



## **Cisco Broadband Policy Manager Operations Guide**

Software Release 1.6

### **Corporate Headquarters**

Cisco Systems, Inc.  
170 West Tasman Drive  
San Jose, CA 95134-1706  
USA  
<http://www.cisco.com>  
Tel: 408 526-4000  
800 553-NETS (6387)  
Fax: 408 526-4100

Customer Order Number:  
Text Part Number: OL-7765-01



THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

CCVP, the Cisco Logo, and the Cisco Square Bridge logo are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn is a service mark of Cisco Systems, Inc.; and Access Registrar, Aironet, BPX, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, Follow Me Browsing, FormShare, GigaDrive, GigaStack, HomeLink, Internet Quotient, IOS, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, LightStream, Linksys, MeetingPlace, MGX, Networking Academy, Network Registrar, *Packet*, PIX, ProConnect, RateMUX, ScriptShare, SlideCast, SMARTnet, StackWise, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0609R)

*Cisco Broadband Policy Manager Operations Guide*  
© 1999-2004 Cisco Systems, Inc. All rights reserved.

# Contents

<b>Contents .....</b>	<b>iii</b>
<b>Preface.....</b>	<b>ix</b>
Introduction.....	ix
Scope.....	ix
Audience .....	ix
Conventions .....	ix
Text .....	ix
Icons.....	xi
Documentation Set.....	xi
Cisco Broadband IP Service Module User Guide .....	xi
Cisco Broadband Policy Design Studio User Guide .....	xi
Cisco Broadband Policy Manager Installation and Configuration Guide .....	xi
Cisco Broadband Policy Manager Operations Guide.....	xii
Cisco Broadband Policy Manager Release Notes .....	xii
Cisco Capacity Admission Control Manager User Guide.....	xii
Organization.....	xii
Chapter 1 - Introduction.....	xii
Chapter 2 - Routine Tasks.....	xii
Chapter 3 - Command Line Interface .....	xii
Chapter 4 - Maintenance Tasks.....	xii
Chapter 5 - Troubleshooting Tasks .....	xii
Appendix A - Glossary .....	xii
Appendix B - Statistics .....	xii
<b>1 Introduction.....</b>	<b>1</b>
Overview.....	1
ACM Architecture .....	2
Topology Awareness Function.....	4
Topology Store Function .....	4
Path Computation Function .....	5
Admission Control Function .....	5
Signaling Interface Function.....	5
Session Awareness Function.....	6
Session Store Function .....	6
Context Store Function .....	6
Context Timer Function.....	6
Accounting Log Function.....	6

Statistics Function.....	7
Alarm Notification Function.....	7
Pool Function.....	7
SM Architecture.....	8
NM Architecture.....	9
Deployment Components.....	11
Relations to Functional Interfaces.....	11
Director.....	11
Resource Controller.....	11
Topology Database Server.....	12
Firewall Requirements.....	12
What's Next?.....	12
<b>2 Routine Tasks.....</b>	<b>13</b>
Overview.....	13
Starting and Stopping the BPM System.....	14
Procedure: Starting the Domain Controller BPM.....	14
Procedure: Starting the Other Systems.....	14
Procedure: Stopping the Domain Controller BPM.....	14
Procedure: Stopping the Other Systems.....	14
BPDS or BPS Installation Procedure.....	15
Procedure: Installing the BPDS or BPS.....	15
BPDS or BPS Login.....	17
Procedure: Logging in to the BPDS or BPS System.....	17
Observing Component Status.....	19
Network Administration Tree Pane.....	20
Service Engine Configuration.....	20
Agent Configuration.....	21
Service Design Tree Pane.....	21
Procedure: Viewing Information about Policy Functions.....	21
Procedure: Viewing Information about Rules.....	23
Statistics.....	25
Agent Statistics.....	25
Other Statistics.....	26
SNMP Traps.....	27
Accounting.....	28
Director Accounting Log Messages.....	31
Resource Controller Accounting Log Messages.....	32
Topology Database Server Accounting Log Messages.....	33
Session Manager Accounting Log Messages.....	37
Network Manager Accounting Log Messages.....	37
Error Messages.....	38
Log Pane.....	38
Log Messages.....	38

Log File Format.....	38
Severity Definitions.....	39
Example Log Entry.....	39
Changing Logging Level for Service Engine.....	39
System Status.....	40
Status Bar.....	40
Role-Based Access Control (RBAC).....	41
Elements of the BPM RBAC.....	41
Roles.....	41
Users.....	41
Resources.....	41
Actions.....	41
Permissions.....	41
Using RBAC.....	42
Starting RBAC.....	42
Roles.....	43
Users.....	48
Importing and Exporting RBAC Data.....	57
Backup and Restore.....	59
What's Next?.....	59
<b>3 Command Line Interface.....</b>	<b>61</b>
Overview.....	61
Realms.....	61
Domain Realm.....	62
Director Realm.....	62
Resource Realm.....	62
Network Realm.....	62
Session Realm.....	62
tash Command Line Interface.....	62
Invoking tash.....	62
Example: Invoking tash without Arguments.....	63
Example: Invoking tash with Arguments.....	63
Environment Variables.....	63
Commands.....	63
Common Commands Available from Both ACM and SM/NM.....	64
CAC-Specific Commands in ACM Extension Library.....	68
Session- and Adaptation-specific Commands in SM/SM Extension Library.....	70
Statistics.....	76
TDS Statistics.....	76
Director Statistics.....	76
Resource Controller Statistics.....	77
Session Manager and Network Manager Statistics.....	77
What's Next?.....	78

<b>4</b>	<b>Maintenance Tasks.....</b>	<b>79</b>
	Overview.....	79
	Log File Maintenance.....	79
	Realms.....	80
	Domain Realm.....	80
	Director Realm.....	80
	Resource Realm.....	80
	Network Realm.....	80
	Session Realm.....	80
	Resource Controller Procedures.....	80
	Adding Resource Controller.....	81
	Removing a Resource Controller.....	82
	Starting Resource Controller.....	83
	Director Procedures.....	84
	Adding a Director.....	84
	Removing a Director.....	85
	Deploying Components.....	86
	Procedure: Deploying an Agent.....	86
	Procedure: Undeploying an Agent.....	86
	Procedure: Deploying a Service.....	87
	Procedure: Undeploying a Service.....	89
	Configuring Components.....	91
	Procedure: Configuring a Standby Service Engine.....	91
	Procedure: Changing Logging Level for Service Engine.....	92
	Procedure: Changing the Name of a Component.....	93
	Procedure: Changing Agent Configuration.....	94
	Agent Properties Dialog Box.....	95
	Agent Devices Dialog Box.....	96
	Procedure: Patching.....	96
	Clusters and Failover.....	97
	Procedure: Using BPS to Create a Cluster.....	97
	Procedure: Verifying Cluster Health.....	100
	Viable Cluster.....	100
	Failed Cluster.....	100
	Procedure: Dissolving a Cluster.....	101
	Failover.....	102
	Manual Failover.....	102
	Automatic Failover.....	103
	Failed Node Recovery.....	103
	Resolving Failover.....	103
	What's Next?.....	105
<b>5</b>	<b>Troubleshooting Tasks.....</b>	<b>107</b>
	Overview.....	107

---

Logging in.....	107
System Information.....	107
Network Administration Tree Pane.....	107
Service Engine Status.....	107
Agent Status.....	108
Service Status.....	109
Log Pane.....	110
Isolating the Problem.....	110
Investigating Hardware Issues.....	111
Investigating Software Issues.....	111
Troubleshooting Considerations.....	111
Is the BPM Running?.....	111
Is the Service Engine Running?.....	112
Are Services and Agents Deployed?.....	112
When Did Problem Occur?.....	112
<b>Appendix A - Glossary.....</b>	<b>113</b>
<b>Appendix B - Statistics.....</b>	<b>131</b>
Overview.....	131
<b>Index.....</b>	<b>141</b>





# Preface

## Introduction

This guide presents information for the Broadband Policy Manager (BPM) and covers the following topics:

- Operations
- Maintenance
- Troubleshooting

This guide contains procedures that the network administrator can perform using the Broadband Policy Studio (BPS) or Broadband Policy Design Studio (BPDS). It also gives command line procedures where there are no BPS or BPDS equivalents.

## Scope

This guide provides instructions for conducting routine tasks, ongoing maintenance tasks, and troubleshooting problems.

## Audience

This guide is for the network or service administrator who performs operations, maintenance, and troubleshooting tasks.

## Conventions

This guide may use the text and icon conventions described in this section.

### Text

The table below contains documentation text conventions.

**Table 1. Text Conventions**

Convention	Explanation	Example
alternate mouse button	Usually indicates the right mouse button.	Click the agent with the alternate mouse button.
arrow -->	Indicates the selection order of menu items.	<b>File --&gt; Save</b> This indicates go to the File menu and choose the Save function.
bar brackets [ ]	Indicate the default.	Choose your Name Service type [2]: This indicates the default is 2.
<b>bold</b>	Indicates user input or button selection.	<b>poweron</b>
<b><i>bold italic</i></b>	Indicates objects, attributes, pin names, and service flows.	Right-click the <b><i>request</i></b> function.

**Table 1. Text Conventions**

<b>Convention</b>	<b>Explanation</b>	<b>Example</b>
Ctrl+X	Indicates the quick access key for a menu option.	Ctrl+M This indicates open the Object Manager.
default mouse button	Usually indicates the left mouse button.	Click the agent with the default mouse button.
<i>italic</i>	Indicates an application, chapter, directory, document, header, section, or title names.	For more information, refer to the section entitled <i>Creating Services</i> .
<KEYNAME>	Indicates press the named key.	Supply the required information, then press the <ENTER> key.
screen display	Represents system output.	This agent does not have any agent-specific properties.

## Icons

The following icon conventions provide additional information to indicate special conditions or possible risks:



**Note:** *A note is an informational message containing a tip or suggestion.*



**Caution:** *A caution indicates a risk of damage to equipment or a loss of data.*

## Documentation Set

The documentation for your Broadband Policy Manager (BPM) system includes the following documents:

- Cisco Broadband IP Service Module User Guide
- Cisco Broadband Policy Design Studio User Guide
- Cisco Broadband Policy Manager Installation and Configuration Guide
- Cisco Broadband Policy Manager Operations Guide
- Cisco Broadband Policy Manager Release Notes
- Cisco Capacity Admission Control Manager User Guide

### Cisco Broadband IP Service Module User Guide

This document discusses the Broadband IP Service Module for session management and network adaptation. It discusses its architecture, components, access methods, and functions.

### Cisco Broadband Policy Design Studio User Guide

This guide provides instructions for installing the Broadband Policy Design Studio (BPDS). It discusses how to use the BPDS to create, deploy, and manage network services and topologies.

### Cisco Broadband Policy Manager Installation and Configuration Guide

This guide describes how to install the software for the BPM. It describes how to install and configure the Solaris operating system for use by the BPM. It also includes procedures to install and configure the BPM software and the procedures to install and log into the BPDS.

## **Cisco Broadband Policy Manager Operations Guide**

This guide describes the use of the BPDS to obtain information, conduct day-to-day operations, perform maintenance tasks, and troubleshoot problems with the BPM system. These tasks include use of the Log Messages addendum, the Application Log Messages addendum, and the Statistics addendum.

## **Cisco Broadband Policy Manager Release Notes**

This document describes new features, known limitations, and other important information about the BPM system.

## **Cisco Capacity Admission Control Manager User Guide**

This document discusses the architecture and components for the Capacity Admission Control Manager product.

## **Organization**

This guide contains the following chapters:

### **Chapter 1 - *Introduction***

Introduction to the operations guide.

### **Chapter 2 - *Routine Tasks***

Procedures for performing day-to-day operations tasks.

### **Chapter 3 - *Command Line Interface***

Procedures for performing tasks using the command line interface.

### **Chapter 4 - *Maintenance Tasks***

Procedures for performing necessary maintenance tasks.

### **Chapter 5 - *Troubleshooting Tasks***

Suggestions for dealing with problems.

### ***Appendix A - Glossary***

Glossary of BPM terms.

### ***Appendix B - Statistics***

System statistics that are available.

# Introduction

## Overview

At the highest level, the Broadband Policy Manager (BPM) consists of a visual development environment, a deployment manager, and execution environments. These elements cooperate to provide a product-line architecture for service-oriented systems. This environment enables and simplifies the repeated production of many related solutions for real-time network policy management.

In the BPM product-line architecture, dataflow programming is the primary compositional element underlying all solutions. Dataflow programming is a well-established programming methodology that promotes the explicit description of data movement and transformation in program execution. Using the development environment component, users develop dataflow programs interactively with a drag-and-drop visual programming tool, the Broadband Policy Design Studio (BPDS). These dataflow programs describe the movement of data between operators, which are black-box programming elements exposed by software agents. These agents generally encapsulate specific implementations of abstractions like protocols, network devices, data sources, or logic capabilities. The generated dataflow programming artifacts combine into collections of services, referred to as applications, which provide complete solutions.

The deployment manager publishes these applications into the execution environment. The execution environment is a distributed domain of networked processing nodes. A node is a computer or some other device on a network. Every node has a unique network address. A link is a line or channel over which data is transmitted.

Each node runs a highly concurrent graph-traversal engine, coupled with a fast data switching fabric. Once published to an execution environment on a node, application services are available for execution and monitoring. The execution environment also provides resiliency and failover capabilities to published applications.

The Admission Control Manager (ACM) provides Capacity Admission Control (CAC) services over broadband access networks. Admission control monitors, controls, and enforces the use of network resources and services with policy-based management. The criteria for policy-based management include identifying users and applications, or identifying traffic based on how, when, and where it enters the network.

The Session Manager (SM) product tracks user sessions connecting to the broadband access network. The SM product offers the ability to add per-subscriber session management storage capability to a policy control solution. It also supplies information for mapping subscribers to physical network devices and ports and provides valuable information to the topology information model. It deploys alongside the NM and CAC products to support them.

The Network Manager (NM) product provides facilities for controlling and querying elements in the network. The NM product offers a variety of interfaces, including Netconf (CLI), SNMP, and RADIUS COA. It deploys alongside the SM and CAC products to support them.

The Broadband Policy Manager (BPM) is a product suite used by service providers to create and deploy advanced services on broadband networks. The administrator of the BPM performs routine, maintenance, and troubleshooting tasks, including starting or stopping the BPM, monitoring status, obtaining statistics or other information, managing user access, configuring components, and solving problems.

The administrator has various tools available to perform these tasks. These include a graphical user interface and a command line interface.

The Broadband Policy Studio (BPS) is a graphical user interface to the BPM. The BPS includes the following:

- Network Administration screen, which shows BPM data from a network point of view, including service engines, agents, and services

The Broadband Policy Design Studio (BPDS) is a similar graphical user interface to the BPS. It has the same Network Administration screens that the BPS has, and also includes:

- Service Design screen, which shows the structure of existing services and allows you to create new services.

You can use the BPDS or BPS to obtain BPM information, conduct routine operations, perform necessary maintenance tasks, and troubleshoot problems.

In addition, for some tasks, there is a command line interface (CLI) that allows you to enter certain commands directly. To enter CLI commands, first `ssh` to the correct machine, log in as the correct user, then enter the commands at the command line.

## ACM Architecture

The delivery of Telephony CAC utilizes generalized internal Cisco components as part of the Admission Control Manager (ACM):

- Topology Awareness Function
- Topology Interaction Function
- Topology Store Function
- Path Computation Function
- Admission Control Function
- Signaling Interface Function
- Session Awareness Function
- Session Store Function
- Context Store Function
- Context Timer Function
- Accounting Log Function
- Statistics Function

The functional components connect through well-defined interfaces to comprise the Director BPMs and Resource Controller BPMs.

**Figure 1. ACM Architecture: Director BPM**

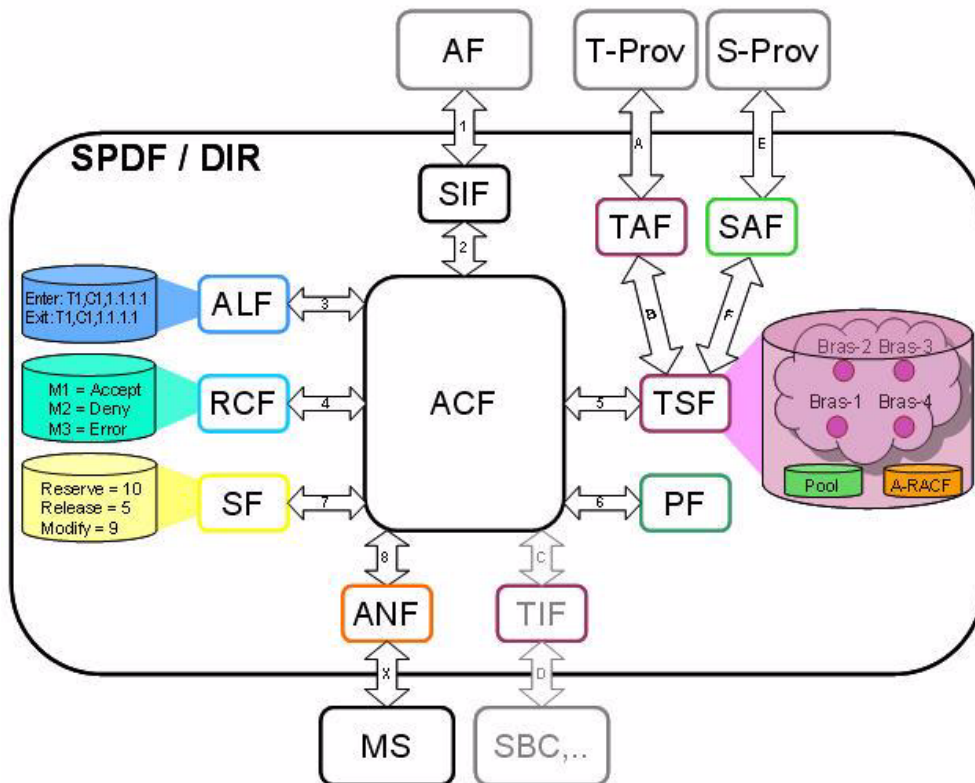
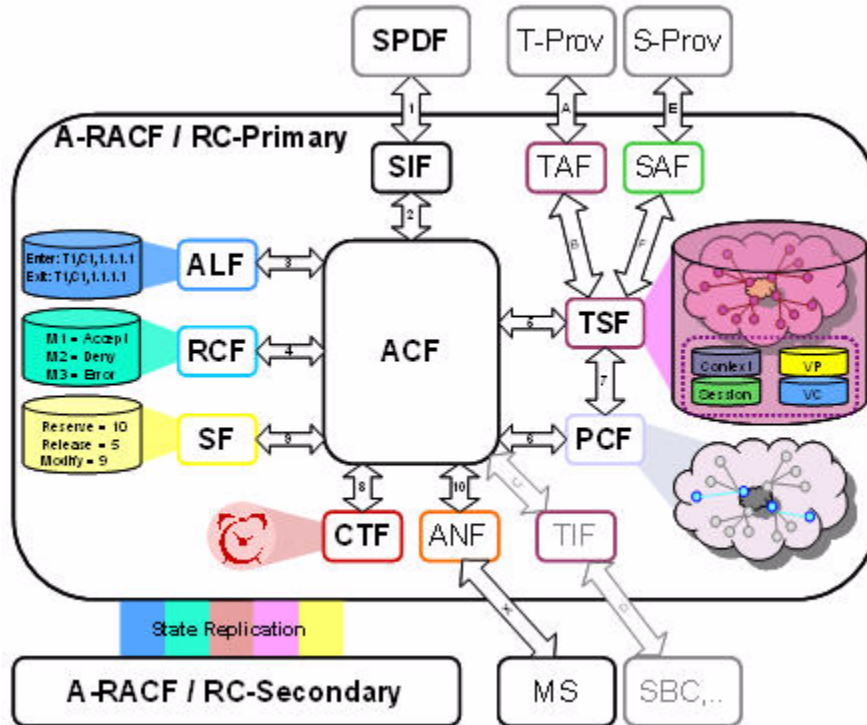


Figure 2. ACM Architecture: Resource Controller BPM



## Topology Awareness Function

The Topology Awareness Function (TAF) extracts changes in the node and underlying network details by various methods. It stores the information and maintains the information model in a data store. A TAF (possibly a different TAF) detects and uses node and link failures to dynamically update and maintain the information model.

The TAF has two discrete interface functions:

- Interface to the network over which network discovery detail is extracted
- Interface to the Topology Store Function (TSF) to record topology data within the Topology Store information model

## Topology Store Function

The Topology Store Function (TSF) maintains the Topology Information Model (TIM). Clients acting as TAFs populate the TIM. The database stores the entire network node and link detail, or the needed local subset of this information, depending on deployment configuration. Node information describes their capabilities. Each node has a unique identifier (deployment dependent). Each link is a directed association between two (or more) nodes. Other elements using the TSF, such as the PCF or ACF, may use the stored Topology Information Model to determine the path between two end points, or update resources in the TIM.

Depending on the deployment, links and nodes need not describe the physical network topology. For tunnels use (for example, MPLS), the description may depict the tunnel head end and remote nodes and ignore the core topology underlying this path. This virtualization allows the admission control criteria to dictate the requirements on the TIM. The TAF in use understands the same virtualization to populate and maintain the TIM appropriately.

The TSF has three separate interfaces to the other system functional components (ACF, PCF, and TAF). The same interface supports the requirements of this interconnectivity.



