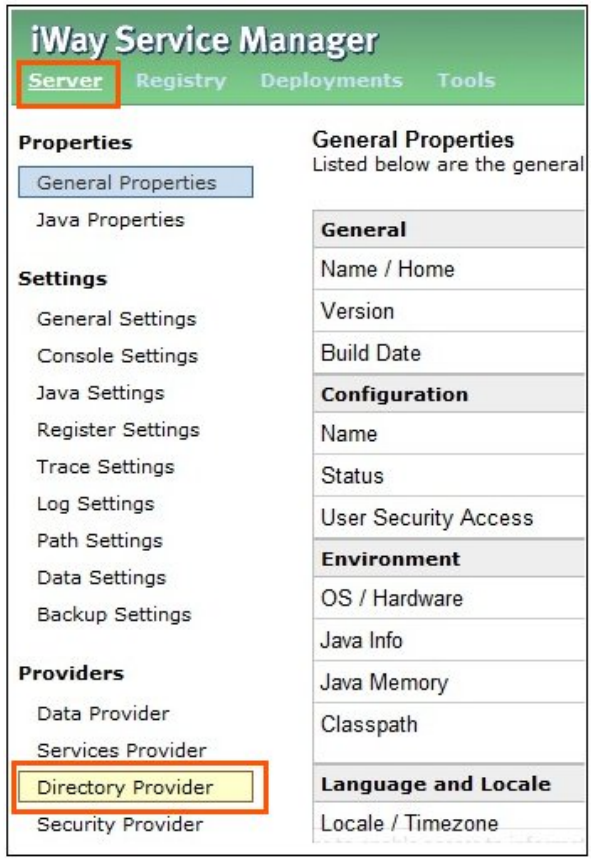
Procedure: How to Create an LDAP Directory Provider

1. Connect to the iSM Administration Console using the following URL:

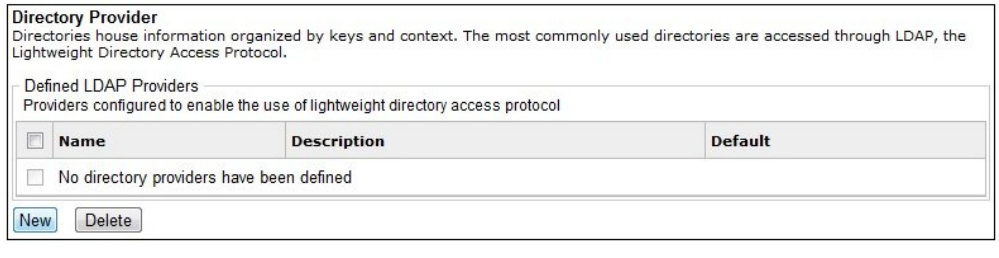
   `http://localhost:9999`

2. Log on to the console using valid iSM credentials (user name and password).

3. If it is not already selected, ensure *Server* is selected in the top pane of the console and then click *Directory Provider* in the Providers section on the left pane, as shown in the following image.

