

# **HP OpenView Smart Plug-in for TIBCO**

## **User's Guide**

**Version: A.01.04**

**HP-UX and Solaris**



**March 2005**

© Copyright 2004-2005 Hewlett-Packard Development Company, L.P.

## Legal Notices

### Warranty

*Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.*

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

### Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company  
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

### Copyright Notices

© Copyright 2004-2005 Hewlett-Packard Development Company, L.P., all rights reserved.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

### Trademark Notices

Java™ is a U.S. trademark of Sun Microsystems, Inc.  
Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation  
UNIX® is a registered trademark of The Open Group  
Linux is a U.S. registered trademark of Linus Torvalds

## Support

Please visit the HP OpenView web site at: <http://www.managementsoftware.hp.com/>. There you will find contact information and details about the products, services, and support that HP OpenView offers.

You can go directly to the HP OpenView support Web site at: <http://support.openview.hp.com/>. As a valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter discussions with other software customers
- Research and register for software training

## Table of Contents

<b>1 Introduction .....</b>	<b>1-1</b>
Audience .....	1-2
Prerequisites .....	1-2
Quick Start Instructions .....	1-2
All Users .....	1-2
First Time Users.....	1-2
Repeat Users .....	1-3
Chapters Summary .....	1-3
Related Documents.....	1-3
Overview of OpenView-TIBCO Business Management .....	1-4
Standard Management Functionality .....	1-4
Advanced Management Functionality (Optional) .....	1-5
TIBCO SPI Conceptual Overview .....	1-5
TIBCO SPI Backend Service .....	1-6
TIBCO SPI Frontend Subagent .....	1-6
Resource Explorer .....	1-6
TIBCO SPI Reports.....	1-7
OpenView Service Effect Analysis (SEA).....	1-7
TIBCO Enterprise Management Advisor .....	1-7
Deployment Scenarios .....	1-9
Consolidated Scenario .....	1-9
Scenario 1: Collocated Components Scenario .....	1-10
Scenario 2: Remote Frontend Subagent.....	1-10
Scenario 3: Fully Distributed (Recommended) .....	1-10
WSDM Overview .....	1-11

XML interfaces .....	1-12
<b>2 Installation and Configuration .....</b>	<b>2-1</b>
Installation Process Summary .....	2-1
Requirements .....	2-2
Third Party License Review .....	2-2
Software Requirements for Standard TIBCO SPI Installation .....	2-2
Hardware Requirements .....	2-3
Patches .....	2-3
Kernel Parameters .....	2-3
Installing and Configuring the TIBCO SPI Components .....	2-4
Install the TIBCO SPI Depot on HP-UX .....	2-4
Install the TIBCO SPI Depot on Solaris .....	2-5
Configure the Management Server .....	2-5
Configuring the opc_adm operator .....	2-5
Register the Backend Service .....	2-5
Install the TIBCO SPI Frontend Subagent Component .....	2-6
Add a Managed Node to the TIBSPI-UNIX Node Group .....	2-6
Install the Monitors, Commands, and Actions on a Managed Node .....	2-7
Install the Frontend Subagent .....	2-7
Verifying the Frontend Subagent Install .....	2-8
Configure the Backend and the Frontend .....	2-8
Assign TIBCO SPI Templates .....	2-9
Configure Data Sources for Metric Data Collection .....	2-10
Assign Backend Service Templates .....	2-10
Deploy Backend Service Templates .....	2-10
Assign Frontend Subagent Templates .....	2-11
Deploy templates to the Frontend Subagent Node .....	2-11
Assign the TIBSPI-UNIX- V1 Template Group .....	2-12
Deploy the TIBSPI-UNIX- V1 Template Group to the UNIX Nodes .....	2-12
Assign the TIBSPI-Windows-V1 Template Group .....	2-13
Deploying the TIBSPI-Windows-V1 Template Group to the Windows Nodes .....	2-13
Verifying Deployed Templates .....	2-14
Start the TIBCO SPI .....	2-14

Restarting the TIBCO SPI .....	2-14
Stopping the TIBCO SPI .....	2-15
Checking the TIBCO SPI Status .....	2-15
Verifying a Start Status .....	2-16
Verify a Stop Status .....	2-17
Install the Java Console .....	2-17
Starting the Java Console .....	2-18
Install the HP Resource Explorer .....	2-18
Starting the HP Resource Explorer from the Java Console .....	2-19
Starting the HP Resource Explorer form the Command Line.....	2-19
Install TIBCO SPI Reports.....	2-20
Uninstalling the TIBCO SPI .....	2-20
Uninstalling the Frontend Subagent.....	2-20
Remove the TIBCO SPI Bits on HP-UX.....	2-21
Remove the TIBCO SPI Bits on Solaris .....	2-21
Remove TIBCO SPI Regroup Condition .....	2-22
Remove TIBCO SPI Message Groups.....	2-22
Remove the TIBCO SPI Application Group .....	2-22
Remove the TIBCO SPI User Profile .....	2-22
Remove the TIBCO SPI User .....	2-23
Remove TIBCO SPI Message Templates.....	2-23
Reinstall the Templates Monitor and Commands .....	2-23
Remove the TIBCO SPI Node Groups From the OVO Database .....	2-24

### **3 Performing Standard Management Functions ..... 3-1**

Service Management .....	3-1
Viewing TIBCO Managed Resources .....	3-2
Linking Other Service Maps.....	3-2
Automatically Linking .....	3-2
Manually Linking .....	3-2
Filtering Unwanted MO Types .....	3-3
Event Management.....	3-4
Viewing TIBCO Events .....	3-4
Automatically Responding to Events.....	3-4

Monitoring and Data Collection .....	3-5
Metric Definition Configuration Files .....	3-5
Modifying Data Collection .....	3-7
Collecting Data for Custom Adapters.....	3-7
Configuring Multi-Instance Metric Data .....	3-9
Example: All Instance Data (Collect) .....	3-10
Example: Specific Instance Data (Collect) .....	3-11
Configuring a Threshold for Multi-Instance Metric Data .....	3-11
Collection Data for Specific Metrics .....	3-12
Monitoring Custom Adapter Metric Thresholds with OVO.....	3-13
Changing the Metric Data Collection Interval .....	3-14
Changing the Metric Threshold Value .....	3-15
Changing which Logfile to Monitor .....	3-15
Reporting and Performance Graphs .....	3-16
Using CODA .....	3-16
CODA Logging.....	3-16
View TIBCO SPI Reports.....	3-17
Defining a User Friendly Name for an MO .....	3-17
Monitoring Performance Metrics with OVPM.....	3-18
Configure OVPM.....	3-18
Create a Bytes Sent OVPM Graph for RVDs .....	3-19
View the RVD Bytes Sent OVPM Graph.....	3-19
Graphing Instance Metric Data in OVPM .....	3-20
TIBCO SPI Self Management .....	3-20
Modify Logging and Tracing Levels.....	3-21
Changing Log Levels .....	3-21
Changing Frontend Log Levels.....	3-21
Changing Backend Log Levels .....	3-21
Changing Trace Levels .....	3-21
Changing Frontend Trace Levels.....	3-22
Changing Backend Trace Levels .....	3-22
<b>4 Using Service Effect Analysis (Optional) .....</b>	<b>4-1</b>
Overview .....	4-1
Service Composer Tool .....	4-2

Requirements.....	4-3
Prerequisites .....	4-3
Apache Tomcat Setup .....	4-4
Environment Setup .....	4-4
Running the SEA Installer .....	4-5
Running the SEA Component .....	4-6
Enabling MSI .....	4-6
Starting the SEA Component.....	4-6
Running the SEA Configuration Tool .....	4-7
Stopping the SEA Component .....	4-7
Creating an Event Definition.....	4-7
SEA Logging and Trace Information .....	4-10
Changing Logging Levels .....	4-11
Changing the MO Update Interval.....	4-12
Uninstalling the SEA Component.....	4-12
<b>5 Using the NNM Integration (Optional).....</b>	<b>5-1</b>
Overview .....	5-1
Service Views of Network Dependencies .....	5-1
Collecting SNMP Trap Events .....	5-2
Performance Reports.....	5-2
Requirements.....	5-4
General Setup Requirements .....	5-4
Enabling and Configuring the NNM Integration .....	5-5
NNM Topology Integration Settings .....	5-5
Changing the OVI JAVA Setting .....	5-6
Changing the OVI Port Setting.....	5-6
RVD Data Collection .....	5-6
<b>6 Implementing Failover.....</b>	<b>6-1</b>
Overview .....	6-1
Adding Multiple Location Candidates .....	6-2

<b>7 Security Features and Configuration.....</b>	<b>7-1</b>
Overview .....	7-1
Conceptual Architecture.....	7-1
What is Secured .....	7-2
What is not Secured .....	7-2
Current Limitations.....	7-3
Configuring HTTPS Communication .....	7-3
Configuring the Frontend Subagent.....	7-3
Configuring SSL/HTTPS.....	7-4
Setting up a Keystore and Truststore for the Frontend Subagent.....	7-4
Configuring the Resource Explorer.....	7-6
Using the Frontend Subagent's Client Certificate .....	7-6
Using a Separate Client Certificate for the Resource Explorer .....	7-7
Starting the Resource Explorer.....	7-7
Customizing HTTPS Configuration Parameters.....	7-8
<b>8 Troubleshooting.....</b>	<b>8-1</b>
Runtime Problems.....	8-1
Frontend Subagent Does Not Stop .....	8-1
Frontend Subagent Does Not Start.....	8-1
Frontend Subagent is Unable to Connect to EMA .....	8-1
Backend Service is Not Found.....	8-3
TIBCO SPI Uses Backup EMA Instead of Primary EMA.....	8-3
TIBCO Service is Not Visible .....	8-4
Missing Operational Notification.....	8-4
Verify TIBSPI-EventService-MSG-V1 Template is Configured .....	8-4
Verify OVO Messages Display in Message Browser .....	8-5
Verify TIBCO EMA Agent is Sending Notifications .....	8-5
Verify Frontend Subagent receives notifications.....	8-6
Verify Resource Host Name .....	8-6
Verify Source Object.....	8-7
Verify Credentials .....	8-7
Verify Communication with the OVO Management Server .....	8-8
Cleanup the OVO Message Queues .....	8-9
Performance Agent Does Not Start Up.....	8-10



TIBCO SPI Fails to Detect MeasureWare Agent.....	8-10
Configure TIBCO SPI Application Errors.....	8-10
Configure TIBCO SPI Application Does Not Start.....	8-10
Configure TIBCO SPI Application Does Not Display Content .....	8-10
Configure TIBCO SPI Application Fails to Transfer WCConfig.xml.....	8-11
Configure TIBCO SPI Application Throws AWTEException.....	8-11
Frontend Logfile Errors .....	8-12
Error Starting Notification HTTP Server .....	8-12
Error Adding Service to Map.....	8-12
Frontend does not Connect to the Backend.....	8-12
Problem Logging Metric Data .....	8-13
SEA Component Runtime Problems .....	8-13
Shutdown Problems.....	8-13
Viewing SEA WSMF Events .....	8-13
Check if OVO Events Work .....	8-13
Check the SEA Log File for Captured Events.....	8-13
Check the SEA Log for Event Subscription ERRORS/WARNINGS.....	8-14
Check the OVComposer GUI.....	8-14
ECS Correlation Problems.....	8-14
Clean the Fact Store .....	8-14
No Managed Objects are Deployed .....	8-15
Mgmt_sv Directory not Present.....	8-15
Event Subscriptions Persistence.....	8-17
Managed Object Discovery Problems.....	8-18
Service Engine Related Problems .....	8-18
Service Engine Error.....	8-18
<b>Appendix A: TIBCO SPI Configuration .....</b>	<b>A-1</b>
Editing WCConfig.xml .....	A-1
Using the Configuration Tool .....	A-1
WCConfig.xml Configuration Parameters.....	A-2
<FrontendSection> .....	A-2
<BackendSection>.....	A-5
<NetworkingInfo>.....	A-5

## **Appendix B: SEA Configuration.....B-1**

Editing AIA.cfg.....	B-1
Using the Configuration Tool .....	B-1
AIA.cfg Configuration Parameters .....	B-2
<ManagedObjectInformation> .....	B-2
<EventSubscription> .....	B-2
<Event_Reporting> .....	B-2
<OVSBA_Configuration> .....	B-2

## **Appendix C: List Templates and Reports.....C-1**

Message Groups.....	C-1
Applications.....	C-1
Templates .....	C-2
TIBCO SPI Self Management .....	C-6
Performance Metrics .....	C-7
Reports.....	C-10

## **Glossary**

## **Index**

# Introduction

The HP OpenView Smart Plug-In (SPI) For TIBCO User's Guide provides detailed information that is used to set up and configure the management systems that enable you to manage a TIBCO environment. The management solution is based on two different product packages: the HP OpenView SPI for TIBCO, and the TIBCO® Enterprise Management Advisor software, a product of TIBCO Software Inc. Together, the two products give IT and application managers the ability to distinguish between significant infrastructure events and events that impact business processes and applications, so that they can take more immediate, appropriate action to maintain uptime and keep mission-critical applications running efficiently.

HP's strategic focus is to enable an adaptive infrastructure for our customers. Our efforts with TIBCO extends OpenView's core competency of service management to include critical business process information and the ability to adaptively manage the enterprise. Together, HP and TIBCO are delivering on a multipart project that begins with a rich, robust SPI to manage the TIBCO environment and ends with the ability for our applications to provide intelligent, adaptive management of the TIBCO environment.

HP OpenView Operations for UNIX (OVO-U) provides a fully integrated management solution for networks, systems, databases, and applications found in heterogeneous distributed IT environments. This comprehensive product suite represents a complete set of tools enabling IT organizations to improve overall availability and reliability, maintain the highest degree of management flexibility, and establish management control over virtually all aspects of an enterprise environment. This guide contains instructions for installing, configuring, and using the OpenView Smart Plug-in for TIBCO. The instructions for using the OVO-U Management Console, including the JavaConsole and Service Navigator are only specific to the TIBCO SPI and not meant as a comprehensive guide for the OVO-U Management Console or Service Navigator. For documentation specific to these products, follow the link provided in the "Related Documents" section below.

TIBCO Software is a world-leading independent business integration software company that enables real-time business by helping companies become more cost-effective, more agile, and more efficient. TIBCO solutions empower customers to improve their business performance by enabling interoperability between diverse computer systems and helps them streamline activities that span their extended enterprise. The Enterprise Management Advisor software allows customers to actively manage TIBCO solutions.

## Audience

This User's Guide is intended for anyone who is responsible for operating and administering OVO. In particular, the guide is for OVO administrators, OVO operators, System Administrators, IT operators, and TIBCO application managers. It is expected that TIBCO application managers will work as OVO Operators and will use the JavaConsole or Service Navigator to manage the TIBCO environment.

## Prerequisites

Users of this Guide should have basic knowledge of the OVO Management Console, and OpenView management solutions, including SPIs. In addition, users should have basic knowledge of TIBCO application management and the TIBCO software environment. Familiarity with Java™, HP-UX, and SOLARIS are useful when completing some of the instructions. It is also recommended, but not required, that a user of this guide have general knowledge of management principals.

## Quick Start Instructions

This section includes general guidelines that can facilitate installing and using the TIBCO SPI.

### All Users

All users should:

- complete the steps of the “Installation and Configuration” chapter, which provides instructions for installing the TIBCO SPI with standard management features
- verify and validate that your installation is working
- add advanced management functionality as needed by following the instructions in Chapter 4 and Chapter 5

### First Time Users

First time users should read this chapter completely to gain an understanding of the:

- standard and advanced functionalities of the TIBCO management solution
- conceptual overview and the dependencies required for the various solution components
- recommended deployment scenarios to help you decide an appropriate deployment configuration for your environment

## Repeat Users

Repeat users should read the *Release Notes* to find out new information about the TIBCO SPI and any documented known issues.

## Chapters Summary

The documentation in this guide is organized in a layered approach from generic instructions about the TIBCO Smart Plug-In to detailed instructions about TIBCO management. Once you have completed and are familiar with the instructions in this guide, you can reference specific instructions as needed.

- **Chapter 1 – Introduction:** Contains overview and contextual information about the WSMO-TIBCO integration, including a brief introduction to WSDM.
- **Chapter 2 – Installation and Configuration:** Contains detailed steps needed to install the standard management features of the OpenView SPI For TIBCO.
- **Chapter 3 – Performing Standard Management Functions:** Contains information and tasks that are used to perform standard management functions, such as service management, event management, and performance reports.
- **Chapter 4 – Using Service Effect Analysis:** Contains instructions for installing and using service effect analysis.
- **Chapter 5 – Using the NNM Integration:** Contains instructions for configuring and using the NNM integration.
- **Chapter 6 – Implementing Failover:** Contains instructions for setting up failover between the TIBCO SPI and the TIBCO EMA software.
- **Chapter 7 – Security Features and Configuration:** Contains instructions for securing the communication channels between the TIBCO SPI and the TIBCO EMA software.
- **Chapter 8 – Troubleshooting:** Contains general procedures used to troubleshoot common errors encountered when using the TIBCO SPI.
- **Appendix A – TIBCO SPI Configuration:** Contains a reference of all configuration parameters used to configure the TIBCO SPI.
- **Appendix B – SEA Configuration:** Contains a reference of all configuration parameters used to configure the SEA component.
- **Appendix C – List Templates and Reports:** Contains a reference of the templates and the reports for the TIBCO SPI, SEA component, and NNM integration.

## Related Documents

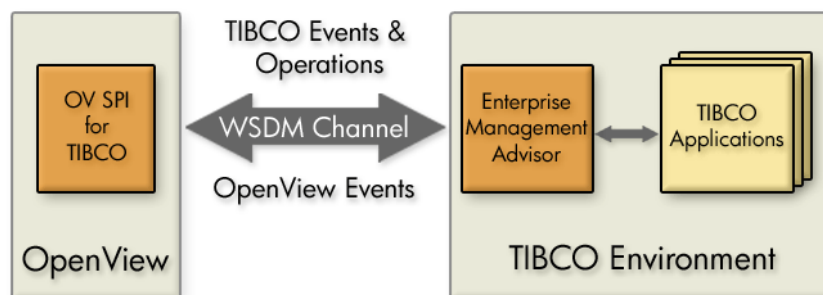
The TIBCO SPI contains integrations with several OV components. OV product guides can be downloaded from the [product manuals](#) page on the OpenView Web site.

The following TIBCO documents, in HTML and PDF formats, are available on the TIBCO Enterprise Management Advisor installation CD.

- TIBCO Enterprise Management Advisor User's Guide
- TIBCO Enterprise Management Advisor Release Notes

## Overview of OpenView-TIBCO Business Management

TIBCO provides a suite of enterprise integration products to bring together disparate and previously incompatible systems. Most of the TIBCO products and applications are management enabled using the TIBCO Hawk™ API. As part of the OpenView-TIBCO integration, TIBCO has created the Enterprise Management Advisor (EMA), which is responsible for discovering the TIBCO software resources and publishing management data for these resources. More importantly, the status of the resources is kept current and up to date. The TIBCO SPI bridges OVO and the TIBCO Enterprise Management Advisor software using a Web Services Distributed Management (WSDM) channel. As the result, the TIBCO SPI enables OpenView to manage TIBCO specific applications and products, as well as other elements in the customer's environment such as computing and network infrastructure and non-TIBCO applications. Figure 1-1 shows a high level view of the integrated system:



**Figure 1-1: OpenView-TIBCO Business Management General Architecture**

### Standard Management Functionality

Standard management functionality that is provided by the TIBCO SPI includes:

- Service Discovery Management
- Event and Notifications
- Monitoring and Thresholding
- Performance Reporting and Graphing

These features are available by installing and configuring the TIBCO SPI using the instructions in Chapter 2 "Installation and Configuration".

## Advanced Management Functionality (Optional)

The TIBCO Business Management Solution also includes several advanced features that can be used to gain additional management features. These features are optional and are not required to use the standard management features of the TIBCO SPI. Advanced management functionality includes:

- Network Node Manager (NNM) Integration
- Other SPI Integration
- Metric Data Collection
- Service Effect Analysis (see the “OpenView Service Effect Analysis” section)

## TIBCO SPI Conceptual Overview

The TIBCO SPI is comprised of five components. The components are distributed and are used to manage the TIBCO environment and its applications.

The components are:

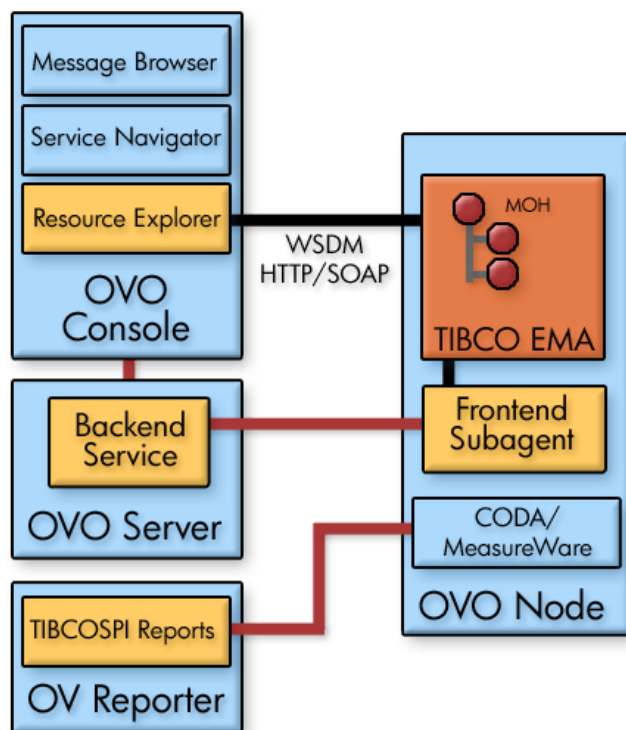
- TIBCO SPI Frontend Subagent
- TIBCO SPI Backend Service
- Resource Explorer
- TIBCO SPI Reports

The components

Figure 1-2 shows a conceptual overview of the TIBCO SPI components and how they interact. While the TIBCO EMA component is not part of the TIBCO SPI, it is included in the figure for clarification. For additional information on the EMA component, see the “TIBCO Enterprise Management Advisor” section below.



Figure 1-2 shows a typical deployment scenario that emphasizes the distributed nature of the TIBCO SPI solution. For additional deployment scenario options, see the “Deployment Scenarios” section below.



**Figure 1-2: Conceptual Overview of the TIBCO SPI Components**

## TIBCO SPI Backend Service

The backend software receives management data and information from the frontend subagent, converts the information to an OpenView recognized form and allows OpenView to manage a TIBCO Environment using traditional OpenView tools. The Backend Service must be located on an OVO-U Management Server.



The Frontend and Backend can be collocated on the same systems. If the Frontend and Backend are on the same system, they must run on the OVO Management Server system. This may be a likely scenario during testing.

## TIBCO SPI Frontend Subagent

The frontend software is responsible for communicating with the TIBCO EMA component to gather management data and information. The Frontend Subagent is essentially a WSDM client able to communicate with MOs exposed as Web services by TIBCO EMA.

## Resource Explorer

The Resource Explorer is a UI tool that is used to view and interact with MOs. The tool is only available for Windows. The tool can be either started from the OVO Java Console (if the Console is installed on Windows) or as a standalone application from the Windows command line.



## TIBCO SPI Reports

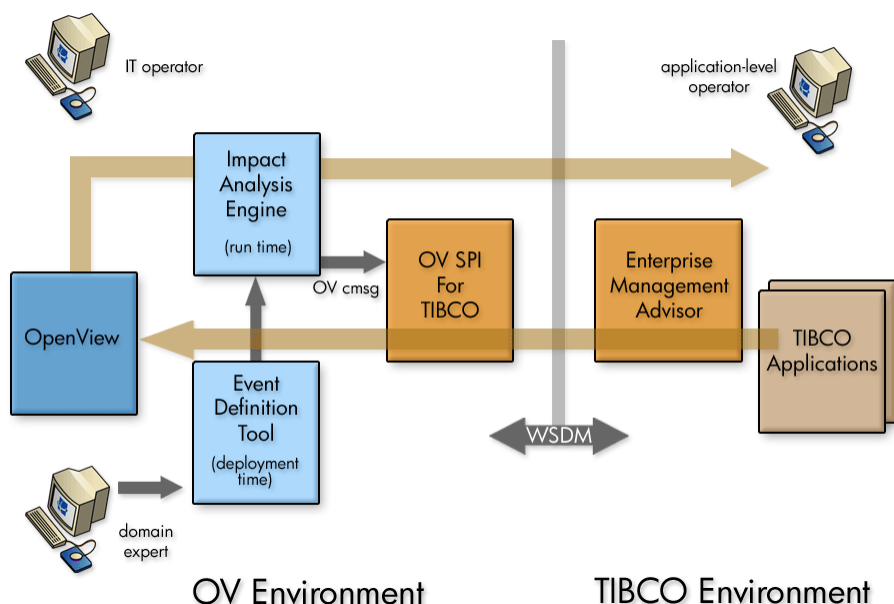
The TIBCO SPI provides various customized reports that provide performance data of the TIBCO Environment. The reports are implemented using the HP OpenView Reporter product, which can interface with both MeasureWare and CODA.

## OpenView Service Effect Analysis (SEA)

SEA is an optional plug-in to OpenView that enables adaptive management. It analyzes and correlates both OpenView native events such as those from computing and network infrastructure and application events such as those from TIBCO managed objects. The SEA module allows an application to receive events by either subscribing to any existing event types or by defining new event types and then subscribing to them. Application developers use the event definition tool (ECS) and SEA methodology to create new event types.

In SEA, a system or application administrator can use ECS to create event types of interest. After the event types are defined and made known to OpenView, the applications can subscribe to the events of certain types, and make corrective actions based on events received asynchronously.

Figure 1-3 shows a high-level conceptual view of the SEA system:



**Figure 1-3: Conceptual View of SEA**

## TIBCO Enterprise Management Advisor

TIBCO EMA software is the instrumentation piece that resides in the TIBCO environment. EMA is the gateway through which the TIBCO environment is managed.



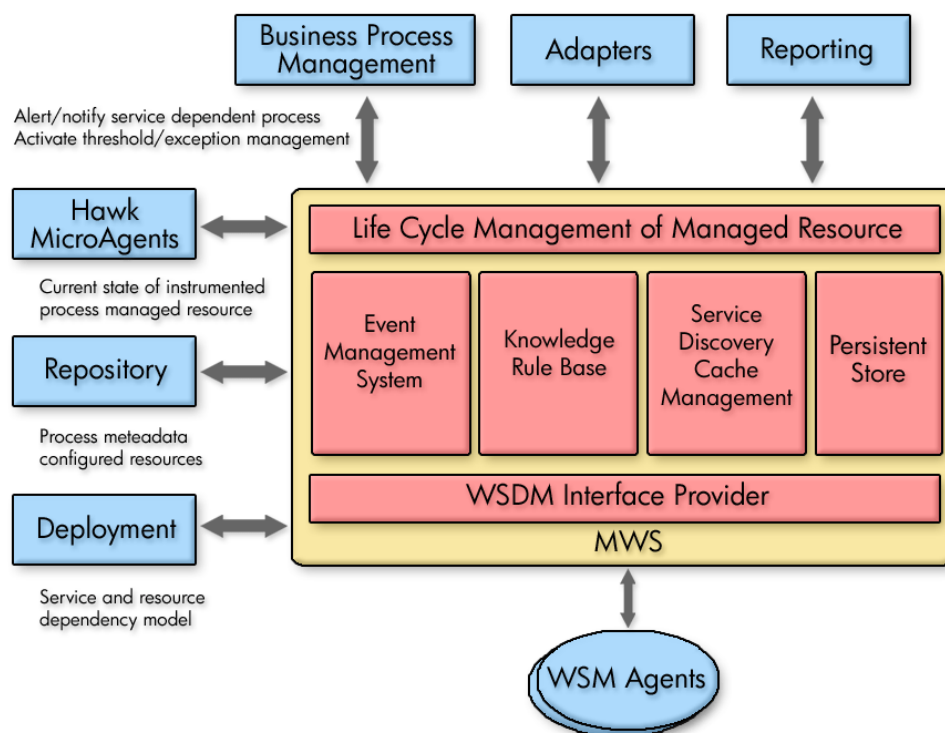
The TIBCO EMA software is developed and distributed by TIBCO. The information presented in this section is included to provide an end-to-end view of the management integration. For detailed implementation information about the EMA software, see the *TIBCO Enterprise Management Advisor User's Guide* included with the TIBCO software.

EMA is responsible for:

- Discovering TIBCO resources.
- Exposing TIBCO resources as MOs.
- Communicating topology changes to the TIBCO SPI using the WSDM Channel.
- Communicating TIBCO resource alerts to the TIBCO SPI.
- Invoking operations on TIBCO resources that are initiated by OVO Operators.
- Communicating performance metrics to the TIBCO SPI.

The agent communicates with the OpenView management platform through the WSDM channel. Managed resources are instrumented as MOs using the TIBCO Hawk API. These MOs are dynamically discovered through the Hawk Console API at run time, and they communicate with the managed resources through the Hawk enabled MicroAgents. The TIBCO SPI can communicate with multiple EMAs, and there is generally one EMA per Hawk domain.

Figure 1-4 shows a high-level architecture for the TIBCO EMA software.



**Figure 1-4: TIBCO Enterprise Management Advisor Architecture**

Figure 1-4 shows a container managing the life cycle of a resource. The state of the managed resource is instrumented through the Hawk API in the TIBCO environment, and the managed resource itself is exposed to OpenView. Each of the TIBCO components has to enlist the set of management aware resources used by it in order to provide the service as defined by the component's contract. This information can very well be an acyclic graph, which the NSM layer can later modify and create a representative template. The repository and the deployment provide this information. For more information on the TIBCO EMA architecture, see the *TIBCO Enterprise Management Advisor User's Guide*

## Deployment Scenarios

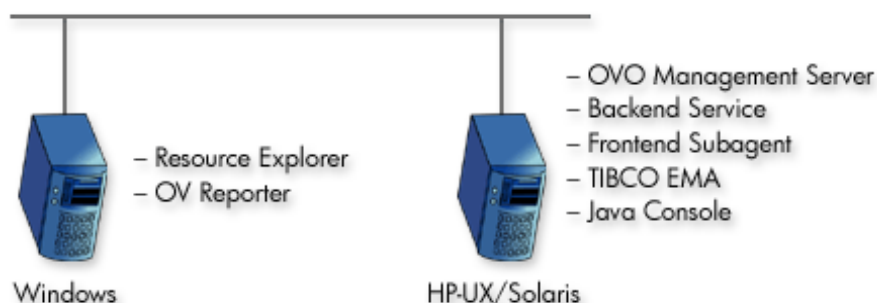
As discussed in the “TIBCO SPI Conceptual Overview” section, the TIBCO SPI is a distributed solution. As such, its components can be distributed in several different deployment configurations. This section provides some common deployment scenarios and does not represent every possible configuration.

- A fully distributed scenario is the recommended deployment scenario. See the “Scenario 3: Fully Distributed” section.

### Consolidated Scenario

It is possible to have a single computer that hosts all of the TIBCO SPI components (in addition to the TIBCO EMA component) except for the Resource Explorer and OV Reporter, which is only available for Windows. In this scenario, the Resource Explorer must be started from the Windows command line and can not be started from within the Java Console. This scenario uses minimal hardware, but results in heavy loads on a single computer. Therefore, it is not recommended for production environments.

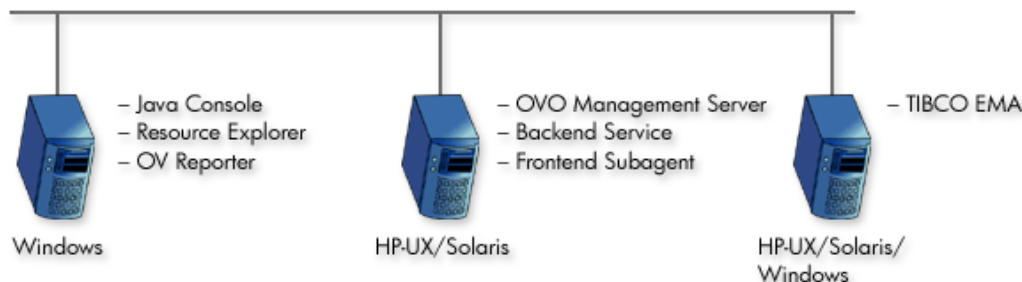
- You may need to reconfigure the default ports used by the components if port conflicts occur.