## Path Computation Function

The Path Computation Function (PCF) determines the path taken through the topology for any given end-to-end session for which the ACF requires information. The PCF determines the nodes and links used for session interconnectivity, according to the request. The result may return both symmetrical and asymmetrical paths for certain topology sections. The PCF synchronizes with the TIM so that it computes valid paths. It must model the underlying network behavior to ensure that its results accurately reflect the path that traffic takes across the network. The ACF utilizes the PCF to subdivide the underlying network into the discrete segments required for any given admission control decision. The PCF provides the information necessary to permit the ACF to subdivide the admission control tasks across a number of underlying admission control functions, though this information comes from the TIM.

The PCF has two discrete, internal interfaces:

- An interface to the ACF through which the ACF requests path details on a given session request
- An interface to the TIM via the TSF through which the PCF can extract raw topology data and over which to determine path calculations. This interface is the same for ACF, TSF, PCF, and TSF.

## Admission Control Function

The Admission Control Function (ACF) receives and responds to session set-up requests from the Signaling Interface Function (SIF). Based on specific requests from the SIF, the ACF initiates the appropriate measures to determine the capability within the TIM to satisfy the requirements of any given session request. Using classical event *condition* mode, the ACF can support sophisticated policy decision rules. When the network (and network device) capability permits dynamic resizing of link bandwidth and queue capacity, the ACF can attempt to dynamically resize links for a positive admission control decision. The TIF handles this. The ACF relies on the PCF to determine the specific path to take through the topology. When a session is accepted, the ACF updates the Topology Information Model to reflect the available capacity left after accepting this session.

The ACF has several (internal) interfaces:

- An interface over which the SIF delivers call set-up requests.
- An interface to the PCF to calculate a path through the topology.
- An interface to the TSF to update the capacity of links in the TIM (when capacity is a resource attached to a link). This can utilize the same interface methods as the PCF TSF and TAF TSF interface functions.
- An interface to the TIF to handle requests for dynamic resizing.

## Signaling Interface Function

The Signaling Interface Function (SIF) handles messages to and from the requesting application. It provides the ACF (ACF) with the session parameters that allow the ACF to accept or deny the call request. The SIF supports signaling protocols at its northbound interface into the requesting application. The interface specifications closely follow the standardization efforts across all appropriate standards organizations. Its southbound interface into the ACF provides an abstraction layer from the choice of actual application signaling method.

The SIF has two discrete interfaces:

- An interface to the requesting application
- An interface into the ACF that provides an abstracted messaging model to the ACF

## Session Awareness Function

The Session Awareness Function (SAF) obtains dynamic session information. This relates to both network and device sessions. Sessions at the edge of the service provider network can provide elements that the overall admission control framework uses. The information supplied by the SAF helps map network sessions to physical network elements (devices and ports) and can therefore provide valuable information to the overall Topology Information Model.

The SAF has two discrete interfaces:

- An interface to the network over which it gleans session awareness (i.e., RADIUS, DHCP, e4 ETSI/TISPAN protocol)
- An interface to the Session Store Function (SSF) to allow storing session state within the session database

## Session Store Function

The Session Store Function (SSF) maintains session state information in the Session Information Model (SIM). Various Cisco functional components can interrogate the SSF. Data from the SSF populates dynamic topology element attributes in the Topology Information Model, or associates network sessions with particular topology elements. The SSF communicates with the other Cisco functional components through its interface.

## Context Store Function

The Context Store Function (CSF) determines how to store media contexts. A media context is a collection of media flows, each of which describes an IP stream via its endpoints, direction, capacity, and various other attributes. The CSF stores information about the TIM resources in use by each flow and the flow characteristics. This information releases the appropriate capacity from the TIM when handling a QoS release message. The CSF communicates with the other Cisco functional components through its interface.

## Context Timer Function

The Context Timer Function (CTF) is the mechanism that accomplishes the soft-state semantics for contexts and the resources they consume. Its responsibility is to periodically release expired contexts, and recoup their resources. It does this by interfacing with the ACF (for QoS release capability) and with the CSF (to retrieve and remove contexts).

The CTF has two interfaces and interacts with other components.

- An interface to register timeouts with the CTF, or initiate an immediate timeout check
- An interface to handle or initiate a timeout interval
- The CTF interfaces with the ACF to initiate a QoS release action for expired contexts. It uses a reserved Application Function Identifier to differentiate itself from an external QoS release.
- The CTF interfaces with the CSF to retrieve contexts to validate their expiration and retrieve necessary inputs to QoS release.

## Accounting Log Function

The Accounting Log Function (ALF) records entrance parameters, internal decisions, and exit responses. A session initiation event provides accounting-pertinent information and a correlation identifier, and the ALF stores the information in an appropriate manner. The ALF receives accounting information through its defined interface.

## Statistics Function

The Statistics Function (SF) records and queries system statistics. It provides a location for various components to store statistics concerning their runtime state and for other clients to inspect those statistics.

The system provides the following statistics:

- -<AF>.qos.<action>.count
- -<AF>.qos.<action>.accept.count
- -<AF>.qos.<action>.deny.count
- -<AF>.qos.<action>.error.count
- -<AF>.qos.<action>.replay.count

where,

<AF> is the Application Function ID (also, a reserved key for the sum total of statistics for all the AFs)

<action> is the request name (i.e., *reserve*, *modify*, *refresh*, *release*)

Certain statistics, such as *release deny*, do not increment. The SF interacts with its clients over its defined interface.

## Alarm Notification Function

The Alarm Notification Function (ANF) alerts external systems of aberrant behavior in the BPM by issuing SNMP traps. The ANF allows various BPM components to consistently report unexpected conditions, behavior, and life cycle changes.

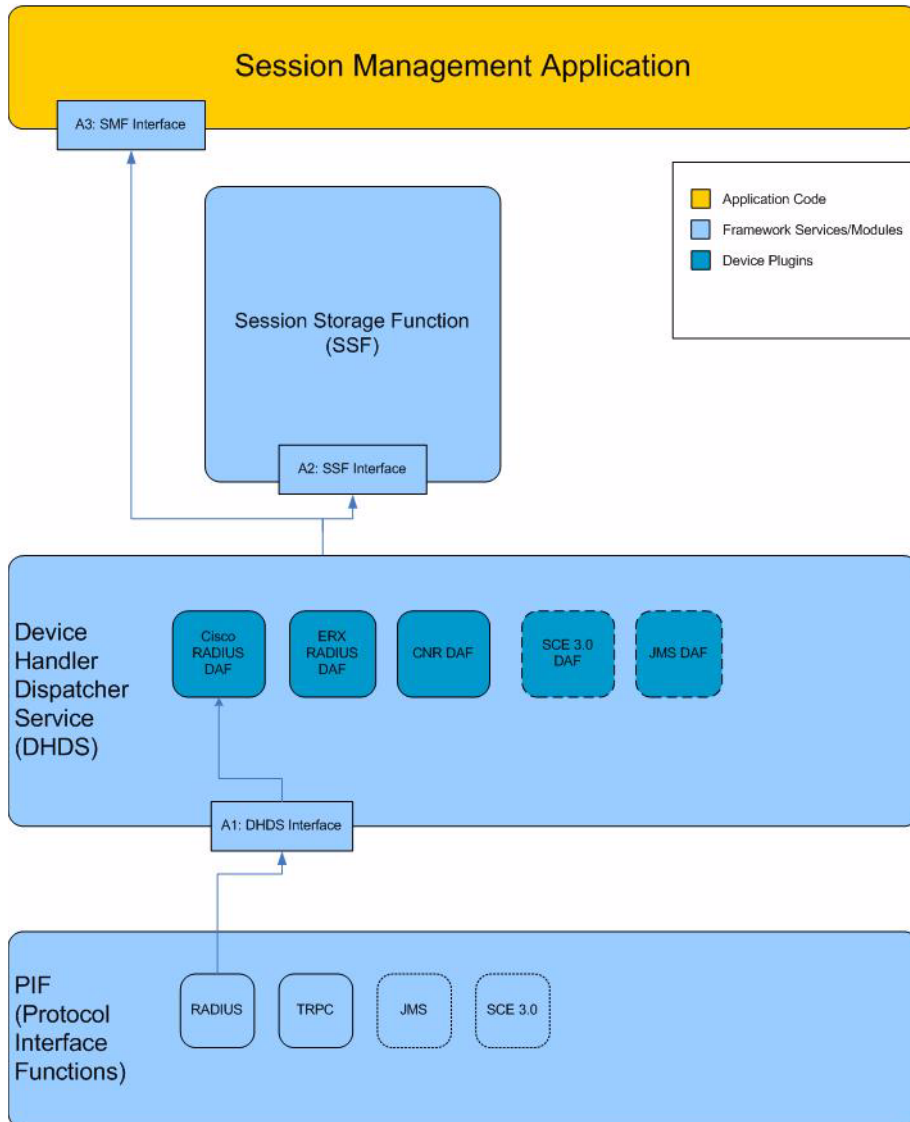
## Pool Function

The Pool Function (PF) encapsulates logic to resolve an IP address to a particular IP pool within the application domain. It segments pool sets based on virtual private networks (VPNs), so that each VPN defines a particular realm of pool information. The PF can add, update, and remove IP pools from its internal data structure. It also matches individual IP addresses against those pools. This matching allows other elements, such as the ACF in the Director to resolve dynamic IP addresses to their pools. Coupled with pool location knowledge, this determines the Resource Controller that is responsible for the realm containing the IP address.

## SM Architecture

Figure 3 shows the architecture of the Session Manager (SM):

**Figure 3. Session Manager Architecture**  
SESSION MANAGER



The Device Handler Dispatcher Service (DHDS) takes an incoming network message from a Protocol Interface Function (PIF), and determines which Device Adapter Function (DAF) to invoke.

The Session Storage Function (SSF) stores and retrieves session data in the repository.

The Session Manager consists of several system components:

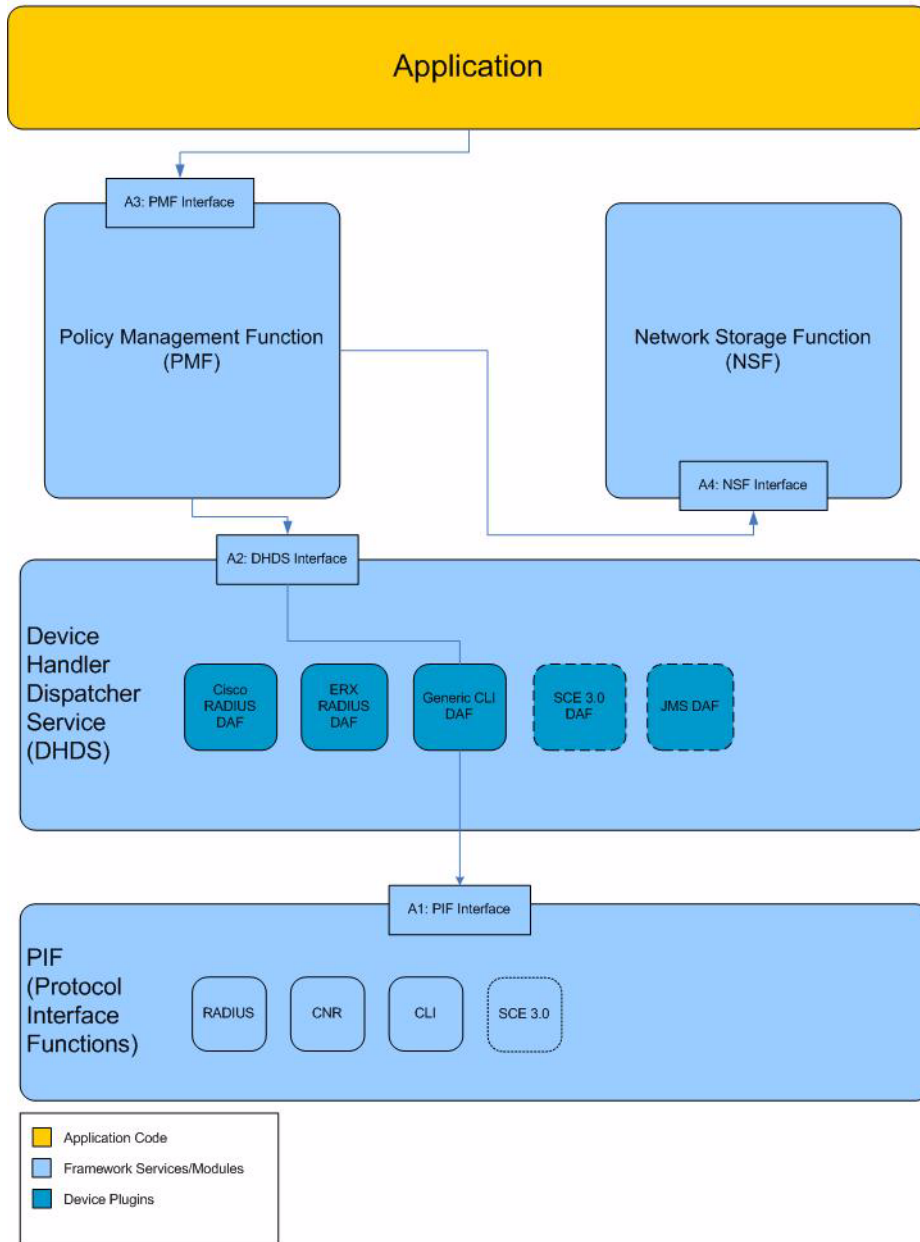
- Protocol Interface Function: Encapsulates a protocol interface towards external systems.
- Device Adapter Function: Defines how to handle events to and from PIFs, based on the specific device sending or receiving the event.

- Session Management Function: Notifies applications about changes in session state. Encapsulates customer-specific business logic applied to network sessions. SMFs are abstracted from specific protocols and devices used in the network through the DAF and PIF layers.

## NM Architecture

Figure 4 shows the architecture of the Network Manager (NM):

**Figure 4. Network Manager Architecture**  
NETWORK MANAGER



The Protocol Interface Function is an interface for extensions for NM, used by Network Adaptation functions to invoke commands on network elements.

The Device Handler Dispatcher Service (DHDS) takes a request from the Profile Management Function (PMF), and determines which NM Device Adapter Function (DAF) to invoke.

The Profile Management Interface is used by Session Management Applications to manage stateful network profiles associated with a user session. Profiles are defined by groups of actions that run against network elements to activate and deactivate the profile.

The Network Storage Function (NSF) stores and retrieves device and profile data in the repository.

Network Manager consists of several system components:

- **Device Adapter Function:** Defines how to handle events to and from PIFs, based on the specific device sending or receiving the event.
- **Network Manager API:** Provides an interface for applications to request services from the PMF or to invoke commands on the DAF itself.

## Deployment Components

There are some very specific high-level functional elements. These are the Director, Resource Controller, and Topology Database Server.

The Director is the functional element responsible for determining which Resource Controller component is responsible for a particular QoS request, coordinating the decisions that the Resource Controller component returned, and formulating a unified response.

The Resource Controller is the targeted element that is responsible for QoS requests for a particular part of the topology information model (for example, a particular access network or node).

The Topology Database Server is responsible for holding the global topology information model, from which Resource Controller components retrieve relevant segments, and the Director retrieves address pool information and the Resource Controller component mappings. This information is also for provisioning Resource Controllers and the Director.

## Relations to Functional Interfaces

### Director

This is one or more stateless installations that takes requests from the application function and routes them to appropriate Resource Controllers, to handle the specific incoming requests. Each Director system can determine which Resource Controller is appropriate to handle resource tracking for a given request (or given endpoint of a request) and route the request appropriately. Also, the Directors detect failures at the Resource Controller level and perform failovers from active of a pair to its standby, as needed.

The Director encapsulates a very specific type of ACF, one that uses address pool information to determine the responsible Resource Controller components, augments the request with local context (which segment of the path the Resource Controller is responsible for), and forwards to the Resource Controller components. The Director ACF also collects the responses and aggregates the results into a unified response for the SIF. In the case of differing responses from Resource Controllers (for example, one Resource Controller accepts one side of a call, but another rejects it), the ACF is responsible for restoring appropriate state to the Resource Controller components.

Thus, the Director represents:

- SIF to interface with the external application function (AF)
- ACF to perform a unified admission control decision (ACD)
- TSF to store the Director TIM, which holds pool and Resource Controller entries for each aggregating access node (BRAS), as well as a node role resource to distinguish BRAS nodes from CPEs
- TAF to react to external updates to the TIM, and to filter them into the internal TSF
- PF to determine the BRAS node associated with a particular IP address, via IP pool matching
- ALF for recording accounting information
- SF for recording statistics
- ANF for reporting alarms

### Resource Controller

The Resource Controller encapsulates an ACF that is responsible for performing the access network-level admission control decision (ACD) based on resource utilization.

Thus, the Resource Controller represents:

- SIF to interface with the external application function, which is the Director in this case (because this is internal, the SIF is transparent)
- TSF to maintain the TIM that the Resource Controller is responsible for and that resource consumption is checked against, as well as session and context information
- TAF to interact with the TSF, to react to topology changes (link or node activation or deactivation) by correctly cleaning up internal state and resource utilization
- SAF to interact with the TSF and react to session changes by correctly cleaning up internal state and resource utilization
- PCF to determine the path through the local topology that the context (call) transits
- CTF to enact the soft-state reservation model, to automatically remove orphaned or otherwise stale contexts and the resources they consume
- ACF to perform local ACD, based on TIM resource utilization
- ALF for recording accounting information
- SF for recording statistics
- ANF for reporting alarms

### Topology Database Server

The Topology Database Server maintains the global topology information model. It provides a centralized configuration location, and the functionality for distributing the information to appropriate clients. The Topology Database Server is responsible for pushing information about IP address pools and Resource Controller to BRAS mappings to all Directors in the Director role. Also, it holds the ownership indication for each segment of the network (that is, which Resource Controller component is responsible for a particular set of links and nodes).

The topology database also holds the responsibility of enacting repartitioning, which involves coordinating all other elements and migrating responsibility from one Resource Controller component to another for a piece of the network.

Thus, the topology database represents:

- TAF to interact with the TAF of Directors in the Director, as well as to push topology to Resource Controller components
- TSF to enact the local storage of the global TIM
- ALF for recording accounting information
- SF for recording statistics
- ANF for reporting alarms

### Firewall Requirements

To effectively communicate between nodes, any firewalls between the nodes must allow for all ports to be open. To do this, add a rule to each firewall that allows any port from the source node to the destination node, and vice versa.

### What's Next?

Once you finish this section, you can continue with [Chapter 2, Routine Tasks](#), which discusses day-to-day operations.



# Routine Tasks

## Overview

Operations tasks include day-to-day running of the Broadband Policy Manager (BPM). This may include examining information and monitoring components, such as service engines, agents, or services. You can perform many operations tasks using the BPM Broadband Policy Studio (BPS) or Broadband Policy Design Studio (BPDS). The BPS or BPDS gives a visual picture of the current BPM configuration. The BPDS or BPS uses standard windowing techniques and editing features.

This chapter discusses the following topics:

- *Starting and Stopping the BPM System*
- *BPDS or BPS Installation Procedure*
- *BPDS or BPS Login*
- *Observing Component Status*
- *Statistics*
- *SNMP Traps*
- *Accounting*
- *Error Messages*
- *System Status*
- *Role-Based Access Control (RBAC)*
- *Backup and Restore*

## Starting and Stopping the BPM System

To start or stop the BPM system, you must enter command line instructions.



**Note:** *When starting a domain, start the components in this order:*

- Domain Controller BPM
- Standby Topology Database Server BPM
- Active Topology Database Server BPM
- Standby Resource Controller BPM
- Active Resource Controller BPM
- Director BPM

### Procedure: Starting the Domain Controller BPM

To start the Domain Controller BPM, enter this command as user tazadmin:

```
/<tazz-install-directory>/bin/domaincontroller -start -clean
```

### Procedure: Starting the Other Systems

To start systems other than the Domain Controller BPM, enter this command as user tazadmin:

```
/<tazz-install-directory>/bin/start_tazz -clean
```

### Procedure: Stopping the Domain Controller BPM

To stop the Domain Controller BPM, enter this command as user tazadmin:

```
/<tazz-install-directory>/bin/domaincontroller -stop -force
```

### Procedure: Stopping the Other Systems

To stop systems other than the Domain Controller BPM, first make sure that the system is not part of a cluster, then enter this command as user tazadmin:

```
/<tazz-install-directory>/bin/stop_tazz -force
```



**Note:** *When shutting down a domain, shut down the components in this order:*

- Director BPM
- Standby Resource Controller BPM
- Active Resource Controller BPM
- Standby Topology Database Server BPM
- Active Topology Database Server BPM
- Domain Controller BPM



**Note:** *If the system is part of a cluster, uncluster first.*

## BPDS or BPS Installation Procedure

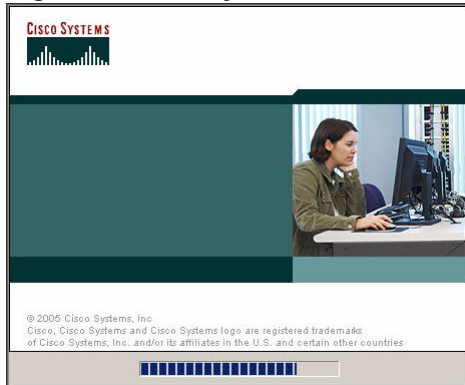
Depending on the product you are using, you may install either the BPDS or the BPS software. The instructions for installing each product are nearly identical.

Follow the steps in this section to install the BPDS or BPS.