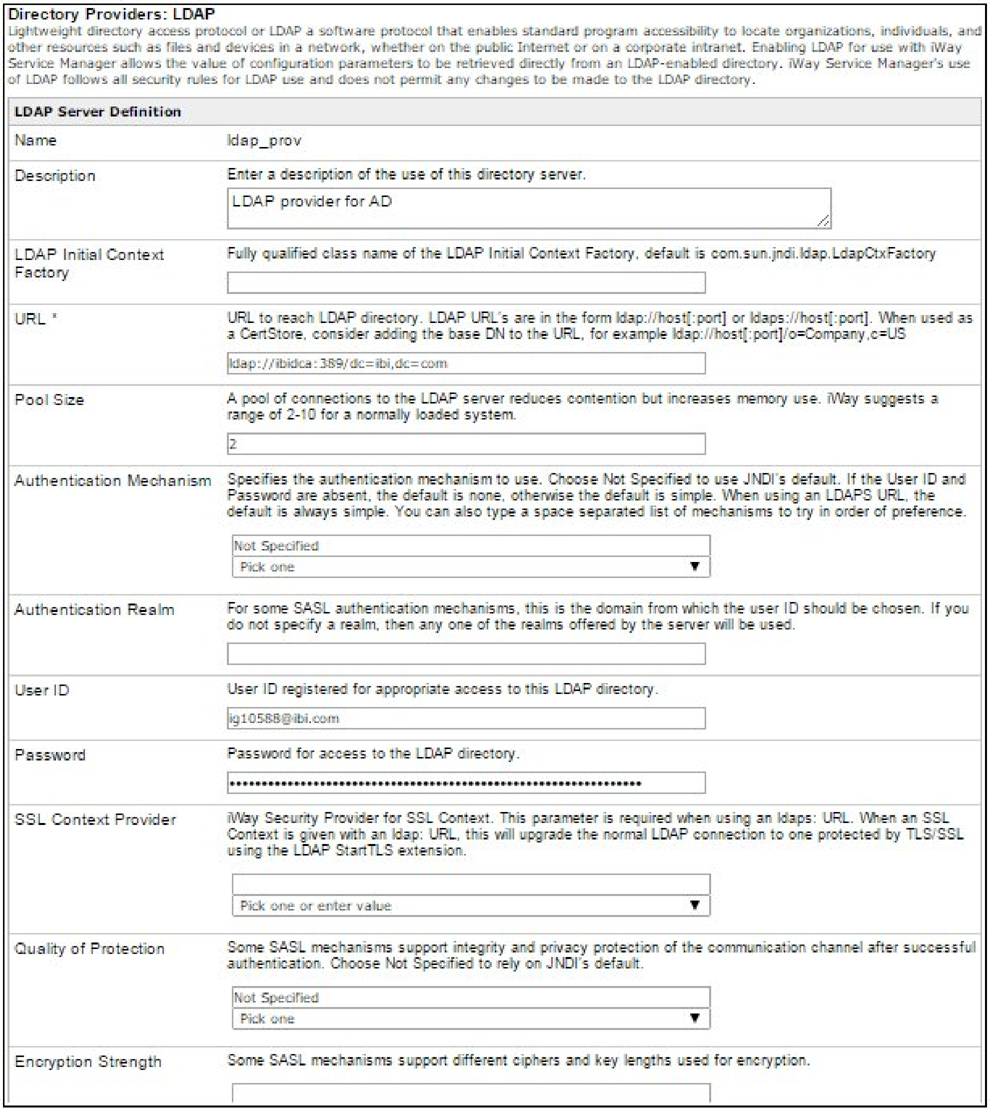


The Directory Provider pane opens, as shown in the following image.



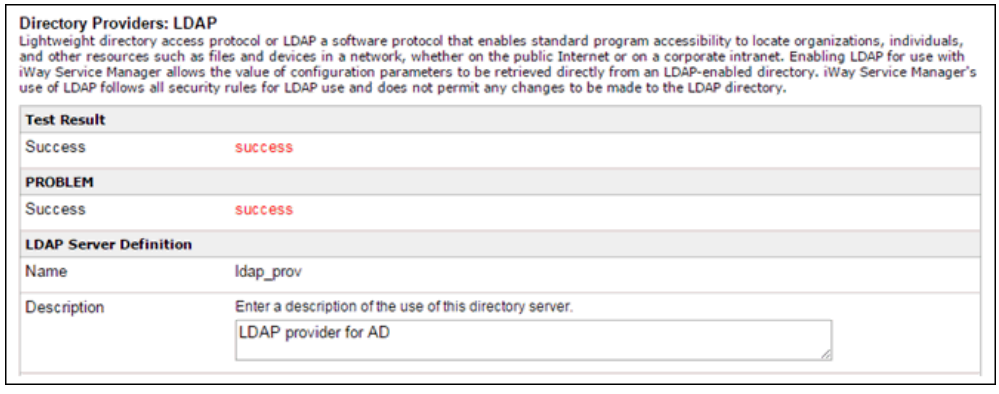
4. Click *New* to create a new Directory Provider for LDAP.

The Directory Providers: LDAP pane opens, as shown in the following image.



5. Provide values for the required parameters to configure a connection to your LDAP system.
6. Scroll down to the bottom of the page and click *Add* to add the LDAP Directory Provider.

Note: You can also click *Test* to verify that you successfully connected to your LDAP system, as shown in the following image.