**Figure 1-5: Consolidated Scenario**

## Scenario 1: Collocated Components Scenario

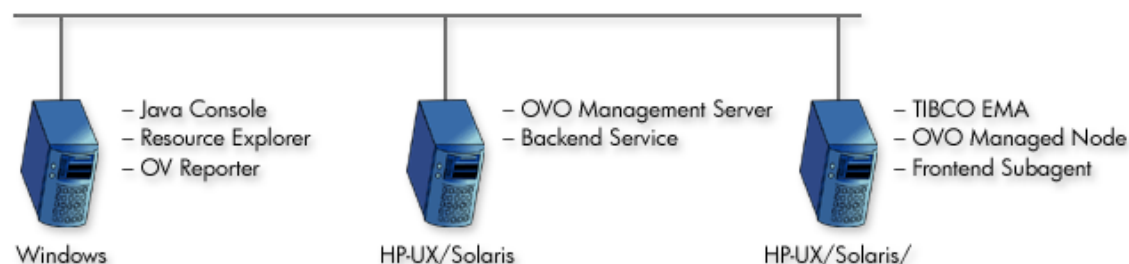
In this scenario, the Frontend Subagent and Backend Service are collocated on the OVO Management Server computer. Management processing is concentrated on a single computer. This scenario is good for testing and is also applicable for less demanding production environments.



**Figure 1-6: Collocated Component Scenario**

## Scenario 2: Remote Frontend Subagent

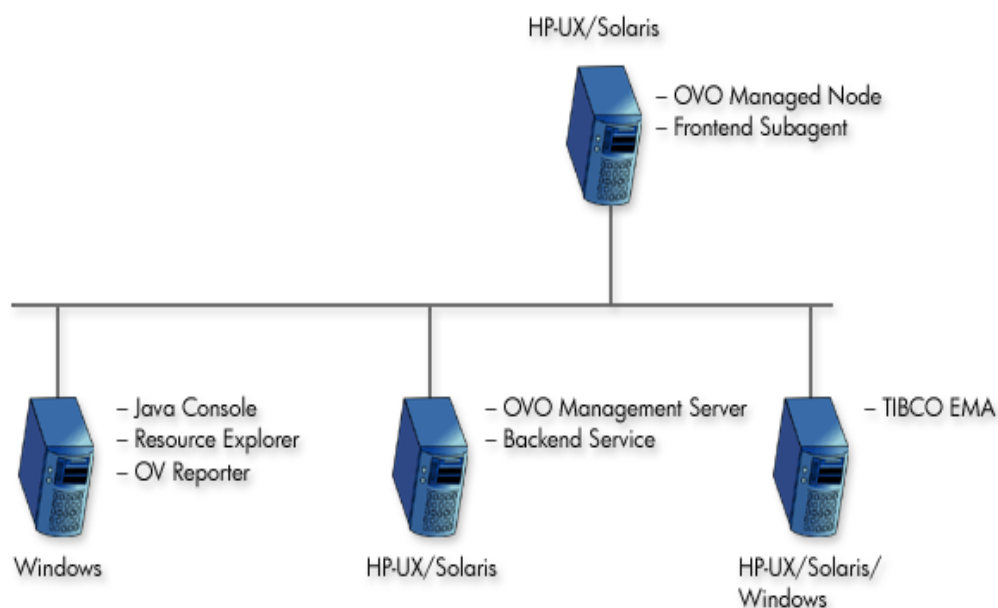
In this scenario, the Frontend Subagent is remote from the OVO Management Server computer and is collocated with the TIBCO EMA component. This scenario effectively separates management processing between a Managed Node and the Management Server. This scenario is ideal for a production environment.



**Figure 1-7: Remote Frontend Subagent Scenario**

## Scenario 3: Fully Distributed (Recommended)

In this scenario, the TIBCO SPI components, as well as the TIBCO EMA component, are separated on different computers. Each computer is relegated to a single task. This scenario provides the most efficient processing and resource utilization. However, this scenario introduces the most overhead. A maximum number of hardware is used and maintaining the solution can become cumbersome. This scenario is ideal for a production environment.



**Figure 1-8: Fully distributed Scenario**

## WSDM Overview

WSDM is a management specification for managing application resources using Web services technology as well as managing Web services using Web services. The specification allows a common integration channel between an ISV and a management station such as OVO. The WSDM standard is defined by the WSDM Technical Community (TC) at OASIS. More information on the standard can be found at the [WSDM TC](#) site at OASIS.

► The WSDM implementation in the TIBCO SPI is based on an HP-authored preliminary version of WSDM known as the Web Service Management Framework (WSMF).

WSDM provides a way to represent and realize management models and is based on Managed Objects (MOs) and the relationships between them. MOs provide management capabilities by implementing management interfaces. The management interfaces are described using WSDL. Hence, WSDM provides the architecture for defining management interfaces for MOs.

The foundational interfaces defined by WSDM can be extended in order to better manage resources in specific domains. In the TIBCO SPI, there are two types of extensions: OpenView domain extensions that provide OpenView data and events to TIBCO applications, and TIBCO domain extensions that enable OpenView to access TIBCO specific manageability.

## XML interfaces

WSDM defines a set of domain agnostic management interfaces, as well as domain specific management interfaces for Web services. Vendors are free to extend WSDM to their specific domain in order to manage resources effectively.

WSDM interfaces are categorized. Each category of interfaces corresponds to one port type in WSDLs.

- **Discovery** – a set of interfaces for discovering MOs and their relationships in the managed environments. There is a set of default relationship types defined in WSDM, and vendors can extend these relationship types as well.
- **Configuration** – a set of interfaces that allows a manager to find out the configuration information about a managed object.
- **Control** – a set of interfaces to control MOs / resources.
- **Performance** – a set of interfaces that allows a manager find out performance information about MOs.
- **EventPush & Event Pull** – a set of interfaces for subscribing to events in either push or pull fashion.
- **EventCallback** – a set of interfaces for listening to events after subscription.

# Installation and Configuration

This chapter provides installation instructions for the various components of the OV TIBCO SPI. The SPI is tightly integrated with OVO for UNIX. If you are not familiar with OVO, you should consult the OVO documentation during the SPI installation. In addition, it is recommended that a development-time installation of OVO be used to test the OV TIBCO SPI integration before installing and running the SPI on a production installation of OVO.

## Installation Process Summary

The following list highlights the installation and configuration process and provides links to the individual installation and configuration tasks. The installation process proceeds as follows:

- 1 Review the “Deployment Scenarios” section in Chapter 1 and select a deployment scenario.
- 2 Review the [Requirements](#) section for hardware and software requirements.
- 3 Install the [TIBCO SPI Depot](#).
- 4 [Configure the Management Server](#)
- 5 [Install the TIBCO SPI Frontend Subagent Component](#)
- 6 [Configure the Backend and the Frontend](#)
- 7 [Assign TIBCO SPI templates](#)
- 8 [Start the TIBCO SPI](#)
- 9 [Install the Java Console](#)
- 10 [Install the HP Resource Explorer](#)
- 11 [Install TIBCO SPI Reports](#)

## Requirements

This section details the requirements for installing and running the TIBCO SPI. Make sure you meet the requirements before you begin the installation.



TIBCO SPI A.01.04 can only be used with TIBCO EMA 2.0. Before installing the TIBCO SPI, make sure the TIBCO EMA 2.0 software has been installed in the TIBCO environment and is operational. For additional TIBCO EMA 2.0 requirements, see the TIBCO EMA documentation.

For HP-UX: `ps -ef | grep -l ema`

For Windows: Use the Services Administrative tool to check if EMA is running.

## Third Party License Review

The TIBCO SPI utilizes third party code. You must review the third party licenses before installing the TIBCO SPI. The licenses are located in the `OV_DOC/TIBCO_SPI_A.01.04.xxx/License` directory. If you disagree with a particular license, you should not install the TIBCO SPI.

## Software Requirements for Standard TIBCO SPI Installation

The following table provides a list of software that is used to run the TIBCO SPI when performing a standard installation.



Additional requirements when using advanced management functions are located in Chapter 4 and Chapter 5.

**Table 2-1: Software Requirements**

Software	Platforms
OVO-U 7.10 and OVO-U 8.x	HP-UX 11.x, Solaris 2.8/5.8/8.0
JavaConsole A.07.00 (included with OVO)	HP-UX 11.x, Windows 2K / NT
DSI2DDF A.01.23 or later for OVO 7.x and DSI2DDF A.02.00.00 or later for OVO 8.x	HP-UX 11.x, Solaris 2.8/5.8/8.0
JRE 1.4.x	HP-UX 11.x, Solaris 2.8/5.8/8.0
OV Reporter 3.5 or later (required to view TIBCO SPI performance reports and graphs)	Windows 2K / NT



## Hardware Requirements

The following recommended hardware requirements should be met when installing and running the TIBCO SPI.

**Table 2-2: Hardware Requirements**

Requirement	Minimum
RAM	<ul style="list-style-type: none"> <li>– 512 MB: System running Frontend Subagent</li> <li>– 512 MB: System running the Backend Service</li> </ul> <p>512 MB is sufficient to run both the Frontend Subagent and Backend Service if they are co-located on the same system.</p>
Disk Space	<ul style="list-style-type: none"> <li>– 50 MB: Front End Subagent</li> <li>– 250 MB: Backend Service</li> </ul>

## Patches

It's important that all recommended patches for OVO-U and Java are installed prior to running any components of the TIBCO SPI. To find the recommended OVO patches, go to <http://support.openview.hp.com/patches>. In the browse by product version section, select **operations for UNIX** and click the >> button. To find the recommended Java patches for HP-UX, go to <http://www.hp.com/products1/unix/java>. From the left side of the screen, click **Patches**. To find the recommended Java patches for Solaris, go to <http://sunsolve.sun.com/pubpatch> for the latest J2SE patch cluster.



Patches must be installed on all systems on which the TIBCO SPI components are running. In addition, patches are updated regularly, so it is a good idea to check for updates every few months.

## Kernel Parameters

Verify that the kernel parameters on both the Backend Service and Frontend Subagent nodes meet the recommended settings to run a Java program. If you are running the Backend Service and/or Frontend Subagent on an HP-UX node, you can use the HPjconfig tool, which provides the recommended kernel parameter settings for your system. To download the HPjconfig tool, go to <http://www.hp.com/products1/unix/java/java2/hpjconfig>.

# Installing and Configuring the TIBCO SPI Components

This section includes detailed steps for installing the TIBCO SPI. The installation includes installing and configuring the Backend Service on the OVO management server, as well as installing and configuring the Frontend Subagent. You must have OVO for UNIX installed before you can install the TIBCO SPI. To install OVO for UNIX, see the OVO documentation. In addition, you must be logged onto the Management Server and Managed Nodes as `root` when completing the instructions in this section.



Make sure that JRE 1.4 is installed on both the Management Server and the Managed Node. On the Management Server, set an environment variable `JAVA_HOME` to the JRE installation directory.

Remove any versions of the TIBCO SPI before completing the installation.

## Install the TIBCO SPI Depot on HP-UX

The TIBCO SPI Depot contains all of the TIBCO SPI components. The installation is performed on an OVO Management Server and automatically installs the TIBCO SPI Backend Service. However, the TIBCO SPI Frontend Subagent and Resource Explorer must be installed after the TIBCO SPI Depot is installed. In addition, if you want to install the Frontend Subagent on a Managed Node, you can do so from the Management Server after you install the Depot.



The Frontend and Backend can be collocated on the same systems. If the Frontend and Backend are on the same system, they must run on the OVO Management Server system. This may be a likely scenario during testing.

To install the TIBCO SPI Depot:

- 1 From a command prompt on the OVO management server, run `swinstall`.
- 2 In the Specify Source dialog box **Source Depot Path...** field, enter the full path to the HP-UX SPI depot on the product CD. For example:

`/<mount_point>/OV_DEPOT/11.0HPUX.sdtape`

- 3 Click the **OK** button.
- 4 In the SD Install - Software Selection dialog box, select **SPITIBCO**.
- 5 Select **Actions** | **Install...** The Install Analysis dialog box displays.
- 6 Click the **OK** button. An Install Window dialog box displays.



The TIBCO SPI may take a few minutes for to be installed. The installation status should report that the installation completed successfully. If the installation status is “Completed with Warnings”, this is expected and you can proceed.

- 7 Click the **Done** button.
- 8 In the SD Install - Software Selection dialog box, select **File** | **Exit**.

## Install the TIBCO SPI Depot on Solaris

When installing the TIBCO SPI on Solaris, run `swinstall` from the command line and provide the complete path to the Solaris SPI depot on the product CD. In addition, include the TIBCO SPI SD install name (`SPITIB`) in the command. For example:

```
swinstall -x reinstall=true -s /<mount_point>/OV_DEPOT/SOLARIS.sdtape
SPITIB
```

## Configure the Management Server

The steps in this section configure the OVO Management Server to use the TIBCO SPI. All of the steps in this section are completed from the OVO Management Server.

### Configuring the `opc_adm` operator

The TIBCO SPI user profile must be assigned to the OVO administrator. This step is required in order for the administrator to see the TIBCO SPI notifications.

To configure the `opc_adm` operator:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From OVO window, select **Window | User Bank**.
- 3 From the User Bank window, right-click **opc\_adm** and select **Modify...** from the popup menu.
- 4 In the Modify User: `opc_adm` window, click the **Profiles...** button.
- 5 In the Profiles of User: `opc_adm` window, verify that the TIBSPI User Profile icon appears. If it's not in the window:
  - a Open the user profile bank window by selecting **Window | User Profile Bank**.
  - b Drag the TIBSPI User Profile from the VPO User Profile Bank window and drop it into the Profiles of User: `opc_adm` window.
  - c Close the VPO User Profile Bank window.
- 6 Close the Profiles of user: `opc_adm` window.
- 7 Close the Modify User: `opc_adm` window by clicking on the **OK** button.
- 8 Close the User Bank window.

### Register the Backend Service

To register the Backend Service:

- 1 From the OVO admin GUI, select the OVO management server in the Node Bank window.
- 2 Add the OVO management server into the TIBSPI-UNIX node group:
  - a From any OVO window, bring up the Node Bank window by selecting **Window | Node Bank**.

- b From any OVO window, bring up the Node Group Bank window by selecting **Window | Node Group Bank**. If the TIBSPI-UNIX node group is currently part of the Node Group Bank, you can skip the remaining steps and go to step 3.
        - c Double-click the **TIBSPI-UNIX** node group in the Node Group window.
        - d Drag the OVO management server node from the Node Bank and drop it onto the Node Group: TIBSPI-UNIX window.
        - e Click the **Close** icon in the Node Group: TIBSPI-UNIX window to close the window.
  - 3 Install the Monitors and Commands to the OVO management server node:
    - a Select the OVO management server in the Node Bank window.
    - b Select **Actions | Agents | Install / Update SW & Config...**
    - c In the Install / Update VPO Software and Configuration dialog box, select **Monitors, Commands, and Actions** in the Components frame.
    - d Select **Force Update** in the Options frame.
    - e Click the **OK** button.
  - 4 On the OVO management server, register the Backend Service by running:

```
cd /opt/OV/bin/  
./tib-perl tib-register-backend add
```



You must invoke the command from the /opt/OV/bin directory.

## Install the TIBCO SPI Frontend Subagent Component

The Frontend Subagent collects management data from the TIBCO EMA software and makes the data available to the Backend Service. The Frontend Subagent can run on the Management Server system or on a remotes system configured as a Managed Node. For detailed steps on installing and configuring OVO managed nodes, please consult your OVO documentation.



If you are running the Frontend Subagent on the Management Server, you can skip the first two sections: “Add a Managed Node to the TIBSPI-UNIX Node Group” and “Install the Monitors, Commands and Actions on a Managed Node”.

### Add a Managed Node to the TIBSPI-UNIX Node Group

If you are installing the Frontend Subagent on a Managed node, the node must be configured as part of the TIBSPI-UNIX Node Group.



If you are installing Frontend Subagent on the Management Server, you can skip this section.

To add a managed node to the TIBSPI-UNIX Node Group:

- 1 From any OVO window, open the Node Bank window by selecting **Window | Node Bank**.
- 2 From any OVO window, open the Node Group Bank window by selecting **Window | Node Group Bank**.
- 3 From the Node Group window, double-click the **TIBSPI-UNIX** node group.
- 4 Drag the node where the Frontend Subagent will be installed from the Node Bank and drop it onto the Node Group: TIBSPI-UNIX window.
- 5 Click the **Close** icon in the Node Group: TIBSPI-UNIX window to close the window.

## Install the Monitors, Commands, and Actions on a Managed Node

If you are installing the Frontend Subagent on a Managed Node, the node must be configured to use monitors, commands, and actions.



If you are installing Frontend Subagent on the Management Server, you can skip this section.

To install monitors, commands, and actions, on a managed node:

- 1 From the Node Bank window, select the node where the Frontend Subagent will be installed.
- 2 Select **Actions | Agent | Install / Update SW & Config**.
- 3 In the Install / Update VPO Software and Configuration dialog, select **Monitors, Commands, and Actions** in the Components frame.
- 4 Select **Force Update** in the Options frame.
- 5 Click the **OK** button.

## Install the Frontend Subagent

To install the Frontend Subagent:

- 1 From the Node Bank window, select the node where you want to install the Frontend Subagent.
- 2 Select **Actions | Subagents | Install/Update...**
- 3 In the Install / Update Subagents dialog box, select the **TIBCO SPI Subagent** in the Subagents frame on the left.
- 4 Click the **OK** button. The Install Subagent Packages window displays. It might take a couple of minutes for the Frontend Subagent to be installed. After installation, the message “SubAgent successfully installed and configured on Managed Node xxx.” displays.
- 5 Press **Enter**.

You cannot start the Frontend Subagent until you configure it. See “Configure the Backend and the Frontend” below. You might get the following critical message in the OVO message browser:



TIBCO SPI Subagent (Frontend) of subagent 25 aborted; process did an exit 0 (OpC30-1040).

You can ignore this message since the Frontend Subagent can’t start until you configure it.

## Verifying the Frontend Subagent Install

You can verify that the Frontend Subagent is installed and registered. On the Frontend system, if it’s an OVO7.x or OVO8.x DCE agent node, run:

```
opcagt -status -id 25
```

The following message displays:

```
VPO Managed Node status:
-----
Control Agent /opt/OV/bin/OpC/opcctl   (xxx) is running
...
TIBCO SPI Subagent (Frontend)/opt/OV/bin/tib-perl /opt/OV/bin/tib-
start-frontend isn't running.
```

If the Frontend Subagent is installed on an OVO8.x HTTPS agent node, run:

```
ovc -status -id TIB
```

A message displays similar to the following:

```
tib-frontend TIB SPI Frontend Subagent      TIB      Stopped
```

## Configure the Backend and the Frontend

Various configuration options for the Backend Service and Frontend Subagent need to be defined. The options are configured using the TIBCO SPI Configuration Editor, which is available from the TIBCO SPI Tools application group.

To configure the Backend and the Frontend:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 In the Application Bank window, double-click the **TIBCO SPI Tools** application group.
- 4 In the Application Group: TIBCO SPI Tools window, right-click **Configure TIBCO SPI** application and select **Execute** from the popup menu.
- 5 From the Configuration Editor, click the icon next to **Top of tree**.
- 6 Click the icon next to **Back End Configuration**.
- 7 Select **OVO Management Server Info**.

- 8 Modify the Host Name and any other fields if required. Make sure the default RMI port is not already in use.
- 9 Click the icon next to **Front End Configuration**.
- 10 Select **Managed Object Information**.
- 11 Select **WSDL Location Information**.
- 12 In the Location Candidates list, add a WSDL location for a managed object you want the Frontend Subagent to monitor. Managed objects are exposed by the TIBCO EMA software. See the TIBCO EMA documentation for instructions on finding a managed object's published WSDL location.



If the WSDL location uses HTTPS, then you must configure the Frontend Subagent's security settings. Please refer to the “Configuring HTTPS Communication” section in Chapter 7 “Security Features and Configuration”.

- 13 To configure an additional WSDL location, create a new Location Candidate by selecting **Managed Object Information** and from the menu click **Edit | New**.
- 14 In the new Location Candidates list, add a WSDL location for the managed object you want the Frontend Subagent to monitor.
- 15 Repeat steps 13 and 14 to add additional WSDL locations.



The WSDL locations that are configured in the Managed Object Information area must be accessible before starting up the Frontend Subagent.

- 16 Select **OVO Management Agent Info**.
- 17 Modify the HostName and any other fields if required. Make sure the default RMI port is not already in use.
- 18 Select **File | Save** to save your changes to the configuration file.
- 19 Select **File | Exit**.
- 20 In the Output of Application window, you should see the message ‘TIBCO SPI configuration completed.’ It may take a few minutes for the message to appear. Don't close the Output of Application window until you see the completion message.
- 21 Click the **Close** button in the Output of Application window.

## Assign TIBCO SPI Templates

OVO-based templates are used to capture management data and metrics. The templates are specific to the TIBCO SPI and are included with the TIBCO SPI installation. In general, the templates collect and monitor data. The data includes metric data, operational notifications, logging data, and UDM metrics. The data is used to effectively manage the TIBCO environment within OVO. See Appendix B “List Templates and Reports” for a complete reference of the templates included with the TIBCO SPI.

Before assigning TIBCO SPI templates, you must configure the data sources that are used to store the collected data metrics.

## Configure Data Sources for Metric Data Collection

The metrics collected by the Frontend Subagent templates are stored in a data source. A script is provided that creates and configures the data source.

To configure data source for metric data collection:

- 1 From a command prompt, change directories to /opt/OV/bin.
- 2 Run the command:  

```
./tib-perl tib-create-datasources
```

## Assign Backend Service Templates

The TIBSPI-UNIX-Backend-V1 template group contains the templates that monitor the TIBCO SPI log files on the OVO management server. This template group is assigned and deployed on the Management Server.

To assign the TIBSPI-UNIX-Backend-V1 template group to the Management Server:

- 1 From the OVO admin GUI, open the Node Bank window by selecting **Window | Node Bank**.
- 2 In the Node Bank window, select the Management Server node where the Backend Service is running.
- 3 Select **Actions | Agents | Assign Templates...**
- 4 In the Define Configuration dialog box, click the **Add...** button.
- 5 In the Add Configuration dialog box, click the **Open Template Window...** button.
- 6 In the Message Source Templates dialog box, double-click the **SPI for TIBCO** template group in the Template Groups from the left.
- 7 Select the **TIBSPI-UNIX-Backend-V1** template group.
- 8 In the Add Configuration dialog box, click the **Get Template Selections** button.
- 9 Click the **OK** button in the Add Configuration dialog box.
- 10 Close the Message Source Templates dialog box by selecting **Templates | Close Window**.
- 11 Click **OK** in the Define Configuration dialog box.

## Deploy Backend Service Templates

To deploy the Backend Service Templates:

- 1 In the Node Bank window, select the Management Server node where the Backend Service is running.
- 2 Select **Actions | Agents | Install / Update SW & Config**.



- 3 In the Install / Update VPO Software and Configuration dialog box, select **Templates** in the Components frame.
- 4 Select **Force Update** in the Options frame.
- 5 Click the **OK** button.

## Assign Frontend Subagent Templates

The Frontend Subagent template groups must be assigned and deployed to the node where the Frontend Subagent is installed before the templates can be used.

To assign Frontend template groups to a node:

- 1 From the OVO admin GUI, open the Node Bank window by selecting **Window | Node Bank**.
- 2 In the Node Bank window, select the node where the Frontend Subagent is running.
- 3 Select **Actions | Agents | Assign Templates....**
- 4 In the Define Configuration dialog box, click the **Add...** button.
- 5 In the Add Configuration dialog box, click the **Open Template Window...** button.
- 6 In the Message Source Templates dialog box, double-click the **SPI for TIBCO** template group in the Template Groups from the left.
- 7 Select the **TIBSPI-EventService-V1** template group.
- 8 In the Add Configuration dialog box, click the **Get Template Selections** button.
- 9 If you want to monitor the Frontend Subagent's logfile, repeat steps 8 and 9 for the **TIBSPI-UNIX-Frontend-V1** template group.
- 10 If you're going to log metrics, repeat steps 8 and 9 for the **TIBSPI-Metrics-V1** template group.
- 11 Click the **OK** button in the Add Configuration dialog box.
- 12 Close the Message Source Templates dialog box by selecting **Templates | Close Window**.
- 13 Click **OK** in the Define Configuration dialog box.

## Deploy templates to the Frontend Subagent Node

To deploy the templates to the Frontend Subagent Node:

- 1 In the Node Bank window, select the node where the Frontend Subagent is running.
- 2 Select **Actions | Agents | Install / Update SW & Config**.
- 3 In the Install / Update VPO Software and Configuration dialog box, select **Templates** in the Components frame.
- 4 Select **Force Update** in the Options frame.
- 5 Click the **OK** button.

## Assign the TIBSPI-UNIX- V1 Template Group

The TIBSPI-UNIX-V1 template group contains the templates that monitor various TIBCO products that are running on UNIX nodes.



The OVO agent must be deployed onto the UNIX nodes where the TIBCO products are running before you can deploy the TIBSPI-UNIX-V1 template group.

For this procedure, assume that TIBCO is running on all of the nodes in the TIBSPI-UNIX node group. Therefore, assign the TIBSPI-UNIX-V1 template group to the TIBSPI-UNIX node group.

To assign the template group to the UNIX node group:

- 1 From the OVO admin GUI, open the Node Group Window by selecting **Window | Node Group Bank**.
- 2 In the Node Group Bank window, select the **TIBSPI-UNIX** node group.
- 3 Select **Actions | Agents | Assign Templates...**
- 4 In the Define Configuration dialog box, click the **Add...** button.
- 5 In the Add Configuration dialog box, click the **Open Template Window...** button.
- 6 In the Message Source Templates dialog box, double-click the **SPI for TIBCO** template group in the Template Groups from the left.
- 7 Select the **TIBSPI-UNIX-V1** template group.
- 8 In the Add Configuration dialog box, click the **Get Template Selections** button.
- 9 Click the **OK** button in the Add Configuration dialog box.
- 10 Close the Message Source Templates dialog box by selecting **Templates | Close Window**.
- 11 Click the **OK** button in the Define Configuration dialog box.

## Deploy the TIBSPI-UNIX- V1 Template Group to the UNIX Nodes

To deploy the template group to the UNIX node group:

- 1 In the Node Group Bank window, select the **TIBSPI-UNIX** node group.
- 2 Select **Actions | Agents | Install / Update SW & Config**.
- 3 In the Install / Update VPO Software and Configuration dialog box, select **Templates** in the Components frame.
- 4 Select **Force Update** in the Options frame.
- 5 Click the **OK** button.

## Assign the TIBSPI-Windows-V1 Template Group

The TIBSPI-Windows-V1 template group contains the templates that monitor various TIBCO products' logfiles that are running on Windows nodes.



The OVO agent must be deployed onto the Windows nodes where the TIBCO products are running before you can deploy the TIBSPI-Windows-V1 template group.

For this procedure, assume that TIBCO is running on all of the nodes in the TIBSPI-WINDOWS node group. Therefore, assign the TIBSPI-Windows-V1 template group to the TIBSPI-WINDOWS node group.

To assign the template group to the Windows node group:

- 1 From the OVO admin GUI, open the Node Group Window by selecting **Window | Node Group Bank**.
- 2 In the Node Group Bank window, select the **TIBSPI-WINDOWS** node group.
- 3 Select **Actions | Agents | Assign Templates...**
- 4 In the Define Configuration dialog box, click the **Add...** button.
- 5 In the Add Configuration dialog box, click the **Open Template Window...** button.
- 6 In the Message Source Templates dialog box, double-click the **SPI for TIBCO** template group in the Template Groups from the left.
- 7 Select the **TIBSPI-Windows-V1** template group.
- 8 In the Add Configuration dialog box, click the **Get Template Selections** button.
- 9 Click the **OK** button in the Add Configuration dialog box.
- 10 Close the Message Source Templates dialog box by selecting **Templates | Close Window**.
- 11 Click **OK** in the Define Configuration dialog box.

## Deploying the TIBSPI-Windows-V1 Template Group to the Windows Nodes

To deploy the templates to the Windows node group:

- 1 In the Node Group Bank window, select the **TIBSPI-WINDOWS** node group.
- 2 Select **Actions | Agents | Install / Update SW & Config**.
- 3 In the Install / Update VPO Software and Configuration dialog box, select **Templates** in the Components frame.
- 4 Select **Force Update** in the Options frame.
- 5 Click the **OK** button.
- 6 Repeat this procedure for each Windows Node.

## Verifying Deployed Templates

The following procedure can be used to verify that TIBCO SPI template groups have been deployed.

To verify deployed templates:

- 1 From a command prompt, change directories to /opt/OV/bin/OpC.
- 2 Run the command:  
`opctemplate`
- 3 In the output, verify that the deployed TIBSPI templates display.

## Start the TIBCO SPI

The TIBCO SPI can be started from the OVO Administration GUI. Before starting the TIBCO SPI, you must have installed and configured the Frontend Subagent.



Make sure you have set the JAVA\_HOME variable to the JRE installation directory. See the “Configuring the Backend and Frontend” section above.

To start the TIBCO SPI:

- 1 Run the OVO admin GUI on the OVO management server and logon.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 From the VPO Application Bank window, double-click the **TIBCO SPI Tools** icon.
- 4 From the Application Group: TIBCO SPI Tools window, right-click **Start TIBCO SPI** and select **Execute** in the popup menu. An Output of Application window displays. The following message displays:

```
Node ovh072.cup.hp.com:
Starting OpC services...Done.
Starting up the backend...
The backend started.
Starting up the frontend...
```

The frontend startup is complete when the message 'Service Map Created' is emitted in the log file.

- 5 Use the steps in the “Checking the TIBCO SPI Status” section of this chapter to verify that the TIBCO SPI is started.

## Restarting the TIBCO SPI

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.

- 3 From the VPO Application Bank window, double-click the **TIBCO SPI Tools** icon.
- 4 From the Application Group: TIBCO SPI Tools window, right-click **Restart TIBCO SPI** and select **Execute** in the popup menu. An Output of Application window displays. The following message displays:  

```
Node ovh072.cup.hp.com:
Starting OpC services...Done.
The frontend was stopped successfully
The backend was stopped successfully
Starting up the backend...
The backend started.
Starting up the frontend...

The Frontend startup is complete when the message 'Service Map Created' is emitted
in the log file.
```
- 5 Use the steps in the “Checking the TIBCO SPI Status” section of this chapter to verify that the TIBCO SPI is started.

## Stopping the TIBCO SPI

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 From the VPO Application Bank window, double-click the **TIBCO SPI Tools** icon.
- 4 From the Application Group: TIBCO SPI Tools window, right-click **Stop TIBCO SPI** and select **Execute** in the popup menu. An Output of Application window displays the message:  

```
Node xxx:
Shutting down OpC services...Done.
The frontend was stopped successfully.
The backend was stopped successfully.
```
- 5 Click the **Close** button.
- 6 Use the steps in the “Checking the TIBCO SPI Status” section of this chapter to verify that the TIBCO SPI is stopped.

## Checking the TIBCO SPI Status

You can check the status of the TIBCO SPI to see if it is started and operating successfully. You can also check the status to insure that the TIBCO SPI has been stopped successfully. This is helpful when debugging any problems.