### Procedure: Installing the BPDS or BPS

1. Insert the BPDS or BPS Installation CD into a drive on your computer. The BPDS or BPS Client Installer starts automatically and presents the initial **Cisco** screen (Figure 5).

**Figure 5. Cisco Systems Screen**



The system displays the **Introduction** screen.

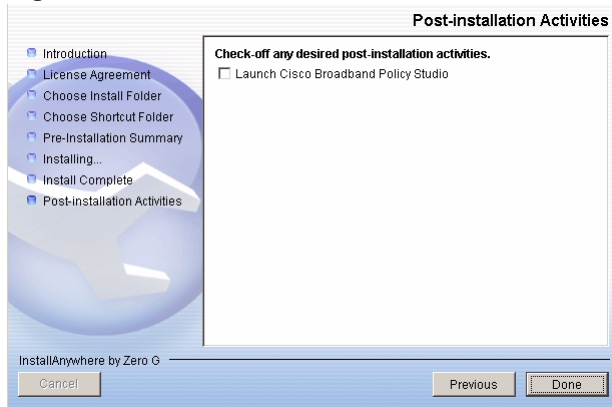
2. Click **Next** to continue. The system displays the license agreement.
3. Click **Next** to continue. The system displays the **Choose Install Folder** Screen.
4. Accept the default location selection and click **Next** to continue. The system displays the **Choose Shortcut Folder** screen.
5. Accept the default location selection and click **Next** to continue. The system displays the **Pre-Installation Summary** screen.
6. Review the installation details and click **Install** to continue.

The system displays the **Installing Broadband Policy Studio** or **Installing Cisco Broadband Policy Design Studio** screen.

7. When the system displays the **Install Complete** screen, click **Next**.

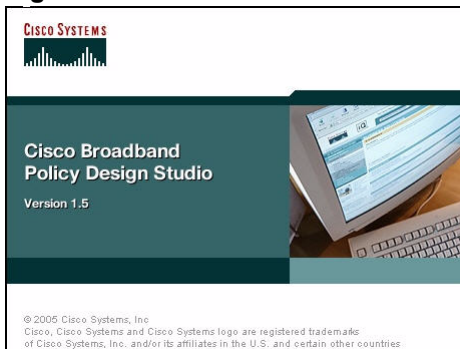
8. The system presents you with options to launch the BPDS or BPS (Figure 6). Check the option to Launch Studio and click **Done**.

**Figure 6. BPDS or BPS Post-installation Activities Screen**



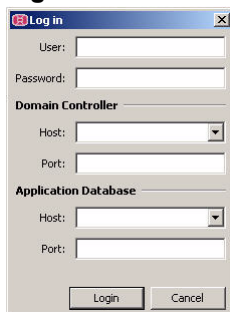
9. The system launches the BPDS or BPS. The system displays the **Welcome** screen (Figure 7).

**Figure 7. BPDS Welcome Screen**



The system displays the **Log in** screen (Figure 8).

**Figure 8. BPDS or BPS Log In Screen**



## BPDS or BPS Login

Depending on the product you are using, you may log in to either the BPDS or the BPS software. The instructions for logging in to each product are nearly identical.

When you see the **Log in** screen (Figure 8), you can log in to the BPDS or BPS. Refer to Table 2 to determine the appropriate authentication details for your system.

**Table 2. Log in Screen Information**

Item	Detail	Default
User	Name that identifies the user.	administrator
Password	Character string that verifies the user name.	administrator
Domain Controller Host	Name or IP address of the host system where the domain controller is running. The Domain Controller is a standalone system responsible for domain management, including application deployment, configuration, and health for all systems in the domain. Only one Domain Controller exists per domain.	<local_machine>
Domain Controller Port	Port number of the TCP port through which the domain controller communicates with the network.	10000
Application Database Host	Name or IP address of the host system where the application database is running. The application database stores information used by the application. Different applications require different data.	<local_machine>
Application Database Port	Port number of the TCP port through which the application database communicates with the network.	10005

### Procedure: Logging in to the BPDS or BPS System

Use this procedure to log in to the BPDS or BPS system.

1. Enter your user name in the **User** text box on the Authentication Screen.
2. Enter your password in the **Password** text box.
3. Enter the domain controller host name or IP address in the **Host** text box. Use the host name or IP address that you assigned to the domain controller.

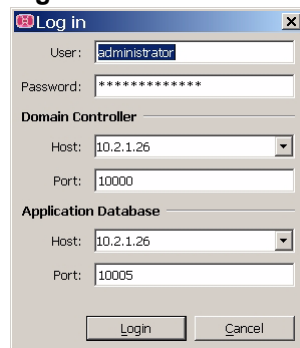
4. Enter the domain controller port number in the **Port** text box. Use the port number through which you want your domain controller to communicate with the network. This should be the number you assigned during installation (base install port).



**Note:** *Ensure that the default port number is open and available.*

5. Enter the application database host name or IP address in the **Host** text box. Use the host name or IP address that you assigned to the application database.
6. Enter the application database port number in the **Port** text box. Use the port number through which you want your application database to communicate with the network. This should be the number you assigned during installation (base install port + 5).
7. The display is similar to [Figure 9](#). Click **Login**.

**Figure 9. BPDS or BPS - Completed Log In Screen**



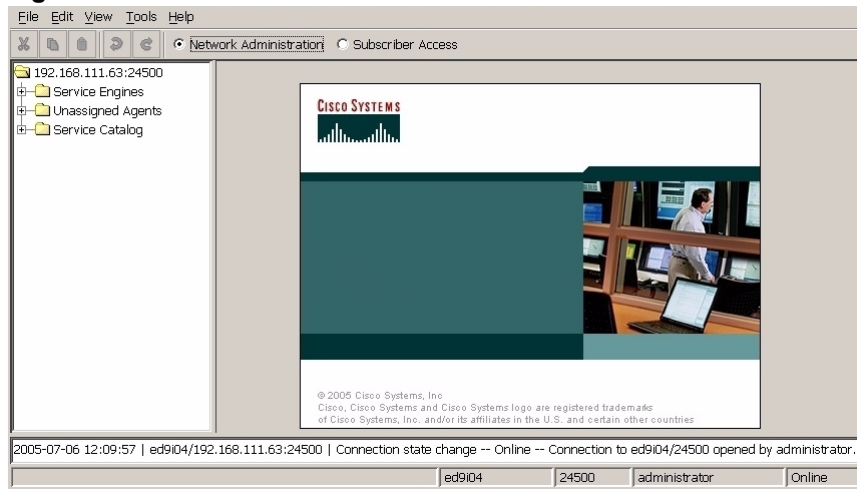
The screenshot shows a 'Log in' dialog box with the following fields and values:

- User: administrator
- Password: [masked]
- Domain Controller:
  - Host: 10.2.1.26
  - Port: 10000
- Application Database:
  - Host: 10.2.1.26
  - Port: 10005

Buttons: Login, Cancel

The system displays the initial BPDS or BPS screen.

**Figure 10. Initial BPDS or BPS Screen**



**Note:** *Your actual screen may look different from this sample screen.*

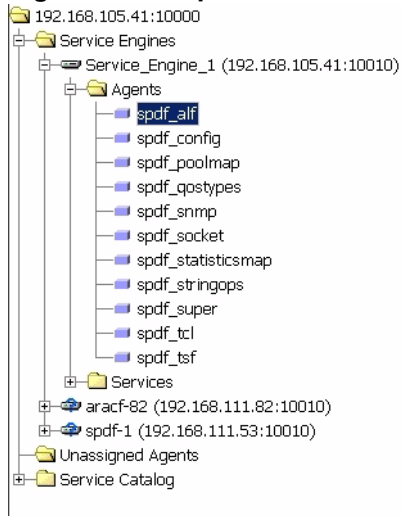
## Observing Component Status

You can observe the status of all BPM components using the BPS, including service engines, agents, and services. This can be useful for determining the installed components, their configuration, and current state.

## Network Administration Tree Pane

The Network Administration Tree pane (Figure 11) presents the network view of BPM objects (service engines, services, and agents) and indicates their status. The Tree pane allows you to choose and view objects in the BPM system.

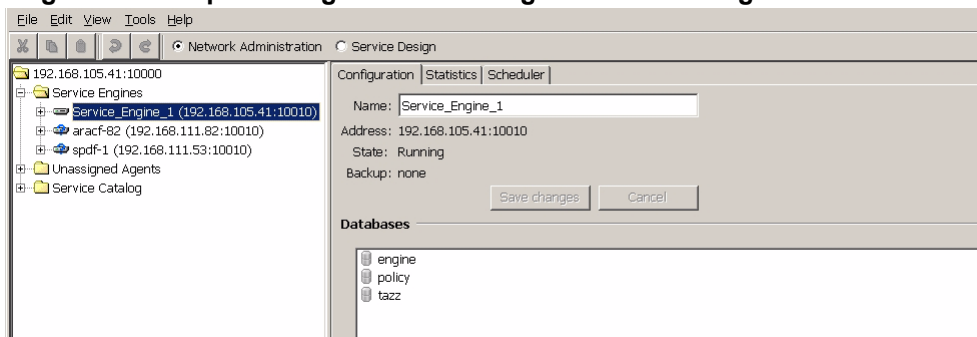
**Figure 11. Sample Network Administration Screen - Tree Pane**



## Service Engine Configuration

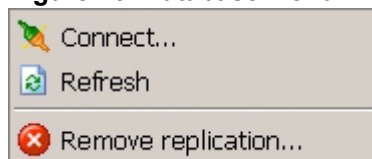
In the Network Administration view, the Configuration tab page for a service engine displays information about a specific service engine. Use this pane to obtain information on service engines and their databases, change values, and save those values. Figure 12 is an example of the display of the configuration tab page for a service engine. You can change the name of the service engine here.

**Figure 12. Sample Configuration Tab Page for Service Engine**



If you right-click one of the databases for a service engine, a drop-down menu appears. From this menu, you can change connections to the selected database, refresh the information for the selected database, or remove the database from replication.

**Figure 13. Database Menu**

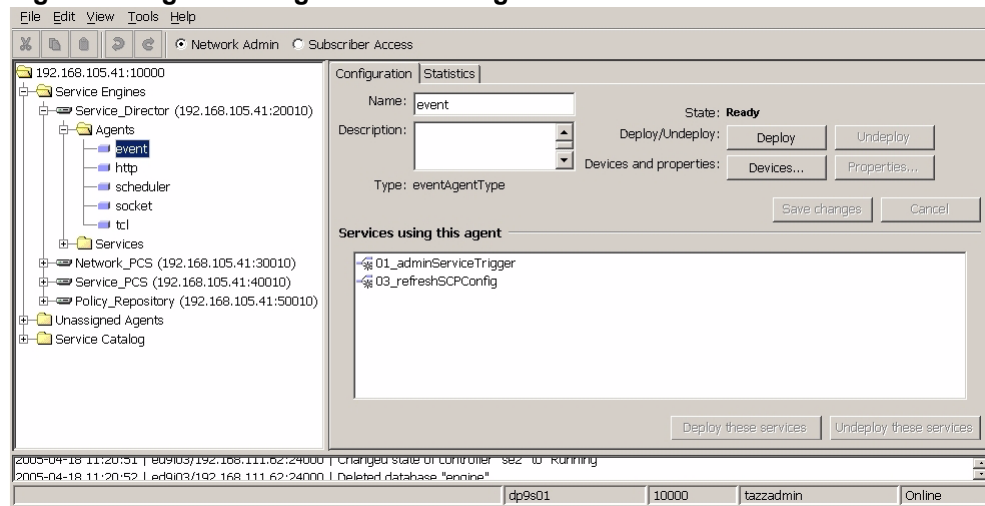




## Agent Configuration

In the Network Administration view, the Agent Configuration tab page displays information about a specific agent. Use this pane to enter or change agent values, access its Properties dialog box or Devices dialog box, and submit those values to the system. [Figure 14](#) is an example of a configuration tab page for an agent. For details on configuring an agent, see [Procedure: Changing Agent Configuration](#) on page 94.

**Figure 14. Agent Configuration Tab Page**



## Service Design Tree Pane

You can view information including Policy Functions and Rules.

### Procedure: Viewing Information about Policy Functions

You can view information about policy functions, including parameters, source rule, and source script. To view information about policy functions:

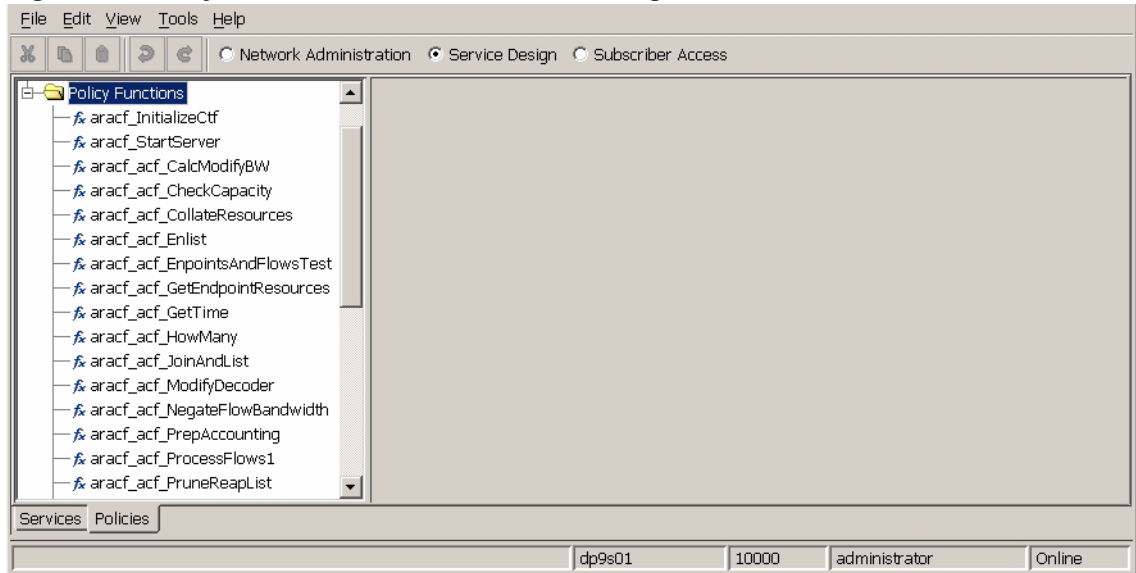
1. On the Service Design pane, click the **Policies** tab under the tree pane. The Policies tree pane appears.

**Figure 15. Policies Tree Pane on Service Design Pane**



- Open the **Policy Functions** folder. The Policy Functions folder opens, showing the currently available policy functions.

**Figure 16. Policy Functions Folder on Service Design Pane**



- Click a policy function. The selected policy function opens. Information on that policy function appears.

**Figure 17. Policy Function Information**

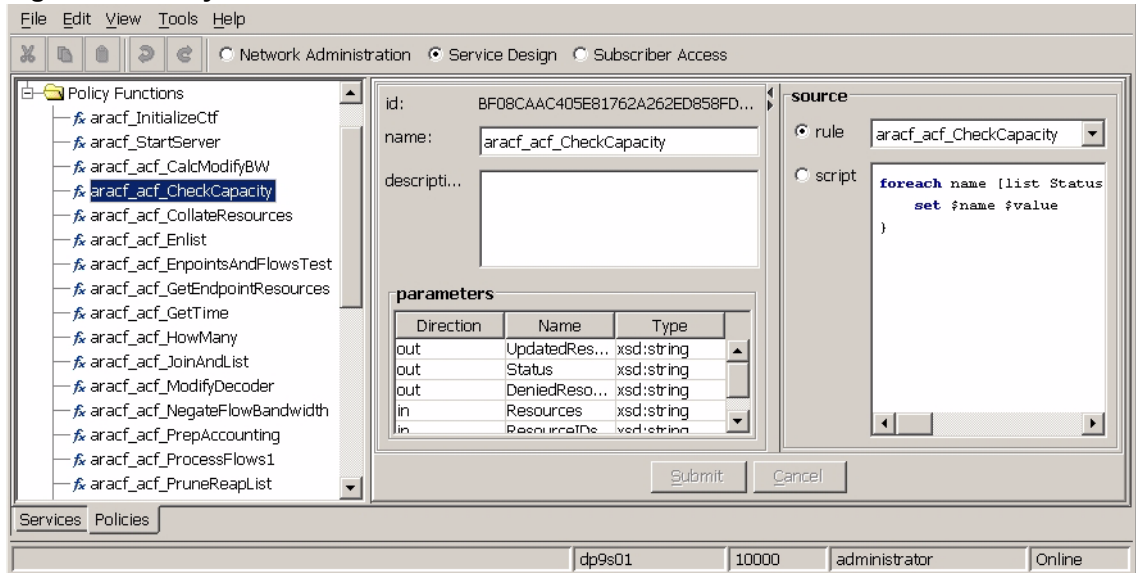


Table 3 presents the available information.

**Table 3. Policy Function Information**

Field Label	Meaning	Data Type	Details
description	description of object	string	
id	unique identifier	string	0-32 bytes
name	name of object	string	1-50 bytes
parameters (policy function)	input and output parameters to the current function		specific to function
source rule	rule that is the source of the policy function		
source script	script logic for the policy function		If checked, user can enter script directly in the window.

### Procedure: Viewing Information about Rules

You can view information about rules, including author, dependencies, parameters, and script.

To view information about rules:

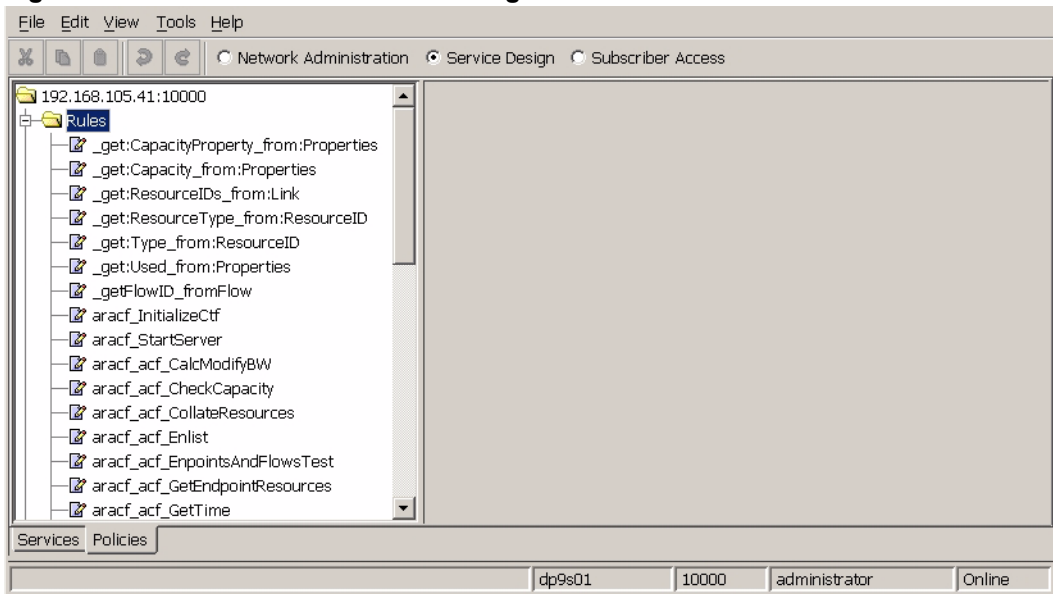
1. On the Service Design pane, click the **Policies** tab under the tree pane. The Policies tree pane appears.

**Figure 18. Policies Tree Pane on Service Design Pane**



- Open the **Rules** folder. The Rules folder opens, showing the currently available rules.

**Figure 19. Rules Folder on Service Design Pane**



- Click a rule. The selected rule opens. Information on that rule appears.

**Figure 20. Rule Information**

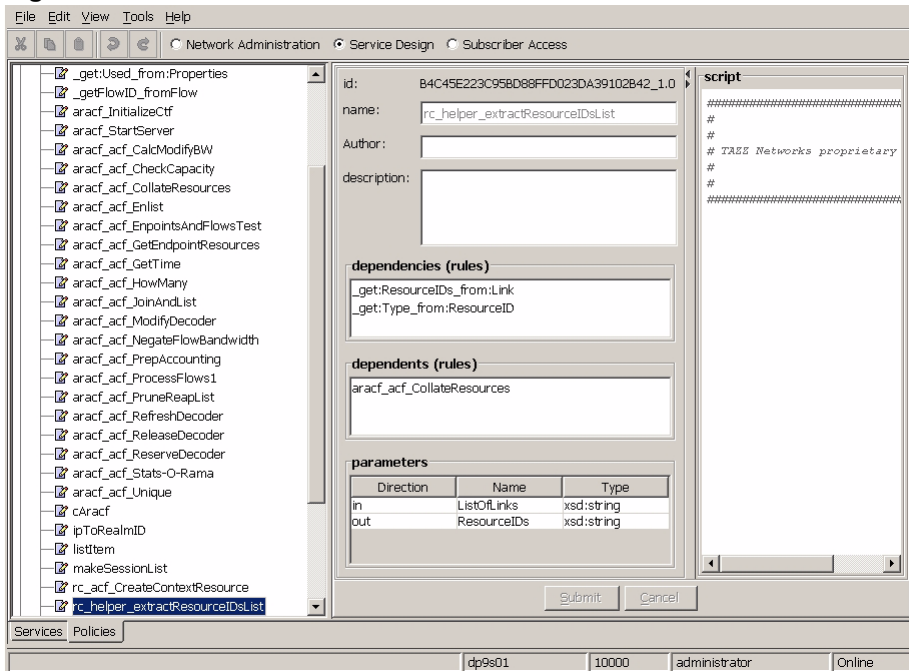


Table 4 presents the available information.