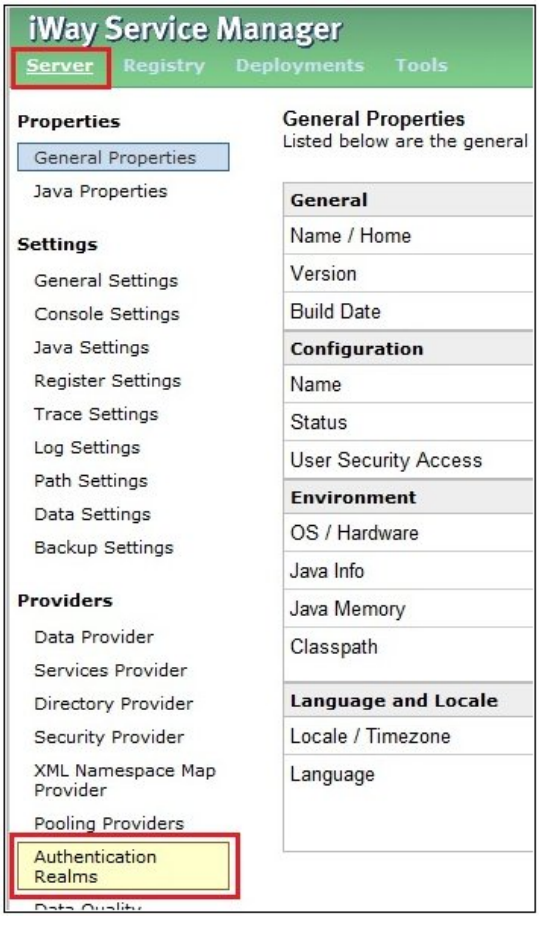
Creating an LDAP Authentication Realm

This section describes how to create an LDAP authentication realm using the iSM Administration Console. An LDAP authentication realm enables authentication against LDAP.

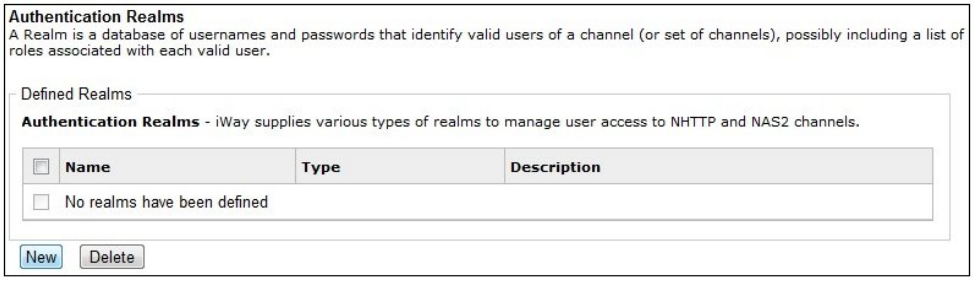
Procedure: How to Create an LDAP Authentication Realm

1. Ensure that you are logged in to the iSM Administration Console.

2. If it is not already selected, ensure *Server* is selected in the top pane of the console and then click *Authentication Realms* in the Providers section on the left pane, as shown in the following image.

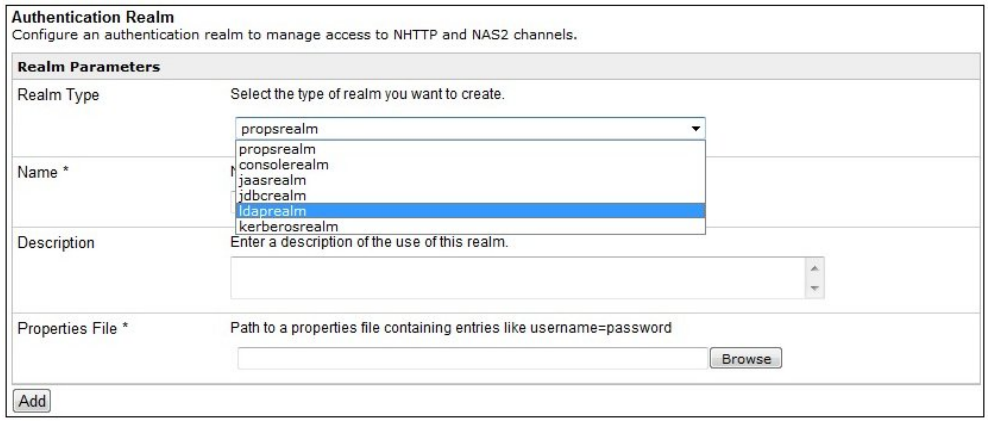


The Authentication Realms pane opens, as shown in the following image.