To check the Status of the TIBCO SPI:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 From the VPO Application Bank window, double-click the **TIBCO SPI Tools** icon.
- 4 From the Application Group: TIBCO SPI Tools window, right-click **Check Status TIBCO SPI** and select **Execute** in the popup menu. An Output of Application window displays the status of both the Backend and Frontend Subagent.
- 5 After reviewing the status, click the **Close** button.

## Verifying a Start Status

If the TIBCO SPI is started without any errors, the following status displays when using the TIBCO SPI Check Status command.


```
object manager name: tib-backend
state:              RUNNING
PID:                4063
last message:       -
exit status:        -

Node ovh072.cup.hp.com:
VPO Managed Node status:
-----
Control Agent       /opt/OV/bin/OpC/opcctl (2428) is running
Message Agent       /opt/OV/bin/OpC/opcmgsa (5944) is running
BBC Local Location Broker /opt/OV/bin/llbserver (5945) is running
Subagent 25:
  TIBCO SPI Subagent (Frontend)
    /opt/OV/bin/tib-perl /opt/OV/bin/tib-start-frontend (4389) is
    running
Done.
```

When using an OVO 8.x HTTPS agent node, the following status displays:

```
object manager name: tib-backend
state:              RUNNING
PID:                4063
last message:       -
exit status:        -

Node ovh072.cup.hp.com:
OVO Managed Node status:
-----
ovcd      OV Control      CORE      (14629)  Running
ovbbccb   OV Communication Broke CORE      (14630)  Running
ovconfd   OV Config and Deploy COREXT     (14631)  Running
Subagent 25:
  TIB SPI Frontend Subagent tib-frontend (4389) is running
Done.
```

 If the Frontend Subagent is not running, it may be still starting up. On the Frontend Subagent node, look in the `/var/opt/OV/log/tib/frontend.log` file for the 'Service Map Created' message. If the message isn't emitted, the Frontend Subagent has not completed its startup.

## Verify a Stop Status

If the TIBCO SPI has stopped without any errors, the following status displays when using the TIBCO SPI Check Status command:

```
object manager name: tib-backend
state:                NOT_RUNNING
PID:                  -
last message:         -
exit status:          Exit(0)

Node ovh072.cup.hp.com:
VPO Managed Node status :
-----
Control Agent        /opt/OV/bin/OpC/opcctl (2428) is running
Message Agent        /opt/OV/bin/OpC/opcmsga (5944) is running
BBC Local Location Broker /opt/OV/bin/llbserver (5945) is running
Subagent 25:
  TIB SPI Subagent (Frontend)
  /opt/OV/bin/tib-perl /opt/OV/bin/tib-stop-frontend isn't running
Done.
```

When using an OVO 8.x HTTPS agent node, the following status displays:

```
object manager name: tib-backend
state:                NOT_RUNNING
PID:                  6677
last message:         -
exit status:          Exit(0)

Node ovh072.cup.hp.com:
OVO Managed Node status:
ovcd      OV Control      CORE      (14629)  Running
ovbbccb   OV Communication Broker CORE      (14630)  Running
ovconfd   OV Config and Deploy COREXT     (14631)  Running
Subagent 25:
  TIBSPI SPI Frontend Subagent tib-frontend isn't running
Done.
```

## Install the Java Console

OVO for UNIX includes a Java-based operator's console that is used to monitor an OVO managed environment. In regards to the TIBCO SPI, the Java console is used to:

- View the status of the TIBCO SPI and TIBCO EMA services.
- Perform impact analysis and root cause analysis for the TIBCO SPI and TIBCO EMA services.
- Interact with TIBCO SPI tools.
- View and manage TIBCO SPI and TIBCO EMA messages.
- Access the HP Resource Explorer

The Java Console can run on any system where a JRE is installed. However, the instructions in this section are specific to installing the Java Console on Windows. See the OVO documentation or the Java Console's Online Help (once you have installed the Java console on Windows) for instructions on additional platform installation instructions.



To access the HP Resource Explorer from within the Java Console, you must install the Java Console on a Windows system. See the “Installing the HP Resource Explorer” section below.

To install the Java console:

- 1 Ftp /opt/OV/www/htdocs/ito\_op/ITO\_JAVA.exe from the OVO management server to the PC where you are installing the console.
- 2 Execute ITO\_JAVA.exe. An install shield displays and guides you through the installation process.

## Starting the Java Console

To start the Java Console:

- 1 Double-click the Java Console icon on the desktop.
- 2 Enter the following information in the HP OpenView Operations Login dialog box (if a different operator was configured, use that operator name instead of “TIBSPI\_Op”):
  - **User Name:** TIBSPI\_Op
  - **Password:** TIBSPI\_Op
  - **Management Server:** <your OVO management server system>
- 3 Click **OK**. The Java Console starts and displays information for your OVO environment.
- 4 From the Object Pane, expand the **Services** node to view TIBCO services.
- 5 From the Object Pane, expand the **Message Groups** node to view the TIBCO and TIBCO SPI message groups.
- 6 From the Object Pane, expand the **Applications** node to view TIBCO SPI Tools.



If you are new to the Java console, see the OVO documentation or the Java Console's Online Help.

## Install the HP Resource Explorer

The HP Resource Explorer is used to interact with TIBCO EMA Managed Objects (MOs) that are exposed for management purposes. The Resource Explorer interacts directly with the MOs and allows the operator to perform many client management tasks. These include:

- Browsing MO published events
- Browsing relationships between MOs



- Browsing and editing MO attributes
- Invoking operations on MOs and editing their parameter values

The HP Resource Explorer must be installed on a Windows platform. In addition, when accessing the Resource Explorer from the Java Console, the Console and Resource Explorer must be installed on the same computer.

To install the HP Resource Explorer:

- 1 FTP tib\_resource\_explorer.zip, located on the management server at /opt/OV/bin, to a location on your Windows PC.
- 2 Unzip the HP Resource Explorer to any directory on the local machine.
- 3 Modify the user's PATH environment variable to include the directory where you installed the HP Resource Explorer. The PATH is used by the Java Console to locate and start the Resource Explorer.

## Starting the HP Resource Explorer from the Java Console

The HP Resource Explorer is typically started from the Java Console. You can start multiple concurrent Resource Explorer sessions.

To start the Resource Explorer from the Java Console:

- 1 Start the HP Java Console.
- 2 From the Object Pane, expand the Services node.
- 3 From the TIBService node, right-click a TIBCO service and then from the popup menu select **Start | Resource Explorer**. The Resource Explorer starts in its own window in a separate process.

Online help is included with the Resource Explorer. From the Explorer's menu bar, select **Help**. The Online Help includes instructions for browsing and editing MOs as well as invoking an MO's operations.

## Starting the HP Resource Explorer form the Command Line

The Resource Explorer can also be started from the command line using the `hp_resource_explorer.bat` file. The file is located in the root directory where the Resource Explorer is installed.

As part of the command, you must pass the URL of the management Web service. Once invoked, this Web service becomes the root of the functional tree in the Resource Explorer. The following is an example command line:

```
hp_resource_explorer.bat -moUri
http://<host>:8888/?wsdl:objid=tibema://www.tibco.com/ema/2005/01/mo/
identity/ServiceInstance/tibtest1/TIBCOServers/ovw001.cup.hp.com-7500
```

## Install TIBCO SPI Reports

The TIBCO SPI includes an integration with HP OpenView Reporter, which is a Windows-based report management system. As part of the integration, a set of reports are included with the TIBCO SPI and used to view the performance of the TIBCO Applications.



HP OpenView Reporter must be installed prior to completing the instructions in this section. For detailed installation instructions, see the HP OpenView Reporter documentation. In addition, the Frontend Subagent system must be running either MeasureWare or CODA. See Chapter 3 “Working with Standard Management Functions” for more information on MeasureWare and CODA.

To install the TIBCO SPI Reports:

- 1 From the HP OpenView Smart Plug-ins for OVO/UNIX CD, change directories to `\OV_REPORTER\TIBCOSPI_A.01.03.xxx`.
- 2 Double-click the **TIBSPI-Reporter.msi**. An InstallShield Wizard displays.
- 3 Click **Next**. The Setup Type screen displays.
- 4 Accept the default option (**Complete**) and click **Next**. The Ready to Install the Program screen displays.
- 5 Click **Install** to initiate the installation.
- 6 After the installation is complete, click **Finish**.

## Uninstalling the TIBCO SPI

To uninstall the TIBCO SPI, use the procedures in this section. The procedures should be completed in the order in which they are listed.



Make sure that you uninstall the Frontend Subagent before you remove the TIBCO SPI from the OVO management server. If you do not, you will encounter problems when reinstalling the TIBCO SPI.

### Uninstalling the Frontend Subagent

The Frontend Subagent must be stopped before you can stop the TIBCO SPI.

To uninstall the Frontend Subagent:

- 1 On the node where the Frontend Subagent is installed, stop the Subagent by running:  

```
opcagt -stop -id 25 (OVO-U 7.x)
```

```
ovc -stop -id TIB (OVO-U 8.x)
```
- 2 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.

- 3 In the Node Bank window, select the node where the Frontend Subagent is installed.
- 4 Select **Actions** | **Subagents** | **Deinstall...**
- 5 In the Deinstall Subagents dialog box, select **TIBCO SPI** Subagent in the Subagents frame on the left.
- 6 Click the **OK** button. The Install Subagent Packages dialog box displays. A confirmation message displays, “SubAgent successfully removed and unconfigured from Managed Node xxx.”
- 7 Press **Enter**.
- 8 If the opcagt is not running on the node where the Frontend Subagent was installed, restart it by running.

```
opcagt -start -id 25 (OVO-U 7.x)
```

```
ovc -start -id TIB (OVO-U 8.x)
```

- 9 Verify that the Frontend Subagent is uninstalled by running the opcagt command on the node where the Frontend Subagent was running:

```
opcagt -status -id 25 (OVO-U 7.x)
```

```
ovc -status TIB (OVO-U 8.x)
```

The output should be:

```
Error opcctl (Control Agent) (xxx) : Subagent 25 not registered
```

The output on OVO 8.x should be:

```
(ctrl-4) No component matches target.
```

## Remove the TIBCO SPI Bits on HP-UX

- 1 Stop the Backend Service on the OVO management server by running `ovstop tib-backend`.
- 2 On the OVO management server, run `swremove`.
- 3 From the SD Remove - Software Selection dialog box, select the **SPITIBCO**.
- 4 Select **Actions** | **Remove...**
- 5 From the Remove Analysis dialog box, click the **OK** button.
- 6 From the Remove Window dialog box, verify that the Status says that it completed successfully. If the status is “Completed with Warnings”, this is expected and you can proceed. If the status indicates errors, click the **Logfile...** button to view the errors.
- 7 Click the **Done** button.
- 8 Select **File** | **Exit** to Close the SD Remove - Software Selection dialog box.

## Remove the TIBCO SPI Bits on Solaris

To remove the TIBCO SPI Bits on Solaris:

- 1 Stop the Backend Service on the OVO management server by running  
`ovstop tib-backend`.
- 2 On the OVO management server, run `swremove` from the command line and include the TIBCO SPI SD install name (SPITIB) in the command.  
  
`swremove SPITIB`

## Remove TIBCO SPI Regroup Condition

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From the Node Bank, select **Actions | Server-Regrouping**.
- 3 In the Regroup Conditions window, select **Send SNMP Traps to Service**.
- 4 Click the **Delete** button.
- 5 Click the **Yes** button to confirm the delete.
- 6 Click the **OK** button to save the change and close the window.

## Remove TIBCO SPI Message Groups

- 1 From any OVO window, select **Window | Message Group Bank**.
- 2 In the Message Group Bank window, right-click the **TIBCO** message group and select **Delete** from the popup menu.
- 3 Click the **Yes** button to confirm the delete.
- 4 Right-click **TIBCO SPI** message group and select **Delete** from the popup menu.
- 5 Click the **Yes** button to confirm the delete.
- 6 Click the **Close** icon to close the window.

## Remove the TIBCO SPI Application Group

- 1 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 2 From the Application Bank window, right-click **TIBCO SPI Tools** and select **Delete** from the popup menu.
- 3 Click the **Yes** button to confirm the delete.
- 4 Click the **Close** icon to close the window.

## Remove the TIBCO SPI User Profile

- 1 From OVO window, select **Window | User Profile Bank**.
- 2 From the User Profile Bank window, right-click the **TIBSPI User Profile** and select **Delete** from the popup menu.

- 3 Click the **Yes** button to confirm the delete.
- 4 Click the **Close** icon to close the window.

## Remove the TIBCO SPI User

- 1 From OVO window, select **Window | User Bank**.
- 2 From the User Bank window, right-click **TIBSPI\_Op** and select **Delete** from the popup menu.
- 3 Click the **Yes** button to confirm the delete.
- 4 Click the **Close** icon to close the window.

## Remove TIBCO SPI Message Templates

- 1 From any OVO window, select **Window | Message Source Templates**.
- 2 In the Message Source Templates window, select **SPI for TIBCO** in the Template Groups frame on the left.
- 3 Select **all of the template groups** in the right frame.
- 4 Click the **Delete From All...** button.
- 5 Click the **Yes** button to confirm the delete.
- 6 Select **SPI for TIBCO** again in the Template Groups frame.
- 7 Click the **Delete From All...** button.
- 8 Click the **Yes** button to confirm the delete.
- 9 Select **[Toplevel]** in the Template Groups frame.
- 10 Select all of the template groups that start with **TIBSPI-** in the right frame .
- 11 Click the **Delete From All...** button.
- 12 Click the **Yes** button to confirm the delete.
- 13 Select **[Toplevel]** again in the Template Groups frame.
- 14 Select all of the templates that start with **TIBSPI-** in the right frame .
- 15 Click the **Delete From All...** button.
- 16 Click the **Yes** button to confirm the delete.
- 17 Close the window by selecting **Templates | Close Window**.

## Reinstall the Templates Monitor and Commands

- 1 From the Node Group Bank window, select the **TIBSPI-UNIX** node group.
- 2 Select **Action | Agents | Install / Update SW & Config...**

- 3 In the Install / Update VPO Software and Configuration dialog box, select **Templates, Monitors and Commands** in the Components frame on the left.
- 4 Select **Force Update** in the Options frame.
- 5 Click the **OK** button.
- 6 In the Node Group Bank window, select the **TIBSPI-WINDOWS** node group.
- 7 Select **Action | Agents | Install / Update SW & Config...**
- 8 In the Install / Update VPO Software and Configuration dialog box, select **Templates, Monitors and Commands** in the Components frame on the left.
- 9 Select **Force Update** in the Options frame.
- 10 Click the **OK** button.

## Remove the TIBCO SPI Node Groups From the OVO Database

- 1 From any OVO window, select **Window | Node Group Bank**.
- 2 In the Node Group Bank window, right-click the **TIBSPI-UNIX** node group and select **Delete** from the popup menu.
- 3 Click the **Yes** button to confirm the delete.
- 4 Right-click **TIBSPI-WINDOWS** node group and select **Delete** from the popup menu.
- 5 Click the **Yes** button to confirm the delete.
- 6 Right-click **TIBSPI-EXTERNAL** node group and select **Delete** from the popup menu.
- 7 Click the **Yes** button to confirm the delete.
- 8 Click the **Close** icon to close the window.

# Performing Standard Management Functions

This chapter provides management tasks that are typically performed when managing a TIBCO environment. In particular, the following sections are included:

- Service Management
- Event Management
- Monitoring and Data Collection
- Reporting and Performance Graphs
- Monitoring Performance Metrics with OVPM
- TIBCO SPI Self Management
- Modify Logging and Tracing Levels

## Service Management

Service management is achieved using the Service Navigator that is included in the OVO Java Console and using the HP Resources Explorer plug-in to the Service Navigator. During runtime, the TIBCO SPI automatically discovers TIBCO managed resources and represents them as a Service Map. Any deployment changes that occur in the TIBCO environment are dynamically synchronized with the Service Map.

The Service Navigator and HP Resource Explorer also allow detailed management of the resources that are presented in the Service Map. The detailed management includes browsing the managed resource hierarchy, their relationships, attributes, metrics, and invocation of methods that are exposed by the resource.

Instructions for installing and starting the Java Console (including the Service Navigator) and the HP Resource Explorer are located in Chapter 2. In addition, Online Help is available for both the Java Console and HP Resource Explorer.



The section is intended only as a “quick start” reference and does not represent a replacement for the Java Console or HP Resource Explorer documentation.

## Viewing TIBCO Managed Resources

To view TIBCO managed resources:

- 1 Start and log in to the Java Console.
- 2 From the Object Pane, expand the **Services** node to view TIBCO resources. The resources are represented in a hierarchy. Expand any node to view any contained resources.
- 3 From the TIBService node, right-click a TIBCO resource and from the popup menu select **Start | Resource Explorer**. The Resource Explorer starts in its own window in a separate process. Like the service node, the Resource Explorer lists the TIBCO resources in a hierarchy form and allows you to select each resource, view its management data, and invoke any available operations.

## Linking Other Service Maps

This feature allows you to link the TIBCO service map to the infrastructure service maps that the TIBCO service nodes depend on. These infrastructure elements could be anything from hardware, to other applications like SAP or Database. To see an integrated service map in the service navigator, the infrastructure components must have a corresponding SPI installed and must have its own service map in the service navigator. There are two methods of linking other SPI service maps to the TIBCO SPI service map:

- Automatic
- Manually (based on a configuration file)



The Automatic linking is currently limited to the service map of the OS SPI. Manually, you can link any service map.

### Automatically Linking

To automatically link the OS SPI service map, the node hosting RVD should be managed using the OS SPI and the OS SPI's service discovery should be enabled.

### Manually Linking

The TIBCO SPI also enables manually linking to other SPIs. For example if there is a DB adapter on the TIBCO EMA agent, then this adapter can be linked to a DB SPI which is already deployed has a service map.

To manually link other service maps:

- 1 Run and log in to the OVO admin GUI on the OVO management server.



- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 In the Application Bank window, double-click the **TIBCO SPI Tools** application group.
- 4 In the Application Group: TIBCO SPI Tools window, right-click **Configure TIBCO SPI** application and select **Execute** from the popup menu.
- 5 Add the following properties:
  - **AliasInfo/ServiceName**: Adapter's service name. The name can be found by clicking **Properties** on the service icon in the Java Console's Service Navigator.
  - **AliasInfo/AliasName**: The SPI's (i.e. DB SPI) service name. The name can be found by clicking **Properties** on the service icon for a SPI in the Java Console's Service Navigator.
  - **AliasInfo/AliasLabel**: The SPI's (i.e. DB SPI) service label. The label can be found by clicking **Properties** on the service icon for a SPI in the Java Console's Service Navigator.
- 6 Restart Frontend Subagent.

## Filtering Unwanted MO Types

You can define a list of MO types that you do not want to be displayed in the Java Console's Service Map. This feature is especially useful if there are many MOs that are being exposed.

To filter out unwanted MO types:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 In the Application Bank window, double-click the **TIBCO SPI Tools** application group.
- 4 In the Application Group: TIBCO SPI Tools window, right-click **Configure TIBCO SPI application** and select **Execute** from the popup menu.
- 5 In the WCConfig.xml Configuration Editor GUI, click the icon next to **Top** of tree.
- 6 Click the icon next to **Front End Configuration**.
- 7 Click the icon next to **Misc. Configuration**.
- 8 Select **Managed Objects to Ignore**
- 9 Fill in the managed object types you want ignored in the General Info section in the right frame.
- 10 Select **File | Save** to save your changes to the configuration file.
- 11 Select **File | Exit**.
- 12 In the Output of Application window, you should see the message 'TIBCO SPI configuration completed.' It may take some time before this message displays. Do not close the Output of Application window until you see the message.

- 13 Verify that the Frontend Subagent startup has completed. Look for the 'Service map created' message in the `/var/opt/OV/log/tib/frontend.log` file.
- 14 Start the Java Console. Logon with the username *TIBSPI\_Op* and password *TIBSPI\_Op* (if a different operator was configured, use that operator name instead of *TIBSPI\_Op*). The MO that you filtered is no longer in the service map.

## Event Management

The TIBCO SPI solution monitors availability and status of TIBCO applications as well as providing the ability to invoke applications manually or automatically when events are received through both an OpenView template, as well as a WSDM channel. You can filter events by adding new conditions using the *TIBSPI-EventService-Msg-V1* template or create your own custom template.

### Viewing TIBCO Events

All events through EMA are captured and displayed in the Java Console's message browser.

To view TIBCO Events:

- 1 Start and log in to the Java Console.
- 2 From the Object Pane, expand the **Message Groups** node to view the TIBCO and TIBCO SPI message groups.
- 3 Right-click a message and select **Properties**. The Message Properties dialog box displays and lists additional details about the event.

### Automatically Responding to Events

A script is provided that allows you to invoke an MO's operations from the command line or from within an OVO template. Typically, the Resource Explorer is used to invoke operations. However, using this script enables you to perform operator automated actions, where actions are performed based on captured events.

To invoke operations from the command line:

- 1 From a command prompt, change directories to `/var/opt/OV/OpC/bin/cmds/`.
- 2 Run the `tib-operation` script using the following parameters:
  - WSDLLocation of the MO
  - NamespaceURI of the PortType
  - PortType Name
  - OperationName

For example:

```
Tib-operation
http://ovw001.cup.hp.com:8888/?wsdl:objid=tibema://www.tibco.com/
ema/2005/01/mo/identity/ServiceInstance/tibtest1/TIBCOServers/ovw0
01.cup.hp.com-7500 http://schemas.hp.com/wsmf/2003/03/Foundation
ManagedObjectConfigurationPT GetName
```

## Monitoring and Data Collection

The Frontend Subagent uses metric definitions to capture TIBCO management data for use in OVPM or OV Service Reporter (i.e., when generating alarms, graphs, and reports). This section provides instructions on how to customize what data is collected and how to change the default collection behavior. A brief overview of the how metrics are defined is also provided.

### Metric Definition Configuration Files

Metrics are configured using two XML configuration files: MetricDefinitions.xml and UDMMetricDefinitions.xml (used for custom adapters) files. These files are located on the Frontend Subagent node in /var/opt/OV/conf/tib. The elements for these files are described below.

- **<MetricDefinitions>** – The MetricDefinitions element is the top-level element within the document. It contains one collection of metrics, consisting of one or more metric definitions.
- **<Metric>** – The Metric element represents one metric. Each metric has a unique ID. If a user-defined metric is an alarming, graphing or reporting metric, the metric ID must be "TIBSPI\_0xxx" where xxx is a number from 700 through 799. Otherwise, if the metric is used only within the calculation of another metric, the metric ID must begin with a letter (case-sensitive) and can be followed by any combination of letters, numbers and underscores. A Metric element contains one more source elements that represent the metric data source. Two data sources are supported: WSM and calculations. The following table lists attributes for the Metric element:

**Table 3-1: Metric Element Attributes**

Attribute	Type/Values	Required	Default	Description
id	ID	yes	N/A	The metric ID.
name	text	no	no	The metric name, used for graphing and reporting. The name can be up to 20 characters in length.
alarm	yes no	no	no	If yes, the metric value is sent to the agent using opcmon.
report	yes no	no	no	If yes, the metric value is logged for reporting.

Attribute	Type/Values	Required	Default	Description
previous	yes no	no	yes	If yes, the metric value is saved in a history file so that deltas can be calculated. If you are not calculating deltas on a metric, set this to "no" for better performance.
graph	yes no	no	no	If yes, the metric is logged for graphing.
description	text	no	""	A description of the metric.

- **<WSM>** – The WSM element is used when the data source of the metric is a TIBCO metric definition. The WSM element contains the following sub-elements:
  - **<MetricName>** – The TIBCO metric definition name.
  - **<ObjectTypeList>** – List of MO types that will have metric value collected.
  - **<ObjectIDList>** – List of MO instances that will have metric values collected.
- **<Calculation>** and **<Formula>** – The Calculation element is used when the data source of the metric is a calculation using other defined metrics. The Calculation element contains a Formula element whose content is a string that specified the mathematical manipulation of other metric values to obtain the final metric value. The metrics are referred to in the calculation expression by their metric ID. The collector can perform calculations that combine one or more metrics to define a new metric. The result of the calculation is the metric value. Calculations must use syntax as follows:
  - Operators supported are +, -, /, \*, and unary minus.
  - Operator precedence and associativity follows the Java model.
  - Parentheses can be used to override the default operator precedence.
  - Allowable operands are metric IDs and literal doubles.

A metric ID can refer to either a WSM metric or another calculated metric. Literal doubles can be specified with or without the decimal notation. The metric ID refers to the id attribute of the Metric element in the metric definitions document. The calculation parser also supports the following functions. All function names are lowercase and take a single parameter which must be a metric ID:

- **delta** – returns the result of subtracting the previous value of the metric from the current value.
- **interval** – returns the time in milliseconds that has elapsed since the last time the metric was collected.

The following example defines a metric whose value is the ratio (as expressed as a percent) of Metric\_1 to Metric\_3:

```
<Formula>(Metric_1/Metric_3)*100</Formula>
```

The following example could be used to define a metric that is a rate (number of times per second) for Metric\_1.

```
<Formula>(delta(Metric_1)/interval(Metric_1))*1000</Formula>
```

## Modifying Data Collection

By default, data is collected for all RVDs, JMS Servers and BWEngines. If you want to collect data for a subset of the RVDs, JMS Servers or BWEngines or a subset of the metrics, you need to modify the `/var/opt/OV/conf/tib/MetricDefinitions.xml` file. For example, if you only want to collect MissedPackets for RVD:

tibema://www.tibco.com/ema/2005/01/mo/identity/ServiceInstance/tibtest1/TIBCO Servers/ovw010-7500, you would change:

```
<Metric id="MissedPackets" alarm="no">
  <WSM>
    <MetricName>Missed Packets</MetricName>
    <ObjectTypeList>
      <ObjectType>
        http://www.tibco.com/ema/2005/01/mo/type/RVD
      </ObjectType>
    </ObjectTypeList>
  </WSM>
</Metric>
```

to:

```
<Metric id="MissedPackets" alarm="no">
  <WSM>
    <MetricName>Missed Packets</MetricName>
    <ObjectIDList>
      <ObjectID>
        tibema://www.tibco.com/ema/2005/01/mo/identity/
        ServiceInstance/tibtest1/TIBCO Servers/ovw010-7500
      </ObjectID>
    </ObjectIDList>
  </WSM>
</Metric>
```

## Collecting Data for Custom Adapters

The following procedure allows you to collect data and metrics for custom adapters

- 1 On the Frontend Subagent node:
 

```
cd /var/opt/OV/conf/tib
cp UDMMetrics-sample.xml UDMMetricDefinitions.xml
```
- 2 Add the metrics for the custom adapters to the `UDMMetricDefinitions.xml` file.
- 3 Bring up the HP Resource Explorer on the custom adapter MO. Follow the instructions in the “Launching the HP Resource Explorer” section.
- 4 Click the + next to the custom adapter MO.
- 5 Write down the value for the **Type** property. This information is required for a later step.
- 6 Click the + next to **ManagedObjectMetricInterface**.
- 7 Right-click **MetricValues** and select **Open** from the popup menu.
- 8 Add each metric definition you want to alarm, graph, or report on into the `UDMMetricDefinitions.xml` file. For example:

Custom Adapter Type:

`http://www.tibco.com/ema/2005/01/mo/type/ServiceInstance/Adapter`

**Metric Definition Name:** A name for the definition.

**Messages Sent:** The total number of messages sent since the adapter was started.

**Message Drop Rate:** The percentage of messages dropped per second.

You want to graph the number of messages sent per collection interval and alarm on the message drop rate. Your `UDMMetricDefinitions.xml` file contains the following entries:

```
<Metric id="TIBSPI_0700" name="MessagesSent" alarm="no"
  graph="yes" report="no">
  <Calculation>
    <Formula>delta(MessagesSentInt)</Formula>
  </Calculation>
</Metric>

<Metric id="MessagesSentInt" alarm="no">
  <WSM>
    <MetricName>Messages Sent</MetricName>
    <ObjectTypeList>
      <ObjectType>http://www.tibco.com/ema/2005/01/mo/type
        /ServiceInstance/Adapter</ObjectType>
    </ObjectTypeList>
  </WSM>
</Metric>

<Metric id="TIBSPI_0701" name="MessageDropRate"
  alarm="yes" graph="no" report="no">
  <WSM>
    <MetricName>Message Drop Rate</MetricName>
    <ObjectTypeList>
      <ObjectType>http://www.tibco.com/ema/2005/01/mo/type
        /ServiceInstance/Adapter</ObjectType>
    </ObjectTypeList>
  </WSM>
</Metric>
```

9 Make sure `/var/opt/OV/bin/OpC/monitor` is in your `PATH`.

10 Create the UDM data sources by running the following script:

```
cd to /opt/OV/bin
./tib-perl tib-create-udm-datasources
```

11 Stop the Frontend Subagent by running:

```
opcagt -stop -id 25 (for OVO 7.x)
ovc -stop -id TIB (for OVO 8.x)
```

12 Verify that the Frontend Subagent was really stopped by running:

```
ps -ef | grep java
```

If there's an entry with `rmiregistry 1651`, the Frontend Subagent is still up so run:

```
kill -9 <rmiregistry pid> <rmiregistry parent pid>
```

For example, if the output of `ps -ef | grep java` was:

```
root 5170 5168 0 08:46:32 ? 0:06
```

```
/opt/java1.4/bin/PA_RISC2.0/rmiregistry 1651
```

You need to run:

```
kill -9 5179 5168
```

**13 Start the Frontend Subagent by running:**

```
opcagt -start -id 25 (OVO-U 7.x)
```

```
ovc -start -id TIB (OVO-U 8.x)
```

It's assumed that you've already assigned the TIBSPI-Metrics-V1 template group to the Frontend Subagent node. Modify the TIBSPI-Collect-Mon-V1 template in the TIBSPI-Metrics-V1 template group to collect the UDM metrics.

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Message Source Templates window by selecting **Window | Message Source Templates**.
- 3 Double click SPI for TIBCO in the Template Groups frame on the left.
- 4 Select the TIBSPI-Metrics-V1 template group in the Template Groups frame on the left.
- 5 Select the TIBSPI-Collect-Mon-V1 template in the right frame and click the **Modify...** button.
- 6 In the Modify Threshold Monitor dialog, add the UDM metrics to the Monitor Program or MIB ID field. For example, if you added metrics TIBSPI\_0700, TIBSPI\_0701 and TIBSPI\_0702 to your UDMMetricDefinition.xml file, add '700-702' to the end of the Monitor Program or MIB ID field. Click the **OK** button.
- 7 Redeploy the templates by following the instructions in the "Deploying Templates to the Frontend Subagent Node" section.



The metrics values are not logged to the data source until the TIBSPI-Metric-V1 template group is assigned and deployed. See "Assigning Template Groups to the Frontend Subagent Node" and "Deploying Templates to the Frontend Subagent Node" in this Chapter.

## Configuring Multi-Instance Metric Data

This procedure configures a metric has multi-instance data. The metric is configured using the MetricDefinitions.xml file. The UDMMetricDefinitions.xml file is used for custom adapters. For information on graphing instance metric data, see the "Graphing Instance Metric Data in OVPM" section later in this chapter.

To configure a metric that has multi-instance metric data modify the MetricDefinitions.xml file and add the following:

- For the metric that has multi instance data, add `instanceType="multi"` to the `<WSM>` tag.
- If you want the metric to only match certain instances, add `<InstanceList>` with `<Instance>` values.