**Table 4. Rule Information**

Field Label	Meaning	Data Type	Details
Author	person who wrote the rule	string	
dependencies (rules)	displays all of the Rules that the current Rule calls	list	
dependents (rules)	displays all of the Rules that the current Rule calls	list	
description	description of object	string	
id	unique identifier	string	0-32 bytes
name	name of object	string	1-50 bytes
parameters (rule)	input and output parameters for the Rule		specific to Rule
script			

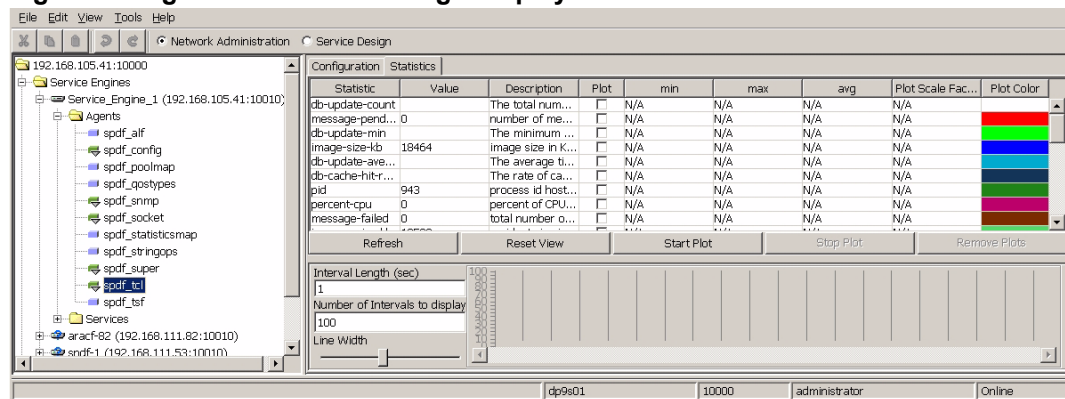
## Statistics

The BPS or BPDS allows you to display a variety of system statistics.

### Agent Statistics

The Network Administration Agent Statistics tab page displays statistical information about a selected agent on the BPM system. You can choose certain statistics to plot over time. Figure 21 is an example of an agent statistics tab page display.

**Figure 21. Agent Statistics Tab Page Display**



[Table 5](#) describes sample agent statistics.

**Table 5. Sample Agent Statistics and Details**

Statistic	Detail
reference-count	component reference count.
repository-name	Name of object in engine repository.
message-pending	Number of messages currently processing.
object-creation-date	Time at which object constructor called.
message-total	Total number of messages processed.
pid	Process id hosting agent.
message-failed	Total number of messages processed unsuccessfully.

## Other Statistics

You can examine statistics for other components by clicking the component in the Network Administration Tree pane, and clicking the Statistics tab. Other useful statistics that are available appear in [Appendix B - Statistics](#).

## SNMP Traps

The Simple Network Management Protocol (SNMP) TCP/IP networks provides a means to monitor and control network devices, and to manage configurations, statistics collection, performance, and security. Using SNMP, the BPM can integrate with a variety of industry standard network monitoring and fault management systems to report statistics, health and fault conditions.

The SNMP prefix used is 1.3.6.1.4.1.10782. The BPM uses sub category 10. Below that category is an index for the sub-component category, and then another (1) to indicate trap, and the trap type. Traps from the BPM are of the form:

```
1.3.6.1.4.1.10782.10.<component>.1.<ID>
```

Table 6 shows the components.

**Table 6. SNMP Components**

20	TDS
30	Director
40	Resource Controller
50	Presence Director

The SNMP codes for the ACM are:

```
1.3.6.1.4.1.10782.10.20.1.1 // TDS configure
1.3.6.1.4.1.10782.10.20.1.2 // TDS startup
1.3.6.1.4.1.10782.10.20.1.4 // TDS clear configuration,
externally initiated
1.3.6.1.4.1.10782.10.20.1.5 // TDS update or clear statistics
externally initiated
1.3.6.1.4.1.10782.10.20.1.6 // initiated failover of a
Resource Controller
1.3.6.1.4.1.10782.10.20.1.7 // notified of Resource Controller
startup
1.3.6.1.4.1.10782.10.20.1.8 // TDS failover occurred
1.3.6.1.4.1.10782.10.20.1.9 // TDS cluster dissolved

1.3.6.1.4.1.10782.10.30.1.1 // Director configure
1.3.6.1.4.1.10782.10.30.1.2 // Director startup
1.3.6.1.4.1.10782.10.30.1.4 // Director clear configuration,
externally initiated
1.3.6.1.4.1.10782.10.30.1.5 // Director update or clear
statistics externally initiated
1.3.6.1.4.1.10782.10.40.1.1 // Resource Controller configure
1.3.6.1.4.1.10782.10.40.1.2 // Resource Controller startup
1.3.6.1.4.1.10782.10.40.1.4 // Resource Controller clear
configuration, externally initiated.
```

The SNMP codes for the SM product are:

```
1.3.6.1.4.1.10782.10.40.0.30 // DHDS - no handler for session
Parameters: ControlPoint, Protocol, Action

1.3.6.1.4.1.10782.10.40.0.31 // Handler-Failed Creating Session
Parameters: ControlPoint, Protocol, Action
```

```

1.3.6.1.4.1.10782.10.40.0.32 // Handler-Failed Executing Action
Parameters: ControlPoint, Protocol, Action

1.3.6.1.4.1.10782.10.40.0.33 // Session Management Failure
Parameters: SessionID

1.3.6.1.4.1.10782.10.40.0.34 // Invalid DHDS URL
1.3.6.1.4.1.10782.10.40.0.35 // Error invoking DHDS
handler
    
```

The SNMP codes for the NM product are:

```

1.3.6.1.4.1.10782.10.40.0.61 // NM Apply Profile failure -
error removing profile
Parameters: SessionID, ProfileID, Step

1.3.6.1.4.1.10782.10.40.0.62 // NM Apply Profile failure -
error updating session
Parameters: SessionID, ProfileID, Step
    
```

The system also makes use of the Solaris SNMP mechanism to issue traps, in particular for the LinkDown network connectivity loss trap. This trap is useful to detect interface problems on the Link that connects to the SIP Server.

## Accounting

Access to the QosServer through the application function (AF) is logged into the application function log file `/tazz/logs/<type>.qos.log`, where `<type>` is SPDF (for Directors) or ARACF (for Resource Controllers). Each event includes one or more standard values. [Table 7](#) gives the values.

**Table 7. Accounting Events**

Value	Description
<time>	Time of event, in microseconds, since the epoch (Jan 1, 1970 at 12:00 AM), expressed as a decimal number.
<ReqID>	This is an integer request ID.
<Config-name-prefix>	This is the prefix given to the ClearConfiguration API. The service clears all configuration entries that match the prefix. For example, if you have a configuration with "a.b.c=1" and "a.b.d=2", and the ClearConfiguration is called with <Config-name-prefix> set to "a.b", then the call clears both entries out of the configuration, and the logged message includes "a.b" in this field.
<Result>	This is a result, in the form of an integer code, which, when converted to Hex, is identifiable. The high order byte represents error type: 0 = success 1 = warning 2 = failure
<Config-comma-separated>	This is a comma-separated list of the configuration values. Each entry is a name/value pair with a space between the name and the value.



**Table 7. Accounting Events** *(continued)*

<Stats-comma-separated>	This is a comma-separated list of the statistics values. Each entry is a name/value pair with a space between the name and the value.
Version	This is "1.0" for this release.
ApplicationId	This identifier differentiates various Application Function implementations to the Qos Server. The value is "TAZZ.Qos".
TransactionId	This uniquely identifies a message sent to a particular Interface. This identifier provides correlation capability for failure management and mediation. The uniqueness of a transaction ID is possibly different per interface.
ContextId	This parameter uniquely identifies a context containing the media streams. This identifier should be globally unique.
Src	This is the source of the operation, in the form ip@VPN. The IP address is in IPv4 format. Thus a sample src value is 192.9.100.3@VPN.
Dst	This is the destination of the operation, in the form ip@VPN. The IP address is in IPv4 format. Thus a sample dst value is 192.9.100.4@VPN.
Endpoints	The Endpoints field is a list of endpoint pairs. Each pair of endpoints is itself a list composed of a source endpoint followed by a destination endpoint. Each of the source and destination endpoints is a list of 4 items as described below: 1) Address: This is the address of the endpoint of the media. The flows are forward/reverse/bi-directional with respect to this address. This is an IP address (IPv4). 2) VPN: This is the identifier of the endpoint's VPN. This provides localized uniqueness to the Source Address if needed, and is not necessary if the Application does not require this level of segmentation. 3) ID: This identifies the endpoint in the TIM. This is not initially known, and may not be specified. When resolved later, the information can help bypass additional duplicate resolution, if the information is known to be relevant to the local topology. 4) Realm: This identifies the realm to which this endpoint belongs. If this information is present in a request, it represents the calling function's concept of realm, which the receiver may refine when handling the request.

**Table 7. Accounting Events** *(continued)*

Lease	This is the number of seconds for the reservation to remain valid. For a refresh message, the lease parameter is ignored. During a refresh, the lease is accepted. After refreshing the reservation, the original lease value is returned. For a modify, the lease is changed to the new passed value, provided it meets internal criteria. The request is always accepted, but if 0 is requested, the default of 200 is returned. The actual value kept internally is then returned. The lease is refreshed during any API call involving a context. It is valid for "Lease" more seconds after the reserve, modify, or refresh.
Priority	This is the specified priority of the context. This is a string.
Flows	A flow descriptor is a set of attributes describing a unidirectional media flow from defined source to defined destination. Each flow contains flow-id, direction, source-port, destination-port, flow-capacity-descriptor, dscp, and carried application id. Flows are not valid for Refresh and Release messages, and appear as empty parameters.
ResourceId	Indicates the resource type (vp or vc) and an id unique to that type of resource.
SessionId	This is an address in the form ip@VPN, similar to the definition of the Src and Dst fields.
FullContextId	This is a string of the form "realm RESOURCE {context <ContextID>}"
Role	"none", "active" or "inactive"
Status	This is the status code for the operation. These are base-10 values.

A minimum of two messages appear in the log for each function call to the application function (AF). Messages appear in the log on ingress and egress to an application function. If an error occurred an additional error message is logged that contains additional diagnostic information.

The format for each message type is described below. In general, each message format has one of three forms:

```
<time>, ENTER, <reqType>, <reqID>, <input1>, ..., <inputN>
<time>, EXIT, <reqType>, <reqID>, <output1>, ..., <outputN>
<time>, INTERNAL, <reqType>, <reqID>, <event>, <prop1>, ...,
<propN>
```

Particular log entries vary depending upon the inputs to the function; details are in the per-message sections below. Additionally, one of the following statements should also appear as the last entry in any rolled log file, to indicate the particular cause of the rollover:

```
<time>, FAILOVER
<time>, SYNC
<time>, ROLLOVER
<time>, UNDEPLOY
<time>, RESET
```

The FAILOVER statement occurs only on the active system of a Resource Controller cluster, when a failover occurs and the system is shutting down.

The SYNC statement occurs only on the standby system of a Resource Controller cluster, after successfully connecting to the active system and just prior to synchronizing their log file contents.

The ROLLOVER statement occurs whenever the log file is rolled over during normal operating procedures, whether due to the configured automatic rollover settings or due to a manual or scheduled invocation of the rolloverLog operator.

The UNDEPLOY statement occurs whenever the LogAgent is undeployed.

The RESET statement occurs if the configured log file already exists when the agent is deployed.

## Director Accounting Log Messages

The following messages go to the Director log file:

```

<time>, ENTER, ClearConfiguration, <ReqID>,
    <Config-name-prefix>
<time>, EXIT, ClearConfiguration, <ReqID>, <Result>

<time>, ENTER, ClearStatistics, <ReqID>, <Config-name-prefix>
<time>, EXIT, ClearStatistics, <ReqID>, <Result>

<time>, ENTER, Configure, <ReqID>, <Config-comma-separated>
<time>, EXIT, Configure, <ReqID>, <Result>

<time>, ENTER, IncrementStatistics, <ReqID>,
    <Stats-comma-separated>
<time>, EXIT, IncrementStatistics, <ReqID>, <Result>

<time>, ENTER, UpdateStatistics, <ReqID>,
    <Stats-comma-separated>
<time>, EXIT, UpdateStatistics, <ReqID>, <Result>

<time>, ENTER, StartServer, <ReqID>
<time>, EXIT, StartServer, <ReqID>, <Result>

<time>, ENTER, QosReserve, <ReqID>, <Version>, <ApplicationID>,
    <TransactionID>, <ContextID>, <Src>, <Dst>, <Lease>,
    <Priority>, <Flows>
<time>, EXIT, QosReserve, <ReqID>, <Status>, <Lease>

<time>, ENTER, QosModify, <ReqID>, <Version>, <ApplicationID>,
    <TransactionID>, <ContextID>, <Src>, <Dst>, <Lease>,
    <Priority>, <Flows>
<time>, EXIT, QosModify, <ReqID>, <Status>, <Lease>

<time>, ENTER, QosRefresh, <ReqID>, <Version>, <ApplicationID>,
    <TransactionID>, <ContextID>, <Src>, <Dst>, <Lease>,
    <Priority>, <Flows>
<time>, EXIT, QosRefresh, <ReqID>, <Status>, <Lease>
<time>, ENTER, QosRelease, <ReqID>, <Version>, <ApplicationID>,
    <TransactionID>, <ContextID>, <Src>, <Dst>, <Lease>,
    <Priority>, <Flows>
<time>, EXIT, QosRelease, <ReqID>, <Status>, <Lease>
<time>, ENTER, RCStartup, <ReqID>, RC-Resource
<time>, EXIT, RCStartup, <ReqID>, <Result>

```

## Resource Controller Accounting Log Messages

The following messages go to the Resource Controller log file:

```

<time>, ENTER, ClearConfiguration, <ReqID>,
    <Config-name-prefix>
<time>, EXIT, ClearConfiguration, <ReqID>, <Result>
<time>, ENTER, ClearStatistics, <ReqID>, <Config-name-prefix>
<time>, EXIT, ClearStatistics, <ReqID>, <Result>
<time>, ENTER, Configure, <ReqID>, <Config-comma-separated>
<time>, EXIT, Configure, <ReqID>, <Result>
<time>, ENTER, IncrementStatistics, <ReqID>,
    <Stats-comma-separated>
    <time>, EXIT, IncrementStatistics, <ReqID>, <Result>
<time>, ENTER, UpdateStatistics, <ReqID>,
    <Stats-comma-separated>
<time>, EXIT, UpdateStatistics, <ReqID>, <Result>

<time>, ENTER, StartServer, <ReqID>
<time>, INTERNAL, StartServer, <ReqID>, InitializeCtf, <Result>
<time>, EXIT, StartServer, <ReqID>, <Result>

<time>, ENTER, StartSession, <ReqID>, AccessAddress
<time>, INTERNAL, StartSession, <ReqID>, CreateSessionID,
    <SessionID>
<time>, EXIT, StartSession, <ReqID>, <Result>

<time>, ENTER, StopSession, <ReqID>, AccessAddress
<time>, INTERNAL, StopSession, <ReqID>, CreateSessionID,
    <SessionID>
<time>, EXIT, StopSession, <ReqID>, <Result>

<time>, ENTER, ContextTimeout, <ReqID>, <Full-ContextID>
<time>, INTERNAL, ContextTimeout, <ReqID>, TimestampMatch,
    <ContextID>
<time>, EXIT, ContextTimeout, <ReqID>, <Result>

<time>, ENTER, Failover, <ReqID>, <State>
<time>, INTERNAL, Failover, <ReqID>, CurrentRole, <Role>
<time>, INTERNAL, Failover, <ReqID>, InitializeCtf, <Result>
<time>, EXIT, Failover, <ReqID>, <Result>

<time>, ENTER, QosReserve, <ReqID>, <Version>, <ApplicationID>,
    <TransactionID>, <ContextID>, <Endpoints>, <Lease>,
    <Priority>, <Flows>
<time>, EXIT, QosReserve, <ReqID>, <Status>, <Lease>

<time>, ENTER, QosModify, <ReqID>, <Version>, <ApplicationID>,
    <TransactionID>, <ContextID>, <Endpoints>, <Lease>,
    <Priority>, <Flows>
<time>, EXIT, QosModify, <ReqID>, <Status>, <Lease>

```

```

<time>, ENTER, QosRefresh, <ReqID>, <Version>, <ApplicationID>,
  <TransactionID>, <ContextID>, <Endpoints>, <Lease>,
  <Priority>, <Flows>
<time>, EXIT, QosRefresh, <ReqID>, <Status>, <Lease>

<time>, ENTER, QosRelease, <ReqID>, <Version>, <ApplicationID>,
  <TransactionID>, <ContextID>, <Endpoints>, <Lease>,
  <Priority>, <Flows>
<time>, EXIT, QosRelease, <ReqID>, <Status>

```

## Topology Database Server Accounting Log Messages

The following messages go to the TDS log file:

```

<time>, ENTER, ClearConfiguration, <ReqID>,
  <Config-name-prefix>
<time>, EXIT, ClearConfiguration, <ReqID>, <Result>

<time>, ENTER, ClearStatistics, <ReqID>, <Config-name-prefix>
<time>, EXIT, ClearStatistics, <ReqID>, <Result>

<time>, ENTER, Configure, <ReqID>, <Config-comma-separated>
<time>, EXIT, Configure, <ReqID>, <Result>

<time>, ENTER, IncrementStatistics, <ReqID>,
  <Stats-comma-separated>
<time>, EXIT, IncrementStatistics, <ReqID>, <Result>

<time>, ENTER, UpdateStatistics, <ReqID>,
  <Stats-comma-separated>
<time>, EXIT, UpdateStatistics, <ReqID>, <Result>

<time>, ENTER, StartServer, <ReqID>
<time>, EXIT, StartServer, <ReqID>, <Result>

<time>, ENTER, StopServer, <ReqID>
<time>, EXIT, StopServer, <ReqID>, <Result>

<time>, ENTER, AddSystem, <ReqID>, <basePort>, <if1>, <ip1>,
  <state1>
  [, <if2>, <ip2>, <state2>][, <if3>, <ip3>, <state3>]
  [, <if4>, <ip4>, <state4>]
<time>, INTERNAL, AddSystem, <ReqID>, FailedAdd, <Errors>

```

where <Errors> is any one of the following ("\*" indicates that the previous set of values can appear multiple times):

- <nodeID> <addNode-Result>
  - for failed "add node" operations
- <resourceID> <addResource-Result> \*
  - for failed "add resource" operations
- <nodeID> <resourceID> <addResourceToNode-Result> \*
  - for failed "add resource to node" operations

```
<time>, EXIT, AddSystem, <ReqID>, <Result>
```

```
<time>, ENTER, DeploySystem, <ReqID>, <managementIP>,
  <basePort>, <role>
```

```
<time>, INTERNAL, DeploySystem, <ReqID>, FailedAddToNode,
  <Errors>
```

where <Errors> is ("\*" indicates that the previous set of values can appear multiple times):

```
•<nodeID> <resourceID> <addResourceToNode-Result> *
<time>, INTERNAL, DeploySystem, <ReqID>, FailedGetConfigs,
  <Result>
<time>, INTERNAL, DeploySystem, <ReqID>, FailedConfigure,
  <Result>
<time>, INTERNAL, DeploySystem, <ReqID>, FailedGetRealms,
  <Result>
<time>, INTERNAL, DeploySystem, <ReqID>, FailedGetCluster,
  <Result>
<time>, INTERNAL, DeploySystem, <ReqID>, FailedCopyRealms,
  <Errors>
```

where <Errors> is any one of the following ("\*" indicates that the previous set of values can appear multiple times):

```
•copy <realm> <copyRealm-Result> *
  for failed "copy realm" operations
•refresh <realm> <refreshRealm-Result> *
  for failed "refresh realm" operations
```

```
<time>, INTERNAL, DeploySystem, <ReqID>, FailedStartServer,
  <Result>
```

```
<time>, INTERNAL, DeploySystem, <ReqID>, FailedUpdateHealth,
  <Result>
```

```
<time>, INTERNAL, DeploySystem, <ReqID>, FailedUpdateDirectors,
  <Result>
```

```
<time>, EXIT, DeploySystem, <ReqID>, <Result>
```

```
<time>, ENTER, RemoveSystem, <ReqID>, <managementIP>,
  <basePort>
```

```
<time>, INTERNAL, RemoveSystem, <ReqID>, FailedGetResources,
  <Result>
```

```
<time>, INTERNAL, RemoveSystem, <ReqID>, FailedRemove, <Errors>
```

where <Errors> is any one of the following ("\*" indicates that the previous set of values can appear multiple times):

```
•<nodeID> <removeNode-Result>
  for failed "remove node" operations
•<resourceID> <removeResource-Result> *
  for failed "remove resource" operations
```

```
<time>, EXIT, RemoveSystem, <ReqID>, <Result>
```

```
<time>, ENTER, ResetSystem, <ReqID>, <managementIP>, <basePort>
<time>, INTERNAL, ResetSystem, <ReqID>, FailedReset,
  <Operation>, <Errors>
```

where <Errors> is any one of the following ("\*" indicates that the previous set of values can appear multiple times):