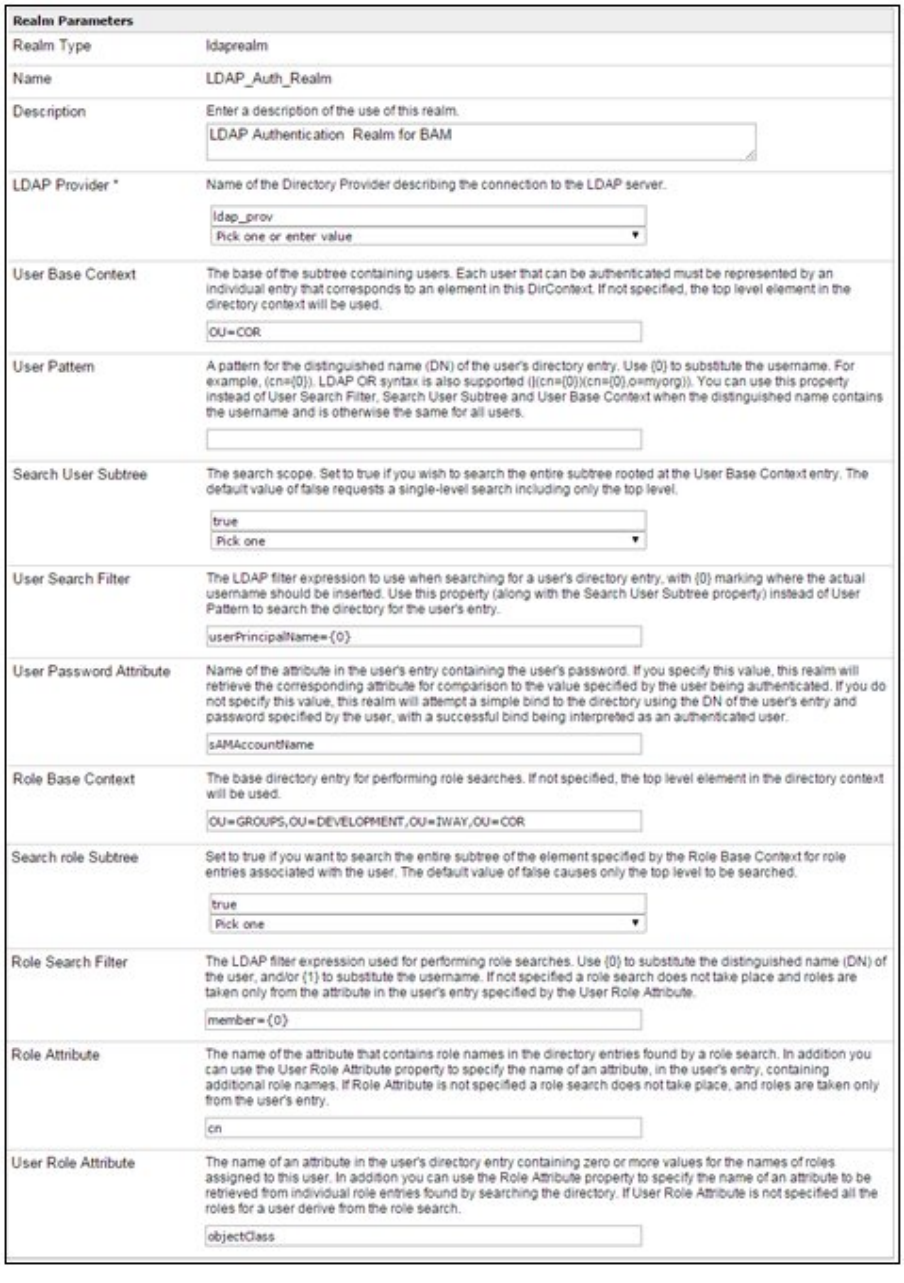
3. Click *New* to create a new LDAP authentication realm.

   The Authentication Realm pane opens, as shown in the following image.



4. From the Realm Type drop-down list, select *ldaprealm*.

The Authentication Realm pane refreshes with your selection and provides configuration parameters that are specific to an LDAP authentication realm, as shown in the following image.