- If you want the metric to match all instance values, don't include the `<InstanceList>` tag. For example:

### Example: All Instance Data (Collect)

```
<Metric id="WSFSPI_0011" name="SpotAll" alarm="yes" graph="yes"
  report="yes">
  <WSM instanceType="multi">
    <MetricName>getColorAssignmentCount-Count</MetricName>
    <ObjectTypeInfo>
      <Type>http://www.tibco.com/ema/2005/01/
        mo/type/ServiceInstance/Custom </Type>
    </ObjectTypeInfo>
  </WSM>
</Metric>
```

Metric WSFSPI\_0011 is used to collect all instance data for the TIBCO Spot application. The `getColorAssignmentCount-Count` operation returns the count for each color.

```
return[0].row[0].Color = red
return[0].row[0].Count = 5
return[1].row[0].Color = black
return[1].row[0].Count = 1
return[2].row[0].Color = green
return[2].row[0].Count = 0
return[3].row[0].Color = orange
return[3].row[0].Count = 0
return[4].row[0].Color = darkGray
return[4].row[0].Count = 0
return[5].row[0].Color = pink
return[5].row[0].Count = 0
return[6].row[0].Color = yellow
return[6].row[0].Count = 2
return[7].row[0].Color = blue
return[7].row[0].Count = 9
return[8].row[0].Color = lightGray
return[8].row[0].Count = 0
return[9].row[0].Color = gray
return[9].row[0].Count = 0
return[10].row[0].Color = cyan
return[10].row[0].Count = 0
return[11].row[0].Color = magenta
return[11].row[0].Count = 2
```

The metric is configured to:

- Call `opcmon` for each instance with object as `<servername>:<instancename>` and the following options: `servername`, `instancename`, and `serverhost`.
- Log data for graphs. For graphs, only one value is logged and it's the sum of the instance values. This is a limitation in the data collector. For the customer to graph instance data, they actually need to use the data from the reports. For the above example, the graph data logged is 19 for `Spot-ovw022` (server name).
- Log data for reports. Data is logged for each instance. For the above example, the report data logged is:



red	5
black	1
green	0
orange	0
darkGray	0
pink	0
yellow	2
blue	0
lightGray	0
gray	0
cyan	0
magenta	2

### Example: Specific Instance Data (Collect)

```
<Metric id="WSFSPI_0012" name="SpotBlue" alarm="yes" graph="no"
  report="no">
  <WSM instanceType="multi">
    <MetricName>getColorAssignmentCount-Count</MetricName>
    <ObjectTypeList>
      <ObjectType>http://www.tibco.com/ema/2005/01
        /mo/type/ServiceInstance/Custom</ObjectType>
    </ObjectTypeList>
    <InstanceList>
      <Instance>blue</Instance>
    </InstanceList>
  </WSM>
</Metric>
```

Metric WSFSPI\_0012 is used to collect the count for blue for the TIBCO Spot application. The metric is configured to call `opcmon` for the blue instance and no graph or report data is logged.

### Configuring a Threshold for Multi-Instance Metric Data

Configuring a threshold on multi instance metric data is achieved using a monitor template. The ISV developer creates a monitor template with the same name as the metric id. The developer then adds conditions for each instance. A Condition that matches all instances can also be added. For example:

For the WSFSPI\_0011 metric, the ISV developer creates a monitor template with the following information:

- Monitor Name: WSFSPI\_0011
- Monitor: External
- Condition: WSFSPI\_0011: Magenta
  - Object Pattern: magenta
  - Threshold: 1
  - Severity: minor
  - Message Text: <\$OPTION(instancename)> count (<\$VALUE>) too high (>=<\$THRESHOLD>)
  - Service Name: <\$OPTION(serverhome)>
- Condition: WSFSPI\_0011: All

- Object Pattern: <\*>
- Threshold: 4
- Severity: warning
- Message Text: <\$OPTION(instancename)> count (<\$VALUE>) too high (>=<\$THRESHOLD>)
- Service Name: <\$OPTION(serverhome)>

Assuming that opcmn is called with the values defined in the All Instance Data (Collect) example and the WSFSPI\_0011 monitor template is deployed. The following messages would be in the OVO message browser:

- OVO message
  - Severity: Min
  - Message Text: 'magenta' count (2.00) too high (>=1.00)
- OVO message
  - Severity: Warn
  - Message Text: 'red' count (5.00) too high (>=4.00)
- OVO message
  - Severity: Warn
  - Message Text: 'blue' count (9.00) too high (>=4.00)

## Collection Data for Specific Metrics

The TIBSPI-Collect-Mon-V1 monitor template is used to log the metric data for graphs. You can modify the template to collect data for a subset of the available metrics. For example, to collect data for metrics TIBSPI\_0001 – TIBSPI\_0004 and TIBSPI\_0026, follow these steps:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Message Source Templates window by selecting **Window | Message Source Templates**.
- 3 Double click SPI for TIBCO in the Template Groups frame on the left.
- 4 Select the TIBSPI-Metrics-V1 template group in the Template Groups frame on the left.
- 5 Select the TIBSPI-Collect-Mon-V1 template in the right frame and click the **Modify...** button.
- 6 In the Modify Threshold Monitor dialog, modify the Monitor Program or MIB ID field to contain:
 

```
Tib-perl -s TIBSPI-Collect-data -c TIBSPI-Collect-Mon-V1 -m 1-4, 26
```
- 7 Click the **OK** button in the Modify Threshold Monitor dialog.
- 8 Redeploy the templates by following the instructions in the “Deploying Templates to the Frontend Subagent Node” section.

## Monitoring Custom Adapter Metric Thresholds with OVO

To monitor custom adapter metric thresholds with OVO:

- 1 The metric needs to be in the `/var/opt/OV/conf/tib/UDMMetricDefinitions.xml` file on the Frontend Subagent node. The alarm attribute for the metric needs to be set to yes. If you are currently not collecting data for the metric, follow the instructions in the “Logging Metrics for Custom Adapters” section.
- 2 If you made modifications to the `UDMMetricDefinitions.xml` file in the above step then do the following:
  - a Remove the `/var/opt/OV/conf/tib/UDMMetricDefinitions.ser` file on the Frontend Subagent node.
  - b Stop the Frontend Subagent by running:
 

```
opcagt -stop -id 25 (OVO-U 7.x)
ovc -stop -id TIB (OVO-U 8.x)
```
  - c Start the Frontend Subagent by running:
 

```
opcagt -start -id 25 (OVO-U 7.x)
ovc -start -id TIB (OVO-U 8.x)
```
- 3 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 4 From any OVO window, bring up the Message Source Templates window by selecting **Window | Message Source Templates**.
- 5 Double click SPI for TIBCO in the Template Groups frame on the left.
- 6 Select the TIBSPI-Metrics-V1 template group in the Template Groups frame on the left.
- 7 Select TIBSPI\_0009 in the right frame. Click the **Copy...** button.
- 8 In the Copy Threshold Monitor dialog, update the Monitor Name and Description fields. The Monitor Name value must be the id value for the appropriate metric in the `/var/opt/OV/conf/tib/UDMMetricDefinitions.xml` file. For example, if you want to monitor the message drop rate and your `UDMMetricDefinitions.xml` file contains the following entry:

```
<Metric id="TIBSPI_0701" name="MessageDropRate"
  alarm="yes" graph="no" report="no">
  <WSM>
    <MetricName>Message Drop Rate</MetricName>
    <ObjectTypeList>
      <ObjectType>http://www.tibco.com/ema/2005/01/mo/type/
        ServiceInstance/Adapter</ObjectType>
    </ObjectTypeList>
  </WSM>
</Metric>
```

Use TIBSPI\_0701 as the Monitor Name value.

- 9 Click the **OK** button in the Copy Threshold Monitor dialog.
- 10 In the Message Source Templates window, click the **Conditions...** button.

- 11 Modify the conditions in the Message and Suppress Conditions dialog. Click the **Close** button.
- 12 Close the Message Source Templates window by selecting **Templates | Close Window**.
- 13 The TIBSPI-Collect-Mon-V1 template needs to be assigned and deployed on the Frontend Subagent node. The template also needs to be collecting the custom adapter metric's data.
  - a Refer to the instructions at the end of the “Logging Metrics for Custom Adapters” section on how to verify that the custom adapter metric's data is being collected.
  - b Run `optemplate` on the Frontend Subagent node. Refer to the “Assigning Template Groups to the Frontend Subagent Node” and “Deploying Templates to the Frontend Subagent Node” sections if the TIBSPI-Collect-Mon-V1 template is not in the returned list.

## Changing the Metric Data Collection Interval

To change the metric data collection interval, change the Polling Interval in the TIBSPI-Collect-Mon-V1 and TIBSPI-Graph-Mon-V1 templates. For example, to change the metric data collection from 5 minutes to 10 minutes, follow these steps:

- 1 Run the OVO admin GUI on the OVO management server and logon as `opc_adm`.
- 2 From any OVO window, bring up the Message Source Templates window by selecting **Window | Message Source Templates**.
- 3 Double-click SPI for TIBCO in the Template Groups frame on the left.
- 4 Select the TIBSPI-Metrics-V1 template group in the Template Groups frame on the left.
- 5 Select the TIBSPI-Collect-Mon-V1 template in the right frame and click the **Modify...** button.
- 6 In the Modify Threshold Monitor dialog, modify the Polling Interval from 5m to 10m. Click the **OK** button.
- 7 Select the TIBSPI-Graph-Mon-V1 template in the right frame and click the **Modify...** button.
- 8 In the Modify Threshold Monitor dialog, modify the Polling Interval from 5m to 10m.
- 9 Click the **OK** button.
- 10 Redeploy the templates by following the instructions in the “Deploying Templates to the Frontend Subagent Node” section.

## Changing the Metric Threshold Value

By default, there are a couple of monitor templates monitoring individual metrics data. For example, the TIBSPI\_0009 monitor template monitors RVD retransmitted packet rate. If the retransmitted packet rate is  $\geq 5$ , a message of major severity appears in the OVO message browser. If the retransmitted packet rate is  $\geq 2$  and  $< 5$ , a message of minor severity appears in the OVO message browser. To change to generate a minor severity message when the retransmitted packet rate is  $\geq 1$  and  $< 5$ , perform the following steps:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Message Source Templates window by selecting **Window | Message Source Templates**.
- 3 Double click SPI for TIBCO in the Template Groups frame on the left.
- 4 Select the TIBSPI-Metrics-V1 template group in the Template Groups frame on the left.
- 5 Select the TIBSPI\_0009 template in the right frame and click the **Conditions...** button.
- 6 In the Message and Suppress Conditions dialog, select the TIBSPI\_0009.2 condition and click the **Modify...** button.
- 7 Select the TIBSPI-Graph-Mon-V1 template in the right frame and click the **Modify...** button.
- 8 In the Modify Threshold Monitor dialog, modify the Threshold from 2 to 1. Click the **OK** button.
- 9 Close the Message and Suppress Conditions dialog by clicking on the **OK** button.
- 10 Close the Message Source Templates window by selecting **Templates | Close Window**.
- 11 Redeploy the templates by following the instructions in the “Deploying Templates to the Frontend Subagent Node” section.

## Changing which Logfile to Monitor

If you installed the TIBCO EMA Agent in *c:\tibco\ema* and TIBCO Hawk in *c:\tibco\hawk* then the templates work as is. Otherwise you need to modify the location of the logfile.

Follow these steps to change the location of the TIBCO Hawk logfile if TIBCO Hawk is installed in the same directory on all Windows Nodes:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Message Source Templates window by selecting **Window | Message Source Templates**.
- 3 Double-click SPI for TIBCO in the Template Groups frame on the left.
- 4 Double-click TIBSPI-Windows-V1 template group in the Template Groups frame on the left.

- 5 Select the TIBSPI-WIN-HAWK-Agent-V1 template group in the Template Groups from on the left.
- 6 Select the TIBSPI-Hawk-WIN-Log-V1 template in the right frame and click the **Modify...** button.
- 7 In the Modify Logfile dialog, modify the Logfile to contain the location of the TIBCO Hawk Agent logfile.
- 8 Click the **OK** button in the Modify Logfile dialog.
- 9 Redeploy the templates by following the instructions in the “Deploying the Template Group to the Windows Nodes” section.

## Reporting and Performance Graphs

This section provides instructions for using CODA for the purpose of generating reports and performance graphs. In addition, instruction for viewing reports and graphs in OV Reporter are also provided.

### Using CODA

Coda is a performance subagent that is bundled for free with the OVO agent in OVO for UNIX 7.0 and higher. It is a light-weight performance agent comparable to MeasureWare. CODA only holds 5 weeks worth of data where as MeasureWare can potentially hold years worth of data. The CODA Database is stored in the <OVAgentDataDir>/databases directory. There is a coda.db and coda##### logs.

To check if CODA is running, from the command prompt run:

```
opcagt -status -id 12 (OVO-U 7.x)
```

```
ovc -status -id coda (OVO-U 8.x)
```

The output should verify that the Performance Agent /opt/OV/bin/coda is running. You can also run the CODA utility program. On HP-UX, the command is

```
/opt/OV/bin/codautl -support (OVO-U 7.x)
```

```
/opt/OV/bin/ovcodautl -support (OVO-U 8.x)
```

The result is a list of the last logged interval for all of the standard metrics and their values.

### CODA Logging

Coda keeps up to 5 weeks of data. Every Sunday at 12:00am (midnight), a new log file is created. Coda will continue to create a new logfile each week until it has accrued 5 weeks of logfiles. When the sixth logfile is created, the oldest file is deleted. The CODA log file (coda.log) is located in <OVAgentDataDir>/log/. For example on HP-UX the file is /var/opt/OV/log/coda.log. To check if CODA is logging data use the following procedure:

- 1 Open /var/opt/OV/log/coda.log

- 2 At the end of the coda.log file, you should see the Staring message, which files were opened, deleted and/or created and finally the "Waiting for requests..." message. As coda logs data, the timestamp for the newest coda##### log changes.

## View TIBCO SPI Reports

The steps in this section demonstrate how to use HP OpenView Reporter to gather TIBCO SPI Metric data and generate TIBCO SPI reports. The section is intended only as a “quick start” reference and does not represent a replacement for the Reporter documentation.

To view TIBCO SPI reports:

- 1 From the Windows Taskbar, select **Start | Programs | HP OpenView | reporter | Reporter**.
- 2 From the left tree, right-click **Discover Area** and select the **Add Single System** command. The Add Single System dialog box displays.
- 3 In the System field, enter the full DNS name of the computer where the Frontend Subagent is installed (i.e., hostname.mycompany.com).
- 4 Click **Add**. The Reporter’s discovery program runs, discovers the system, and automatically gathers metric data (collected by either MeasureWare, or CODA) from the system.
- 5 From the Main toolbar, click the **Generate Reports** button. The TIBCO SPI reports are generated. This may take several minutes to complete.
- 6 From the Main toolbar, click the **Show Reports** button. A browser displays and lists all of the TIBCO SPI reports. The reports are organized into 4 categories that show the metric data over different time ranges: TIBCO Full Range, TIBCO Last Full Month, TIBCO Last Full Week, and TIBCO Yesterday.

## Defining a User Friendly Name for an MO

You can define a user friendly name for the MOs that appears on TIBCO graphs and reports. The name is also used as the name of the file where the data is logged for graphs before it’s sent to MeasureWare/CODA. Therefore, the name must be a valid file name.

In order to use the OV Service Reporter reports that compare network and TIBCO RVD performance metrics, a user friendly name for each RVD is automatically defined in the following format as part of the out-of-box solution:

```
<RVD fully qualified host name>--<RVD port>--RVD
```

For example:

```
ovw010.cup.hp.com-7500-RVD
```

To define a user friendly name:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.

- 3 In the Application Bank window, double-click the **TIBCO SPI Tools** application group.
- 4 In the Application Group: TIBCO SPI Tools window, right-click **Configure TIBCO SPI application** and select **Execute** from the popup menu.
- 5 From the GUI, click the icon next to **Front End Configuration**.
- 6 Click **Report Information**.
- 7 In the menu bar, select **Edit | New**.
- 8 Use the **ReportGroupObjectID** and the **ReportGroupName** fields to enter the group object ID and the group name. Remember that the **ReportGroupName** must be able to be used as a valid file name. For example:

Assume:

The WSDL location for the RVD using port 7500 on ovw010.cup.hp.com is  
`http://ovw001.cup.hp.com:8888/?wsdl=objid=tibema://www.tibco.com/ema/2005/01/mo/identity/ServiceInstance/tibtest1/TIBCOServers/ovw010-7500`

Then:

`ReportGroupObjectID =`  
`http://ovw001.cup.hp.com:8888/?wsdl=objid=tibema://www.tibco.com/ema/2005/01/mo/identity/ServiceInstance/tibtest1/TIBCOServers/ovw010-7500`

`ReportGroupName = ovw001.cup.hp.com-7500-RVD`

- 9 Continue to add new entries for each MO.
- 10 Select **File | Save**.
- 11 Select **File | Exit**.
- 12 Click the Close button in the Output of Application window.

## Monitoring Performance Metrics with OVPM

As mentioned earlier, you can monitor performance using OVPM. For detailed instructions on OVPM, see the OVO documentation.



The following procedures must be completed in the order listed.

### Configure OVPM

On the system where OVPM is installed, add the Frontend Subagent host name into the `<install_dir>/Data/systemCODA.txt` file if using CODA or `<install_dir>/Data/systemMWA.txt` if using MeasureWare. For more information on using CODA, see the “Using CODA” section below.



For example, if OVPM were installed on ovw001.cup.hp.com in C:\Program Files\HP OpenView, then the Frontend Subagent is running on ovh001.cup.hp.com. If you're using coda, then you need to add ovh001.cup.hp.com into the C:\Program Files\HP OpenView\Data\systemCODA.txt file on ovw001.cup.hp.com.

## Create a Bytes Sent OVPM Graph for RVDs

To create an OVPM graph (bytes sent) for an RVD:

- 1 Start the Performance Manager GUI by Select **Start | Programs | HP OpenView | performance manager | Performance Manager**.
- 2 In the Performance Manager home page, click the **Web Forms Interface** button.
- 3 Click the **Design** tab and enter a title.
- 4 Select the Frontend Subagent node as the Data Source.
- 5 Select a Date Range. You can use 1 Hour if you're testing recently logged data.
- 6 Select **Ending Now** if you're testing recently logged data.
- 7 Select **TIBSPI\_METRICS:TIBSPI\_METRICS** as the class.
- 8 Select **B003\_BYTESSENT** as the metric and enter a label.
- 9 Scroll up and select **SERVERNAME** as the Metric Filter.
- 10 Select **=** and Fill in the RVD name you want in the graph. To figure out the RVD name, you can look in the /var/opt/OV/conf/tib/SiteConfig file on the Frontend Subagent node.
- 11 Scroll down and enter **TIBCO** as the User Template File.
- 12 Enter a name for the Save Graph Name.
- 13 Click the **Save Graph** button.
- 14 Scroll up and click the **Draw Graph** button.

## View the RVD Bytes Sent OVPM Graph

To view the RVD bytes sent OVPM graph:

- 1 Start the Performance Manager GUI by Select **Start | Programs | HP OpenView | performance manager | Performance Manager**.
- 2 In the Performance Manager home page, click the **Web Forms Interface** button.
- 3 Select the **Display** tab if it is not already selected.
- 4 Select **UserTIBCO** for the Template File.
- 5 Select the RVD graph for the Graph Name.
- 6 Select the Frontend Subagent node in the System Names list box.
- 7 Select the desired date range information
- 8 Click the **Draw Graph** button

## Graphing Instance Metric Data in OVPM

OVPM is used to graph instance metric data. When graphing metric data, you need to use the report data source and specify the appropriate values for `SERVERNAME` and `OBJECTNAME` in the Metric Filter.

To graph instance metric data in OVPM

- 1 Start the Performance Manager GUI by Select **Start | Programs | HP OpenView | performance manager | Performance Manager**.
- 2 In the Performance Manager home page, click the **Web Forms Interface** button.
- 3 Click on the **Custom** Tab.
- 4 Enter the following information:

- **Data Source** – select the frontend system. If you don't see your frontend system, add it into either systems CODA.txt or systemsMWA.txt in C:\Program Files\HP OpenView\Data\. Click the **Custom** Tab again.
- **Date Range** – to see the latest data, select 1 Hour in Date Range and select the **Ending Now** option box.
- **Metric Filter** – select “SERVERNAME” and “=”. Enter the name of the MO you used in the SiteConfig file as `SERVER<#>_NAME` then `&&OBJECTNAME=<instance name>`. For example:

If we want to graph the blue count for the Spot application and our SiteConfig contained the following entry:

```
SERVER2_NAME=Spot-ovw022
```

```
SERVER2_HOME=tibema%3A%2F%2Fwww.tibco.com%2Fema%2F2005%2F01%2Fmo%2Fidentity%2FServiceInstance%2Ftibtest%2FDefaultDeployment%2FSpot-ovw022
```

```
SERVER2_PORT=0
```

The Metric Filter value would be:

```
Spot-ovw022&&OBJECTNAME=blue
```

- **Class** – select `<SPName>_RPT_METRICS:<SPName>_RPT_METRICS`.
  - **Metric** – select VALUE.
  - **Label** – Enter in a label name.
- 5 Click Draw Graph (near the top of the page).

## TIBCO SPI Self Management

The TIBCO SPI is capable of managing itself and all Frontend and Backend processes. The SPI has templates that need to be pushed to that node for this feature to be activated. In addition, the TIBCO SPI also monitors the EMA agent process. For this EMA agent monitor feature to be activated, the host where the EMA agent is running needs to be a managed node, and the templates to monitor the process need to be pushed to that node. Refer to “Appendix C: List Templates and Reports” for a detailed reference of the individual templates.

# Modify Logging and Tracing Levels

This section provides instruction for changing log and trace levels for the TIBCO SPI. Logging and tracing levels can be changed for both the Frontend Subagent and the Backend Service and are useful for debugging and auditing purposes.

## Changing Log Levels

The TIBCO SPI supports two log levels (`ERROR` and `INFO`) for both the Frontend Subagent and Backend Service log messages. By default the log level is set to `INFO` for both the Frontend and the Backend. Log levels can be customized by modifying the log property files. The property files are

- `/var/opt/OV/log/tib/frontend.properties`
- `/var/opt/OV/log/tib/backend.properties`

### Changing Frontend Log Levels

To change the Frontend's log level from `INFO` to `ERROR`:

- 1 Open the `/var/opt/OV/conf/tib/Frontend.properties` file.
- 2 Set the `java.util.logging.FileHandler.level` to `ERROR`. For example:  

```
java.util.logging.FileHandler.pattern = %h/frontend.log
java.util.logging.FileHandler.level = ERROR
```
- 3 Save and close the file.
- 4 Restart the TIBCO SPI. The Frontend log file is `/var/opt/OV/log/tib/frontend.log`

### Changing Backend Log Levels

To change the Backend's log level from `ERROR` to `INFO`:

- 1 Open the `/var/opt/OV/conf/tib/Backend.properties` file.
- 2 Set the `java.util.logging.FileHandler.level` to `ERROR`. For example:  

```
java.util.logging.FileHandler.pattern = %h/backend.log
java.util.logging.FileHandler.level = ERROR
```
- 3 Save and close the file.
- 4 Restart the TIBCO SPI. The Backend log file is `/var/opt/OV/log/tib/backend.log`

## Changing Trace Levels

The TIBCO SPI provides a mechanism that collects and stores all trace data in a trace file. By default, tracing is configured to show `INFO` messages. To get a more detail level of trace messages the trace level has to be set to `FINE`. Trace levels can be customized by modifying the trace property files. The property files are:

- /var/opt/OV/log/tib/frontend.trace
- /var/opt/OV/log/tib/backend.trace

## Changing Frontend Trace Levels

To change Frontend's trace levels from `INFO` to `FINE`:

- 1 Open the `/var/opt/OV/conf/tib/Frontend.properties` file.
- 2 For the Frontend trace file, set the `java.util.logging.FileHandler.level` value to `FINE`. For example:  

```
java.util.logging.FileHandler.pattern = %h/frontend.trace
java.util.logging.FileHandler.level = FINE
```
- 3 Save and close the file.
- 4 Restart the TIBCO SPI. The Frontend trace file is `/var/opt/OV/log/tib/frontend.trace`.

## Changing Backend Trace Levels

To change the Backend's trace levels from `INFO` to `FINE`:

- 1 Open the `/var/opt/OV/conf/tib/Backend.properties` file.
- 2 For the Backend trace file, set the `java.util.logging.FileHandler.level` to `FINE`. For example:  

```
java.util.logging.FileHandler.pattern = %h/backend.trace
java.util.logging.FileHandler.level = FINE
```
- 3 Save and close the file.
- 4 Restart the TIBCO SPI. The Backend trace file is `/var/opt/OV/log/tib/backend.trace`.

## Using Service Effect Analysis (Optional)

This chapter provides instructions for using the Service Effect Analysis (SEA) component and creating event definitions using the service composer tool (ECS composer). The SEA component, including the ECS tool, has a separate installation than the TIBCO SPI. In addition, refer to the *CorrelationComposer Guide* included with the NNM documentation.

### Overview

The SEA component provides advanced manageability features. The SEA component enables the end user to monitor and manage a TIBCO environment, by receiving correlated management events that combine infrastructure and TIBCO application level alerts. The SEA component is an add-on to the TIBCO SPI and is not required to use the standard management features of the TIBCO SPI.



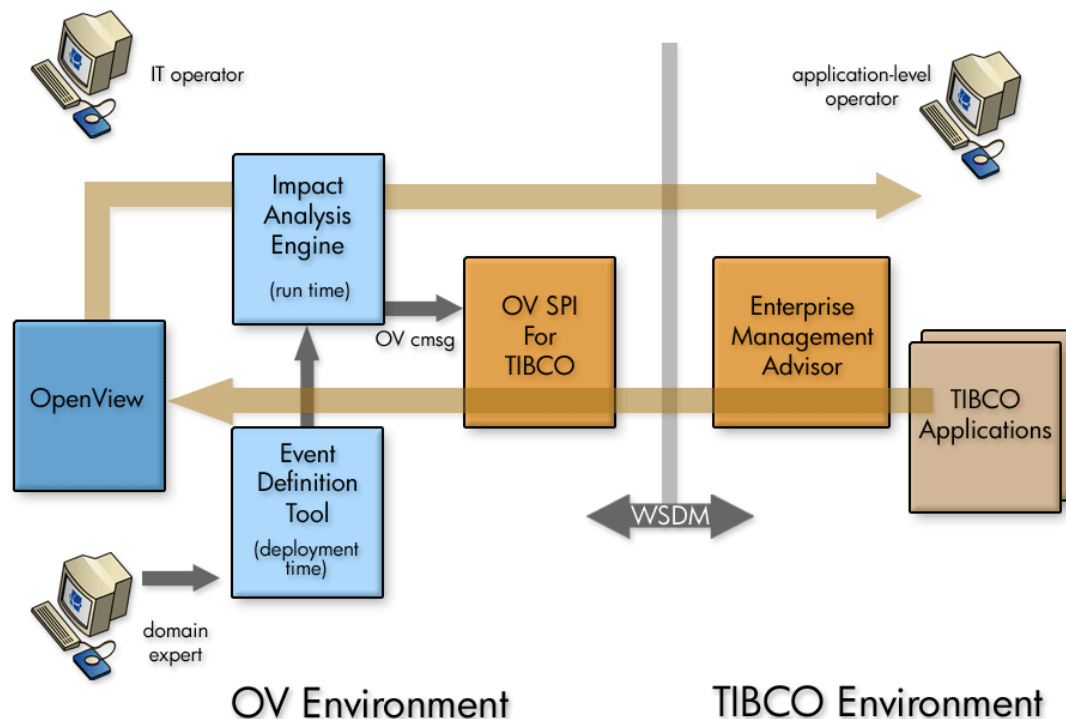
The SEA Component must be installed on the OVO Management Server system.

The SEA component has both deployment time and run time components. During deployment time, domain experts use the rules/correlation definition tool (Service Composer) to define the higher level rules that are eventually triggered by the events as they are generated at run time.

The various User/Actor roles in this solution architecture are as follows:

- **Network Operator** – views the solution from a basic management perspective. (The typical OV Network Operator)
- **Domain Expert** – defines the higher-level SEA events at deployment time.
- **Application manager** – receives and views run time SEA events and employs adaptive management in the enterprise.

Figure 4-1 shows a high-level conceptual view of the SEA system:



**Figure 4-1: Conceptual View of SEA**

## Service Composer Tool

SEA Events are created by the domain expert using the OV ECS Composer tool. These events are then uploaded into the ECS engine and enable context sensitive, higher level alerts to be generated at run-time. These alerts are WSDM-based alerts that are subscribed to by the end user, who can then take appropriate corrective action.

The OV ECS Composer comes packaged with an ECS circuit and a graphic user interface to parameterize and define correlation rules to perform event correlation. It provides various pre-defined correlation models. The various types of correlations are as follows:

- Blacklist correlation is used in order to discard a specific category of alarms.
- Enrich correlation is used when information in an alarm is insufficient. This type allows add/delete/modify of event attributes.
- Multi-Source correlation is used to identify a relationship between arbitrary alarms, potentially from different sources that together form a logical set that identifies a problem.
- Flapping correlation is used to block events that are a logical consequence of another event.
- Rate correlation is used to measure the number of events occurring in a defined window of time.
- Repeated correlation is used to suppress duplicate alarms arriving within a specified period of time.

## Requirements

The following table lists the software that is required to run the SEA component and perform service effect analysis.

**Table 4-1: SEA Requirements**

Requirement	Suggested Minimum
Disk Space on the OVO Management Server	100 MB
OS	HP-UX 11.x Solaris SPARC 8.0
Java	JRE 1.4.x or later <ul style="list-style-type: none"> <li>HP-UX: <a href="http://www.hp.com/go/java">http://www.hp.com/go/java</a></li> </ul>
Management Software	OVO-U 7.1x (including JavaGUI) NNM 6.41
Tomcat	4.1.24 or later <ul style="list-style-type: none"> <li><a href="http://jakarta.apache.org/tomcat/">http://jakarta.apache.org/tomcat/</a> (Tomcat 5 is not currently supported)</li> </ul>
Correlation Composer 3.2 (bundled with NNM)	HP-UX 11.x, Solaris 2.8/5.8/8.0
ECS OVO patch	HP-UX: PHSS_29270 for 10x, and PHSS_29971 for 11.x Solaris: PSOV_03296
Axis 1.1	HP-UX 11.x, Solaris 2.8/5.8/8.0

## Prerequisites

The following prerequisites are required before installing the SEA component:

- The user should have some general idea of OVO, ECS, NNM and some basic understanding of HP OpenView SPIs.
- You will need some sort of X session, to view the ECS GUI. If you want to run in a telnet session, you will need to export the DISPLAY to run it in GUI mode.
- The SEA installation will install in a fixed directory. (The directory is /opt/OV/nonOV/SEA.) Ensure that there is enough space in the folder before starting your install.
- The SEA solution works with certain versions of OV and third party products. To verify the versions of the software installed on your system, here are a few pointers:
  - JRE 1.4: From the command prompt, type `java -version`.

- For the Open view related suite of products: from the command prompt, use the `swlist` command.
  - To see the GUI use `swlist -i`.
  - To see the command line output, use `swlist`.
  - For a list of all the filesets that are installed on your system, use `swlist -R`. (You could also use this command along with `grep` to find out if a particular product/fileset/patch is installed on your system. For example, `swlist -r | grep PHSS_29722`.)
- For the SEA installation, it is imperative to have super user privileges because the SEA installation includes the installation of certain OV components, which requires a start and stop of OV services.
- The SEA component is configured to use port 4444. If you need to change the default port, use the SEA configuration tool. Re-start the SEA components after a new port has been defined. See “Running the SEA Configuration Tool” in this chapter.

## Apache Tomcat Setup

Install Tomcat Server. Make sure not to install to a directory with any spaces in the path. Set the Environment variable `CATALINA_HOME` to the directory where Tomcat is installed, henceforth referred to as `%CATALINA_HOME%`.

It is assumed that your environment is set to:


- `host=localhost` - This can be left localhost, or may be changed for your machine name.
- `port=8080` – The port to start Tomcat on. Edit the entries in `%WSMF_HOME%/axis/build.properties` to match your environment, if different.
- `port=8005` – The port to shutdown Tomcat on. Edit the entries in `%WSMF_HOME%/axis/build.properties` to match your environment, if different.



To configure the Tomcat shutdown port, please do an SEA Deploy, and then go into the `%WSMF_HOME%/axis/tomcat_base/conf` and edit the `server.xml` to change that port. This change has to be made in this file each time after SEA deploy is run. See “Running the SEA Component”.