- health <updateResource-Result>  
for failed "update health property" operations
- configuration <updateResource-Result>  
for failed "update configuration property" operations
- aracfs <updateResources-Result>  
for failed "update A-RACF property" operations
- directors <distributeUpdates-Result> [<result2> <result3> ...  
(per failure)]  
for failed "distribute to directors" operations
- removeDirector <removeDirector-Result>  
for failed "remove director" operations
- StopServer <StopServer-Result>  
for failed "stop server" operations
- realm <realm> <dropRealm-Result> \*
- <resourceID> <removeResourceFromNode-Result> \*

```
<time>, EXIT, ResetSystem, <ReqID>, <Result>
```

```
<time>, ENTER, CreateCluster, <ReqID>, <mode>, <activeKey>,
  <standbyKey>
<time>, INTERNAL, CreateCluster, <ReqID>, FailedCreate,
  <Errors>
```

where <Errors> is any one of the following ("\*" indicates that the previous set of values can appear multiple times):

- <linkID> <addLink-Result>  
for failed "add link" operations
- <clusterID> <addResource-Result>  
for failed "add resource" operations
- <activeID> <getResourcesOfNode-Result>  
for failed "get resources of node" operations
- <activeID> <clusterID> <addResourceToNode-Result>  
for failed "add cluster ID to active node" operations
- <standbyID> <configurationID> <updateResource-Result>  
for failed "update standby configuration" operations
- <standbyID> <resourceID> <addResourceToNode-Result> \*

<time>, EXIT, CreateCluster, <ReqID>, <Result>

<time>, ENTER, DissolveCluster, <ReqID>, <clusterKey>  
 <time>, INTERNAL, DissolveCluster, <ReqID>, FailedDissolve,  
 <Errors>

where <Errors> is any one of the following ("\*" indicates that the previous set of values can appear multiple times):

- <linkID> <removeLink-Result>  
for failed "remove link" operations
- <clusterID> <removeResource-Result>  
for failed "remove cluster ID" operations
- <standbyID> <getResourcesOfNode-Result>  
for failed "get standby resources" operations
- <standbyID> <realmID> <removeResourceFromNode-Result> \*  
for failed "remove standby resources" operations

<time>, EXIT, DissolveCluster, <ReqID>, <Result>

<time>, ENTER, FailoverCluster, <ReqID>, <clusterKey>  
 <time>, INTERNAL, FailoverCluster, <ReqID>, FailedGet, <Result>  
 <time>, INTERNAL, FailoverCluster, <ReqID>, FailedSwap,  
 <Errors>

where <Errors> is any one of the following:

- <activeID> <getResourcesOfNode-Result>  
for failed "get resources of the active" operations
- <activeKey> <getPrimaryIP-Result>  
for failed "get primary IP of active" operations
- <standbyKey> <getPrimaryIP-Result>  
for failed "get primary IP of standby" operations

<time>, INTERNAL, FailoverCluster, <ReqID>, FailedUpdate,  
 <Errors>

where <Errors> is any one of the following:

- <linkID> <updateLink-Result>  
for failed "update link" operations
- <clusterID> <updateResource-Result>  
for failed "update cluster ID" operations
- aracfs <updateResources-Result>  
for failed "update A-RACF resources" operations
- directors <distributeUpdates-Result> [<result2> <result3> ...  
(per failure)]  
for failed "distribute updates to directors" operations

<time>, EXIT, FailoverCluster, <ReqID>, <Result>



```

<time>, ENTER, TsfTransactionExpired, <ReqID>, <TransactionID>
<time>, EXIT, TsfTransactionExpired, <ReqID>, <Prefix>

<time>, ENTER, AracfFailure, <ReqID>, <Host>
<time>, EXIT, AracfFailure, <ReqID>, <Result>

<time>, ENTER, AttemptFailover, <ReqID>, <NodeKey>
<time>, INTERNAL, AttemptFailover, <ReqID>, NoStandby, <NodeID>
<time>, INTERNAL, AttemptFailover, <ReqID>, StandbyNotHealthy,
    <StandbyNodeKey>
<time>, INTERNAL, AttemptFailover, <ReqID>, InitiateFailover,
    <Status>
<time>, EXIT, AracfFailure, <ReqID>, <Result>

```

## Session Manager Accounting Log Messages

The following messages go to the Resource Controller log file:

```

<time>, ENTER, HandlerEvent, <TraversalID>, <ControlPoint>,
    <Protocol>, <Event>
Description: Created by DHDS when it is invoked
<time>, EXIT, HandlerEvent, <TraversalID>, <Result>
Description: Created by DHDS when it completes processing an
    event

<time>, ENTER, SessionStart, <SessionID>, <Reason>
Description: Created by Device Adapter before calling Session
    Management Application Start
<time>, EXIT, SessionStart, <SessionID>, <Result>
Description: Created by Device Adapter after calling SMA::Start

<time>, ENTER, SessionModify, <SessionID>, <Reason>
Description: Created by Device Adapter before calling
    SMA::Modify
<time>, EXIT, SessionModify, <SessionID>, <Result>
Description: Created by Device Adapter after calling
    SMA::Modify

<time>, ENTER, SessionStop, <SessionID>, <Reason>
Description: Created by Device Adapter before calling SMA::Stop
<time>, EXIT, SessionStop, <SessionID>, <Result>
Description: Created by Device Adapter after calling SMA::Stop

```

## Network Manager Accounting Log Messages

The following messages go to the Resource Controller log file:

```

<time>, ENTER, PMFApplyProfile, <SessionID>, <Profile>
Description: Created by PMF when ApplyProfile is called
<time>, INTERNAL, PMFApplyProfile, <SessionID>, <Reason>
Description: Created by Device Adapter if profile apply fails
<time>, EXIT, PMFApplyProfile, <SessionID>, <Result>
Description: Created by Device Adapter after apply profile
    completes

```

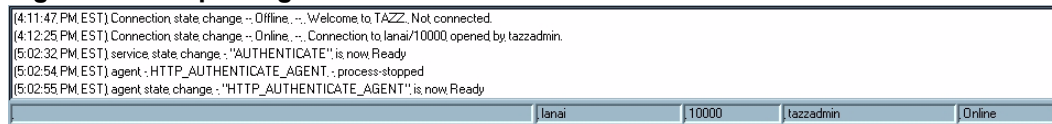
## Error Messages

The BPS displays certain BPM error messages.

## Log Pane

The Log pane (Figure 22) presents informational and error messages. This information can be useful for reviewing operations and for investigating problems.

**Figure 22. Sample Log Pane Content**



## Log Messages

In addition to the messages that the BPS displays, there are many log warning, error and fatal messages from components of the BPM system.

The BPM system, or *backend*, is composed of many components. Each component has the ability to log information into the BPM system log file. As of version 1.5 of the backend, this log file is named `tazz.log`, and is located in the `/opt/tazz/logs` directory. To view the file, enter the command:

```
tail -f /opt/tazz.log
```

or:

```
more tazz.log
```

A complete list of log messages is available in the separate *Log Messages* addendum and the *Application Log Messages* addendum.

## Log File Format

The format of the log file is uniform to facilitate automated processing. The format is as follows:

```

<log-file> ::= <log-line>*
<log-line> ::= <log-entry><crlf>
<crlf> ::= '\r\n'
<log-entry> ::= <date> <sp> <host> <sp> <obr> <port> <cbr> <sp>
               <obr> <severity> <cbr> <sp> <obr> <class> <cbr> <msg>
<date> ::= DDD MMM HH:MM:SS TZ YYYY (HH is 24 hr)
<sp> ::= ' '
<host> ::= {host on which BPM is installed}
<obr> ::= '['
<cbr> ::= ']'
<port> ::= {port}
<severity> ::= ('DEBUG' | 'INFO' | 'WARN' | 'ERROR' | 'FATAL')
<class> ::= {fully qualified namespace of component logging
              event}
<msg> ::= <text-line>+
<text-line> ::= <text><crlf>
<text> ::= {ascii text}
  
```

## Severity Definitions

Log entries receive a classification of one of the following severities:

**DEBUG:** BPM internal instrumentation

**INFO:** subsystem has changed state (e.g., user has logged in)

**WARN:** an abnormal condition arose, which did not affect the processing of the current request.

**ERROR:** an abnormal condition arose, which did or may prevent the completion of the current request. The system is able to continue processing requests.

**FATAL:** an abnormal condition arose, which prevents the subsystem from processing further requests. The affected subsystem is being shut down and restarted.

## Example Log Entry

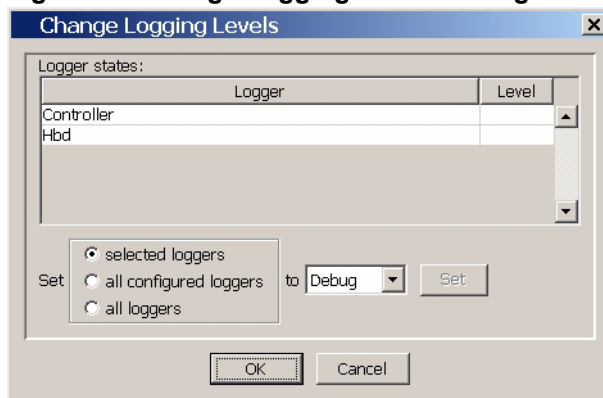
```
Mon Dec 20 17:12:50 EST 2004 localhost [4010] [ERROR]
[com.tazz.eventdispatcher.SocketAdapter] Error occurred accepting
connection: Socket closed
```

## Changing Logging Level for Service Engine

The Change Logging Levels dialog allows you to change the logging levels of loggers in a service engine:

1. Right-click the service engine in the Network Administration tree pane.
2. Choose **Change Log Level** from the drop-down list. The Change Logging Levels dialog opens.

**Figure 23. Change Logging Levels Dialog**



3. To change selected loggers, perform these steps:
  - a. Click desired loggers in the list.
  - b. Choose **selected loggers**.
  - c. Click the down-arrow, then choose the logging level from the drop-down list.
  - d. Click **Set**.
4. To change configured loggers, perform these steps:
  - a. Choose **all configured loggers**.
  - b. Click the down-arrow, then choose the logging level from the drop-down list.
  - c. Click **Set**.

5. To change all loggers, perform these steps:
  - a. Choose **all loggers**.
  - b. Click the down-arrow, then choose the logging level from the drop-down list.
  - c. Click **Set**.
6. To change individual loggers, right-click the **Level** column beside the logger, then choose the logging level from the drop-down list.
7. When done changing logging levels, click **OK**.

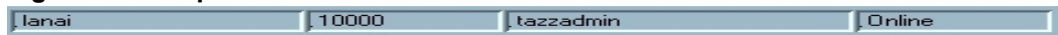
## System Status

You can observe system status information.

### Status Bar

The Status bar ([Figure 24](#)) provides system information.

**Figure 24. Sample Status Bar Content**



---

## Role-Based Access Control (RBAC)

The Role-Based Access Control (RBAC) function of the Broadband Policy Manager (BPM) allows administrators to control user access to actions and data in the BPM BPS. This section describes the elements of the BPM RBAC and shows how administrators can use RBAC for managing user permissions.

### Elements of the BPM RBAC

The BPM RBAC allows access control based on the roles that individual users are assigned. This section describes the elements of the BPM RBAC model:

- Roles
- Users
- Resources
- Actions
- Permissions

#### Roles

A role is a job function within the context of an organization, along with the authority and responsibility conferred on the user assigned to the role. For example, an *agent monitor* role is granted the permission to view agent statistics, but is denied permissions to perform other actions on agents, for example, deploy, undeploy, and modify.

#### Users

A user is a person using the BPM Broadband Policy Studio (BPS). Users are assigned to one or more roles.

#### Resources

A resource is anything in the BPM BPS that is subject to access control. Examples of resources are Report Manager and service engines.

#### Actions

An action is any task that a user can perform in the BPM BPS that is subject to access control. An action is performed on a resource. Examples of such actions are: *deploy Agents*, *modify Agents*, and *Use Network Admin View*.

#### Permissions

Roles are granted or denied permission to perform an action. For example, the *agent monitor* role is denied permission to perform the *deploy Agents* action.

#### *Per-user Overrides*

By default, a user has the same permissions as the roles to which he is assigned. However, user permissions can be different from those of his assigned roles. For example, a user with the *agent monitor* role, by default, is denied permission to perform the *deploy Agents* action. An administrator can grant a user the *deploy Agents* permission without granting the same permission to every user with the same roles.

### User Belonging to Multiple Roles

If a user belongs to more than one role, and these roles have conflicting permissions on the same action, then a *deny* takes precedence over a *grant*. For example, a user may have two roles, *agent monitor* and *agent deployer*. Although the *agent monitor* role denies permission to *deploy Agents*, the *agent deployer* role grants permission to *deploy Agents*. In this case, the user is denied permission to *deploy Agents* (unless there is a separate per-user override granting permission to *deploy Agents*).

## Using RBAC

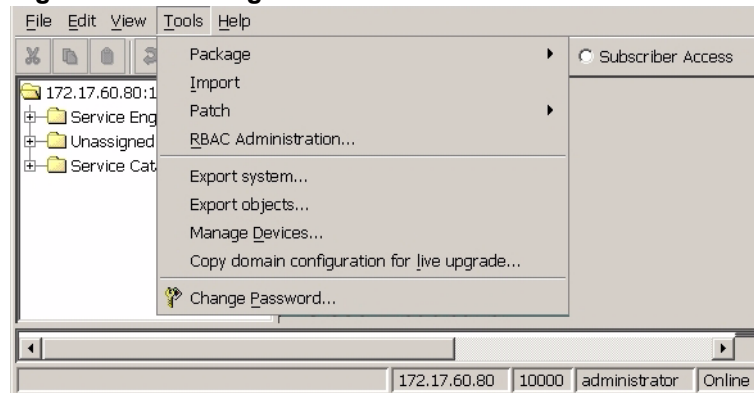
You can use RBAC to arrange access in a variety of ways. This section describes the RBAC display and how to use it to accomplish common tasks.

### Starting RBAC

To start RBAC, follow these steps:

1. From the BPS, choose **Tools --> RBAC Administration** (Figure 25).

**Figure 25. Selecting RBAC Administration**



2. The RBAC window appears (Figure 26).

**Figure 26. Main RBAC Display**



This window displays a tree diagram that can include current RBAC domains, users, and roles.

## Roles

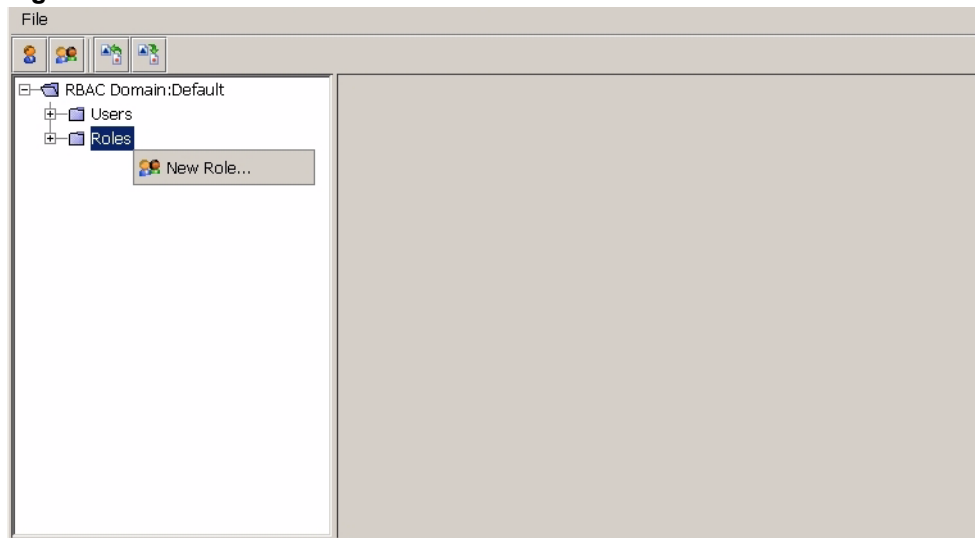
You can view all the currently defined roles and the access details that each role defines. You can also create new roles, and modify or delete existing roles.

### Creating Roles

To create a new role, follow these steps:

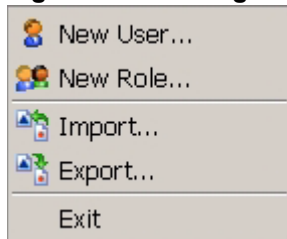
1. In the RBAC window, right-click on the **Roles** folder (Figure 27).

**Figure 27. Roles Folder**



2. Click **New Role**. Alternatively, you can choose **File --> New Role** (Figure 28).

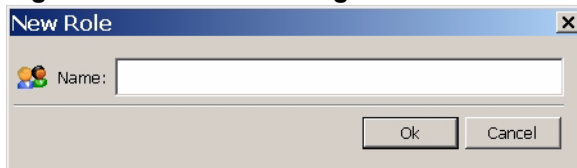
**Figure 28. Creating New Role**



Alternatively, you can click the **New Role** icon, above the tree diagram.

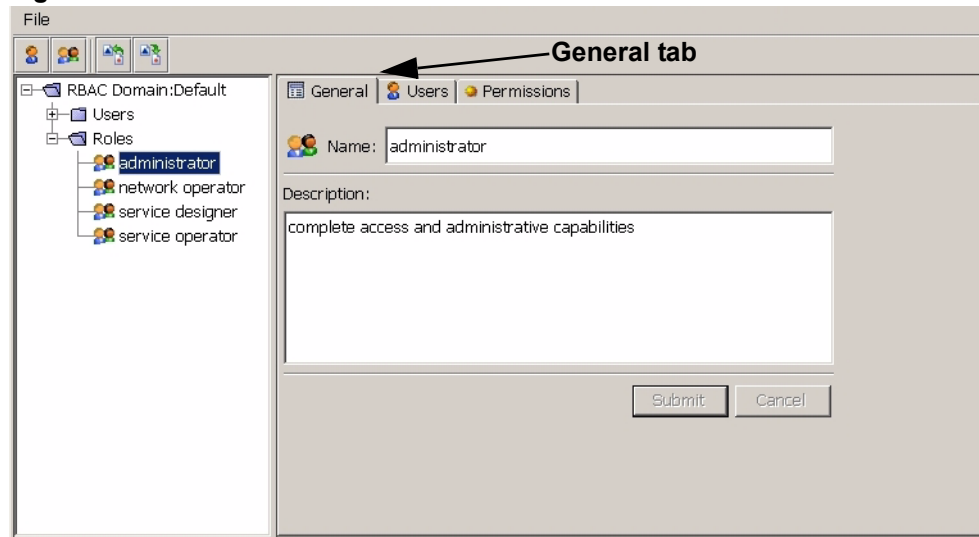
3. The **New Role** dialog appears. Enter a **Name** for the new role, and click **OK** (Figure 29).

**Figure 29. New Role Dialog**



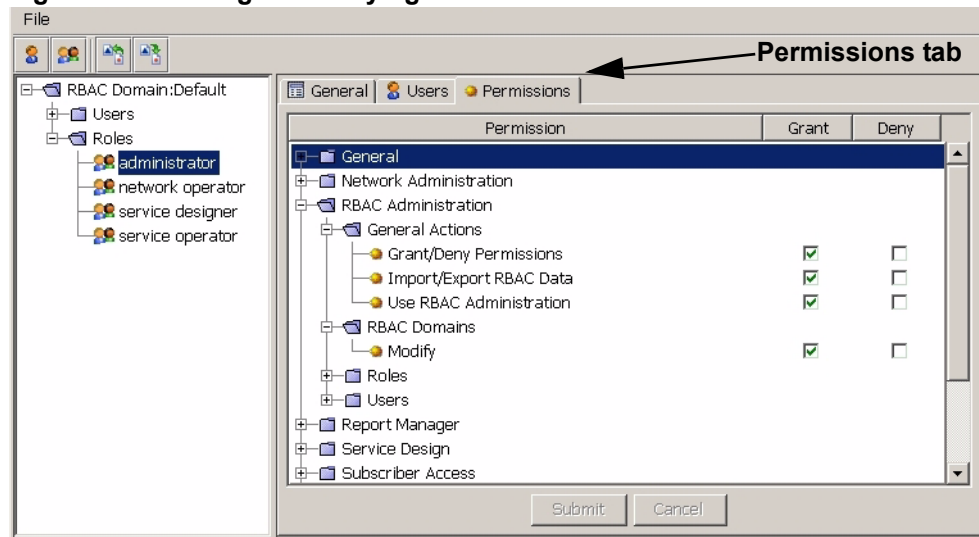
- The new role is created. You can optionally enter a description of this role in the **Description** field on the **General** tab, then click **Submit** (Figure 30).

**Figure 30. General Tab for Roles**



- To grant and deny permissions for the new role, click the **Permissions** tab (Figure 31).

**Figure 31. Granting and Denying Permissions**



A tree of available permissions appears. For each permission, check **Grant** or **Deny**.

Permissions are color coded as follows:

- Green: the permission is explicitly granted or denied.
  - Gray: the default value of the permission.
- Click a gray or blank permission. It becomes green to show that you selected it explicitly. Click a green permission, to reset the default permission to gray.
  - When finished with permissions, click **Submit**.