Realm Parameters	
Realm Type	ldaprealm
Name	LDAP_Auth_Realm
Description	Enter a description of the use of this realm. LDAP Authentication Realm for BAM
LDAP Provider *	Name of the Directory Provider describing the connection to the LDAP server. ldap_prov Pick one or enter value
User Base Context	The base of the subtree containing users. Each user that can be authenticated must be represented by an individual entry that corresponds to an element in this DirContext. If not specified, the top level element in the directory context will be used. OU=COR
User Pattern	A pattern for the distinguished name (DN) of the user's directory entry. Use {0} to substitute the username. For example, (cn={0}). LDAP OR syntax is also supported (|(cn={0})(cn={0},o=myorg)). You can use this property instead of User Search Filter, Search User Subtree and User Base Context when the distinguished name contains the username and is otherwise the same for all users.
Search User Subtree	The search scope. Set to true if you wish to search the entire subtree rooted at the User Base Context entry. The default value of false requests a single-level search including only the top level. true Pick one
User Search Filter	The LDAP filter expression to use when searching for a user's directory entry, with {0} marking where the actual username should be inserted. Use this property (along with the Search User Subtree property) instead of User Pattern to search the directory for the user's entry. userPrincipalName={0}
User Password Attribute	Name of the attribute in the user's entry containing the user's password. If you specify this value, this realm will retrieve the corresponding attribute for comparison to the value specified by the user being authenticated. If you do not specify this value, this realm will attempt a simple bind to the directory using the DN of the user's entry and password specified by the user, with a successful bind being interpreted as an authenticated user. sAMAccountName
Role Base Context	The base directory entry for performing role searches. If not specified, the top level element in the directory context will be used. OU=GROUPS,OU=DEVELOPMENT,OU=IWAY,OU=COR
Search role Subtree	Set to true if you want to search the entire subtree of the element specified by the Role Base Context for role entries associated with the user. The default value of false causes only the top level to be searched. true Pick one
Role Search Filter	The LDAP filter expression used for performing role searches. Use {0} to substitute the distinguished name (DN) of the user, and/or {1} to substitute the username. If not specified a role search does not take place and roles are taken only from the attribute in the user's entry specified by the User Role Attribute. member={0}
Role Attribute	The name of the attribute that contains role names in the directory entries found by a role search. In addition you can use the User Role Attribute property to specify the name of an attribute, in the user's entry, containing additional role names. If Role Attribute is not specified a role search does not take place, and roles are taken only from the user's entry. cn
User Role Attribute	The name of an attribute in the user's directory entry containing zero or more values for the names of roles assigned to this user. In addition you can use the Role Attribute property to specify the name of an attribute to be retrieved from individual role entries found by searching the directory. If User Role Attribute is not specified all the roles for a user derive from the role search. objectClass

5. Provide values for the required parameters to configure an LDAP authentication realm.
6. Scroll down to the bottom of the page and click *Add* to add the configured LDAP authentication realm.

   Note the name of the newly configured LDAP authentication realm (for example, LDAP_Auth_Realm), as this value will be used during the iWay Business Activity Monitor configuration.

Mapping LDAP Roles to iWay Business Activity Monitor Roles

This section describes how to map LDAP roles to iWay Business Activity Monitor roles using the iWay Business Activity Monitor console. As defined roles in an LDAP system are unlikely to match corresponding roles in iWay Business Activity Monitor, you must map the roles between the LDAP system and iWay Business Activity Monitor.

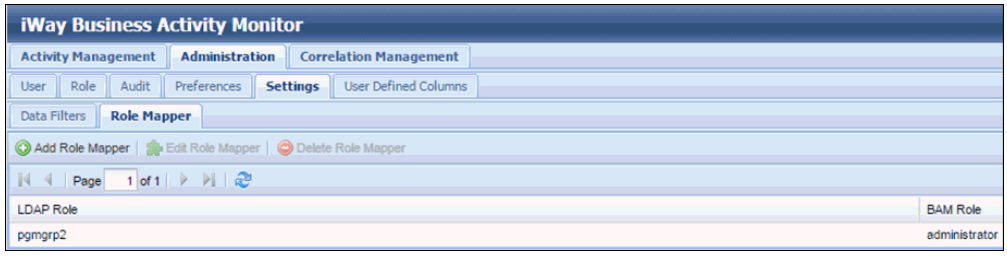
Procedure: How to Map LDAP Roles to iWay Business Activity Monitor Roles

1. Login to the iWay Business Activity Monitor console as an *administrator* while it is under standard authentication schema against the database.