## Environment Setup

This section describes setting up the required system environment variables. The following variables must be setup:

- `%WSMF_HOME%` = `/opt/OV/nonOV/SEA` (The WSMF Smart Business Agent is an internal component of the SEA module, which enables the creation of a WSDM-based channel).
- 
 The WSDM implementation in the TIBCO SPI is based on an HP-authored preliminary version of WSDM known as the Web Service Management Framework (WSMF).
- `%JAVA_HOME%` = the location of JRE installation.



- Please ensure that /opt/OV/bin is in the \$PATH.

## Running the SEA Installer

SEA is installed using an InstallAnywhere wizard. Before starting the install make sure:

- opc services and OVO is operational. (Opcsv –status should show all services as running.)
- Log in to OVO as *opc\_adm*.
- Log in with super user privileges.

To install SEA:

- 1 Change directories to /opt/OV/bin.
- 2 From this directory, type:

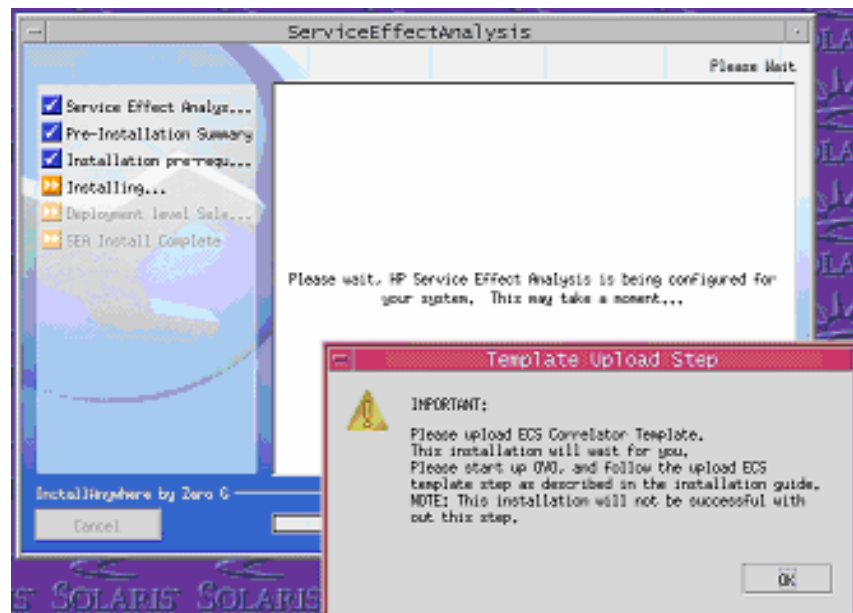
```
sh ./ SEAInstallation.bin
```

The InstallAnywhere wizard steps through the installation process. Follow the installation instructions and accept the default settings.

When prompted by the Template Upload Step dialog box (shown below), complete the following steps before continuing with the installation. This manual step is required to upload SEA related templates that enable the creation and correlation of the high level SEA events.



The SEA installer waits until you finish the manual steps.



- 3 Log in to OVO as *opc\_adm*.
- 4 From the Menu bar in the VPO Node bank, select **Actions | Server | Assign Templates**.

- 5 Click **open template window**.
- 6 Select **Correlation Composer Template**.
- 7 From the Template Configuration window, click **Get Template Selections**.
- 8 Click **OK**.
- 9 From the Menu bar in the VPO Node bank, select **Actions | Server | Install/Update server templates**.
- 10 Return and finish the installation. To verify if the above steps were successful, check the `/var/opt/OV/conf/OpC/mgmt_sv` directory. It should contain an `ecs_comp.eco` file.

## Running the SEA Component

The following procedures must be completed in the order in which they are presented.

### Enabling MSI

You must enable MSI before you can run the SEA component:

- 1 Log in to OVO as `opc_adm`.
- 2 From the Menu bar in the VPO Node bank, select **Actions | Server | Configure**.
- 3 From the Configure Management Server Window, select the following options:
  - enable output
  - send all messages to Server MSI
  - Divert messages
- 4 Click **OK**.

### Starting the SEA Component

To Start the SEA:

- 1 Run the OVO admin GUI on the OVO management server and log on as `opc_adm`.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank | TIBCO SPI Tools**.
- 3 Click the **Deploy SEA** icon. (This step only needs to be performed once after installation.)
- 4 Click the **Start Tomcat** icon.
- 5 Click the **Start SEA** icon.
- 6 Verify that the SEA has been deployed by pointing a browser to the wsmf services. For example, <http://localhost:8080/wsmf/services>. It should show all the WSDLs' of the services available to the SEA component.

## Running the SEA Configuration Tool

The SEAConfiguration tool is used to configure various aspects of the SEA configuration file. To launch the SEA configuration tool:

- 1 Run the OVO admin GUI on the OVO management server and log on as *opc\_adm*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 Click the **SEA Configuration Tool** icon in the TIBSPI application bank. The SEA Configuration Tool launches.

## Stopping the SEA Component

Stopping the SEA component includes stopping Tomcat as well. This will affect any other applications that are running on the same instance of Tomcat.

To stop SEA, use the following command:

- 1 Run the OVO admin GUI on the OVO management server and log on as *opc\_adm*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 Double-click on the StopSEA icon. A window displays and gives the status of the application after it has stopped. This operation may take some time to complete.

If SEA does not shutdown properly, type `ps -ef | grep java` and do a `kill -9` on the process id.



You can also stop SEA using the following command:  
`/opt/OV/bin/stopSEA.sh`

## Creating an Event Definition

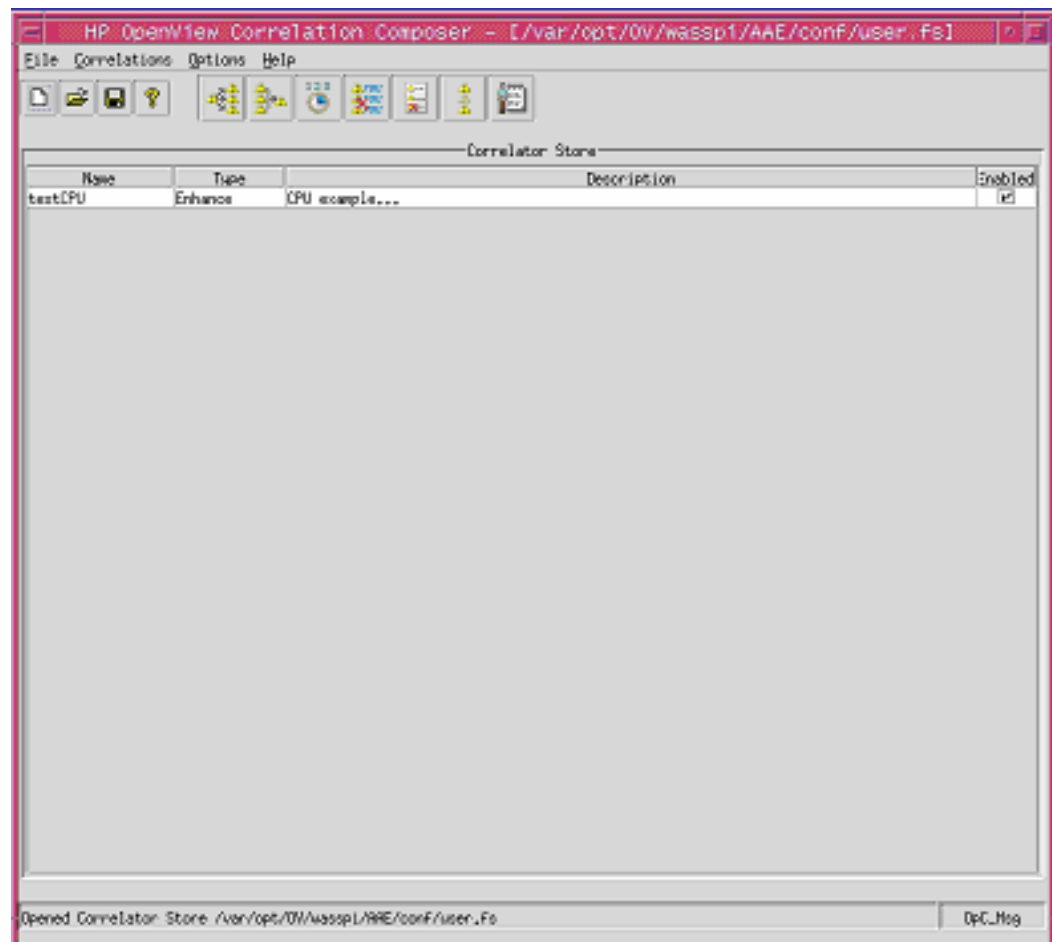
This section demonstrates the steps involved in creating an event definition. The demonstration creates an event definition for a CPU\_HIGH message generated by OV as part of the Smart Plug-in. The predefined ECS Composer enrichment correlation model has been chosen for this example.



Always use the user.fs file enclosed with the SEA Solution to create your correlators. Do not update the testCPU correlator. (This correlator is used to check the validity of the installation etc.)

To create an event definition:

- 1 From the OVO Application Bank, click the SEA Event Definition icon. The HP OpenView Correlation Composer displays.



- 2 Specify the attributes that will identify the raw event. In the example, the CPU event that arrives at the engine is identified by OBJECT=CPU.
- 3 In the advanced filter, use the following:  
`AIA_EVENT != Constants.IS_AN_AIA_EVENT`
- 4 Create and assign a variable called **name**. In the example, the event name is "testCPU". This variable has to be assigned to the AIA\_EVENT\_NAME attribute in the new alarm definition.

**Enhance**

Name: testCPU

Definition   **New Alarm**   CallBacks

**Alarm Signature**

Field	Operator	Value
OBJECT	matches	"CPU"

**Variables**

Name	Type	Value
name	Constant	"testCPU"

**Advanced Filter**

Name	Operator	Value
AIA_EVENT	!=	CONSTANTS.IS_AIA_EVENT

**Parameters**

Want Original ☐   Enhance Always ☐

OK   Cancel   Help

- 5 In the new alarm section, select **New Alarm Selection**. Fill in the fields of the new alarm. Make sure that AIA\_EVENT is set to **CONSTANTS.IS\_AIA\_EVENT** and assign a variable for the **AIA\_EVENT\_NAME**. These fields need to be set otherwise the event will not be forwarded to OV SBA.
- 6 Select the **Feedback** button (shown below) to allow the event to be fed-back into the correlation engine.

Enhance

Name: testCPU

Description Definition **New Alarm** CallBacks

New Alarm Specification ▼

Alarm No1

**New Alarm Definition**

Name	Value
AIA_EVENT	CONSTANTS.IS_AIA_EVENT
AIA_EVENT_NAME	name
APPLICATION	APPLICATION
GROUP	GROUP
MSGTEXT	MSGTEXT
OBJECT	OBJECT
NODENAME	NODENAME
SERVICE_NAME	SERVICE_NAME
SEVERITY	SEVERITY

New Previous Next Delete

☒ Feedback

OK Cancel Help

- 7 Save the correlator in the same user.fs and exit the OVComposer GUI. The custom correlator is now deployed.
- 8 To configure SEA, use the OVO application bank SEA Configuration Tool application, then use the deploy SEA, followed by Start SEA to start SEA.



Refer to “Appendix B: SEA Configuration” for details on the SEA Configuration properties.

Proceed to the TIBCO component whose higher level management alerts you have factored in to your custom correlator, and configure the rule bases for it. This enables the relevant TIBCO event to be generated. Please refer to the appropriate TIBCO product manual for instructions.

At run time, when the specified sequence of management alerts occur, the SEA event is generated.

The TIBCO SEA component that has subscribed to these alerts receives and forwards this SEA event on the RVD Bus. Any TIBCO or other higher level applications that are listening on that subject is then capable of receiving this event and taking corrective action. See the TIBCO documentation for more information about how applications can listen for SEA events on the RVD bus.

## SEA Logging and Trace Information

The SEA component utilizes the Apache Commons Logging Framework and Log4J as the logging system. The Logging feature is used to emit log messages. Log messages are used to record component activity and troubleshoot problems.

When you install the SEA component, the logging feature is preconfigured. You can change the level of detail you require. The logging feature is configured through a properties file:

/opt/OV/nonOV/SEA/wsmf-sba/internal/newconfig/log4j.properties.



Logging affects performance. Different logging settings discussed in this section should only be used for troubleshooting when you are in a test environment.

## Changing Logging Levels

A log level is used to constrain a log messages based on the type of information that you want to emit. Logging levels are set in the logging properties file. By default, only warning (WARN) messages are emitted. Table 4-2 describes each of the logging levels.

**Table 4-2: Logging Levels**

Logging Level	Description
<b>DEBUG</b>	Emits output that is used to track events that are occurring in the server. The output includes problems and non-problems.
<b>INFO</b>	The default setting. Emits output that is used to track any non-problem events.
<b>WARN</b>	Emits output that is used to track errors that have occurred but that do not stop the Network Services from continuing.
<b>ERROR</b>	Emits output that is used to track errors that may result in reduced functionality.

To change logging levels:

- 1 Stop the SEA component if it is currently started.
- 2 Using a text editor, open /opt/OV/nonOV/SEA/wsmf-sba/internal/newconfig/log4j.properties.
- 3 Set a new logging level for all packages or for a specific package. For example:

```
log4j.rootCategory=DEBUG, STDOUT_MSG_ONLY, ROLL_FILE
```

Or

```
log4j.category.com.hp.ovms=DEBUG
```



Levels assigned for a specific package overrides the overall log level

- 4 Save and close the properties file.
- 5 Restart the SEA Component.

## Changing the MO Update Interval

By default, managed objects are dynamically updated (i.e., added or deleted) every five minutes. You can change this default value by assigning a new polling frequency

To change the polling frequency:

- 1 Using a text editor, open `/opt/OV/nonOV/SEA/wsmf-sba/internal/newconfig/axis/sea.xml`.
- 2 In the `<DiscoveryObjects name="SEA-OVSBA">` node, change the `<PollFrequency>` element to a value in Milliseconds.
- 3 Save and close the file.
- 4 Run Deploy SEA and Start SEA from the Application bank.

## Uninstalling the SEA Component

To uninstall the SEA Component:

- 1 Log in to OVO as *opc\_adm*.
- 2 From the Menu bar in the VPO Node bank, select **Actions | Server | Assign Templates**.
- 3 Select **Group Correlation Composer** from the list.
- 4 Click **Remove Templates**.
- 5 From the Menu bar in the VPO Node bank, select **Actions | Server | Install/Update server templates**.
- 6 From a command prompt change directories to `/opt/OV/nonOV/SEA` directory and type `sh ./ Uninstall_HP_Service_Effect_Analysis`.

Files that are placed in `/opt/OV/bin/SEA` directory while using SEA, either by the user or by the SEA Component (e.g., log files) will not be deleted. After the uninstallation is complete, you must manually delete such files.



If the installation aborts prematurely, the installation does not uninstall itself before quitting. If the uninstaller was copied over to the filesystem, you can follow the uninstallation steps. Otherwise, manually delete the SEA directory.



## Using the NNM Integration (Optional)

This section describes how to enable the TIBCO SPI NNM integration and how to configure its management features.

### Overview

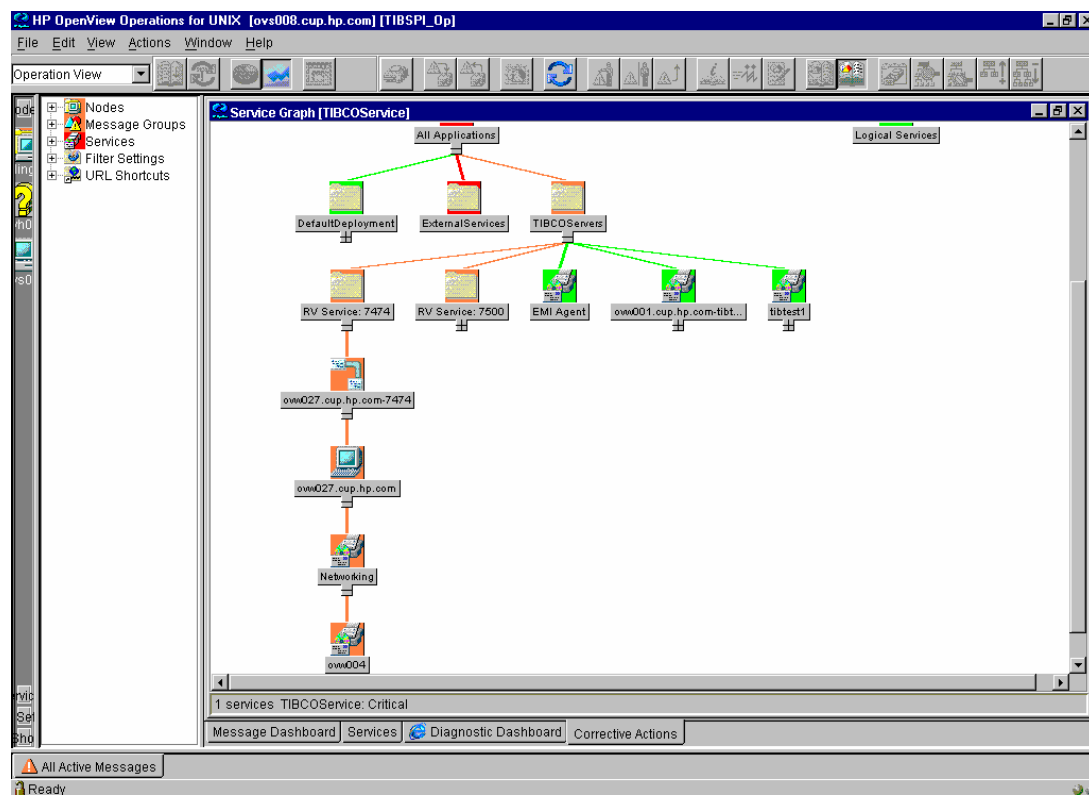
The TIBCO SPI contains an NNM integration that enables an operator to view the connection between managed applications and the network infrastructure. Application data is obtained from TIBCO EMA; while, the network infrastructure data is retrieved from OpenView NNM. The integration provides the following features:

- Service Views of Network Dependencies
- Collecting SNMP Trap Events
- Performance Reports

### Service Views of Network Dependencies

The Service Navigator's Service viewer provides a graphical view of the TIBCO SPI service map and shows an application's network dependencies. The Service viewer contains the networking device directly connected to systems running the TIBCO RVD or RVRD. Typically, this device is a switch.

The following screen capture shows the service view of applications and their underlying infrastructure. In the service view, a switch named `ovw004` is connected to system `ovw027.cup.hp.com`, which is running a TIBCO RVD.



## Collecting SNMP Trap Events

Network events are forwarded from NNM to OVO and update the status of network devices in the TIBCO SPI service map. Some examples of NNM SNMP trap events which might be forwarded to OVO include:

- <\$MIB\_OBJECT> threshold exceeded. Sampled high of <\$HIGH\_VALUE> at <\$TIME> low of <\$LOW\_VALUE> at <\$TIME>
- <\$MIB\_OBJECT> threshold rearmed. Sampled high of <\$HIGH\_VALUE> at <\$TIME> low of <\$LOW\_VALUE> at <\$TIME>
- Agent up with Possible Changes (coldStart Trap) ...
- Agent up with No Changes (warmStart Trap) ...
- Agent interface down (linkDown Trap) ...
- Agent interface up (linkUp Trap) ...

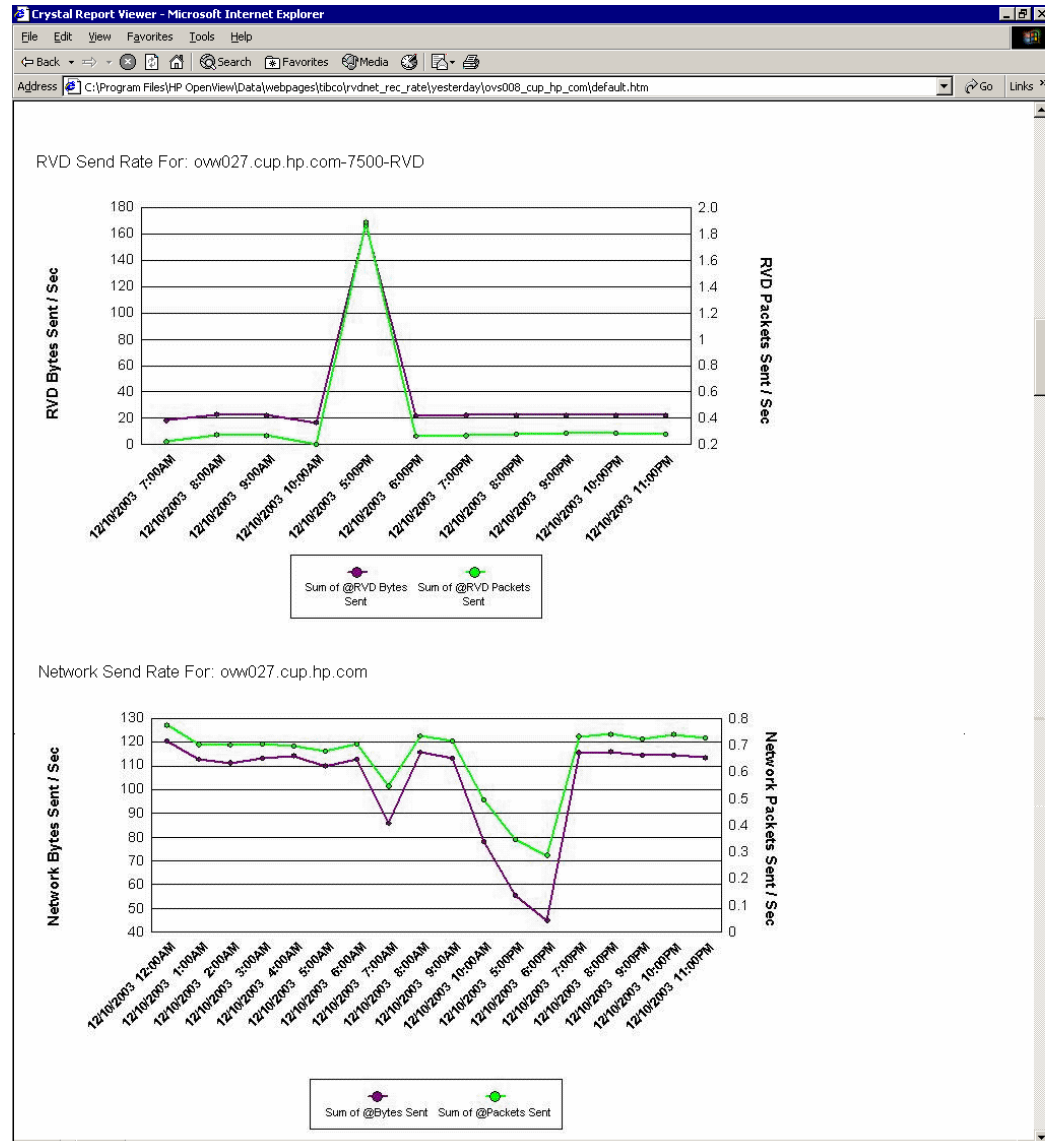
## Performance Reports

Lastly, the NNM integration allows an operator to generate and view OV Service Reporter reports that compare network and TIBCO RVD performance metrics. These reports include:

- TIBCO RVD & Network Interface Receive Rates (Yesterday, Last Full Week, Last Full Month, and Full Range)

- TIBCO RVD & Network Interface Send Rates (Yesterday, Last Full Week, Last Full Month, and Full Range)
- TIBCO RVD Retransmissions & Network Interface Errors (Yesterday, Last Full Week, Last Full Month, and Full Range)

The following screen capture is an example performance report generated through Service Reporter.



## Requirements

The TIBCO SPI and NNM integration has several software dependencies. It is assumed that OVO and the TIBCO SPI have been installed and are running properly.

**Table 5-1: NNM Integration Requirements**

Requirement	Required Version
NNM	6.4.1 with the following patch: HP_UX – PHSS_30031: s700_800 11.x NNM6.4x/ET2.0x Solaris – PSOV_03301: sparc SOL 2.x OV NNM6.4x/ET2.0x NNM 7.01 is required when using OVO 8.x NNM 7.5 not supported.
NNM Extended Topology (ET)	2.01
OV Reporter	3.5 or later MS Access is not supported as the OV Reporter database (MSDE, MS SQL Server, and Oracle are supported).

## General Setup Requirements

The following lists general setup instructions that must be performed when using the NNM integration. In addition, consult the individual component's documentation for installing components such as NNM, NNM ET, and OV Reporter.



If you install OVPA (MeasureWare) after starting the TIBCO SPI front-end, you need to restart the front-end if you want network metrics logged to OVPA instead of CODA.

- NNM must be installed on the OVO management server.
- Enable SNMP agents on network devices and switches. SNMP agents allow NNM to discover the network topology.
- Perform an NNM discovery.
- Install NNM ET on the OVO management server where NNM is installed.
- Perform an NNM ET discovery.
- Change the TIBSPI\_Op user or user profile to include responsibility for the SNMP message group and the TIBSPI-External node group.
- Install and distribute the SNMP 6.20 Trap template on the system running OVO and NNM. If you are already using a custom template for NNM traps, make sure this template is distributed to the system running OVO and NNM. You can use an OVO tool, /opt/OV/bin/OpC/utlis/ovtrap2opc, to upload customized NNM SNMP traps into OVO.

- Follow the instructions in chapter 2 for installing the TIBCO SPI reports. If you are unfamiliar with OV Reporter, see the “View TIBCO SPI Reports” section in Chapter 3.
  - Assign the following per-system reports: TIBCO RVD & Network Interface Receive Rates (FR, LFM, LFW, and Y), TIBCO RVD & Network Interface Send Rates (FR, LFM, LFW, and Y), TIBCO RVD Missed Packets (FR, LFM, LFW, and Y), and TIBCO RVD Retransmissions & Network Interface Errors.

## Enabling and Configuring the NNM Integration

The TIBCO SPI NNM integration is disabled by default.

To enable and configure the NNM integration:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_admin*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 In the Application Bank window, double-click the **TIBCO SPI Tools** application group.
- 4 In the Application Group: TIBCO SPI Tools window, right-click **Configure TIBCO SPI** application and select **Execute** from the popup menu.
- 5 From the application’s left navigation panel, expand **Networking Info** to view the NNM integration parameters.
- 6 Select **Network Monitoring**.
- 7 In the right panel, complete the fields. See Appendix A for a description of the parameters.
- 8 Select **Network Reporting**.
- 9 In the right panel, complete the fields. See Appendix A for a description of the parameters.
- 10 Select **File | Save** to save your changes to the configuration file.
- 11 Select **File | Exit**.
- 12 In the Output of Application window, verify the message “TIBCO SPI configuration completed”. It may take some time for the message to appear. Don’t close the Output of Application window until you see the completion message.
- 13 Click the **Close** button in the Output of Application window.

## NNM Topology Integration Settings

The NNM Integration utilizes OpenView Interconnect (OVI). OVI is bundled with the SPI and is part of the standard installation. OVI is installed with default settings that may not be applicable for your system. These settings include the OVI JAVA setting and the OVI port setting.

## Changing the OVI JAVA Setting

To change the OVI JAVA setting:

- 1 Open `var/opt/OV/conf/tib/ICO_SPI_TIBCO.env`
- 2 Change the line after the JAVA executable to point your JRE installation. For example:  
`JAVA=<your jre / bin location>`
- 3 Save and close the file.

## Changing the OVI Port Setting

The default OVI port is set to 16835. If this port is already being used on your system, you must change the port number.

To change the OVI Port setting:

- 1 Open `/var/opt/OV/conf/tib/WCConfig.xml`.
- 2 Change the `<OviPort>` number.
- 3 Save and close the file.
- 4 Open `/var/opt/OV/conf/tib/ResponderProxyPluglet.http.config`
- 5 Change the port number to the same port number you defined in `WCConfig.xml`.
- 6 Save and close the file.
- 7 Restart the TIBCO SPI.

## RVD Data Collection

You must be collecting RVD data to be able to compare the TIBCO RVD and network performance metrics. If you are not already collecting RVD data, follow the steps in the “Collecting Metric Data” and “Creating Data Sources for Logged Metric Values” sections.

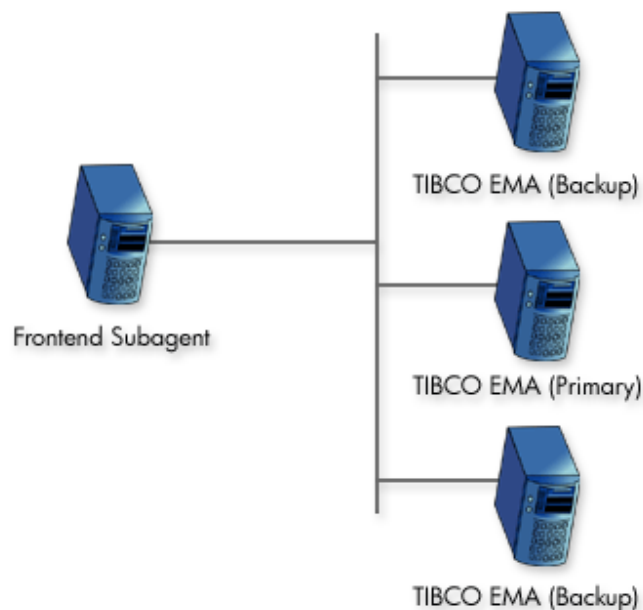
# Implementing Failover

This chapter provides instructions for implementing failover between the Frontend Subagent and 2 or more TIBCO EMA components.

## Overview

The Frontend Subagent can be configured to use multiple EMA installations. This allows the Frontend Subagent to continue collecting management data of a TIBCO environment even when a TIBCO EMA stops responding. The failover solution depends on multiple installations of the TIBCO EMA.

Specifically, the Frontend subagent uses one or more Location Candidates. Each candidate contains WSDL locations for managed objects that are exposed by a particular EMA. The first candidate is considered the primary candidate. Any subsequent candidates are considered backups for the primary candidate.



**Figure 6-1: OpenView-TIBCO Business Management General Architecture**

When the Frontend Subagent starts, it tries the first location candidate. If the location candidate responds, it is used. If there is no response, the Frontend tries the next location candidate. This process is repeated until a location candidate responds. For example, you could provide two location candidates:

```
http://hostA.domain.com:8888/?wsdl:objid=tibema://www.tibco.com/ema/  
2005/01/mo/identity/Domain/tibtest
```

```
http://hostB.domain.com:8888/?wsdl:objid=tibema://www.tibco.com/ema/  
2005/01/mo/identity/Domain/tibtest
```

In this example, the EMA on hostA is serving as the primary EMA and hostB is the backup. If hostA fails, then the EMA on hostB becomes the primary. When a failover takes place, the Frontend performs the following actions:

- Discards all managed object information for the EMA on hostA
- Discovers all managed objects for the EMA on hostB
- Reconstructs the service map
- Subscribes the notifications to the EMA on hostB

## Adding Multiple Location Candidates

Managed object WSDL locations are configured using the TIBCO SPI Configuration Editor, which is available from the TIBCO SPI Tools application group. If you want to implement failover, you must provide multiple location candidates for the same WSDL location.

To add multiple location candidates:

- 1 From the TIBCO SPI Configuration Editor, click the icon next to **Top of tree**.
- 2 Click the icon next to **Front End Configuration**.
- 3 Select **Managed Object Information**.
- 4 Select **WsdL Location Information**.
- 5 In the menu bar, select **Edit | New**.
- 6 In the Location Candidates list, add additional WSDL locations for the managed objects you want the Frontend Subagent to monitor.



If the WSDL location uses HTTPS, then you must configure the Frontend Subagent's security settings. Please refer to the “Configuring HTTPS Communication” section in Chapter 7 “Security Features and Configuration”.