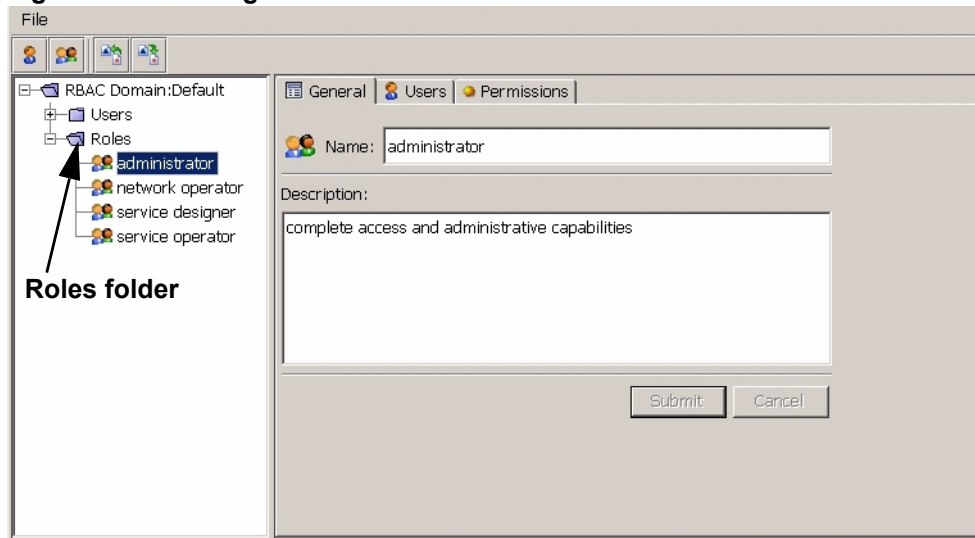


## Viewing and Modifying Roles

To view or modify existing roles, follow these steps:

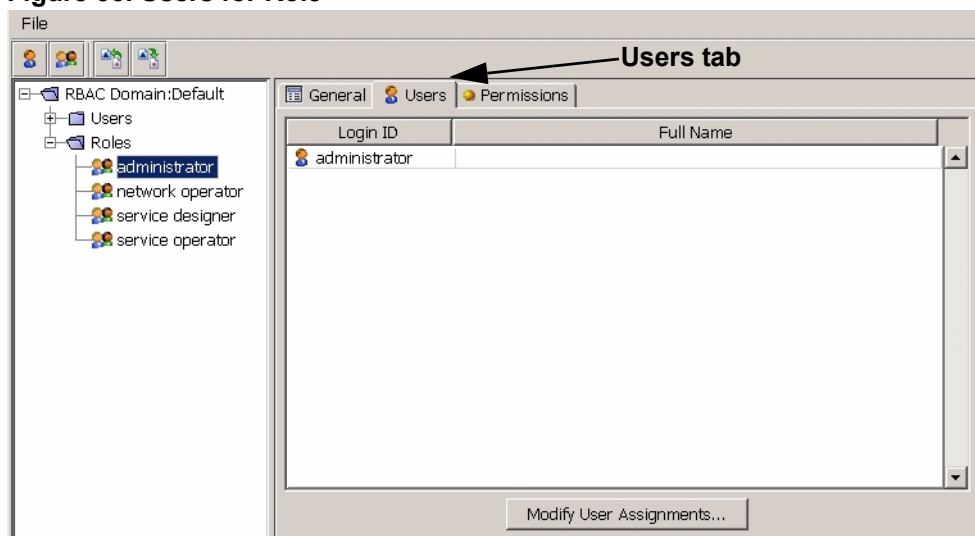
1. From the **Roles** folder, choose the role to view or modify (Figure 32).

**Figure 32. Selecting Role**



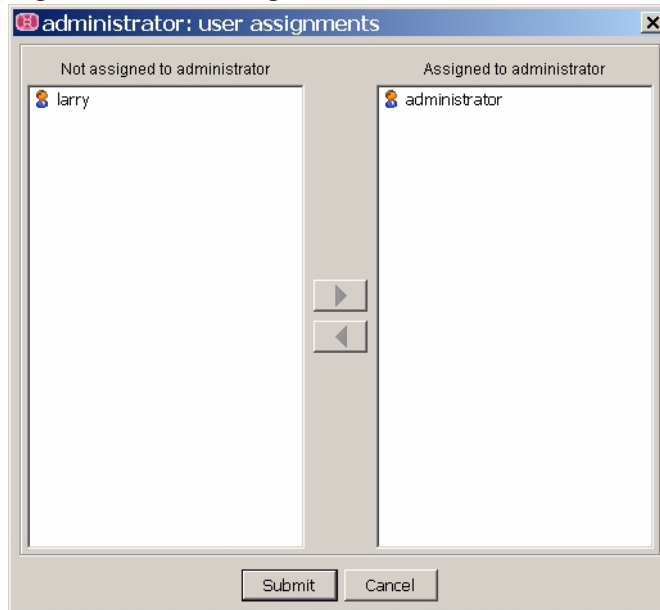
2. If not selected, click the **General** tab. This shows the current **Name** and **Description** of the role. You can change the name of the role by modifying the text in the **Name** field. You can create or modify the description of the role by entering the text in the **Description** field. Click **Submit** when finished.
3. If not selected, click the **Users** tab. This shows the users who currently have this role (Figure 33).

**Figure 33. Users for Role**



4. To add or remove users for a role, click **Modify User Assignments** (Figure 33). The **user assignments** window for this role appears (Figure 34).

**Figure 34. User Assignments Window**



5. To add a user to this role, choose the user from the **Not assigned** list and click the right-arrow (Figure 34). To remove a user from this role, choose the user from the **Assigned** list and click the left-arrow.

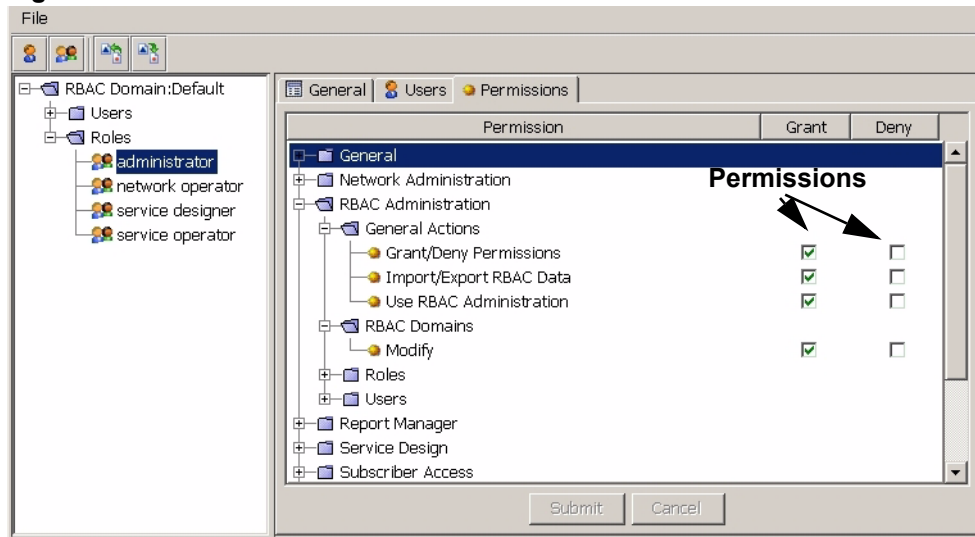


**Caution:** *Removing a user from a role may prevent that user from performing necessary tasks.*

Click **Submit** when finished. The result of your changes appears on the **Users** tab.

6. If not selected, click the **Permissions** tab. This shows the permissions for this role (Figure 35).

**Figure 35. Permissions for Role**



Permissions are color coded as follows:

- Green: the permission is explicitly granted or denied.
  - Gray: the default value of the permission.
7. To change permissions for this role, click **Grant** or **Deny** for the permission you wish to change (Figure 35).



**Caution:** *Changing permissions for a role may prevent users with that role from performing necessary tasks.*

If you click a gray or blank permission, it becomes green to show that you selected it explicitly. If you click a green permission, the default permission resets to gray.

8. Click **Submit** when finished.

### Deleting Roles

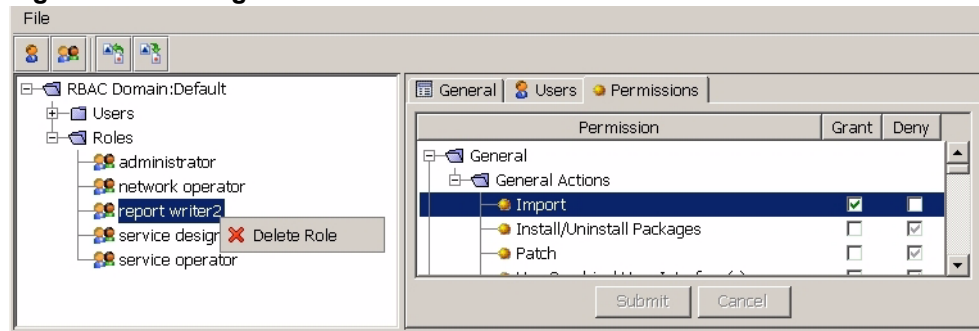
To delete a role, follow these steps:

1. Right-click the role to delete from the tree list, and choose **Delete Role** (Figure 36).



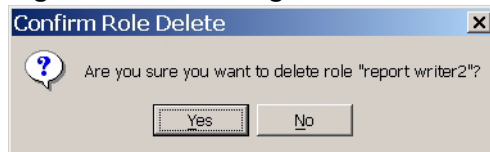
**Caution:** *Deleting a role may prevent users with that role from performing necessary tasks.*

**Figure 36. Deleting Role**



2. The **Confirm Role Deletion** dialog appears (Figure 37). Click **Yes**. The role is deleted.

**Figure 37. Confirming Role Deletion**



### Users

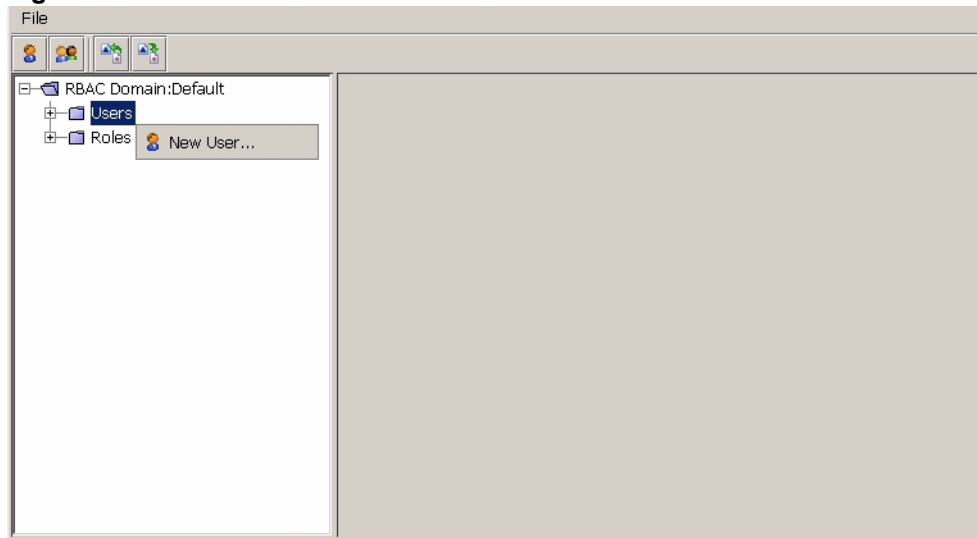
You can view all the users and their roles and permissions. You can also create new users and modify or delete existing users.

## Creating Users

To create a new user, follow these steps:

1. In the RBAC window, right-click on the **Users** folder (Figure 38).

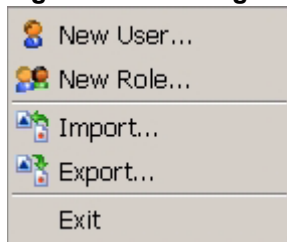
**Figure 38. Users Folder**



2. Click **New User**.

Alternatively, you can choose **File --> New User** (Figure 39).

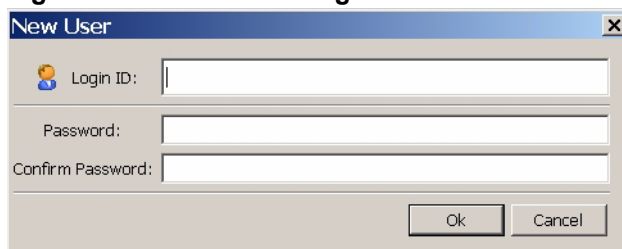
**Figure 39. Creating New User**



Alternatively, you can click the **New User** icon, above the tree diagram.

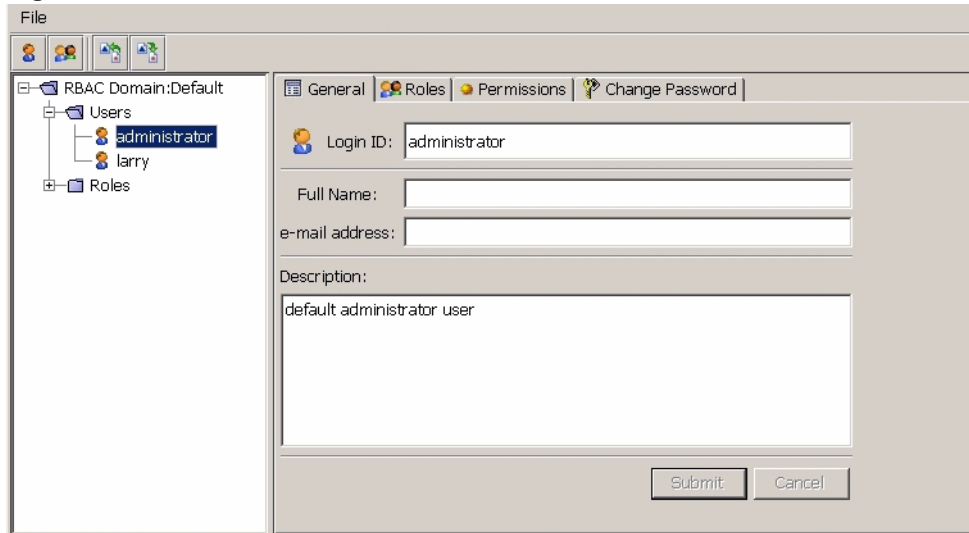
3. The **New User** dialog appears (Figure 40). Enter a **Name** for the new user. Enter and confirm a **Password** for the new user. When finished, click **OK**.

**Figure 40. New User Dialog**



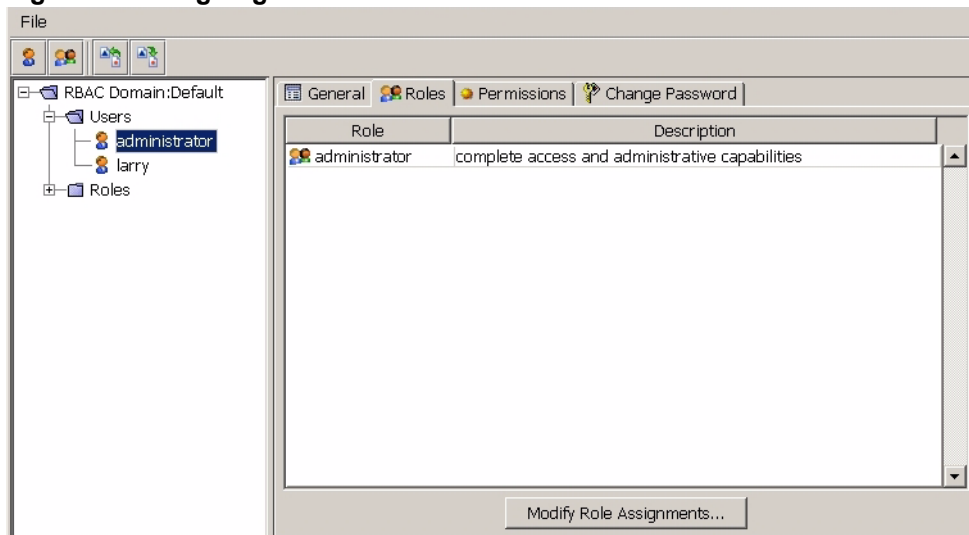
- The new user is created. On the **General** tab, you can optionally enter the name of the user in the **Full Name** field, the user e-mail address in the **e-mail address** field, or a description of this user in the **Description** field, then click **Submit** (Figure 41).

**Figure 41. General Tab for Users**



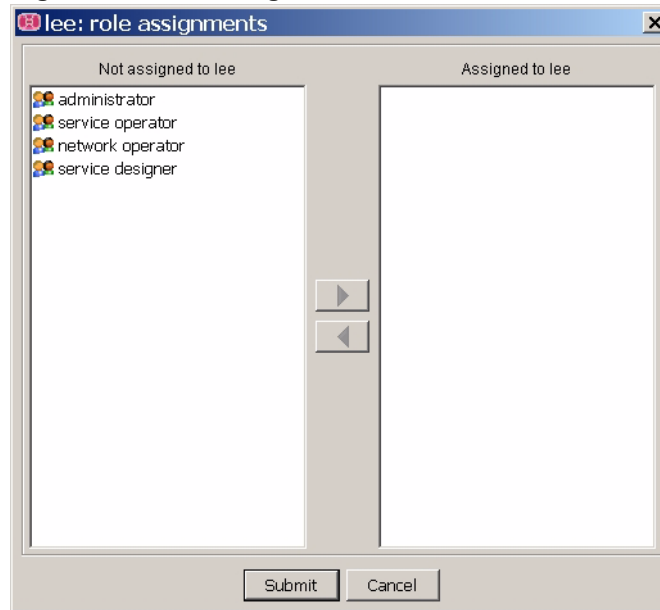
- To assign roles for the new user, click the **Roles** tab (Figure 42).

**Figure 42. Assigning Roles**



- To add or remove roles for a user, click **Modify Role Assignments**. The **role assignments** window for this user appears (Figure 43).

**Figure 43. Role Assignments Window**



- To add a role for this user, choose the role from the **Not assigned** list and click the right-arrow. To remove a role from this user, choose the role from the **Assigned** list and click the left-arrow.

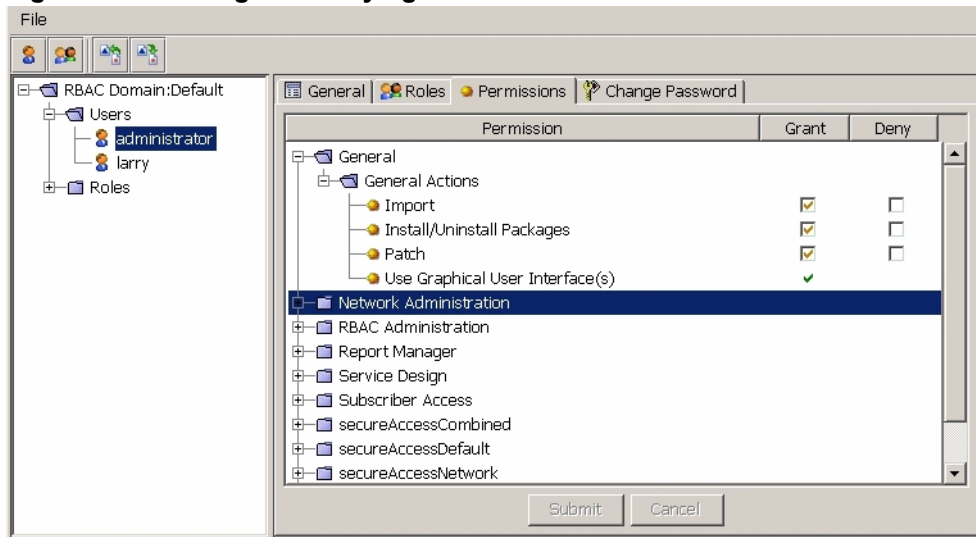


**Caution:** *Removing a role from a user may prevent that user from performing necessary tasks.*

- Click **Submit** when finished. The result of your changes appears on the **Roles** tab.

- To grant and deny permissions for the new user, click the **Permissions** tab (Figure 44).

**Figure 44. Granting and Denying Permissions**



A tree of available permissions appears.

- For each permission, check **Grant** or **Deny**. This granting or denying applies only to the selected user. If you want to grant or deny the same permission to everyone with the same role as this user, change the permission for the role, not for the individual user.

Permissions are color coded as follows:

- Orange: the permission is inherited from one of the roles of the user.
- Green: the permission is overridden for this user.
- Gray: the default value of the permission.

In addition, some permissions appear without a checkbox. These permissions are read-only, and you cannot change them. This is done to prevent, say, an administrator from locking himself out of the system.

If you click an orange, gray, or blank permission, it becomes green to show that you selected it explicitly. If you click a green permission, the original inherited or default permission resets to orange or gray.

- When finished with permissions, click **Submit**.