   The default URL for the iWay Business Activity Monitor Console is:

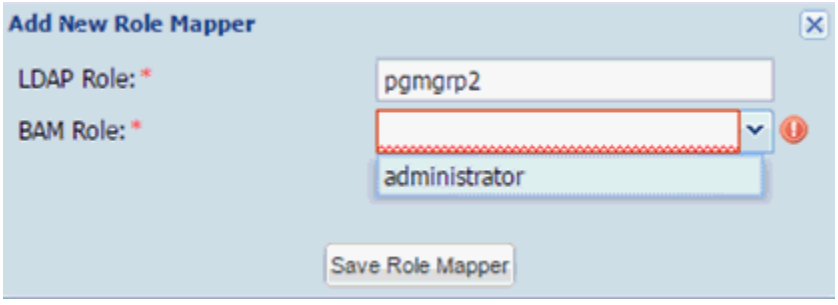
   `http://localhost:8087`

2. Click the *Administration* tab, select the *Settings* tab, and then click the *Role Mapper* tab, as shown in the following image.



3. Click *Add Role Mapper* to add a new mapping between an LDAP system role and an iWay Business Activity Monitor role.

The Add New Role Mapper dialog box opens, as shown in the following image.



4. Specify an LDAP role in the LDAP Role field, which would be returned for those users requiring access to the iWay Business Activity Monitor console, and map it to the iWay Business Activity Monitor role by selecting it from the *BAM Role* drop-down list.

   You can create multiple mappings between an LDAP and iWay Business Activity Monitor roles.

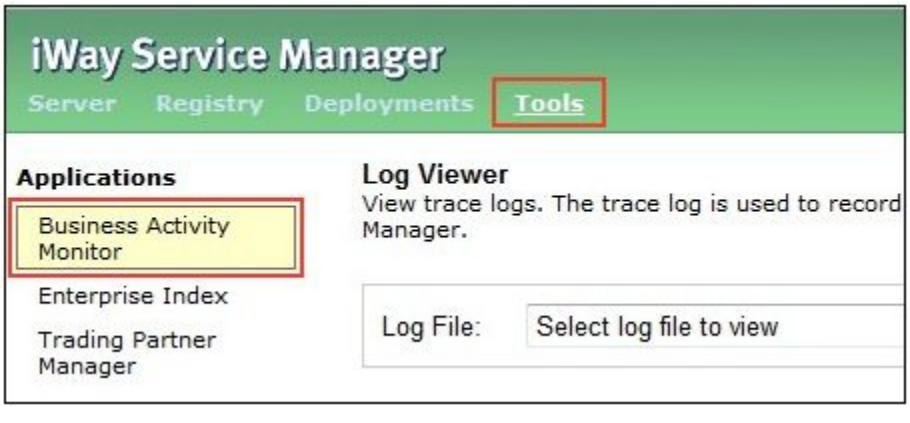
5. Click *Save Role Mapper*.

Enabling iWay Business Activity Monitor Authentication Through an Authentication Realm

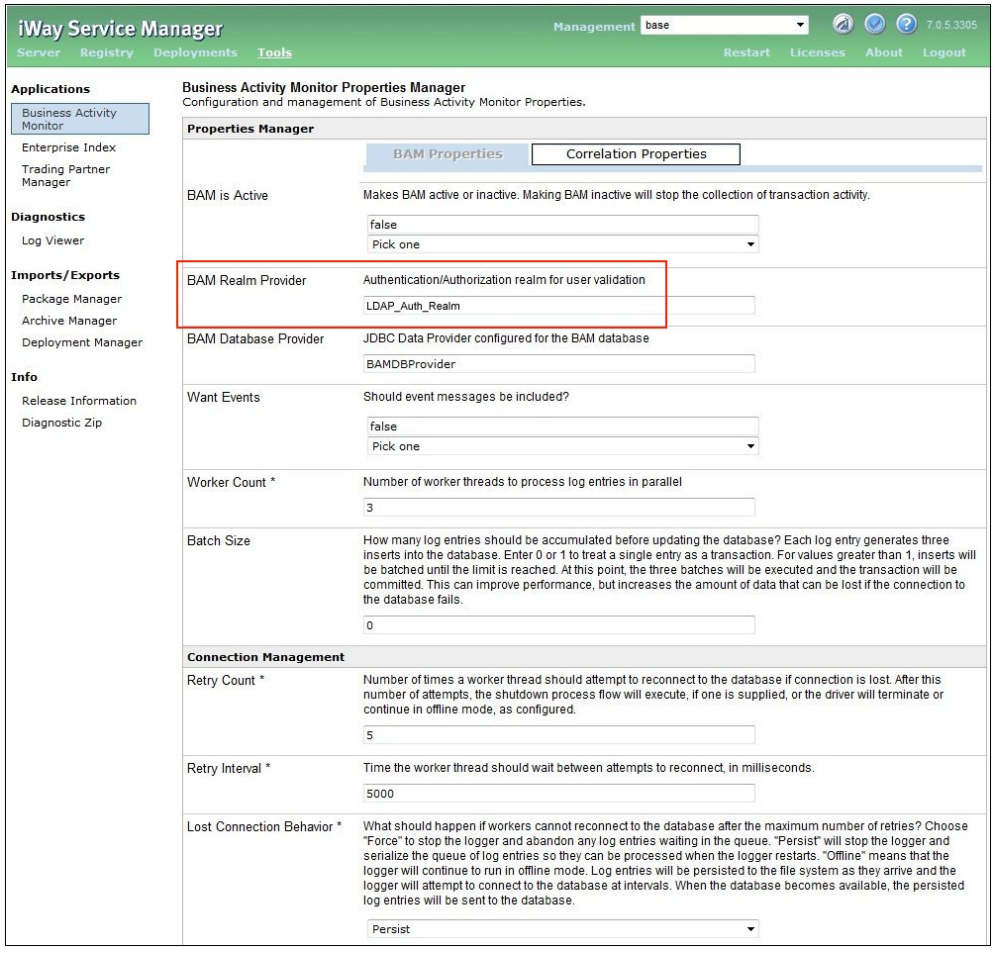
This section describes how to enable iWay Business Activity Monitor authentication through an authentication realm using the iSM Administration Console.