- 7 Select **File | Save** to save your changes to the configuration file.
- 8 Select **File | Exit**.
- 9 In the Output of Application window, you should see the message ‘TIBCO SPI configuration completed.’ It may take a few minutes for the message to appear. Don’t close the Output of Application window until you see the completion message.
- 10 Click the **Close** button in the Output of Application window.



# Security Features and Configuration

This chapter provides instructions for securing the management channels that are used by the TIBCO SPI. Knowledge of SSL and HTTPS security principals are required to complete some of the instructions in this chapter.

## Overview

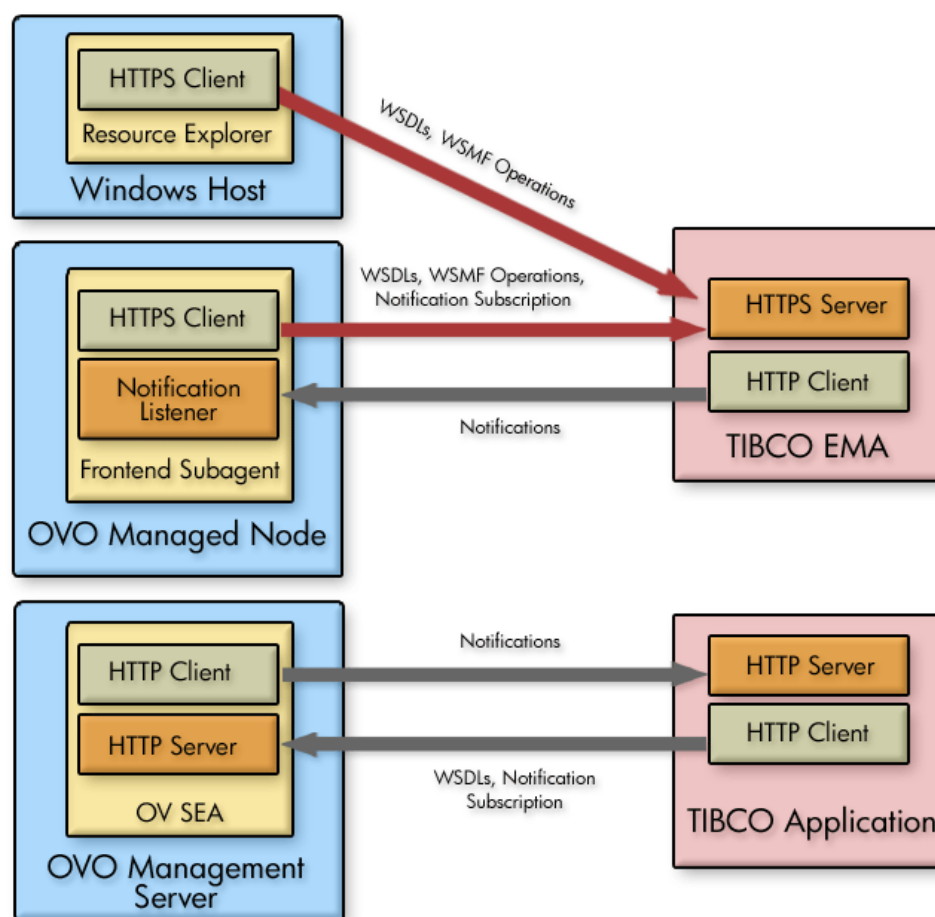
The TIBCO SPI security features are used to prevent unauthorized access to the TIBCO Managed Resources exposed by TIBCO EMA. This ensures that only trusted/authorized users are able to perform any actions on TIBCO Managed Resources in the managed environment. These features do not focus on data protection (i.e., data being passed to the TIBCO SPI from the TIBCO EMA as part of Event Notifications).

## Conceptual Architecture

There are various communication channels between the TIBCO EMA and TIBCO SPI components as shown in Figure 7-1. These communication channels are secured using HTTPS (SSL over HTTP) to ensure that the TIBCO Resources exposed by EMA are securely accessed from various components of the TIBCO SPI.

The standards supported for HTTPS include:

- X.509v3 for the certificate format
- SSLv3/TLS1.0
- Cipher suites available from JDK 1.4 JSSE library



**Figure 7-1: Communication Channel Security**

## What is Secured

As shown in Figure 7-1, HTTPS is used to secure the TIBCO resources and any operations exposed to these managed resources through the EMA WSMF channel.

## What is not Secured

OV SEA exposes to any WSMF client correlated events and root cause analysis information, but does not expose any operations that affect changes in OVO. This channel is not secured using HTTPS.

The TIBCO SPI also has a notification listener to which TIBCO EMA sends asynchronous notifications. This channel is not secured using HTTPS.

Since in both of these instances there is a possibility of “denial of service” attacks, future releases of the TIBCO SPI may secure these channels based on customer feedback.

## Current Limitations

The security features of the TIBCO SPI have the following limitations:

- Notifications from TIBCO SPI (OV SEA) can be subscribed by any client including TIBCO EMA over HTTP. The HTTPS protocol is not supported for this communication channel.
- Notifications from TIBCO EMA are received by the Frontend Subagent over HTTP. The HTTPS protocol is not supported for this communication channel.
- An Open View operator can perform all operations exposed for various Managed Resources from the Resource Explorer after successfully authenticating itself to TIBCO EMA for HTTPS communication. There is no mechanism to establish roles that allow only a specific set of operations to be accessible to a specific operator depending upon the role of the operator, or only allow access to limited set of Managed Resources, etc...
- If the Keystore that contains the client certificate and the associated private key for the Frontend Subagent also contains certificates for other entities (this may be applicable for an enterprise where all the certificates used by various applications are stored in a central Keystore), then the TIBCO SPI requires that all the private keys associated with corresponding certificates are protected by the same password. Otherwise, the TIBCO SPI will not be able to recover the private key for the Frontend Subagent from this Keystore.

## Configuring HTTPS Communication

This section provides instructions for implementing HTTPS communication for the Frontend Subagent and the Resource Explorer. If you do not currently have a Keystore or Truststore, instructions are also included for creating a Keystore and Truststore as well as importing signed certificates.

### Configuring the Frontend Subagent

Digital certificates are used for mutual authentication for HTTPS communication between the Frontend Subagent and TIBCO EMA. The certificate for the Frontend is stored in a Keystore. The default location of the Keystore is `/var/opt/OV/conf/tib/frontend.ks` but can be changed to a different location. By default the Keystore format is assumed to be JKS but other Keystore types can be used. See the “Customizing HTTPS configuration Parameters” section below.



TIBCO EMA uses the term “Identity File” when referring to a Keystore. The term Keystore in this documentation is the same as Identity File in the EMA documentation.

The CA's root certificate for TIBCO EMA is stored in the JDK Truststore. The default location of the Truststore is `/$JAVA_HOME/jre/lib/security/cacerts` but can be changed to some other location.

## Configuring SSL/HTTPS

To configure SSL/HTTPS for TIBCO SPI Frontend:

- 1 Start the TIBCO SPI Configuration Editor.
- 2 Under the Frontend Configuration node on the left panel, click **SSL Configuration**
- 3 On the right panel, enter the following information:

- **KeyStore Location:** The full path to a Keystore file. The Keystore contains the client certificate for the Frontend Subagent. The path is set by default to `/var/opt/OV/conf/tib/frontend.ks`.



The Keystore does not exist, but needs to be created using a tool described in the “Setting up a Keystore and Truststore for TIBCO SPI Frontend” section. If you have an existing Keystore, enter the path to your existing Keystore. When using an existing Keystore, the Frontend Subagent certificate must be imported into the Keystore.

- **Keystore Password:** The password to access the Keystore.
- **Private Key Password:** The password used to protect the private key.
- **TrustStore Location:** The file path to the TrustStore. The default path is set to `$JAVA_HOME/jre/lib/security/cacerts`.

- 4 Select **File** | **Save**.
- 5 Select **File** | **Exit**.

## Setting up a Keystore and Truststore for the Frontend Subagent

HTTPS communication for the Frontend Subagent requires the use of a Keystore and Truststore as well as a signed certificate. The setup is typically completed in the following manner:

- Obtain (from your preferred Certificate Authority) a client side certificate for the Frontend Subagent.
- Import this certificate to the Frontend Subagent's Keystore.
- Import the certificate of the CA who issued the TIBCO EMA certificate to the Frontend Subagent's Truststore if it does not already exist in the Truststore. You can get the list of all the CA certificates that are already imported to the Truststore by running the following command:

```
keytool -list -v -keystore <full path to Truststore file> -storepass
<truststorepassword>
```

If you have your own key management tool to create public key-private key pair and import certificates to an existing Keystore, then follow your procedure to obtain a client certificate for the Frontend Subagent and import it to the Frontend Subagent's Keystore. In addition, obtain the CA certificate of the issuer of the TIBCO EMA certificate and import it to the Frontend Subagent's Truststore. You will have to specify the alias name and the Keystore type used for the TIBCO SPI Frontend. See the “Customizing HTTPS configuration Parameters” section below.

If the Frontend Subagent Keystore does not exist or you do not have your own tool for managing Keystores, the following section provides instructions for creating a Keystore, getting a certificate, and importing certificates to the Keystore and Truststore.

## Creating a Keystore and Importing Certificates

A utility is provided that facilitates creating a Keystore and importing client-side certificates. The utility is typically used when you do not have an existing Keystore to use for the Frontend Subagent.

The utility is the same as the Java Keytool, but uses the information that is stored in WCCfg.xml (keystore location, keystore password, private key password and trust store location etc. ) and will use the default information specified in the Frontend.properties file (e.g aliasname, keystore type) while generating a key and importing certificates.

To create a Keystore and import certificates:

- 1 From a command prompt, change directories to /opt/OV/bin.
- 2 Enter the following command:

```
./tib-perl tib-keymgrutil -genkey -dname "<distinguished name>" [-keyalg <keyalg>] [-sigalg <sigalg>] [-keysize <keysize>]
```

- **dname**: This argument is entered in the form CN=<your name/hostname> OU=<org unit> O=<organization> L=<locality/city> ST=<state/province> C=<two letter country code>.
- **keyalg** (optional): The key algorithm to be used. If this argument is not specified, the default key algorithm is DSA.
- **sigalg** (optional): The digital signature algorithm to be used. If this argument is not specified, the default signature algorithm is SHA1withDSA if the key algorithm is DSA and MD5withRSA if the key algorithm is RSA.
- **keysize** (optional): if this argument is not specified, the default key size is 1024 bits.

For example:

```
./tib-perl tib-keymgrutil -genkey -dname "CN=tibfrontend OU=OV O=HP L=Cupertino ST=CA C=US"
```

This will create the frontend.ks (if that's the name specified for the Keystore) which contains the private key and the self signed certificate for Frontend Subagent.

- 3 Create a certificate-signing request into the specified CSR file.
- 4 Send the CSR file (typically emailed) to a CA, and save the reply from CA in some file <client\_cert\_file>. This file contains your client certificate.
- 5 Import the client certificate to the Truststore:

```
./tib-perl tib-keymgrutil -importcerts -clientcert <client_cert_file> -rootCA <root_cert_file> [-trustcert <trust_cert_file>]
```

- **clientcert:** This argument is the file returned from the CA for the Frontend Subagent's certificate, which can be a single certificate in X.509 format or a certificate chain in PKCS7 format. This argument imports the reply into the Keystore as specified during the SSL Configuration.
- **rootCA:** This argument is optional depending on the format of the client certificate.

This argument is required if the certificate reply is in X.509 format. In this case, you must import the Root Certificate of the CA who issued the Client certificate for the Frontend Subagent into the Keystore. This argument is the file that contains the Root Certificate of the CA

This argument is not required if the client cert is returned by the certificate authority in the PKCS7 format.

- **trustcert:** This argument is the certificate of the CA that issued the server certificate of TIBCO EMA. If the truststore that the user specified already has the CA certificate of the issuer of the TIBCO EMA certificate, this option is not required. Otherwise the CA certificate included in *<trust\_cert\_file>* will be imported to the Truststore specified during SSL Configuration. If the specified Truststore does not exist already, it will be created with a password “changeit”.

## Configuring the Resource Explorer

As mentioned in the Overview section, the communication channel between the Resource Explorer and the TIBCO EMA can be secured using HTTPS. It is recommended that you reuse the Frontend Subagent client certificate as your Resource Explorer client certificate so that you do not need to maintain a separate client certificate for the Resource Explorer. However, a section for creating a separate client certificate for the Resource Explorer is also provided in this section.

### Using the Frontend Subagent's Client Certificate

To use the Frontend Subagent client certificate for the Resource explorer:

- 1 Copy the Keystore file from the Frontend Subagent machine to the Resource Explorer machine and rename the Keystore file if required.



If you have an existing Keystore on the Resource Explorer machine which you want to use, you can import the client certificate you obtained for the Frontend Subagent into this existing Keystore using your preferred key management tool.

- 2 Stop the Resource Explorer if it is currently started.
- 3 Using a text editor, open the Resource Explorer script located at `%RE_HOME%\hp_resource_explorer\hp_resource_explorer.bat`.
- 4 In the script, specify the Keystore and Truststore locations for the Resource Explorer as JVM system parameters. The defaults are set to the security directory under your `%JAVA_HOME%` as below:

```
set JVM_ARGS=%JVM_ARGS% -Ddeployment.user.certs=
%JAVA_HOME%\jre\lib\security\client_certs
```

The `-Ddeployment.user.certs` argument refers to the full path to the Keystore file.

```
set JVM_ARGS=%JVM_ARGS% -
Ddeployment.system.certs=%JAVA_HOME%\jre\lib\security\cacerts
```

the `-Ddeployment.system.certs` argument refers to the full path to the Truststore file. In addition, the file `%JAVA_HOME%\jre\lib\security\client_certs` does not exist and needs to be created using some key management tool.



You will also have a chance to set / change these values in the security dialog when the Resource Explorer starts.

- 5 Save and close the Resource Explorer script.
- 6 Start the Resource Explorer.

## Using a Separate Client Certificate for the Resource Explorer

If you do not want to use the Frontend Subagent's client certificate for the Resource Explorer, you can use a separate client certificate for the Resource Explorer. It is assumed that you are using your own key management tool to perform these steps.

To use a separate client certificate for the Resource Explorer:

- 1 Obtain a client certificate for the Resource Explorer from your preferred Certificate Authority.
- 2 Import the client certificate for the Resource Explorer into the Keystore on the Resource Explorer machine. The Keystore file should be located at the location specified in the `hp_resource_explorer.bat` file as explained in the previous section.
- 3 Import the certificate of the CA who issued the TIBCO EMA certificate into the Truststore that is specified in the `hp_resource_explorer.bat` file as explained in the previous section. This step needs to be performed only if the CA certificate of the issuer of the TIBCO EMA certificate does not already exist in the Truststore. To verify if a CA certificate already exists in the Truststore, you can run the following command:

```
keytool -list -v -keystore <full path to truststore file> - storepass
<truststorepassword>
```

## Starting the Resource Explorer

When the Resource Explorer starts, if the URI to the root managed object uses the HTTPS protocol (`https://...`), a security dialog displays. Enter the following information:

- Password for the private key
- Keystore location
- Truststore location

Default locations of the Kestore and Truststore are shown in the dialog, modify the values as necessary.

## Customizing HTTPS Configuration Parameters

The `Frontend.properties` file that is located in the `/var/opt/OV/conf/tib` directory contains the following properties for customizing various SSL/HTTPS settings:

- `com.hp.wsmf.ssl.client.aliasname = <aliasname>`

This value is set to `tibFrontEnd` by default. If you use a different alias name for the Frontend Subagent in your Keystore, then specify that name.

- `com.hp.wsmf.ssl.keyStoreType = <type>`

This value is set to `JKS` by default. If you're using a Keystore of different type, specify the type here.

- `com.hp.wsmf.ssl.socketTimeout = <timeout in milliseconds>`

This value is set to `120000` by default.



If you get a `Connection Timeout Socket Exception` at runtime, increase this value.

- `com.hp.wsmf.certCheckWarnLevel1Days = <days>`

This value is set to `15` by default. If a certificate is found to expire within the days specified, a major severity message is sent to the OpenView Console's Message browser.

- `com.hp.wsmf.certCheckWarnLevel2Days = <days>`

This value is set to `30` by default. If a certificate is found to expire within the days specified, a warning severity message is sent to the OpenView Console's Message browser.



# Troubleshooting

This chapter provides common troubleshooting tasks when using the TIBCO SPI. In addition, refer to the *TIBCO SPI Release Notes* for the latest information about the TIBCO SPI.

## Runtime Problems

### Frontend Subagent Does Not Stop

The Frontend subagent doesn't stop if there is a problem canceling the subscription.

**Solution:**

You must manually stop the Frontend Subagent Java processes.

### Frontend Subagent Does Not Start

The Frontend Subagent is not starting up.

**Solution:**

To locate the error, look in the `/var/opt/OV/log/tib/frontend.log` file. Refer to the “Errors In Frontend Logfile” section later in this chapter for explanations on resolving the error.

### Frontend Subagent is Unable to Connect to EMA

The Frontend Subagent may fail to connect to EMA for various reasons. Depending on the specific scenario, The Frontend logs specific message into the `frontend.log` file when it fails to connect to EMA. The section below describes the various error messages logged into `frontend.log` when Frontend Subagent fails to connect to EMA:

```
WARNING WSF-1097: Problem accessing
https://ovw009.cup.hp.com:8888/?wsdl:objid=tibema://www.tibco.com/ema
/2005/01/mo/identity/Root, WSDLException: faultCode=OTHER_ERROR:
Unable to connect to server. Cause: Connection refused
```

**Solution:**

The reason for this error is the EMA Gateway is down. Make sure that EMA is started successfully.

```
SEVERE TIBSPI-105 Fronted Startup Failed : Exception: WSF-1202:
Failed to retrieve the Client Certificate for tibFrontEnd at the
KeyStore /var/opt/OV/conf/tib/frontend.ks. Exception: Keystore was
tampered with, or password was incorrect
```

**Solution:**

The reason for this error is the password for the Frontend KeyStore is wrong. Verify that you have specified the correct password for the Frontend KeyStore when specifying the SSL Configuration Settings.

```
SEVERE TIBSPI-105 Fronted Startup Failed : Exception: WSF-12105:
Failed to load the Keystore /var/opt/OV/conf/tib/frontend.ks.
Exception: Cannot recover key
```

**Solution:**

The reason for this error is the password for the Private Key is wrong. Verify that you have specified the correct password for the Private Key when specifying the SSL Configuration Settings.

```
WARNING WSF-1097: Problem accessing
https://ovw009.cup.hp.com:8888/?wsdl:objid=tibema://www.tibco.com/ema
/2005/01/mo/identity/Root, WSDLException: faultCode=OTHER_ERROR:
Unable to read WSDL document from
'https://ovw009.cup.hp.com:8888/?wsdl:objid=tibema://www.tibco.com/em
a/2005/01/mo/identity/Root':.
sun.security.validator.ValidatorException: No trusted certificate
found
```

**Solution:**

The reason for this error is the CA Certificate of issuer of EMA server certificate is not present in the TrustStore. Make sure to import the CA Certificate of issuer of EMA server certificate into the TrustStore that you specified in SSL Configuration Settings.

```
SEVERE TIBSPI-105 Fronted Startup Failed : Exception: WSF-1209:
Client Certificate for tibFrontEnd has expired on Mon Feb 21 15:33:21
PST 2005. Client Certificate has to be renewed for successful SSL
communication.
```

**Solution:**

The reason for this error is the certificate for Frontend has expired and requires to be renewed.

```
WARNING: WSF-1097: Problem accessing
https://ovw009.cup.hp.com:8888/?wsdl:objid=tibema://www.tibco.com/ema
/2005/01/mo/identity/Root, WSDLException: faultCode=OTHER_ERROR:
Unable to read WSDL document from
'https://ovw009.cup.hp.com:8888/?wsdl:objid=tibema://www.tibco.com/em
a/2005/01/mo/identity/Root'. :
java.security.cert.CertificateExpiredException: NotAfter: Fri Jan 07
16:01:27 PST 2005
```

**Solution:**

The reason for this error is the EMA Server Certificate has expired and requires to be renewed.

```
WARNING ><WSF-0004: Problem with push subscribe:
http://ovw009.cup.hp.com:8888/?wsdl:objid=tibema%3A%2F%2Fwww.tibco.co
m%2Fema%2F2005%2F01%2Fmo%2Fidentity%2FExternalService%2FtibcoDomain%2
Fovw009-C%3A%5CTibcoTestApp%5CDebugTest%5CWork; nested exception is:
java.net.SocketTimeoutException: Read timed out
```

**Solution:**

If Frontend is connecting to TIBCO EMA using HTTP, then edit the `/var/opt/OV/conf/tib/Frontend.properties` file and increase the value for `com.hp.wsmf.http.socketTimeout`. The time specified here is in milliseconds.

If Frontend is connecting to TIBCO EMA using HTTPS, then edit the `/var/opt/OV/conf/tib/Frontend.properties` file and increase the value for `com.hp.wsmf.ssl.socketTimeout`. The time specified here is in milliseconds.

## Backend Service is Not Found

After registering the Backend Service, the `ovstatus` and `ovstart` commands indicate that the “tib-backend” cannot be found.

**Solution:**

Stop and restart the OVO server:

```
ovstop
ovstart
```

## TIBCO SPI Uses Backup EMA Instead of Primary EMA

When using failover, the TIBCO SPI uses the backup TIBCO EMA instance instead of the primary TIBCO EMA instance.

**Solution:**

This occurs when the backup TIBCO EMA instance starts before the primary TIBCO EMA instance. Make sure all TIBCO EMA instances are fully started before starting the Frontend subagent.

## TIBCO Service is Not Visible

The TIBCO Service does not show up in the Resource Explorer.

### Solution:

Verify that the username you used to log into the HP VP Java Console is the same user name that's configured as `<OVOUserName>` in the `WCConfig.xml` file on the Frontend Subagent node.

## Missing Operational Notification

Operational notifications are not appearing in the OVO message browser.

**Solution:** Complete the following procedures.

### Verify TIBSPI-EventService-MSG-V1 Template is Configured

Verify that the TIBSPI-EventService-MSG-V1 template is configured on the OVO managed node where the Frontend Subagent is running.

- 1 From a window on the Frontend Subagent node, find out which templates are configured by running  

```
opctemplate
```
- 2 The list of configured templates is displayed. If the TIBSPI-EventService-MSG-V1 template is in the list, you can skip the rest of the steps in this task.
- 3 If you don't see the TIBSPI-EventService-MSG-V1 template, run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 4 Open the Node Bank window by selecting **Window | Node Bank**.
- 5 In the Node Bank window, select the node where the Frontend Subagent is running.
- 6 Select **Actions | Agents | Assign Templates...**
- 7 In the Define Configuration dialog box, click the **Add...** button.
- 8 In the Add Configuration dialog box, click the **Open Template Window...** button.
- 9 In the Message Source Templates dialog box, double-click the **SPI for TIBCO** template group in the Template Group from the left.
- 10 Click the **TIBSPI-EventService-V1** template group.
- 11 In the Add Configuration dialog box, click the **Get Template Selections** button.
- 12 In the Add Configuration dialog box, click the **OK** button.
- 13 Close the Message Source Templates dialog box.
- 14 In the Define Configuration dialog box, the Frontend node and TIBSPI-EventService-V1 template group should be in the list.
- 15 Click the **OK** button.

## Deploy the Templates

Deploy the templates for the Frontend node. You only need to do this if the TIBSPI-EventService-MSG-V1 template was not already configured.

- 1 In the Node Bank window, select the node where the Frontend Subagent is running.
- 2 Select **Actions** | **Agents** | **Install / Update SW & Config**.
- 3 In the Install / Update VPO Software and Configuration dialog box, select **Templates** in the Components frame.
- 4 Select **Force Update** in the Options frame.
- 5 Click the **OK** button.

## Verify OVO Messages Display in Message Browser

Verify that an OVO test message appears in the OVO message browser.

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm* or *TIBSPI\_Op*.
- 2 From a window on the Frontend Subagent node, send a test message by running
 

```
opcmsg severity=minor application=test object=test msg_text="my test" msg_grp=TIBCO
```
- 3 A message with minor severity should appear in the OVO message browser. If a message does not appear, follow the steps in the “Verify Creditials” section.

## Verify TIBCO EMA Agent is Sending Notifications

Verify that the TIBCO EMA Agent is sending the notification to the Frontend Subagent.

- 1 Turn debugging on for the EMA Agent. On the system where the EMA Agent is installed:
  - a Modify the <EMAAgentDir>/config/config.xml file. Uncomment <role>debugRole</role> at the end of the file.
  - b Restart the EMA Agent.
- 2 Either wait for the Frontend Subagent to detect that the EMA Agent has restarted or restart the Frontend Subagent.
- 3 After the Frontend Subagent has restarted, generate the operational notification. For example if you’re trying to see the status of an adapter reflected in the service map, stop the adapter if it’s already running.
- 4 Look in the <EMAAgentDir>/logs/emaagent.log file. Output similar to the following displays:

```
2004 Feb 11 10:19:28:174 GMT -8 Info [Application] EMA-35411 The
status of Managed Object
'tibema://www.tibco.com/ema/2005/01/mo/identity/ServiceInstance/ti
btest2/DefaultDeployment/Spot' changed from
'http://schemas.hp.com/wsmf/2003/03/Foundation/Status/Operational'
to 'http://schemas.hp.com/wsmf/2003/03/Foundation/Status/Inactive'
```

## Verify Frontend Subagent receives notifications

If the TIBCO EMA Agent is sending the notifications, we need to verify that the Frontend Subagent is receiving the notification.

- 1 Stop the Frontend Subagent.
- 2 Run the Frontend in a command window. On the system where the Frontend is installed, run:

```
cd to /opt/OV/bin
./tib-perl tib-start-frontend
```

The output is written to stdout. Wait until you see the message 'Service map created'.

- 3 Generate the operational notification. For example, start or stop an adapter.
- 4 The message 'Received an operation notification' is printed to stdout along with the information sent to opcmsg. For example if the sample Spot application is started, the following is printed:

Received an operation notification

```
opcmsg:
object=Spot
msg_grp=TIBCO
service_id=<some wsdl>
severity=Normal

msg_text=Status Change from
http://schemas.hp.com/wsmf/2003/03/Foundation/Status/Inactive to
http://schemas.hp.com/wsmf/2003/03/Foundation/Status/Operational
node=<some node>
application=ServiceInstance/Custom

-option
WSMF_EVENT_TYPE=http://www.tibco.com/ema/2005/01/event/
StatusChange

-option
MSG_CORR_ID=tibema://www.tibco.com/ema/2005/01/mo/identity/
ServiceInstance/tibtest2/DefaultDeployment/Spot
```

## Verify Resource Host Name

The node must be in the OVO Node Bank and in the TIBSPI-External or TIBSPI-UNIX node group.



The node value must be an exact match to the Hostname for the node in the OVO Node Bank. The node cannot be an IP address and it must be accessible through DNS. It does not work to put the host name in the /etc/hosts file. The name needs to be in the DNS tables.

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, open the Node Bank window by selecting **Window | Node Bank**.

- 3 The node must be a node in the Node Bank or a node in the Holding Area.
- 4 Close the Node Bank window.
- 5 From any OVO window, open the Node Group Bank window by selecting **Window | Node Group Bank**.
- 6 In the Node Group Bank window, double-click the **TIBSPI-External** node group.
- 7 If the node is in the TIBSPI-External node group window, close the window and skip the following steps.
- 8 If the node is not in the TIBSPI-External node group window, click the arrow pointing up to get back to the Node Group Bank window.
- 9 In the Node Group Bank window, double-click the **TIBSPI-UNIX** node group.
- 10 If the node is not in the TIBSPI-UNIX node group, add it to either the TIBSPI-UNIX or TIBSPI-External node group.

## Verify Source Object

The source object that's filled in as the `service_id` value in the Frontend output must be one of the objects that the Frontend Subagent recognizes.

- 1 Start the HP VP Java Console.
- 2 Click the + icon in front of the services until you get to the service that represents the object that is the `service_id` value
- 3 Right-click the service and select **Properties...** from the popup menu.
- 4 In the Services Properties [XXX] dialog box, the value in the Name field must be an exact match to the `service_id` value.
- 5 Click the **Close** button to close the dialog box.

## Verify Credentials

The OVO operator that you're logged on as is able to see messages in the TIBCO message group. We assume for this example that the OVO operator is `opc_adm`.

- 1 Run the OVO admin GUI on the OVO management server and logon as `opc_adm`.
- 2 From any OVO window, select **Window | User Bank**.
- 3 From the User Bank window, right-click `opc_adm` and select **Modify...** from the popup menu.
- 4 In the Modify User: `opc_adm` window, click the **Profiles...** button.
- 5 In the Profiles of User: `opc_adm` window, verify that the TIBSPI User Profile icon appears. If it's not in the window:
  - a Open User Profile Bank window by selecting **Window | User Profile Bank**.
  - b Drag the TIBSPI User Profile from the VPO User Profile Bank window and drop it into the Profiles of User: `opc_adm` window.
  - c Close the VPO User Profile Bank window.

- 6 Close the Profiles of user: opc\_adm window.
- 7 In the Modify User: opc\_adm window, click the **OK** button.
- 8 Restart the session. From any OVO window, select **Map | Restart Session**.
- 9 In the HP OpenView Windows WARNING dialog, click the **OK** button.

## Verify Communication with the OVO Management Server

If the Frontend is receiving the notification, you now need to check to see if there are communication problems between the Frontend node and the OVO management server. To do this, you need to turn on the OVO message tracing.



Refer to the OVO documentation for instructions on enabling OVO message tracing.

- 1 On the OVO management server, add `OPC_TRACE TRUE` and `OPC_TRACE_AREA MSG` to the `/opt/OV/bin/OpC/install/opcsvinfo` file. Your file should look similar to:

```
OPC_INSTALLED_VERSION A.07.10
OPC_MGMT_SERVER ovh072.cup.hp.com
OPC_MGMTSV_CHARSET iso885915
OPC_INSTALLATION_TIME 03/25/03 15:28:11
OPC_SG FALSE
OPC_TRACE TRUE
OPC_TRACE_AREA MSG
```

- 2 Run `/opt/OV/bin/OpC/opcsv -trace`. The trace information is written to `/var/opt/OV/share/tmp/OpC/mgmt_sv/trace`.
- 3 Run `tail -f /var/opt/OV/share/tmp/OpC/mgmt_sv/trace` to see the trace messages as they are written.
- 4 On the Frontend node, add `OPC_TRACE TRUE` and `OPC_TRACE_AREA MSG` to the `/opt/OV/bin/OpC/install/opcinfo` file.
- 5 Run `/opt/OV/bin/OpC/opcagt -trace`. The trace information is written to `/var/opt/OV/tmp/OpC/trace`.
- 6 Run `tail -f /var/opt/OV/tmp/OpC/trace` to see the trace messages as they are written.
- 7 Send a notification. For example, you can start or stop the TIBCO Spot sample application. In the OVO agent trace file (`/var/opt/OV/tmp/OpC/trace`), you should see some messages similar to:

```
02/11 12:18:20.517 opcmsg(9930:001) [MSG]: Queueing message: TIBCO
'Status Change from h'
02/11 12:18:20.523 opcmsgi(1791:001) [MSG]: Sending message:
6f732be6-5ccf-71d8-0be9-0ff414390000 TIBCO 'Status Change from h'
15.244.60.103
02/11 12:18:20.534 opcmsga(1778:001) [MSG]: Message/Act.Resp.
received from agents: 6f732be6-5ccf-71d8-0be9-0ff414390000 TIBCO
'Status Change from h' 15.244.60.103
```



```
02/11 12:18:20.535 opcmsga(1778:001)[MSG]: OpC mgr for msg:
6f732be6-5ccf-71d8-0be9-0ff414390000 TIBCO 'Status Change from h'
15.244.60.103 opcmgr : ovh072.cup.hp.com 15.244.20.57

02/11 12:18:20.535 opcmsga(1778:001)[MSG]: forwarding msg:
6f732be6-5ccf-71d8-0be9-0ff414390000 TIBCO 'Status Change from h'
15.244.60.103 opcmgr : ovh072.cup.hp.com 15.244.20.57

02/11 12:18:20.538 opcmsga(1778:001)[MSG]: Sending msg (len =
554): Status Change from http://schemas.hp.com/wsmf/2003/03/Founda

02/11 12:18:20.543 opcmsga(1778:001)[MSG]: Message forwarded:
6f732be6-5ccf-71d8-0be9-0ff414390000 TIBCO 'Status Change from h'
15.244.60.103 opcmgr : ovh072.cup.hp.com 15.244.20.57
```

In the OVO management server trace file

(/var/opt/OV/share/tmp/OpC/mgmt\_sv/trace), you should see some messages similar to:

```
02/11 12:18:20.542 opcmsgrd(1595:02f)[MSG]: Message received:
6f732be6-5ccf-71d8-0be9-0ff414390000 TIBCO 'Status Change from h'
15.244.60.103

02/11 12:18:20.543 opcmsgm(1611:001)[MSG]: Message received from
message receiver: 6f732be6-5ccf-71d8-0be9-0ff414390000 TIBCO
'Status Change from h' 15.244.60.103

02/11 12:18:20.552 opcmsgm(1611:001)[MSG]: Message processing:
ip_addr=15.244.60.103 (mapped), node_name='ovw022.cup.hp.com'

02/11 12:18:20.552 opcmsgm(1611:001)[MSG]: csm_db_msg_add called
with msg 6f732be6-5ccf-71d8-0be9-0ff414390000.

02/11 12:18:20.592 opcmsgm(1611:001)[MSG]: csm_db_msg_add finished
for msg 6f732be6-5ccf-71d8-0be9-0ff414390000. Last err: 0-0

02/11 12:18:20.593 opcmsgm(1611:001)[MSG]: Message forwarded to
DM: 6f732be6-5ccf-71d8-0be9-0ff414390000 TIBCO 'Status Change from
h' 15.244.60.103
```

- 8 Turn tracing off on the OVO management server by modifying OPC\_TRACE to FALSE in the /opt/OV/bin/OpC/install/opcsvinfo file.
- 9 Run /opt/OV/bin/OpC/opcsv -trace.
- 10 On the Frontend node, turn tracing off by setting OPC\_TRACE to FALSE in the /opt/OV/bin/OpC/install/opcinfo file.
- 11 Run /opt/OV/bin/OpC/opcagt -trace.