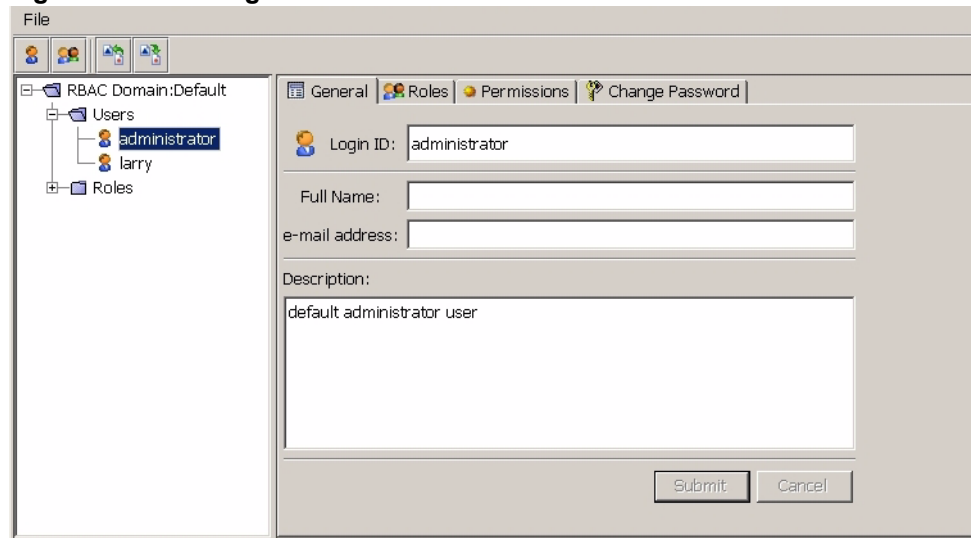


## Viewing and Modifying Users

To view or modify existing users, follow these steps:

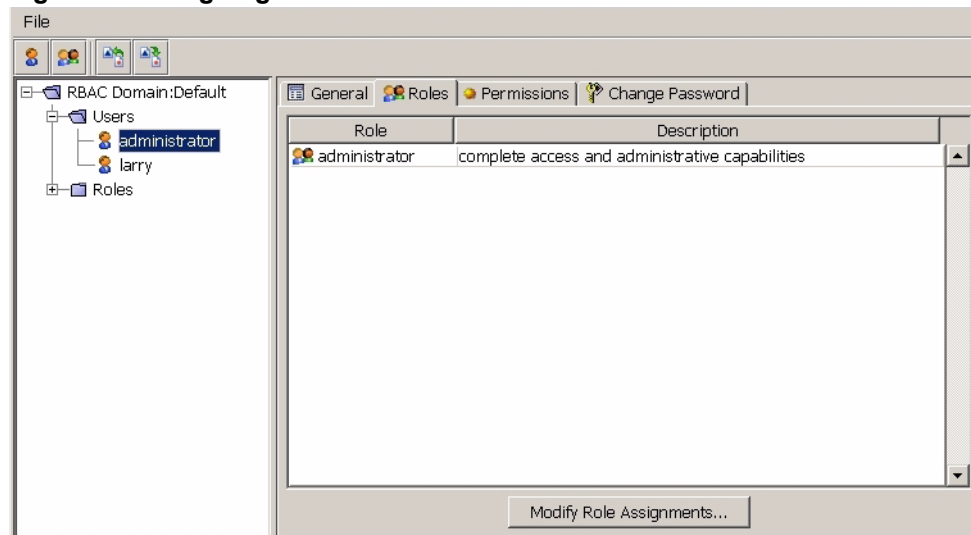
1. From the **Users** folder, click the user to view or modify (Figure 45).

**Figure 45. Selecting User**



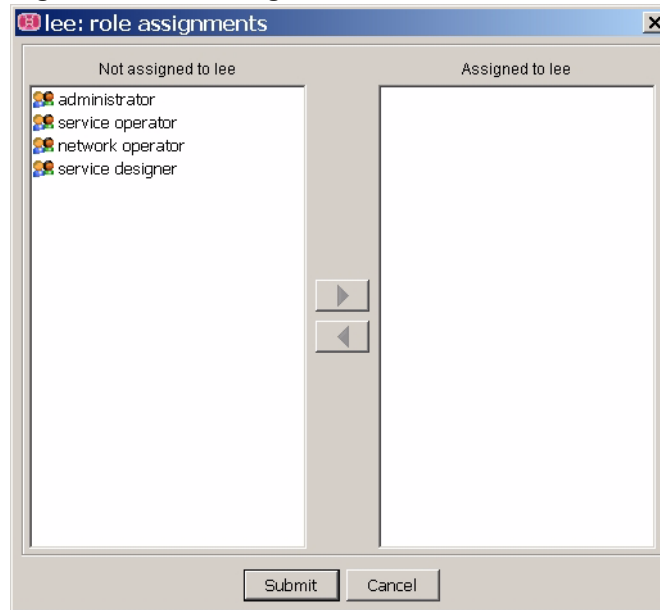
2. If not selected, click the **General** tab (Figure 45). This shows the current **Login ID**, **Full Name**, **e-mail address**, and **Description** of the user. You can change any of these items by modifying the text in the field. Click **Submit** when finished.
3. To view or modify roles for the user, click the **Roles** tab (Figure 46).

**Figure 46. Assigning Roles**



- To add or remove roles for a user, click **Modify Role Assignments**. The **role assignments** window for this user appears (Figure 47).

**Figure 47. Role Assignments Window**



- To add a role for this user, choose the role from the **Not assigned** list and click the right-arrow. To remove a role from this user, choose the role from the **Assigned** list and click the left-arrow.

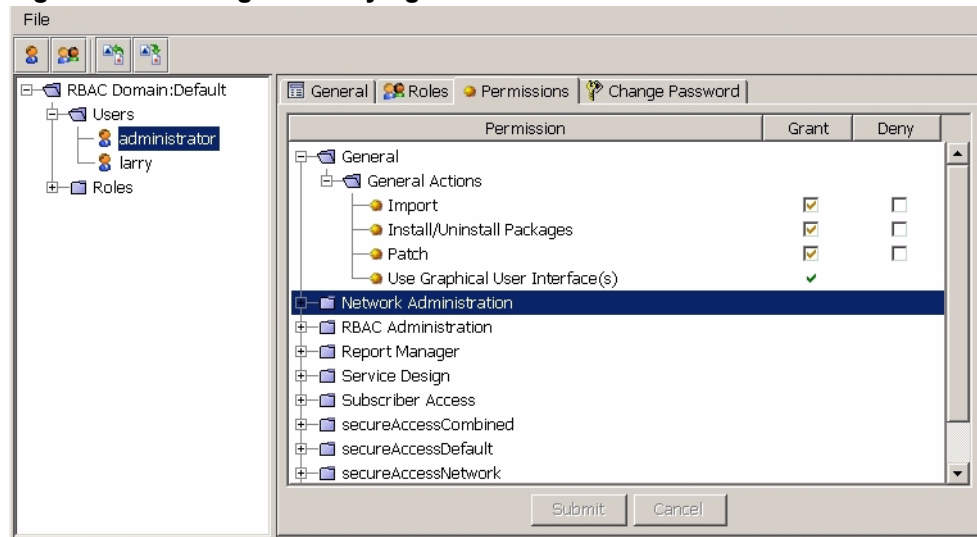


**Caution:** *Removing a role from a user may prevent that user from performing necessary tasks.*

- Click **Submit** when finished. The result of your changes appears on the **Roles** tab.

7. To grant and deny permissions for a user, click the **Permissions** tab (Figure 48).

**Figure 48. Granting and Denying Permissions**



A tree of available permissions appears.

8. For each permission, check **Grant** or **Deny**. This granting or denying applies only to the selected user. If you want to grant or deny the same permission to everyone with the same role as this user, change the permission for the role, not for the individual user.

Permissions are color coded as follows:

- Orange: the permission is inherited from one of the roles of the user.
- Green: the permission is overridden for this user.
- Gray: the default value of the permission.

In addition, some permissions appear without a checkbox. These permissions are read-only, and you cannot change them. This prevents, for example, an administrator from locking himself out of the system.

If you click an orange, gray, or blank permission, it becomes green to show that you selected it explicitly. If you click a green permission, the original inherited or default permission resets to orange or gray.

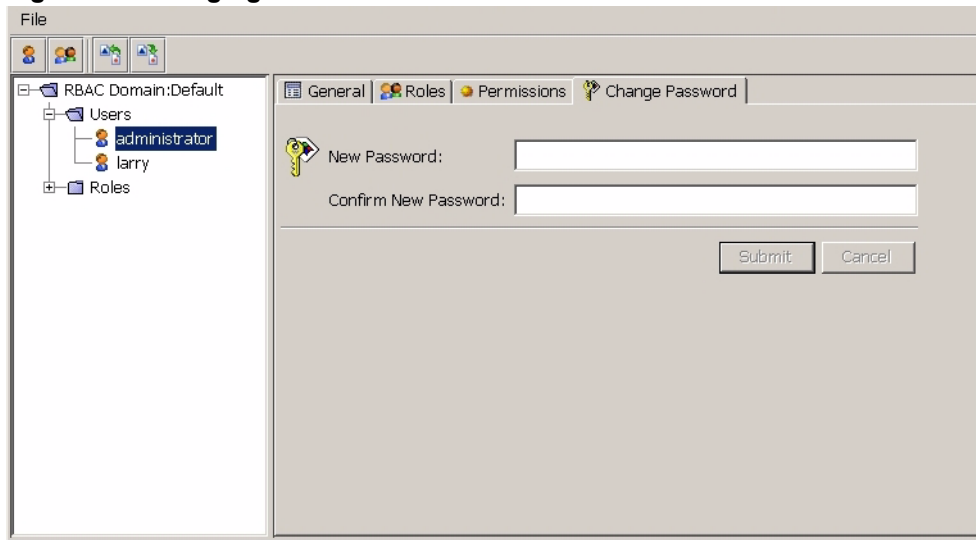
9. When finished with permissions, click **Submit**.



**Caution:** *Changing permissions for a user may prevent that user from performing necessary tasks.*

10. To change the password for a user, click the **Change Password** tab (Figure 49).

**Figure 49. Changing Password**



11. Enter and confirm the **New Password**, then click **Submit**. The password is changed.



**Caution:** *Changing a user password may prevent that user from performing necessary tasks.*

## Deleting Users

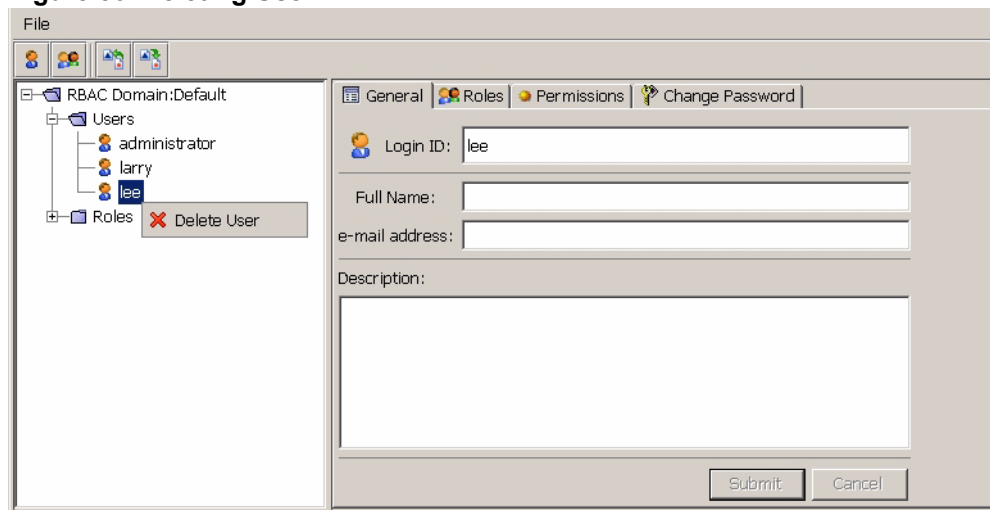
To delete a user, follow these steps:

1. Right-click the user to delete from the tree list, and choose **Delete User** (Figure 50).



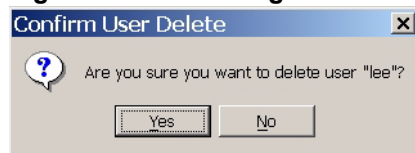
**Caution:** *Deleting a user may prevent that user from performing necessary tasks.*

**Figure 50. Deleting User**



2. The **Confirm User Deletion** dialog appears (Figure 51). Click **Yes**. The user is deleted.

**Figure 51. Confirming User Deletion**



## Importing and Exporting RBAC Data

You can export RBAC data, to preserve the users, roles, and permissions that you have defined. You can also import RBAC data, either to recover changed or lost data, or to duplicate existing definitions.

### Exporting RBAC Data

To export RBAC data, follow these steps:

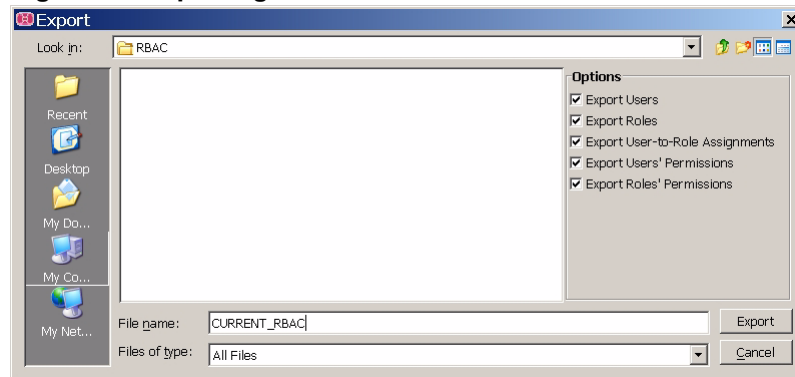
1. Click the **Export Rbac data** icon (on right in Figure 52), or choose **File --> Export**.

**Figure 52. Export RBAC Data Icons**



- The **Export** dialog appears (Figure 53). Move to the directory where you want to save the data. Check which data to export from the following choices:
  - Users
  - Roles
  - User-to-Role Assignments
  - User Permissions
  - Role Permissions
- Enter a name for the data file. Click **Export**.

**Figure 53. Exporting Data**



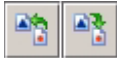
The selected data is exported to the data file.

### **Importing RBAC Data**

To import RBAC data from an existing data file, follow these steps:

- Click the **Import Rbac data** icon (on left in Figure 54).

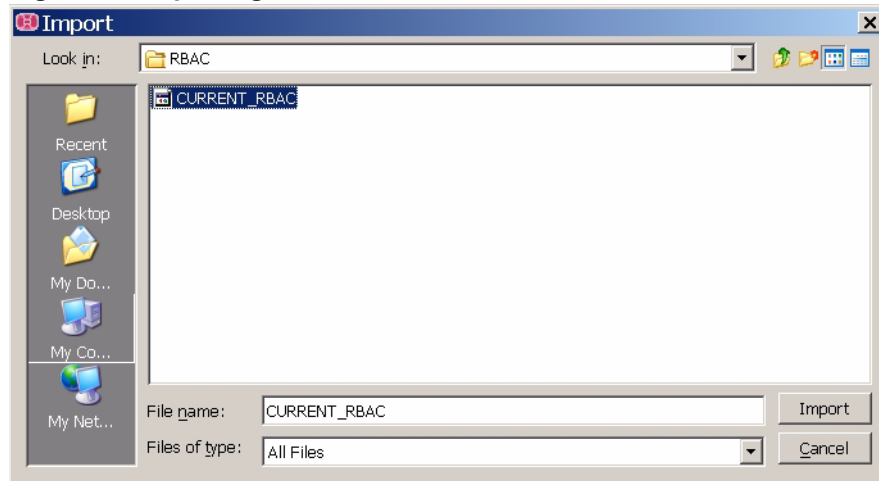
**Figure 54. Import RBAC Data Icons**



Alternatively, you can choose **File --> Import**.

- The **Import** dialog appears (Figure 55). Move to the directory where the data file is located. Choose which data file to import. Click **Import**.

**Figure 55. Importing Data**



The selected data is imported to RBAC.

## Backup and Restore

The export and import functions allow the administrator to save internal data from the database to an external file, or to load data from an external file to the database. They do not backup or restore data, in the sense of copying data to secondary media for safekeeping.

You can backup (export) and restore (import) various sets of data using the BPS. From the Tools Menu, click **Export** or **Import** and choose the specific data set from the drop-down menu. (In the Network Administration view, you can also right-click a service engine, and choose **Import** from the drop-down menu.)

You can export system data. You can also export metadata. You can export data from a specific search engine, by right-clicking the service engine in Network Administration view, then choosing **Export subtree** from the drop-down menu.

To copy the agents and services of one service engine to a second service engine, right-click the original service engine and click **Copy configuration** from the drop-down menu. Follow the prompts to choose the second service engine.

## What's Next?

There are other ways to perform tasks besides using the BPDS. [Chapter 3, Command Line Interface](#), describes these other procedures.





# Command Line Interface

## Overview

This chapter discusses the command line interface that you can use to perform some operational tasks without using the Broadband Policy Design Studio (BPDS). This chapter includes the following topics:

- [Realms](#)
- [tash Command Line Interface](#)
- [Statistics](#)

## Realms

Topology, resources, active sessions, and active contexts exist in an information *realm*. The realm improves performance by restricting lookups and updates against smaller data sets, providing less lock contention and faster search times. It also allows a Resource Controller to comprehend the realms for which it is responsible. If a request involves a realm that the Resource Controller does not own, it can ignore the request (if that is its configured behavior). When a new Resource Controller is introduced, BRAS responsibility migrates from one Resource Controller to a new Resource Controller. The realm concept allows the information model to consider this a block movement of a realm. The migration affects only the realm that is moving. When scaling the BPM to accommodate more hardware and repartitioning the realms, only the realm/BRAS being moved is unavailable to quality of service (QoS) requests. Calls originating or terminating in other realms remain uninterrupted.

A BRAS defines a realm of self-contained information. The mapping of Resource Controllers to BRASs allows a single Resource Controller to handle multiple BRASs and their access. The state maintenance of various components is specified at the granularity of the Resource Controller. Thus, a single Topology Store Function (TSF) element handles more than one BRAS.

### Domain Realm

The Topology Database Server houses the domain realm, which maintains application-level information about the topology. The nodes in the topology are Director and Resource Controller systems. The Topology Database Server uses the Domain Realm to understand the system topology. Links represent connectivity between cluster pairs. This allows the Topology Database Server to act as the central contact point for the application portal and as the main owner on application-level configuration and management. Resources represent interfaces on the component systems, the health of each system, cluster information, and the configuration of each system.

### Director Realm

The Director realm stores information about what Resource Controller is responsible for a given device (such as a BRAS), and what IP address pools a given BRAS handles. The Topology Database Server maintains the realm, and distributes it to each Director when there are updates. Nodes in this realm represent Resource Realms. A Director uses this information to forward an incoming request to the correct Resource Controller.

### Resource Realm

A resource is any device or other item that can be used, such as a printer, disk drive, or memory. Resources, topologies, active sessions, and active contexts exist in a realm. A Resource Realm is a realm distributed to a Resource Controller. The Resource Controller Realm defines the topology of a given device (for example, a BRAS). It represents the ports, VPs, VCs, and assigned CPE devices for that BRAS. The realm improves performance by restricting lookups and updates against smaller data sets.

### Network Realm

The Network Realm stores specific network adaptation information, (such as the devices active on a particular Resource Controller), profiles, devices, and handlers. The Network Realm is centrally provisioned on the Topology Database Server, and distributed to all Resource Controllers.

### Session Realm

A Session Realm, unique to a Resource Controller, improves performance by restricting lookups and updates against smaller data sets.

## tash Command Line Interface

A server shell provides interactive access to the Broadband Policy Manager (BPM).

### Invoking tash

To invoke the shell, execute this command:

```
> <install_dir>/bin/tash [<input-file> [<args>]]
```



**Note:** Before invoking tash, enter the command:

```
stty erase ^H
```

*This ensures that backspace works correctly.*

If you invoke tash without options, it enters an interactive shell. The shell contains commands for interacting with the BPM.



**Note:** In examples below, lines that start with ">" are for the unix command shell (tcsh) and lines that start with "%" are for the tash shell.

### Example: Invoking tash without Arguments

If you invoke tash without options, it enters an interactive shell. Running tash like this:

```
> tash
```

produces these results:

```
[message]
%
```

### Example: Invoking tash with Arguments

If you invoke tash with arguments, the first argument indicates the name of the script file that tash should invoke. Subsequent arguments are concatenated to a list variable named `argv` that is accessible to the shell. The variable `argc` contains the number of arguments in the list.

Consider the following script named "example":

```
for { set i 0 } { $i < $argc } { incr i } {
    puts "arg $i = [lindex $argv $i]"
}
```

Invoking "example" like this:

```
> tash example this is a test
```

produces these results:

```
arg 0 = this
arg 1 = is
arg 2 = a
arg 3 = test
```

### Environment Variables

The tash shell starts up with the set of environment variables. For a list, type:

```
% env | grep ^TAZZ
```

### Commands

Each section describes commands available within tash. Command arguments within angle brackets are required; arguments within square brackets are optional. Arguments within curly braces can be repeated.

Within tash, you can alias commands as needed, using the "alias" command. For example:

```
alias CT ssf::CreateTransaction
```

**Common Commands Available from Both ACM and SM/NM****AddPool**

Adds an IP address pool.

```
Usage: AddPool <vpn> <ip> <mask> <data>
<vpn>: Realm associated with the pool
<ip>: IP address for the pool
<mask>: IP mask (i.e. significant left-most number of bits in
the IP address)
<data>: The node key – the database field that represents the
node of the BRAS
```

Example of adding IP pool with IP address 192.170.1.1 and mask 24 to resource realm r192\_9\_21\_1:

```
AddPool app1 192.170.1.1 24 r192_9_21_1
```

**AddResourceRealm**

Adds a Resource Realm to the topology database. Only available on the TDS.

```
Usage: AddResourceRealm <RealmID> [timeout]
<RealmID>: ID of realm being added
[timeout]: Optional parameter, maximum allowed duration of
operation before failure is assumed
```

Example of adding a resource realm with timeout 30:

```
AddResourceRealm r192_9_21_1 30
```

**AssignResourceRealm**

Assigns a Resource Realm to a particular Resource Controller, and pushes the realm out to that Resource Controller. Only available on the TDS. The realm must not currently be assigned to another Resource Controller.

```
Usage: AssignResourceRealm <RealmID> <RCID> [timeout]
<RealmID>: ID of realm being added
[timeout]: Optional parameter, maximum allowed duration of
operation before failure is assumed
```

Example of assigning realm r192\_9\_21\_1 to Resource Controller 192.168.111.82:10000 with timeout 30:

```
AssignResourceRealm r192_9_21_1 192.168.111.82:10000 30
```

**ClearStats**

Clears all statistics that match a given prefix. If no prefix is given, the command clears all statistics.

```
Usage: ClearStats <Prefix>
```

**ClearTopology**

Clears the nodes, links, and resources for one or more realms. If on a Director system, also removes the IP pools.

```
Usage: ClearTopology [ <RealmID> ... ]
```

**Configure**

Sets configuration settings to a local system, all systems of a particular role, or to a remote system.