Procedure: How to Enable iWay Business Activity Monitor Authentication Through an Authentication Realm

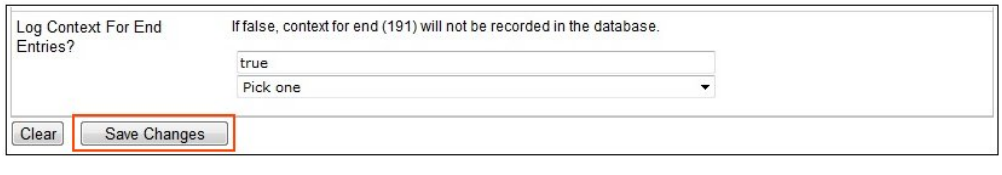
1. Ensure that you are logged in to the iSM Administration Console.
2. Click *Tools* in the top pane, and then under the Applications section in the left pane, select *Business Activity Monitor*, as shown in the following image.



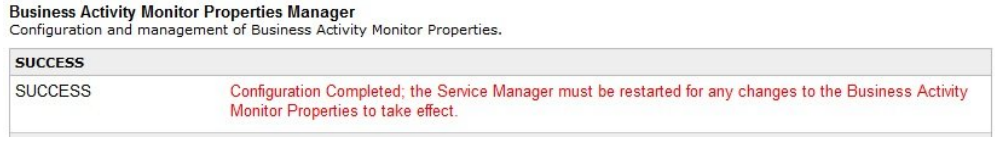
The Business Activity Monitor Properties Manager pane opens, as shown in the following image.



3. In the BAM Realm Provider field, type the name of the configured LDAP authentication realm (for example, LDAP_Auth_Realm) to be used for iWay Business Activity Monitor authentication.
4. Scroll down to the bottom of the page and click *Save Changes*.



A Configuration Completed message displays, as shown in the following image.



5. Restart iSM.
6. Open a new browser window and access the iWay Business Activity Monitor console.

   The default URL for the iWay Business Activity Monitor console is:

   `http://localhost:8087`

   When you are prompted for the user name and password, you can now use the credentials from the LDAP system.

   You will be authenticated through the configured LDAP authentication realm and its role will be mapped to the iWay Business Activity Monitor role.

   Note: The default user (admin/iway) and any other users in the local iWay Business Activity Monitor database repository of the user, will not have access to iWay Business Activity Monitor, unless authentication is switched back to the default setting, and *BAM Realm Provider* is set to an empty value under the Business Activity Monitor configuration section.

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

TIBCO, the TIBCO logo, the TIBCO O logo, FOCUS, iWay, Omni-Gen, Omni-HealthData, and WebFOCUS are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle Corporation and/or its affiliates.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

This software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the readme file for the availability of this software version on a specific operating system platform.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.