## Cleanup the OVO Message Queues

If you don't see the above trace messages, it could be that your OVO message queues are corrupt.

- 1 Stop opcagt by running `opcagt -kill`.
- 2 Remove the temporary files by running `rm -f /var/opt/OV/tmp/OpC/*`.
- 3 Restart the opcagt by running `opcagt -start`.
- 4 Close all OVO GUIs.
- 5 Stop the OVO management server by running `ovstop opc ovoacomm`.

- 6 Remove the temporary files by running  

```
rm -f /var/opt/OV/share/tmp/OpC/mgmt_sv/*
```
- 7 Restart the OVO management server processes by running `opcsv -start`.

## Performance Agent Does Not Start Up

When you run `opcagt -start -id 12` and `opcagt -status` the output shows that the Performance Agent is not running.

### Solution:

Verify that CODA really isn't running by running `ps -ef | grep coda`. If CODA is running it means CODA was either started up standalone, or the OVO control agent died, was restarted and is longer the parent.

If CODA isn't running, look in the `<OVAgentDataDir>/log/coda.log` file for any error messages.

## TIBCO SPI Fails to Detect MeasureWare Agent

The TIBCO SPI fails to detect the MWA even though MWA is running on the system.

### Solution:

Make sure the `systemsMWA.txt` file is located in `/var/opt/OV/conf/perf` directory. If it is not, you must copy `systemsMWA.txt` from its existing directory to `/var/opt/OV/conf/perf`. OVPM should then detect MWA.

# Configure TIBCO SPI Application Errors

## Configure TIBCO SPI Application Does Not Start

The Configure TIBCO SPI application does not start.

### Solution:

Delete `/var/opt/OV/conf/tib/appconfig.xml.ser` and restart the configuration tool.

## Configure TIBCO SPI Application Does Not Display Content

The Configure TIBCO SPI application starts, but no content displays.

### Solution:

This problem occurs when you manually edit `WCConfig.xml` with a text editor and save it with non UTF-8 encoding. This file is expected to be in UTF-8.

Open the file in your text editor and make sure to save it in UTF-8 encoding. If your editor does not have a "save as" option to do this and uses system default encoding, then make sure you set your session's `LANG` variable to UTF-8.

## Configure TIBCO SPI Application Fails to Transfer WCConfig.xml

When you run the Configure TIBCO SPI application, the Output of Application window has the message 'Error by transfer file from /var/opt/OV/conf/tib/WCConfig.xml to /var/opt/OV/conf/tib/WCConfig.xml (OpC40-745)'.

### Solution:

There was a problem transferring the WCConfig.xml file from the OVO management server to the Frontend Subagent node. You need to manually transfer the file and restart the Backend service and Frontend Subagent.

- 1 Ftp the /var/opt/OV/conf/tib/WCConfig.xml file from the OVO management server to /var/opt/OV/conf/tib on the Frontend Subagent node.

- 2 Restart the Backend Service by running:

```
ovstop tib-backend
ovstart tib-backend
```

- 3 Restart the Frontend Subagent by running:

```
opcagt -stop -id 25 (OVO 7.x) or ovc -stop -id TIB for (OVO 8.x)
opcagt -start -id 25 (OVO 7.x) or ovc -start -id TIB for (OVO 8.x)
```

- 4 Check the status:

```
opcagt -status -id 25 (OVO 7.x)
ovc -status -id TIB (OVO 8.x)
```

## Configure TIBCO SPI Application Throws AWTException

When you close the Configure TIBCO SPI application, you get the following exception in the Application Output window:

```
java.awt.AWTException: cannot open XIM
```

### Solution:

By default, you cannot input Asian (Japanese, Chinese, Korean) characters into the Configure TIBCO SPI application. To input Asian characters in a Java GUI an input method is required. This input method can be a pure Java input method (independent of OS) or an input method provided by the OS. An input method is installed with Java 1.4. The Configure TIBCO SPI is a Java GUI. Therefore, you need to modify the JAVA\_HOME variable in the /var/opt/OV/bin/OpC/cmds/tib\_config script on the OVO server system to a JVM on the system that's configured to display Asian characters.

## Frontend Logfile Errors

The Frontend subagent's log file is located in `/var/opt/OV/log/tib/frontend.log`.

### Error Starting Notification HTTP Server

```
NotifMgr: Error: Problem starting notification HTTP server. Please  
make sure port is not already in use: XXXX.
```

**Solution:**

Check to see if the Frontend is already running:

```
opcagt -status -id 26 (OVO 7.x)
```

```
ovc -status WLI (OVO 8.x)
```

If the response comes back with “WLI SPI Subagent (Frontend) ... is running”, that means the Frontend is already running.

If the response comes back with “... isn't running”, `opcagt` thinks the Frontend isn't up, but the Frontend Java process may not have been stopped yet. Wait for few minutes for the port to be released and try to start WLI SPI again.

### Error Adding Service to Map

```
Problem adding services to map: Connection refused to host: XXX;  
nested exception is: java.net.ConnectException: Connection refused
```

**Solution:**

Verify that the Backend process is running. On the OVO management server system, run:

```
ovstatus tib-backend.
```

If outcome is “NOT\_RUNNING”, start the Backend by running:

```
ovstart tib-backend.
```

### Frontend does not Connect to the Backend

```
Frontend is not able to connect to the Backend.
```

**Solution:**

This message may display just after installing the Frontend Subagent because the TIBCO SPI has not been configured yet or the Backend Service is not started.

Make sure that you configure the SPI using the Configure TIBCO SPI application located in the OVO Application Bank. See Chapter 2 for complete instructions.

After the configuration is complete, use the Start TIBCO SPI application located in the OVO Application Bank.

## Problem Logging Metric Data

WSF-0009: ddflow returned an error logging TIBSPI\_RPT\_METRICS:256

This error may occur when the data sources for the TIBCO SPI metrics are not registered successfully. Make sure you register the TIBCO SPI Metric data sources by running the following commands on the Frontend Subagent machine:

```
cd /opt/OV/bin
./tib-perl tib-create-datasources
```

## SEA Component Runtime Problems

### Shutdown Problems

The SEA component does not shutdown.

**Solution:**

The default shutdown port is already being used. To ensure that SEA is shutdown, complete the following:

- 1 Type `ps -ef | grep startSEA.sh` and do a `kill -9` on the process id.
- 2 Check for the port number in the `/opt/OV/bin/SEA/conf/AIA.cfg` file under `<OVSBA_CONFIGURATION>` (Usually 4444, by default).
- 3 Type: `netstat -na | grep <port number>`.
- 4 If you see the port number in the output, it means the port is being used. Change the port number in the AIA.cfg file and restart SBA.

### Viewing SEA WSMF Events

You cannot view SEA WSMF events in your SOAP client.

**Solution:** Complete the following procedures.

#### Check if OVO Events Work

- 1 On the OVO server, execute the command:  

```
"opcmsg a=a o=o n=<nodename> msg_text="testing"
```
- 2 Check OVO browser to see if the message arrives. If it does, then OVO events are OK.

#### Check the SEA Log File for Captured Events

You can check `var/opt/OV/log/SEA/SEA.log` to ensure that events are being captured:

- 1 See if an entry for the event sent in the previous section is present. For example, if you ran one of the pre-packaged tests, like the no-impact test, then you should see an event message with OBJECT="CPU".
- 2 Check if you have connection failed errors. If so, OV and OVSBA may not be running properly, or they are looking at the wrong port. Stop OV, Start OV. Stop OVSBA, Start OVSBA.
- 3 Try sending another event.

## Check the SEA Log for Event Subscription ERRORS/WARNINGS

Check the SEA log file for the following subscription warning:

Zero length ImpactedObjects: No Subscriptions for this event.

If you get this warning, please recreate the event definition. It may be corrupted.

## Check the OVComposer GUI

Check the OVComposer GUI to see if there is a valid service\_name under new alarm definition pane. In addition, see if this service is actually part of the SEA MO. Also make sure there is a valid AIA\_EVENT\_NAME. Save the events and refresh the MOVviewer.

## ECS Correlation Problems

You are experiencing ECS correlation related problems.

**Solution:** Complete the following procedures.

### Clean the Fact Store

- 1 Delete /var/opt/OV/conf/OpC/mgmt\_sv/eco\*.fs file.
- 2 Remove OVO templates:
  - a From the Menu bar in the VPO Node bank, select **Actions | Server | Assign Templates**.
  - b Click **Remove Templates**.
  - c From the Menu bar in the VPO Node bank, select **Actions | Server | Install/Update server templates**.
- 3 Verify that /var/opt/OV/conf/OpC/mgmt\_sv/eco\* file is not there.
- 4 Change directories to /opt/OV/bin, and run:

```
./seaECS.sh -r
./seaECS.sh -i
```
- 5 Log into OVO as opc\_adm/OpC\_adm. (You need to have OV administration privileges)
- 6 From the Menu bar in the VPO Node bank, select **Actions | Server | Install/Update server templates**.
- 7 Click **open template selection window**.

- 8 Select **Correlation ComposerTemplate**.
- 9 Go back to the template configuration window, and click **get Template selections**.
- 10 Click **OK**.
- 11 Verify if the above steps were successful by checking the `/var/opt/OV/conf/OpC/mgmt_sv` directory. It should contain an `ecs_comp.eco` file.
- 12 Change directories to `/opt/OV/bin`, and run:
 

```
./seaECS.sh -u
```

## No Managed Objects are Deployed

No managed objects are deployed

### Solution:

- 1 Check `/var/opt/OV/conf/SEA/AIA.cfg` that the root service name mirrors the high level service name of the TIBSPI in the service navigator.
- 2 Check the obj type, to see if they have the right entries.
- 3 See if service engine is operational by using the command: `opcsv -status`.
- 4 Check for any error messages in `/var/opt/OV/log/SEA/SEA.log`.

## Mgmt\_sv Directory not Present

The `mgmt_sv` directory is not present.

### Solution

To ensure that this directory is created, here are the steps that need to be taken:

The following steps involve installing ECS 3.2 , which includes installing PHSS\_30125 patch. Please follow the "Special Installation Instructions" in the patch documentation. For your convenience it is enclosed below:

[http://www2.itrc.hp.com/service/cki/patchDocDisplay.do?patchId=PHSS\\_30125#Special%20Installation%20Instructions](http://www2.itrc.hp.com/service/cki/patchDocDisplay.do?patchId=PHSS_30125#Special%20Installation%20Instructions)

“Special Installation Instructions:

If you want to use the ECS 3.2 Designer

-----

This patch enables you to use the ECS 3.2 designer but the ECS 3.2 OVO integration misses some files (see B555016387) so you have to copy them manually before you start the ECS designer the first time:

```
ITO_MODULES=/etc/opt/OV/share/conf/OpC/mgmt_sv/ecs/modules
OV_CONF=/etc/opt/OV/share/conf
rm -f ${OV_CONF}/ecs/modules.newconfig/ito 2>/dev/null
mkdir -p ${OV_CONF}/ecs/modules.newconfig/ito 2>/dev/null
if [ -d ${ITO_MODULES} ]
then
  for MODULE in `ls ${ITO_MODULES}/*.ecs`
  do
```

```

        cp ${MODULE} ${OV_CONF}/ecs/modules.newconfig/ito 2>&1
    done
fi

```

This problem will be addressed with the next ECS patch.

PHSS\_30125.

#### (A) Patch Installation Instructions

-----

(A1) Stop all OVO processes on your management server; this includes manager processes, communication processes and user-interface processes on the machine.

1. Stop all OVO GUIs (including all Java GUIs connected to this server). Use the "File: Exit" menu bar item.

2. Stop the OVO manager processes: `# ovstop opc ovoacomm`

If any OVO processes are still running, kill them manually:

```
# kill -9 <pid-of-orphaned-process>
```

If you are running OVO in an MC/ServiceGuard installation, you must apply this patch to all MC/SG cluster nodes.

1. Files on the shared disk volumes at `/var/opt/OV/share` and/or `/etc/opt/OV/share` will be patched. Therefore either put the package in maintenance mode or mount the shared disks manually before installing

- Put the package in maintenance mode and leave the shared disks mounted:

```
# touch /tmp/maint_NNM
```

```
# ovstop opc ovoacomm
```

- Mount the shared disks manually:

```
# cmhaltpkg OpC
```

```
# vgchange -a e <shared_vg>
```

```
# mount <etc_shared_vol> /etc/opt/OV/share
```

```
# mount <var_shared_vol> /var/opt/OV/share
```

2. Install the patch on the first cluster node.

3. On the other cluster nodes, make sure the OpC package is not running and the shared disks are not mounted when PHSS\_30125 is installed.

4. Install the patch on the other cluster nodes.

5. After the patch installation remove the following files from the local disks because they already exist on the shared disk:

```
# rm -rf /etc/opt/OV/share/*
```

```
# rm -rf /var/opt/OV/share/*
```



Note whenever you need to deinstall PHSS\_30125, you need to have the same state of the shared disks as during the patch installation. That is, for the patch deinstallation they must be mounted where they were mounted during the patch installation, and they must not be mounted where they were not mounted at that time.

(A2) Install the patch following the standard installation instructions provided above under "Installation Instructions". Observe that you can use `opc_backup(5)` for backing up your system before installing a patch.

(A3) After installing PHSS\_30125, restart the OVO processes on your Management Server system:

- Restart the OVO Manager processes, and check that the processes are running:

```
# /opt/OV/bin/OpC/opcsv -start
```

```
# /opt/OV/bin/OpC/opcsv -status
```

If you are running OVO in an MC/ServiceGuard installation:

- If you entered maintenance mode, return to full monitoring:

```
# ovstart opc
```

```
# rm /tmp/maint_NNM
```

- If you stopped the OpC package, restart it:

```
# cmrunpkg OpC
```

(B) Patch Deinstallation Instructions

-----

NOTE: Before removing the patch, stop all OVO server processes, as described in the Patch Installation Instructions (A1). If you are running OVO in an MC/ServiceGuard installation, make sure to mount the shared disks on the node and only on the node that had them mounted during the patch installation. Otherwise restoring the original files onto the shared disk will fail. Be sure to deinstall PHSS\_30125 from all cluster nodes so that all cluster nodes have the same revision."

## Event Subscriptions Persistence

I do not wish to persist subscriptions.

### Solution:

There is a fault tolerance feature in the SEA that allows for persistence of the event subscriptions so that they do not get lost in the case of a system crash. If you would like the subscriptions to no longer be in effect, please go to the directory configured to be `java.io.tmpdir`, on your system, and delete the file called `_wsmf-sba_`. Remember to restart SEA after the file has been removed. The directory is typically set to `/var/tmp` on UNIX.

## Managed Object Discovery Problems

```
SEA encounters Managed Object Discovery problems. Error is
[java] --> 02-26-04 01:27:50 ERROR [hp.thread.12.discovery-main-SEA-
OVSBA] MoHConnector: getDependency failed. Received null from service
engine getXML: rcv peer closed connection
```

### Solution:

The SEA functionality requires that the Service Engine component in OpenView is operational.

- 1 Make sure that the Service Engine Component is operational. Type:  
`opcsv -status` at the command prompt.
- 2 Make sure the SEA Configuration port is not in conflict. Just to make sure, change the default 4444 port to something else, and restart OV and the SEA components. To check for the port usage, type:  
`netstat -a | grep <ovsba Port>`

## Service Engine Related Problems

The SEA solution requires the service engine. Make sure service engine is operational, before you re-start the SEA solution.

### Service Engine Error

```
Sometimes the service engine is up as per the opcsv -status command, but doing
:opcservice -list -services results in the following error:
```

```
Error: Failed to connect to AF_UNIX socket
'/var/opt/OV/sockets/OpC/opcsvcm': No such file or directory
```

### Solution:

Remove and recreate the opcsvcm file:

- 1 Stop OpenView service processes  
`# /opt/OV/bin/ovstop -v -c`
- 2 Remove the opcsvcm socket file  
`# rm /var/opt/OV/sockets/OpC/opcsvcm`
- 3 Restart the OpenView service processes  
`# /opt/OV/bin/ovstart -v -c`
- 4 The opcsvcm socket should be recreated with the following permissions:  
`# ll /var/opt/OV/sockets/OpC/`  
`total 0`  
`srwxrwxrwx 1 root sys 0 Feb 5 05:04 opcsvcm`



## Appendix A: TIBCO SPI Configuration

This appendix is a reference for the TIBCO SPI configuration parameters. In particular, this appendix focuses on the WConfig.xml configuration file. The parameters are discussed throughout this guide; however, they are listed here in a reference style so they can be easily accessed.

### Editing WConfig.xml

The WConfig.xml file is located in `/var/opt/OV/conf/tib/` on the system running the TIBCO SPI Front-End Subagent. The WConfig.xml file is structured using XML. You can edit this file using the SPI's configuration tool, or you can manually edit this file using an XML or text editor. If you chose to manually edit WConfig.xml, you must restart the Front-End Subagent before the changes take effect.

### Using the Configuration Tool

To edit parameters using the configuration tool:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 In the Application Bank window, double-click the **TIBCO SPI Tools** application group.
- 4 In the Application Group: TIBCO SPI Tools window, right-click **Configure TIBCO SPI application** and select **Execute** from the popup menu.
- 5 Edit the available parameters.
- 6 Select **File | Save** to save your changes to the configuration file.

# WCConfig.xml Configuration Parameters

The WCConfig.xml configuration file contains parameters that configure the Backend Service, Frontend Subagent, and the NNM integration. The root element of the configuration file is <WcConfiguration>. All elements must be children within the root element.

## <FrontendSection>

This node contains parameters that are used to configure the Frontend Subagent. It contains six child nodes:

- <ManagedObjectInfo>
- <OVOMgmtAgentInfo>
- <NotifHttpServerInfo>
- <ServiceMapInfo>
- <Miscellaneous>
- <ReportInfo>

### FrontendSection/ManagedObjectInfo

This element contains a list of WSDL locations that you want the Frontend Subagent to monitor. For each MO you want the Frontend subagent to monitor, add the MO's WSDL location. You only need to add the 'root' MOs' WSDL locations because the Frontend subagent recursively discovers the children. It retrieves information for the children that are related to the parent MO by the relation configured in the Relations to Use list on the Service Map Information area. By default, all children that are related to the parent MO by `http://schemas.hp.com/wsmf/2003/02/Relations/Contains` and `http://schemas.hp.com/wsmf/2003/03/Relations/DependsOn` types are discovered. You can add more entries to the Relations to Use list but you need to be careful about circular dependencies.



The MOs configured in the Managed Object Information list must be accessible before starting up the Frontend subagent.

### FrontendSection/ManagedObjectInfo/WsdlLocation

MO's WSDL location.

### FrontendSection/OVOMgmtAgentInfo

Parameters for the Frontend Subagent

#### FrontendSection/OVOMgmtAgentInfo/HostName

The Frontend subagent system's host name.

#### FrontendSection/OVOMgmtAgentInfo/JavaHome

The location of JAVA\_HOME on Frontend subagent system.

#### FrontendSection/OVOMgmtAgentInfo/RmiPort

The Frontend subagent RMI port number.

### **FrontendSection/OVOMgmtAgentInfo/AgentType**

The Frontend subagent system type. Choices are: UNIX or Windows. (Windows is not implemented yet.)

### **FrontendSection/NotifHttpServerInfo**

Configures the HTTP server in the Frontend Subagent to receive WSMF notifications

### **FrontendSection/HTTPPort**

The HTTP server's port number that receives WSMF notifications

### **FrontendSection/Service MapInfo**

This node contains parameters that allow you to configure the Service Map

### **FrontendSection/Service MapInfo/ActionsToBeIgnored**

The Service Map will ignore actions that are listed in this node.

### **FrontendSection/Service MapInfo/ActionsToBeIgnored/Action**

This node contains parameters for specifying which actions will be ignored.

### **FrontendSection/Service MapInfo/ActionsToBeIgnored/Action/ActionName**

The name of the action to be ignored

### **FrontendSection/Service MapInfo/ActionsToBeIgnored/Action/NamespaceURI**

Namespace of the action to be ignored

### **FrontendSection/Service MapInfo/RelationsToBeUsed**

List of relations that are used by the Frontend subagent to recursively discover children MOs.

### **FrontendSection/Service MapInfo/RelationsToBeUsed/Relation**

Relation to use in discovery

### **FrontendSection/Service MapInfo/OVOUserForServiceMap**

List of OVO users that can view the TIBCO service.

### **FrontendSection/Service MapInfo/OVOUserForServiceMap/OVOUserName**

OVO user name to associate with TIBCO service

### **FrontendSection/Service MapInfo/ServiceAliasMap**

List of service names associated with TIBSPI that can be substituted with another service name. For example if there is a DB service node as part of DB SPI, the TIBSPI node connected to this DB can use the same service name as the DB service node.

### **FrontendSection/Service MapInfo/ServiceAliasMap/AliasInfo**

### **FrontendSection/Service MapInfo/ServiceAliasMap/AliasInfo/ServiceName**

Service Name of the TIBSPI.

**FrontendSection/Service MapInfo/ServiceAliasMap/AliasInfo/AliasName**

The service name to be replaced with this name.

**FrontendSection/Service MapInfo/ServiceAliasMap/AliasInfo/AliasLabel**

The SPI's (i.e DB SPI) service label. The label can be found by clicking the properties on the service icon for a SPI in the Java Console's Service Navigator.

**FrontendSection/Service MapInfo/ServiceIconMap**

This node contains parameters for defining icons for MO types.

**FrontendSection/Service MapInfo/ServiceIconMap/ServiceIcon**

List of icon/MO type. Displays the specified icon in the service map for an MO of the specified type

**FrontendSection/Service MapInfo/ServiceIconMap/ServiceIcon/ServiceType**

MO type

**FrontendSection/Service MapInfo/ServiceIconMap/ServiceIcon/IconFile**

Fully qualified name of the icon file

**FrontendSection/miscellaneous/**

This node is for miscellaneous parameters.

**FrontendSection/miscellaneous/ManagedObjectsToBeIgnored/**

This node allows you to list of MO types that you want to ignore. These MO types are not discovered and therefore will not appear in the service map.

**FrontendSection/miscellaneous/ManagedObjectsToBeIgnored/  
ManagedObjectType**

MO type to ignore

**FrontendSection/miscellaneous/RootServiceName**

The root of the MO hierarchy

**FrontendSection/ReportInfo**

This node allows you to specify a user friendly name for MOs.

**FrontendSection/ReportInfo/ReportGroupInfo**

For each MO that you're collecting data for, you can define a user friendly name that appears on the graphs and reports to identify the MO. The name is also used as the name of the file where data is logged for graphs before it's sent to the Performance Agent. Therefore the name needs to be a valid file name.

**FrontendSection/ReportInfo/ReportGroupInfo/ReportGroupObjectID**

MO's WSDL location that data is being collected for

**FrontendSection/ReportInfo/ReportGroupInfo/ReportGroupName**

User friendly name of MO

## <BackendSection>

This node contains parameters that are used to configure the Backend Service. It contains a single child node, <OVOMgmtServerInfo>.

### **BackendSection/OVOMgmtServerInfo**

This node contains parameters that configure the Backend Service on the OVO management server.

#### **BackendSection/OVOMgmtServerInfo/HostName**

Backend Service system's host name

#### **BackendSection/OVOMgmtServerInfo/JavaHome**

JAVA\_HOME on the Backend Service system

#### **BackendSection/OVOMgmtServerInfo/RmiPort**

Backend Service RMI port number

#### **BackendSection/OVOMgmtServerInfo/NodeGroupName**

OVO node group that the host name manager adds nodes to

## <NetworkingInfo>

This node allows you to configure the NNM integration. It contains two child nodes:

- <NetworkMonitoring>
- <NetworkReporting>

### **NetworkInfo/NetworkMonitoring**

This node allows you to define parameters that configure the NNM integrations' monitoring behavior.

#### **NetworkInfo/NetworkMonitoring/NNMHostName**

This parameter is the fully qualified hostname of the system running OVO and the NNM server.

#### **NetworkInfo/NetworkMonitoring/OviPort**

This parameter defines the TCP port that the TIBCO SPI Back-End Subagent uses to communicate with the Front-End Subagent NNM integration. This parameter is set to 16835 by default. If another application on the OVO server is already using port 16835, you will need to change the port number. If you change the port, you will also need to update the **serverPort** entry in /var/opt/OV/conf/tib/ResponderProxyPluglet.http.config on the OVO server. The **serverPort** and **OviPort** must be the same value.

#### **NetworkInfo/NetworkMonitoring/DiscoveryInterval**

This parameter determines how often the TIBCO SPI Front-End Subagent checks the NNM topology for changes and modifies the TIBCO SPI service view to include any updates. This parameter is set to 4 hours by default.

### **NetworkInfo/NetworkMonitoring/NNMTopology**

This parameter enables the NNM integration in the TIBCO SPI service map. Set this parameter to **true** to enable the NNM integration.

### **NetworkInfo/NetworkMonitoring/NNMTopologyHosts**

This node allows you to specify the topology that is consumed by the NNM Integration.

### **NetworkInfo/NetworkMonitoring/NNMTopologyHosts/ByObjectType**

This node allows you to configure a topology by object type.

### **NetworkInfo/NetworkMonitoring/NNMTopologyHosts/ByObjectType/ObjectType**

This parameter enables the NNM integration to discover the networking device which is directly connected to each system running the TIBCO RVD. This parameter is set to **RVD** by default and does not need to be set.

### **NetworkInfo/NetworkReporting**

This node allows you to define parameters that configure the NNM integrations' reporting behavior.

### **NetworkInfo/NetworkReporting/ReportingOn**

This parameter enables performance reporting. TIBCO SPI Service Reporter reports compare TIBCO RVD and network performance metrics. Set this parameter to **true** to enable performance reporting.

### **NetworkInfo/NetworkReporting/Interval**

This parameter determines how often the TIBCO SPI Front-End Subagent logs network performance metrics to the OpenView Performance Agent (MeasureWare) or CODA. This parameter is set to 5 minutes by default.

### **NetworkInfo/NetworkReporting/CommunityName**

This parameter represents the community name for the SNMP agent running on each RVD system. Typically, the community name for host systems (as opposed to networking devices) is set to public. This parameter is set to **public** by default.

### **NetworkInfo/NetworkReporting/SnmpPort**

This parameter represents the port used by the SNMP agent running on each RVD system. Typically, the SNMP port for hosts systems is set to 161. This parameter is set to 161 by default.

### **NetworkInfo/NetworkReporting/ByObjectType**

This node allows you to configure a topology by object type.

### **NetworkInfo/NetworkReporting/ObjectType**

This parameter enables the NNM integration to collect network metrics for each system running the TIBCO RVD. This parameter is set to **RVD** by default and does not need to be set.





## Appendix B: SEA Configuration

This appendix is a reference for the TIBCO SPI SEA configuration parameters. In particular, this appendix focuses on the AIA.cfg configuration file. The parameters are discussed throughout this guide; however, they are listed here in a reference style so they can be easily accessed.

### Editing AIA.cfg

The AIA.cfg file is located in `/var/opt/OV/conf/SEA/` on the system running the TIBCO SPI AAE directory. The AIA.cfg file is structured using XML. You can edit this file using the SEA's configuration tool, or you can manually edit this file using an XML or text editor. If you chose to modify AIA.cfg, you must restart Tomcat and restart SEA from the application bank, before the changes take effect.

### Using the Configuration Tool

To edit parameters using the configuration tool:

- 1 Run the OVO admin GUI on the OVO management server and logon as *opc\_adm*.
- 2 From any OVO window, bring up the Application Bank window by selecting **Window | Application Bank**.
- 3 In the Application Bank window, double-click the **TIBCO SPI Tools** application group.
- 4 In the Application Group: TIBCO SPI Tools window, right-click **Configure SEA application** and select **Execute** from the popup menu.
- 5 Edit the available parameters.
- 6 Select **File | Save** to save your changes to the configuration file.

## AIA.cfg Configuration Parameters

The AIA.cfg configuration file contains parameters that configure the SEA solution, and allow for configuration of parameters for Managed Objects, Event Subscriptions, SEA ports, etc.... The root element of the configuration file is <AIA\_CONFIGURATION>. All elements must be children within the root element.

### <ManagedObjectInformation>

This node contains parameters that are used to configure the Root Service name, and the object types for the managed objects that are part of the SEA solution. It has 2 child nodes:

- <ROOT\_SERVICE\_NAME>
- <BY\_OBJTYPES>

#### **ManagedObjectInformation/ROOT\_SERVICE\_NAME**

This element contains the root service name as specified in the Service Navigator view of the TIBCO SPI.

For example, TIBCOService

#### **ManagedObjectInformation/BY\_OBJTYPES**

This element(s) identifies type of service you are interested in. This could be an RVService, a logic service, etc. For Example:

```
<BY_OBJTYPES>
  <OBJTYPE>
    http://www.tibco.com/ema/2005/01/mo/type/Folder/RVServices
  </OBJTYPE>
</BY_OBJTYPES>
```

To get both these values, right-click on the Service Node of the TIBCO tree to get the properties. The ObjectID value identifies the service name and the ObjectType value represents the type of service.

### <EventSubscription>

This node allows you to configure the polling interval for the frequency with which new event definitions will be discovered.

### <Event\_Reporting>

This node contains parameters that are currently not supported.

### <OVSBA\_Configuration>

This section allows you to configure parameters related to the SEA smart business agent. Like the port number where the SEA Smart business agent listens for new events.



## Appendix C: List Templates and Reports

This appendix provides reference information for:

- Message Groups
- Applications
- Templates
- TIBCO SPI Self Management
- Performance Metrics
- Reports

### Message Groups

Name	Description
TIBCO	For messages coming from the TIBCO environment.
TIBCO SPI	For messages coming from TIBCO SPI.
TIBCO SPI SEA	For messages coming from TIBCO SPI SEA.

### Applications

Name	Group	Description
TIBSPI SPI Tools	Group	Contains all of the TIBCO SPI applications
Check Status TIBCO SPI	Application	Checks the status of the Backend and the Frontend

Name	Group	Description
Configure TIBCO SPI	Application	Launches a GUI to modify the TIBCO SPI configuration file. Transfers the configuration file to the Frontend system and restarts the Backend and the Frontend.
Deploy SEA	Application	Deploys the TIBCO SPI SEA
Restart TIBCO SPI	Application	Restarts the Backend and the Frontend
SEA Configuration tool	Application	Launches a GUI to modify the SEA configuration file.
SEA Event Definition	Application	Starts the TIBCO SPI SEA wizard
Start SEA	Application	Starts the TIBCO SPI SEA
Stop SEA	Application	Stops SEA Web service
Start TIBCO SPI	Application	Starts the Backend and the Frontend
Stop TIBCO SPI	Application	Stops the Backend and the Frontend

## Templates

Name	Type	Group	Description
SPI for TIBCO	Group	N/A	Contains all of the TIBCO SPI template groups and templates
TIBSPI-EventService-V1	Group	SPI for TIBCO	Contains templates that intercept the OVO messages that are sent by the TIBCO SPI
TIBSPI-Metrics-V1	Group	SPI for TIBCO	Contains the templates related to metric data collection and metric threshold monitoring
TIBSPI-UNIX-Backend-V1	Group	SPI for TIBCO	Contains the templates that monitor the TIBCO SPI log files on the Backend node
TIBSPI-UNIX-Frontend-V1	Group	SPI for TIBCO	Contains the templates that monitor the TIBCO SPI log files on the Frontend node
TIBSPI-UNIX-V1	Group	SPI for TIBCO	Contains the templates that monitor various TIBCO products that are running on Unix nodes

Name	Type	Group	Description
TIBSPI-Windows-V1	Group	SPI for TIBCO	Contains the templates that monitor various TIBCO products that are running on Windows nodes
TIBSPI-EventService-Msg-V1	Message Template	TIBSPI-EventService-V1	The Frontend intercepts operational notifications from the TIBCO EMA. For each operational notification it receives, it sends an OVO message that represents the notification. This template intercepts these OVO messages that are generated by the Frontend.
TIBSPI-Frontend-MSG-V1	Message Template	TIBSPI-EventService-V1	Intercepts the OVO message that's sent by the Frontend once it's started up. Acknowledges previous OVO messages that indicated that a TIBCO resource was down This template must be assigned and deployed to capture the messages sent from the TIBCOSPI-Frontend-Unix-Sched-V1
TIBSPI-Collect-Mon-V1	Monitor Template	TIBSPI-Metrics-V1	Collects metric data for alarming, reports and graphs. Note that the metric data is logged into the report data source. For the graphs, the metric data is saved in text files in: <code>/var/opt/OV/datafiles/tib/datalog</code> It is logged into the graph data source by the TIBSPI-Graph-Mon-V1 template.
TIBSPI-Graph-Mon-V1	Monitor Template	TIBSPI-Metrics-V1	Logs the metric data collect by the TIBSPI-Collect-Mon-V1 template into the graph data source
TIBSPI_0009	Monitor Template	TIBSPI-Metrics-V1	Monitors the RVD Retransmission Packet Rate
TIBSPI_0010	Monitor Template	TIBSPI-Metrics-V1	Monitors the RVD Missed Packet Rate

Name	Type	Group	Description
TIBSPI-Backend-Log-V1	Logfile Template	TIBSPI-UNIX-Backend-V1	Monitors the Backend log file
TIBSPI-SEA-ECS-Log-V1	Logfile Template	TIBSPI-UNIX-Backend-V1	Monitors the SEA ECS log file
TIBSPI-SEA-Log-V1	Logfile Template	TIBSPI-UNIX-Backend-V1	Monitors the SEA log file
TIBSPI-Frontend-Unix-Log-V1	Logfile Template	TIBSPI-UNIX-Frontend-V1	Monitors the Frontend log file
TIBSPI-UNIX-EMA-V1	Group	TIBSPI-UNIX-V1	Contains templates that monitor the TIBCO Enterprise Management Advisor running on a Unix system
TIBSPI-UNIX-RVRD-V1	Group	TIBSPI-UNIX-V1	Contains templates that monitor TIBCO Rendezvous running on a Unix system
TIBSPI-UNIX-HAWK-Agent-V1	Group	TIBSPI-UNIX-V1	Contains templates that monitor the TIBCO Hawk Agent running on a Unix system
TIBSPI-EMA-Unix-Log-V1	Logfile Template	TIBSPI-TIBCO-EMA-UNIX-V1	Monitors the TIBCO Enterprise Management Advisor log file on a Unix system
TIBSPI-RVRD-Unix-Log-V1	Logfile Template	TIBSPI-TIBCO-RVRD-V1	Monitors the TIBCO Rendezvous Routing Daemon log file on a Unix system
TIBSPI-Hawk-Unix-Log-V1	Logfile Template	TIBSPI-UNIX-Hawk-Agent-V1	Monitors the TIBCO Hawk Agent log file on a Unix system
TIBSPI-HawkHMA-Unix-Log-V1	Logfile Template	TIBSPI-UNIX-Hawk-Agent-V1	Monitors the TIBCO Hawk MicroAgent log file on a Unix system
TIBSPI-HawkTibRendezvous-Unix-V1	Logfile Template	TIBSPI-UNIX-Hawk-Agent-V1	Monitors the TIBCO Hawk Agent Rendezvous log file on a Unix system
TIBSPI- WIN-EMA-V1	Group	TIBSPI-Windows-V1	Contains templates that monitor the TIBCO Enterprise Management Advisor running on a Windows system

Name	Type	Group	Description
TIBSPI-WIN-RVRD-V1	Group	TIBSPI-Windows-V1	Contains templates that monitor TIBCO Rendezvous running on a Windows system
TIBSPI-WIN-HAWK-Agent-V1	Group	TIBSPI-Windows-V1	Contains templates that monitor the TIBCO Hawk Agent running on a Windows system
TIBSPI-EMA-WIN-Log-V1	Logfile Template	TIBSPI-WIN-EMA-V1	Monitors the TIBCO Enterprise Management Advisor log file on a Windows system
TIBSPI-RVRD-WIN-Log-V1	Logfile Template	TIBSPI-WIN-RVRD-V1	Monitors the TIBCO Rendezvous Routing Daemon log file on a Windows system
TIBSPI-Hawk-WIN-Log-V1	Logfile Template	TIBSPI-WIN-HAWK-Agent-V1	Monitors the TIBCO Hawk Agent log file on a Windows system
TIBSPI-HawkHMA-WIN-Log-V1	Logfile Template	TIBSPI-WIN-HAWK-Agent-V1	Monitors the TIBCO Hawk MicroAgent log file on a Windows system
TIBSPI-HawkTibRendezvous-WIN-V1	Logfile Template	TIBSPI-WIN-HAWK-Agent-V1	Monitors the TIBCO Hawk Agent Rendezvous log file on a Windows system

## TIBCO SPI Self Management

Name	Template Group	Description
TIBSPI-Mon-Backend-V1	TIBSPI-UNIX-Backend-V1	Monitors the TIBCO SPI Backend process on the Management Server. It sends a message to the OVO Message Browser if Backend process is found not running. An operator initiated Action is available for that message to restart the Backend process.
TIBSPI-Mon-Frontend-V1	TIBSPI-UNIX-Frontend-V1	Monitors the TIBCO SPI Frontend process on the Management Server. It sends a message to the OVO Message Browser if Frontend process is found not running. An operator initiated Action is available for that message to restart the Frontend process.
TIBSPI-EMA-Unix-Mon-V1	TIBSPI-UNIX-EMA-V1	Monitors the TIBCO EMA process running on Unix. It sends a message to the OVO Message Browser if this process is found not running. An operator initiated Action is available for that message to restart the EMA process. OVO Administrator may have to modify the operator initiated action if they have installed the EMA in a directory other than /opt/tibco/ema/2.0 directory, and provide the location where EMA is installed in the specific environment.
TIBSPI-EMA-WIN-Mon-V1	TIBSPI-WIN-EMA-V1	Monitors the TIBCO EMA process running on Windows. It sends a message to the OVO Message Browser if the process is not running. An operator initiated Action is available for that message to restart the EMA process.



## Performance Metrics

Name	MO Type	Alarm	Graph	Report	Description
B001_MsgsSent	RVD		X	X	Number of messages sent by the RVD in the last polling interval
B002_MsgsRcvd	RVD		X	X	Number of messages received by the RVD in the last polling interval
B003_BytesSent	RVD		X	X	Number of bytes sent by the RVD in the last polling interval
B004_BytesRcvd	RVD		X	X	Number of bytes received by the RVD in the last polling interval
B005_PcktSent	RVD		X	X	Number of packets sent by the RVD in the last polling interval
B006_PcktRcvd	RVD		X	X	Number of packets received by the RVD in the last polling interval
B007_RetranPckt	RVD		X	X	Number of packets retransmitted by the RVD in the last polling interval
B008_MissPcktRVD	RVD		X	X	Number of packets missed by the RVD in the last polling interval

Name	MO Type	Alarm	Graph	Report	Description
B009_RetranPcktRate	RVD	X			The RVD's retransmitted packet rate
B010_MissedPcktRate	RVD	X			The RVD's missed packet rate
NumInboundMsgs	JMS Server		X	X	Number of inbound messages received by the JMS Server
InboundMsgRate	JMS Server	X			The JMS Server's inbound message rate per second
NumOutboundMsgs	JMS Server		X	X	Number of outbound messages sent by the JMS Server
OutboundMsgRate	JMS Server	X			The JMS Server's outbound message rate per second
NumPendingMsgs	JMS Server		X	X	Number of pending messages for the JMS Server
NumJobsCreated	BW Engine		X	X	Total number of jobs created for all process definitions in the BW Engine
NumJobsSuspended	BW Engine		X	X	Total number of jobs suspended for all process definitions in the BW Engine
NumJobsSwapped	BW Engine		X	X	Total number of jobs swapped for all process definitions in the BW Engine

Name	MO Type	Alarm	Graph	Report	Description
NumJobsQueued	BW Engine		X	X	Total number of jobs queued for all process definitions in the BW Engine
NumJobsAborted	BW Engine		X	X	Total number of jobs aborted for all process definitions in the BW Engine
NumJobsCompleted	BW Engine		X	X	Total number of jobs completed for all process definitions in the BW Engine
NumJobsCheckpointed	BW Engine		X	X	Total number of jobs checkpointed for all process definitions in the BW Engine
TotalElapsedTime	BW Engine		X	X	Total elapsed time in milliseconds of all jobs completed by all process definitions in the BW Engine

## Reports

Name	Type	Report Family	Description
TIBCO Full Range	Report Family	N/A	Contains reports with all available data
TIBCO Last Full Month	Report Family	N/A	Contains reports with data for the last full month
TIBCO Last Full Week	Report Family	N/A	Contains reports with data for the last full week
TIBCO Yesterday	Report Family	N/A	Contains reports with data from yesterday
TIBCO RVD & Network Interface Receive Rates (FR)	Report	TIBCO Full Range	This report shows the top 10 RVDs of all servers connected on the system. The top 10 RVDs are selected based on the highest rate of packets received. For each RVD also the Network Interface Receive Rates per second of the server are displayed.
TIBCO RVD & Network Interface Send Rates (FR)	Report	TIBCO Full Range	This reports shows the top 10 RVDs of all servers connected on the system. The top 10 RVDs are selected based on the highest rate of packets sent. For each RVD also the Network Interface Send Rates per second of the server are displayed.
TIBCO RVD Missed Packets (FR)	Report	TIBCO Full Range	This report shows the top 10 RVDs of all servers collected on the system. The top 10 RVDs are selected based on the highest number of missed packets.
TIBCO RVD Retrans & Network Interface Errors (FR)	Report	TIBCO Full Range	This report shows the top 10 RVDs of all servers collected on the system. The top 10 RVDs are selected based on the highest retransmission percentage. For each RVD also the Network Interface error percentages of the server are displayed.

Name	Type	Report Family	Description
TIBCO Top 10 RVD Throughput (FR)	Report	TIBCO Full Range	This report shows the top 10 RVDs based on the highest number of messages sent. The chart displays the daily sum of messages sent for each RVD.
TIBCO RVD & Network Interface Receive Rates (LFM)	Report	TIBCO Last Full Month	This report shows the top 10 RVDs of all servers connected on the system. The top 10 RVDs are selected based on the highest rate of packets received over the last full month. For each RVD also the Network Interface Receive Rates per second of the server are displayed.
TIBCO RVD & Network Interface Send Rates (LFM)	Report	TIBCO Last Full Month	This reports shows the top 10 RVDs of all servers connected on the system. The top 10 RVDs are selected based on the highest rate of packets sent over the last full month. For each RVD also the Network Interface Send Rates per second of the server are displayed.
TIBCO RVD Missed Packets (LFM)	Report	TIBCO Last Full Month	This report shows the top 10 RVDs of all servers collected on the system. The top 10 RVDs are selected based on the highest number of missed packets over the last full month.
TIBCO RVD Retrans & Network Interface Errors (LFM)	Report	TIBCO Last Full Month	This report shows the top 10 RVDs of all servers collected on the system. The top 10 RVDs are selected based on the highest retransmission percentage over the last full month. For each RVD also the Network Interface error percentages of the server are displayed.
TIBCO Top 10 RVD Throughput (LFM)	Report	TIBCO Last Full Month	This report shows the top 10 RVDs based on the highest number of messages sent over the last full month. The chart displays the daily sum of messages sent for each RVD.

Name	Type	Report Family	Description
TIBCO RVD & Network Interface Receive Rates (LFW)	Report	TIBCO Last Full Week	This report shows the top 10 RVDs of all servers connected on the system. The top 10 RVDs are selected based on the highest rate of packets received over the last full week. For each RVD also the Network Interface Receive Rates per second of the server are displayed.
TIBCO RVD & Network Interface Send Rates (LFW)	Report	TIBCO Last Full Week	This report shows the top 10 RVDs of all servers connected on the system. The top 10 RVDs are selected based on the highest rate of packets sent over the last full week. For each RVD also the Network Interface Send Rates per second of the server are displayed.
TIBCO RVD Missed Packets (LFW)	Report	TIBCO Last Full Week	This report shows the top 10 RVDs of all servers collected on the system. The top 10 RVDs are selected based on the highest number of missed packets over the last full week.
TIBCO RVD Retrans & Network Interface Errors (LFW)	Report	TIBCO Last Full Week	This report shows the top 10 RVDs of all servers collected on the system. The top 10 RVDs are selected based on the highest retransmission percentage over the last full week. For each RVD also the Network Interface error percentages of the server are displayed.
TIBCO Top 10 RVD Throughput (LFW)	Report	TIBCO Last Full Week	This report shows the top 10 RVDs based on the highest number of messages sent over the last full week. The chart displays the daily sum of messages sent for each RVD.
TIBCO RVD & Network Interface Receive Rates (Y)	Report	TIBCO Yesterday	This report shows the top 10 RVDs of all servers connected on the system. The top 10 RVDs are selected based on the highest rate of packets received yesterday. For each RVD also the Network Interface Receive Rates per second of the server are displayed.

Name	Type	Report Family	Description
TIBCO RVD & Network Interface Send Rates (Y)	Report	TIBCO Yesterday	This reports shows the top 10 RVDs of all servers connected on the system. The top 10 RVDs are selected based on the highest rate of packets sent yesterday. For each RVD also the Network Interface Send Rates per second of the server are displayed.
TIBCO RVD Missed Packets (Y)	Report	TIBCO Yesterday	This report shows the top 10 RVDs of all servers collected on the system. The top 10 RVDs are selected based on the highest number of missed packets yesterday.
TIBCO RVD Retrans & Network Interface Errors (Y)	Report	TIBCO Yesterday	This report shows the top 10 RVDs of all servers collected on the system. The top 10 RVDs are selected based on the highest retransmission percentage for yesterday. For each RVD also the Network Interface error percentages of the server are displayed.
TIBCO Top 10 RVD Throughput (Y)	Report	TIBCO Yesterday	This report shows the top 10 RVDs based on the highest number of messages sent yesterday. The chart displays the daily sum of messages sent for each RVD.





## **Attributes**

Represents information about an MO as a set of properties.

## **Backend Service**

The backend service is a software component that receives data and information from the frontend subagent, converts the information to an OpenView recognized form, and allows OpenView to render the managed environment based on the management data.

## **Conversation**

A managed object that implements the Conversation management interface which represents one service's view of a series of related messages.

## **Enterprise Management Advisor (EMA)**

The TIBCO software that is the gateway through which the TIBCO environment is managed.

## **Event**

An event is a change in the state of the MO.

## **Event Manager**

An Event Manager manages all events emitting from MOs. The Event Manager is responsible for storing, retrieving and (if persistence is implemented) recovering events.

## **Frontend Subagent**

The Frontend Subagent is a software component that is responsible for communicating with the TIBCO EMA software to gather management data about a TIBCO environment and its hosted applications.

## **Managed Object (MO)**

An MO is a management representation of a resource. An MO implements a management interface to provide a means to monitor and/or control the underlying resource. OpenView manages all managed resources in TIBCO environment through their corresponding MOs. In the context of this document, when we talk about the MO, we also refer to the managed resource itself.

## **Management interface**

A management interface exposes the management capabilities of a resource. A Management interface is presented as a set of attributes, operations, and notifications to be accessed through a set of WSDL portTypes.

## **Managed Resource**

Any TIBCO application, product, or abstract management notion such as a class of business process, is referred to as a managed resource.

## **Model**

A model is a set of objects, properties, and their relations.

## **Namespaces**

Namespaces are used to uniquely associate the port types for an interface with a URI. Namespaces are defined in an MO and used in the WSDL file for an MO.

## **Notification**

A notification is a message that is sent or retrieved by one or more subscribers to inform that an event has occurred.

## **Notification Types**

The set of exceptions and state changes that can be reported by an MO.

## **Operations**

The set of functions that can be provided to support the management of an MO.

## **PortTypes**

A PortType is the atomic unit of management functionality. MOs can choose which management portTypes to implement but cannot partially implement a portType. A portType is defined for each interface category and is used in the WSDL file for an MO.

## **Resource**

A resource is a component of a deployed environment.

## **Resource View**

An OpenView term that describes a UI representation of a computing environment from the system administrator point of view that starts with what applications are running on which hosts.

## **Relation**

A relation is a type of association between MOs.

## **Relationship**

A relationship specifies two managed objects and the relation to define how two specific objects are associated.

## **Service**

An MO that implements the Service management interface which represents the management capabilities of a Web service. This Web service may be acting as the provider and/or the consumer of Web service messages.

**Service Effects Analysis (SEA)**

SEA is a plug-in to OpenView that enables adaptive management.

**Service View**

The service view is a UI representation of a computing environment that is application-centric and describes all application dependencies. This view is the bases for root cause analysis of a failure condition, as well as the initial AIA and simple event correlations.

**Service Map**

An OpenView term to describe a graphic view into a managed environment. This view shows relationships among MOs.

**Simple Object Access Protocol (SOAP)**

The standard for Web services messages. Based on XML, SOAP defines an envelope format and various rules for describing its contents. Seen (with WSDL and UDDI) as one of the three foundation standards of Web services, it is the preferred protocol for exchanging Web services.

**Subscriber**

A subscriber is an entity that is interested in selected notifications from MOs. These notifications contain information about the state change in an MO. For scalability reason, subscription to notifications has an associated timeout. Subscription can be renewed before they expire.

**Web Services Description Language (WSDL)**

The standard format for describing a Web service. Expressed in XML, a WSDL definition describes how to access a Web service and what operations it can perform.

**Web Services Execution Environment (WSEE)**

An MO that implements the WSEE management interface which encapsulates the management capabilities of a Web service execution environment.



## A

- adaptive management, 4-1
- advance management
  - SEA, 4-1
- advanced management
  - nnm integration, 5-1
- agent node, 2-4
  - adding, 2-6
- AIA configuration file
  - editing, B-1
  - reference, B-1
- AIA.cfg, B-1, B-2
- alarms, 4-8
- alerts, 4-2, 4-10
- application group, 2-22
- application manager, 4-1
- assign template groups, 2-11

## B

- backend node
  - deploy template groups, 2-10
- backend service, 1-6
  - configure, 2-8
  - register, 2-5
- backlist correlation, 4-2
- business management
  - architecture, 1-4

## C

- CA root certificate, 7-3, 7-4
- check status
  - command, 2-15
- client certificate, 7-5
- CODA, 5-4
  - errors starting, 8-10
  - logging, 3-17
  - overview, 3-16

- coda.log, 3-16
- configuration
  - backend service, 2-5
  - backend service and frontend subagent, 2-8
  - data sources, 2-10
  - frontend subagent, 2-10
  - SBA polling frequency, 4-12
  - SEA, B-1
  - TIBCO SPI, A-1
  - tool, A-1, B-1
- correlation
  - architecture, 4-1
  - backlist, 4-2
  - enrich, 4-2
  - flapping, 4-2
  - multi-source, 4-2
  - rate, 4-2
  - repeated, 4-10
- correlator
  - custom, 4-7
  - testCPU, 4-7
- custom adapters, 3-7
- custom adapters metric threshold values, 3-13

## D

- data sources, 2-10
- DEBUG log level, 4-11
- deploy template groups, 2-10, 2-11
- deployment scenarios
  - collocated components, 1-10
  - consolidated, 1-9
  - fully distributed, 1-10
  - remote frontend subagent, 1-10
- domain expert, 4-1
- domain extensions
  - OV, 1-11
  - TIBCO, 1-11

**E**

- ECS, 1-7
  - correlation problems, 8-14
- ECS composer, 4-2, 4-7
- ECS engine, 4-2
- EMA. See Enterprise Management Advisor
- emiagent.log, 2-9
- enable MSI, 4-6
- enrich correlation, 4-2
- Enterprise Management Advisor, 1-1
  - architecture, 1-8
  - business management overview, 1-4
  - monitor, 3-20
  - overview, 1-7
- ERROR log level, 4-11
- event
  - invoking operations, 3-4
- event definition
  - creating, 4-7
- event management, 3-4
- events
  - viewing, 3-4

**F**

- failover, 6-1
- flapping correlation, 4-2
- frontend subagent, 1-6
  - assigntemplate groups, 2-11
  - configure, 2-8, 2-10
  - deploy template groups, 2-11
  - error starting, 8-1
  - errors in log file, 8-12
  - install, 2-6
  - uninstall, 2-20
  - verify install, 2-8
- frontend.log, 8-12

**G**

- graphs, 3-16
  - creating bytes sent, 3-19
  - instance metrics, 3-20
  - performance, 5-2
  - viewing, 3-19

**H**

- hardware requirements, 2-3
- Hawk API, 1-4, 1-8
  - console, 1-8
- Hawk MicroAgent, 1-9
- HPjconfig tool, 2-3
- HTTPS, 7-1
  - configuration parameters, 7-8
  - configuring, 7-4

**I**

- INFO log level, 4-11
- installation
  - frontend subagent, 2-6, 2-7
  - resource explorer, 2-19
  - SEA, 4-5
  - TIBCO SPI, 2-4
- instance metrics
  - graphing, 3-20

**J**

- Java console, 1-1
  - installing, 2-17
  - starting, 2-18

**K**

- kernel parameters, 2-3
- keystores, 7-3
  - setting up, 7-4
  - utility, 7-5

**L**

- license
  - open source, 2-2
- location candidates, 6-2
- log levels, 3-21
  - changing, 4-11
- log levels, changing, 4-11
- log4j, 4-10
- logging, 4-10
- logging metrics, 3-5
  - custom adapters, 3-7, 3-13
- logging.properties, 4-10

## M

- manageability, 3-1
- managed node. *See* agent node
- managed resources
  - viewing, 3-2
- management categories
  - configuration, 1-12
  - control, 1-12
  - dsicover, 1-12
  - event callback, 1-12
  - event push and event pull, 1-12
  - performance, 1-12
- management console, 1-1
  - Java console, 2-17
  - resource explorer, 2-18
- management interfaces, 1-12
- management server, 2-4
- MeasureWare, 3-16, 5-4
- message drop rate, 3-8
- message group, 2-22
- message groups
  - reference, C-1
- message templates, 2-23
- metric element attributes, 3-5
- MetricDefinitions.xml, 3-5, 3-7
- metrics, 3-5
  - changing threshold values, 3-15
  - collect data, 3-12
  - configuration file, 3-5
  - data sources, 2-10
  - graphing, 3-20
  - logging for custom adapters, 3-7, 3-13
  - multi-instance data, 3-9, 3-11
  - performance, 5-2
- MO
  - filtering unwanted, 3-3
  - user friendly name, 3-17
- monitor EMA agent, 3-20
- MSI, 4-6
- multi-instance metric data
  - configuring, 3-9
  - configuring threshold, 3-11

- multi-source correlation, 4-2

## N

- network dependencies, 5-1
- network errors, 5-3
- network monitoring, 5-5
- network operator, 4-1
- NNM
  - configuring, 5-5
  - enabling, 5-5
  - integration overview, 5-1
  - parameters, A-2
  - requirements, 5-4
- NNM ET, 5-4
- node groups, 2-24

## O

- opc\_adm operator
  - configuring, 2-5
- opcmsg, 2-11
- opctemplate, 2-14
- open source license, 2-2
- OpenView
  - Composer tool (ECS), 1-7
- OpenView Operations, 1-1
- operational notification, 8-6
  - missing, 8-4
- operator automated actions, 3-4
- OV Reporter, 2-20, 3-5
- OVO, 2-1
- ovpa, 5-4
- OVPM, 3-18
  - RVD graphs, 3-19
  - view graphs, 3-19

## P

- patches
  - software, 2-3
- performance, 3-18
- performance graphs, 3-16
- performance metrics
  - reference, C-1

**R**

- rate correlation, 4-2
- receive rates, 5-2
- regroup condition, 2-22
- reinstall
  - templates monitor and commands, 2-23
- remove
  - TIBCO SPI application group, 2-22
  - TIBCO SPI bits, 2-21, 2-22
  - TIBCO SPI message group, 2-22
  - TIBCO SPI message templates, 2-23
  - TIBCO SPI node groups, 2-24
  - TIBCO SPI regroup condition, 2-22
  - TIBCO SPI user profile, 2-22
  - TIBCO SPI users, 2-23
- repeated correlation, 4-10
- reports, 1-7, 3-16
  - install, 2-20
  - performance, 5-2
  - reference, C-1
  - view, 3-17
- requirements
  - hardware, 2-3
  - NNM, 5-4
  - SEA installation, 2-2, 4-3
  - software, 2-2
  - software patches, 2-3
  - TIBCO SPI installation, 2-2
- resource explorer, 1-6
  - installing, 2-18, 2-19
  - overview, 1-6
  - security configuration, 7-6
  - starting from command line, 2-19
  - starting from Java Console, 2-19
  - using, 2-11
- restarting
  - TIBCO SPI, 2-14
- runtime problems, 8-1
- RVD
  - bus, 4-10
  - collecting, 3-7
  - create graphs, 3-19

- data collection, 5-6
- monitoring, 3-18
- view graphs, 3-19

**S**

- SBA
  - polling frequency, 4-12
- sba-wsms.xml, 4-12
- SEA
  - architecture, 4-1
  - composer tool, 4-2
  - configuration, B-1
  - configuration tool, 4-7
  - default port, 4-4
  - error viewing WSMF events, 8-13
  - event definition, 4-7
  - installation, 4-5
  - installation requirements, 2-2
  - introduction, 1-7
  - MO discovery problems, 8-18
  - no MOs are deployed, 8-15
  - overview, 4-1
  - prerequisites, 4-3
  - roles, 4-1
  - running, 4-6
  - software requirements, 4-3
  - stop, 4-7
  - uninstall, 4-12
  - upload templates, 4-5
- security, 7-1
- self management, for TIBCO SPI, 3-20
- send rates, 5-3
- service
  - problem adding to map, 8-12
- service engine
  - error, 8-18
- service management, 3-1
- service map
  - linking, 3-2
  - viewing, 3-2
- service navigator, 1-1
- service reporter, 5-2
- service views, 5-1



- SNMP, 5-2, 5-4
- software requirements, 2-2
  - patches, 2-3
- SPI (TIBCO). *See* TIBCO SPI
- SSL, 7-1
  - configuring, 7-4
- standard management, 3-1
- stopping
  - TIBCO SPI, 2-15

## T

- template groups
  - assign, 2-11
  - assign to backend node, 2-10
  - assign to Windows nodes, 2-13
  - deploy, 2-11
  - deploy to backend node, 2-10
  - deploy to individual UNIX nodes, 2-12
  - deploy to Windows nodes, 2-13
  - verify, 2-14
- templates
  - TIBSPI\_0009, 3-15
  - TIBSPI-Collect-Mon-V1, 3-14
  - TIBSPI-Hawk-WIN-Log-V1, 3-15
  - TIBSPI-UNIX- V1 template group, 2-12
  - TIBSPI-UNIX-Backend-V1 template group, 2-10
  - TIBSPI-UNIX-Frontend-V1 template group, 2-11
  - TIBSPI-Windows-V1 template group, 2-13
  - TIBSPI-WIN-HAWK-Agent-V1 template group, 3-15
- templates monitor and commands, 2-23
- testCPU correlator, 4-7
- TIBCO Hawk logfile, 3-15
- TIBCO SPI, 1-4
  - adaptive management, 4-1
  - applications reference, C-1
  - backend service, 1-6
  - business management overview, 1-4
  - check status command, 2-15
  - configuration, A-1
  - frontend subagent, 1-6
  - hardware requirements, 2-3

- installation HP-UX and SOLARIS, 2-4
- installation requirements, 2-2
- introduction, 1-1
- message groups, 2-10
- message templates, 2-23
- node groups, 2-24
- remove bits, 2-21, 2-22
- reports, 1-7
- restarting, 2-14
- self management, 3-20
- self management reference, C-1
- software requirements, 2-2
- stopping, 2-15
- template groups, 2-10
- troubleshooting, 8-1
- uninstall, 2-20
- user, 2-23
- user profile, 2-22
- TIBSPI\_0009 template, 3-15
- TIBSPI-Collect-Mon-V1 template, 3-14
- TIBSPI-EventService-V1, 2-11
- TIBSPI-Hawk-WIN-Log-V1 template, 3-15
- TIBSPI-Metrics-V1, 2-11
- TIBSPI-Unix node group, 2-5, 2-6
- TIBSPI-UNIX- V1 template group, 2-12
- TIBSPI-UNIX-Backend-V1 template group, 2-10
- TIBSPI-UNIX-Frontend-V1, 2-11
- TIBSPI-UNIX-Frontend-V1 template group, 2-11
- TIBSPI-Windows-V1 template group, 2-13
- TIBSPI-WIN-HAWK-Agent-V1 template group, 3-15
- trace levels, 3-22
- trap events, 5-2
- troubleshooting, 8-1
- truststores, 7-3
  - setting up, 7-4

## U

- UDM metric definition file, 3-7
- UDM metrics, 2-10
- UDMMetricDefinition.xml, 3-5, 3-9, 3-13
- uninstall

- frontend subagent, 2-20
- SEA, 4-12
- TIBCO SPI, 2-20
- TIBCO SPI bits, 2-21, 2-22

## V

- verify
  - Frontend subagent installation, 2-8
  - start status, 2-16
  - stop status, 2-17
  - template groups, 2-14
  - TIBCO SPI, 2-15

## W

- WARN log level, 4-11

- WC configuration file
  - editing, A-1
  - editor error, 8-11
  - parameters, A-2
  - reference, A-1
- WCConfig.xml, 2-8, A-1
  - transferring fails, 8-11
- WLI SPI
  - reports, 2-20
- WSDM
  - alerts, 4-2
  - channel, 1-4
  - overview, 1-11
  - XML interfaces, 1-12