Specifying just “<role>” configures all systems of that specified role.

Specifying “<role>” and “<host>”/“<port>” applies the change to just the remote system.

Configuration should ONLY be performed from the TDS – it should not be used in normal circumstances from other systems.

Alias: Conf

Usage: Configure { name value } ... [ role [ host port ] ]

### **GetConfiguration**

Returns configuration settings that match the prefix. Specifying “<role>” and “<host>”/“<port>” applies the change to just the remote system.

Aliases: GetC, GetConf

Usage: GetConfiguration <prefix> [ role host port ]

### **GetRealms**

Returns the realms that this system processes.

Alias: GR

Usage: GetRealms <types>

### **GetRealmTypes**

Returns the realms and their types that this system processes.

Alias: GRT

Usage: GetRealmTypes <types>

### **Hash**

Returns the MD5-hashed value of the argument. This is used to translate ExternalSessionID to ContextID, its internal equivalent. (The Director does this.)

Usage: Hash <value>

### **help**

Displays help on a specified topic. If <topic> is omitted, all available help topics are displayed.

Usage: help [<topic>]

Example of help with no parameters:

```
prompt% help
ClearConfiguration
ClearStats
ClearTopology
...
```

Example of help searching for a string:

```
prompt% help *device*
ssf::CreateDeviceAccess
ssf::CreateDeviceAction
ssf::CreateDeviceInstance
ssf::CreateDeviceSession
ssf::CreateDeviceType
...
```

### **IncrStats**

Increments statistics – passed name value pairs.

Usage: UpdateStats { statistic-name value } ...

### **loadSessions**

Loads the session SIM information from a file.

Usage: loadSessions <file>

### **Provision**

Provisions the Topology Database Server with Director and Resource Controller TIM information, from a file. This command is only available on the TDS system.

“<action>” can be used to over-ride the default action as specified in the provisioned file. It can have values “add”, “remove”, or “update”.

Alias: P

Usage: Provision <file> <action>

### **qgrep**

The qgrep command can be used to filter output from the Get commands (described below). The syntax is as follows.

Alias: qg

Usage: qgrep <pattern> <input> [-v]

The -v option negates the qgrep predicate causing results to be displayed that do not match <pattern>.

Example: only display lines that contain the pattern 1.1.1.1

```
% grep 1.1.1.1 [GetSessions]
```

### **ReloadConfiguration (TDS only)**

Reloads the configuration from the database for the specified system and reapplies it.

Usage: ReloadConfiguration <host> <port>

### **ReloadConfigurations (TDS only)**

Reloads the configuration from the database for all systems of a specified role, and reapplies each one to the host.

Usage: ReloadConfigurations <role>

### **RemoveAllLinks**

Removes all the Links from the topology.

Usage: RemoveAllLinks [ { realms } ]

### **RemoveAllNodes**

Removes all the Nodes from the topology.

Usage: RemoveAllNodes [ { realms } ]

### **RemoveAllResources**

Removes all specified resources in the topology store.

Usage: RemoveAllResources [realm...] [resource-type...]

### **RemovePool**

Removes an IP pool map.

Usage: RemovePool <vpn> <ip> <mask>

<vpn>: Realm associated with the pool

<ip>: IP address for the pool

<mask>: IP mask (i.e. significant left-most number of bits in the IP address)

### **RemovePools**

Removes all IP pool maps from one or more VPNs.

Usage: RemovePools [ <vpn> ... ]

<vpn>: Realm associated with the pool

### **RemoveResourceRealm**

Removes a Resource Realm from the topology database. Only available on the TDS. The realm must not currently be assigned to a Resource Controller.

Usage: RemoveRealm <RealmID> [ timeout ]

### **SaveConfiguration (TDS only)**

Saves the current configuration of the specified system to the database. This writes all configuration into the host-specific configuration resource.

Usage: SaveConfiguration <host> <port>

### **SaveConfigurations (TDS only)**

Saves the current configuration of the specified systems (specified by role) to the database. This writes all configuration into the host-specific configuration resource, for all systems of a given role.

Usage: SaveConfigurations <role>

### **SetHealth**

Sets the health of the specified host (to a possibly supplied value).

Usage: SetHealth <Host> <Port> <Cluster-Role> [Health] [-quiet]  
 Cluster-Role: (a)ctive, (s)tandby  
 Health: (u)p {=0}, (d)own {=-1}, (t)ransition {=-2},  
 <numeric>

### **GetLinks**

Displays list of links and attributes

Aliases: GetL, GL

Usage: GetLinks [ {realms..} ] [ {ids..} ]

### **GetNodes**

Displays list of nodes and attributes

Aliases: GetN, GN

Usage: GetNodes [ {realms..} ] [ {ids..} ]

### **GetPools**

Displays list of pools. If a VPN is specified, only the pools for that VPN are displayed.

Aliases: GetP, GP

Usage: GetPools [ vpn ]

### **GetResources**

Displays all specified resources in the topology store.

Aliases: GetR

Usage: GetResources [realm...] [resource-type...]

Example:

```
prompt% GetR App1
Resource Type: aracf
id | health | host | port | qos | saf | fof | _health | _host | _port | _qos | _saf | _fof
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
(0 rows)
Resource Type: pool
id | ip | mask | vpn | data
-----+-----+-----+-----+-----
(0 rows)
```

**GetStats**

Displays statistics for those names provided or all if no statistic names are specified.

Alias: GetSt

Usage: GetStats [statistic-name] ...

**UnassignResourceRealm**

Unassigns a Resource Realm from a particular Resource Controller, causing that Resource Controller to delete its local copy of the realm. The realm does not get removed from the TDS, and is therefore available for assignment to another Resource Controller (or deletion from the TDS. Only available on the TDS. The realm must currently be assigned to a Resource Controller.

Usage: UnassignRealm <RealmID> <RCID> [ timeout ]

**UpdateStats**

Updates statistics – passed name value pairs.

Usage: UpdateStats { statistic-name value } ...

**CAC-Specific Commands in ACM Extension Library**

These commands are part of the ACM Extension Library.

**qos::CleanExpiredContexts**

Examines all current active contexts, and releases any that are sufficiently out of date. A context is considered “sufficiently out of date” if the difference between the current time and the timestamp on the context, plus the lease, is greater than the overage (in seconds). Overage in this case defaults to 12 hours.

Usage: qos::CleanExpiredContexts <timeout> [overage]

**qos::StartServer**

Starts the QoS Server on the local system.

Usage: qos::StartServer

**qos::StartSession**

Indicates that a QoS session has started. Arguments are as described below.

Note that currently “<bras-ip>” and “<bras-id>” are the same.

Usage: qos::StartSession <cpe-ip>@<vpn-id> <cpe-id> <bras-ip>  
 <bras-id> <Props>  
 -<cpe-ip> is the IP address of the CPE  
 -<vpn-id> is a string  
 -<cpe-id> is the Technical Key (BRAS-IP/PORT/VPI/VCI)  
 -<bras-ip> is the IP address of the BRAS  
 -<bras-id> is the identifier for the BRAS  
 -<Props> is the empty string, currently ""

**qos::StopServer**

Stops the QoS Server on the local system.

Usage: qos::StopServer

**qos::GetContexts**

Displays active contexts

Aliases: GetCtx, GC

Usage: qos::GetContexts [ { realms.. } ]



**qos::GetSessions**

Displays a list of active sessions, for one or more realms.

Alias: GetSe

Usage: qos::GetSessions [ { realm } ]

**qos::GetTimeouts**

Displays timeouts that are in affect.

Alias: GetTi

Usage: qos::GetTimeouts

**qos::StopSession**

Indicates that a session has stopped. Arguments are as described below.

Note that currently “<bras-ip>” and “<bras-id>” are the same.

Usage: qos::StopSession <cpe-ip>@<vpn-id> <cpe-id> <bras-ip>  
<bras-id> <Props>  
-<cpe-ip> is the IP address of the CPE  
-<vpn-id> is a string  
-<cpe-id> is the Technical Key (BRAS-IP/PORT/VPI/VCI)  
-<bras-ip> is the IP address of the BRAS  
-<bras-id> is the identifier for the BRAS  
-<Props> is the empty string, currently ""

## Session- and Adaptation-specific Commands in SM/SM Extension Library

Note that the format for these commands differs from the formats of the QoS commands. This is because these commands take a large number of optional parameters, which thereby require keyword argument names to identify them. The arguments can be passed in any order.

### **ssf::CreateTransaction**

Opens a transaction. The `lease` parameter identifies the number of seconds after opening until the transaction expires. No notice of expiration is provided until a `CommitTransaction` call, when it returns an error message.

Usage: `ssf::CreateTransaction lease <value>`

### **ssf::RollbackTransaction**

Rolls back a transaction. Uses the ID of the transaction to roll back.

Usage: `ssf::RollbackTransaction transactionID <value>`

### **ssf::CommitTransaction**

Commits a transaction. Uses the ID of the transaction to commit.

Usage: `ssf::CommitTransaction transactionID <value>`

### **ssf::GetNetworkSessionAppliedProfile**

Retrieves the applied profile for a network session. Displays the contents as output. `TransactionID` defaults to "IMPLICIT" if unspecified.

Usage: `ssf::GetNetworkSessionAppliedProfile networkSessionID <value> profileVersionedID <value> sessionRealmID <value> [ transactionID <value> ] [ networkRealmID <value> ]`

### **ssf::UnsetNetworkSessionProfile**

Unsets the profile for a network session.

Usage: `ssf::UnsetNetworkSessionProfile networkSessionID <value> realmID <value> transactionID <value>`

### **ssf::RemoveNetworkSessions**

Removes one or more network sessions.

Usage: `ssf::RemoveNetworkSessions realmID <value> transactionID <value> [ queryIDs <value> ] [ queryKeys <value> ] [ queryStatuses <value> ]`

### **ssf::CountNetworkSessions**

Counts the current number of network sessions on the local resource controller, and displays the results on the command line.

Usage: `ssf::CountNetworkSessions realmID <value> transactionID <value> [ queryIDs <value> ] [ queryKeys <value> ] [ queryStatuses <value> ]`

**ssf::UpdateNetworkSessions**

Updates one or more network sessions with new information provided on the command line. The command first finds a set of network session using the query keys, then replaces information in those sessions based on the attributes, keys, and statuses values.

```
Usage: ssf::UpdateNetworkSessions realmID <value> transactionID
<value> [ attributes <value> ] [ keys <value> ]
[ profileVersionedID <value> ] [ queryIDs <value> ]
[ queryKeys <value> ] [ queryStatuses <value> ] [ statuses
<value> ]
```

**ssf::FindNetworkSessions**

Finds network sessions and displays them on the command line.

```
Usage: ssf::FindNetworkSessions realmID <value> transactionID
<value> [ lock <value> ] [ queryIDs <value> ]
[ queryKeys <value> ] [ queryStatuses <value> ]
```

**ssf::SetNetworkSessionProfile**

Sets the profile for a network session.

```
Usage: ssf::SetNetworkSessionProfile networkSessionID <value>
profileVersionedID <value> realmID <value>
transactionID <value>
```

**ssf::UnsetNetworkSessionProfiles**

Unsets one or more profiles in a network session, based on query IDs.

```
Usage: ssf::UnsetNetworkSessionProfiles queryIDs <value> realmID
<value> transactionID <value>
```

**ssf::CreateNetworkSession**

Creates a network session.

```
Usage: ssf::CreateNetworkSession realmID <value> transactionID
<value> [ attributes <value> ] [ keys <value> ]
[ networkSessionID <value> ] [ profileVersionedID <value> ] [
statuses <value> ]
```

**ssf::GetNetworkSessionValues**

Retrieves information from a network session, using this expression syntax:

Expression syntax: (<<X>> marks a free variable, [X] marks a choice between several values)

1. Lookup on network session

- a. Lookup on a specific network session with specified namespace and name

```
attribute.<<namespace>>.<<name>>
key.<<namespace>>.<<name>>
status.<<namespace>>.<<name>>
```

- b. Lookup on a specific network session with specified name

```
attribute.<<name>>
key.<<name>>
status.<<name>>
```

- c. Lookup on any network session attribute, key, or status with specified namespace and name

```
<<namespace>>.<<name>>
```

- d. Lookup on any network session attribute, key, or status with specified name  
`<<name>>`
  - e. Lookup on network session status table for the isTerminal value of specified namespace and name  
`status.<<namespace>>.<<name>>.isTerminal`
2. Lookup on device session
- a. Lookup on a specific device session with specified namespace and name  
`role.<<role-name>>.attribute.<<namespace>>.<<name>>`  
`role.<<role-name>>.key.<<namespace>>.<<name>>`  
`role.<<role-name>>.status.<<namespace>>.<<name>>`
  - b. Lookup on a specific network session with specified name  
`role.<<role-name>>.attribute.<<name>>`  
`role.<<role-name>>.key.<<name>>`  
`role.<<role-name>>.status.<<name>>`
  - c. Lookup on any network device attribute, key, or status with specified namespace and name  
`role.<<role-name>>.<<namespace>>.<<name>>`
  - d. Lookup on any network device attribute, key, or status with specified name  
`role.<<role-name>>.<<name>>`
  - e. Lookup on network device status table for the isTerminal value of specified namespace and name  
`role.<<role-name>>.status.<<namespace>>.<<name>>.isTerminal`
3. Lookup on device session access
- a. Lookup on specified attribute  
`role.<<role-name>>.access.<<attribute-name>>`
  - b. Lookup on specified management protocol and attribute  
`role<<role-name>>.access.<<protocol-name>>.<<attribute-name>>`
  - c. An empty string will be returned in the following cases:  
 Expression has invalid syntax.  
 No value found for an expression.  
 More than one value found for an expression.  
 Usage: `ssf::GetNetworkSessionValues expressions <value>`  
`networkSessionID <value> sessionRealmID <value> transactionID`  
`<value> [ networkRealmID <value> ]`

**ssf::VersionProfile**

Creates a new version of a profile.

Usage: `ssf::VersionProfile profileID <value> transactionID`  
`<value> version <value> [ name <value> ] [ realmID <value> ] [`  
`steps <value> ]`

**ssf::GetAppliedProfile**

Gets the profile currently applied to a session.

```
Usage: ssf::GetAppliedProfile deviceInstanceIDs <value>
profileVersionedID <value> transactionID <value>
[ realmID <value> ]
```

**ssf::RemoveProfiles**

Removes one or more profiles from the database.

```
Usage: ssf::RemoveProfiles transactionID <value>
[ queryIDs <value> ] [ realmID <value> ]
```

**ssf::FindProfiles**

Finds profiles based on a set of query criteria.

```
Usage: ssf::FindProfiles transactionID <value> [ lock <value> ] [
queryNames <value> ] [ queryVersionedIDs <value> ]
[ realmID <value> ]
```

**ssf::CreateProfile**

Creates a profile.

```
Usage: ssf::CreateProfile name <value> transactionID <value>
version <value> [ profileID <value> ] [ realmID <value> ]
[ steps <value> ]
```

**ssf::FindDeviceSessions**

Finds device sessions based on a set of query criteria.

```
Usage: ssf::FindDeviceSessions realmID <value> transactionID
<value> [ lock <value> ] [ queryIDs <value> ]
[ queryKeys <value> ] [ queryNetworkSessionIDs <value> ]
[ queryStatuses <value> ]
```

**ssf::RemoveDeviceSessions**

Removes one or more device sessions.

```
Usage: ssf::RemoveDeviceSessions realmID <value> transactionID
<value> [ queryIDs <value> ] [ queryKeys <value> ]
[ queryStatuses <value> ]
```

**ssf::UpdateDeviceSessions**

Updates one or more device sessions. The sessions are selected via a set of query criteria, and updated with new values for attributes, keys, and statuses.

```
Usage: ssf::UpdateDeviceSessions realmID <value> transactionID
<value> [ attributes <value> ] [ deviceInstanceID <value> ] [
keys <value> ] [ queryIDs <value> ] [ queryKeys <value> ]
[ queryNetworkSessionIDs <value> ] [ queryStatuses <value> ] [
statuses <value> ]
```

**ssf::CreateDeviceSession**

Creates a device session.

```
Usage: ssf::CreateDeviceSession networkSessionID <value> realmID
<value> transactionID <value> [ attributes <value> ] [
deviceInstanceID <value> ] [ deviceSessionID <value> ]
[ keys <value> ] [ statuses <value> ]
```

**ssf::RemoveDeviceAccesses**

Removes one or more device accesses.

Usage: `ssf::RemoveDeviceAccesses transactionID <value>`  
`[ queryIDs <value> ] [ realmID <value> ]`

**ssf::UpdateDeviceAccesses**

Updates one or more device accesses.

Usage: `ssf::UpdateDeviceAccesses transactionID <value>`  
`[ managementProtocol <value> ] [ properties <value> ]`  
`[ queryIDs <value> ] [ queryProtocols <value> ]`  
`[ realmID <value> ]`

**ssf::CreateDeviceAccess**

Creates a device access.

Usage: `ssf::CreateDeviceAccess managementProtocol <value>`  
`transactionID <value> [ deviceAccessID <value> ]`  
`[ properties <value> ] [ realmID <value> ]`

**ssf::FindDeviceAccesses**

Finds device accesses given a set of query criteria.

Usage: `ssf::FindDeviceAccesses transactionID <value>`  
`[ lock <value> ] [ queryIDs <value> ]`  
`[ queryProtocols <value> ] [ realmID <value> ]`

**ssf::RemoveDeviceActions**

Removes one or more device actions.

Usage: `ssf::RemoveDeviceActions transactionID <value>`  
`[ queryActions <value> ] [ queryDeviceTypes <value> ]`  
`[ queryIDs <value> ] [ queryProtocols <value> ]`  
`[ realmID <value> ]`

**ssf::FindDeviceActions**

Finds device actions based on a set of query criteria.

Usage: `ssf::FindDeviceActions transactionID <value>`  
`[ lock <value> ] [ queryActions <value> ]`  
`[ queryDeviceTypes <value> ] [ queryProtocols <value> ]`  
`[ queryVersionedIDs <value> ] [ realmID <value> ]`

**ssf::VersionDeviceAction**

Versions a device action.

Usage: `ssf::VersionDeviceAction deviceActionID <value>`  
`transactionID <value> version <value> [ actionName <value> ] [`  
`instructions <value> ] [ managementProtocol <value> ] [ realmID`  
`<value> ]`

**ssf::CreateDeviceAction**

Creates a device action.

Usage: `ssf::CreateDeviceAction actionName <value> deviceTypeID`  
`<value> instructions <value> managementProtocol <value>`  
`transactionID <value> version <value>`  
`[ deviceActionID <value> ] [ realmID <value> ]`

**ssf::RemoveDeviceInstances**

Removes one or more device instances.

```
Usage: ssf::RemoveDeviceInstances transactionID <value>
      [ queryIDs <value> ] [ realmID <value> ]
```

**ssf::FindDeviceInstances**

Finds device instances, based on a set of query criteria.

```
Usage: ssf::FindDeviceInstances transactionID <value>
      [ lock <value> ] [ queryAddresses <value> ]
      [ queryDeviceTypes <value> ] [ queryIDs <value> ]
      [ queryRoles <value> ] [ realmID <value> ]
```

**ssf::UpdateDeviceInstances**

Updates device instances.

```
Usage: ssf::UpdateDeviceInstances transactionID <value>
      [ deviceAccessIDs <value> ] [ ipAddress <value> ]
      [ queryAddresses <value> ] [ queryDeviceTypes <value> ]
      [ queryIDs <value> ] [ queryRoles <value> ]
      [ realmID <value> ] [ roleNames <value> ]
```

**ssf::CreateDeviceInstance**

Creates a device instance.

```
Usage: ssf::CreateDeviceInstance deviceTypeID <value>
      ipAddress <value>
      transactionID <value> [ deviceAccessIDs <value> ]
      [ deviceInstanceID <value> ] [ realmID <value> ]
      [ roleNames <value> ]
```

**ssf::FindDeviceTypes**

Finds device types based on a set of query criteria.

```
Usage: ssf::FindDeviceTypes transactionID <value>
      [ lock <value> ] [ queryIDs <value> ] [ queryRoles <value> ] [
      realmID <value> ]
```

**ssf::CreateDeviceType**

Creates a device type.

```
Usage: ssf::CreateDeviceType transactionID <value>
      [ deviceTypeID <value> ] [ model <value> ] [ realmID <value> ]
      [ roleNames <value> ] [ vendor <value> ]
```

**ssf::RemoveDeviceTypes**

Removes one or more device types.

```
Usage: ssf::RemoveDeviceTypes transactionID <value>
      [ queryIDs <value> ] [ realmID <value> ]
```

**ssf::UpdateDeviceTypes**

Updates one or more device types.

```
Usage: ssf::UpdateDeviceTypes transactionID <value>
      [ model <value> ] [ queryIDs <value> ] [ queryRoles <value> ] [
      realmID <value> ] [ roleNames <value> ] [ vendor <value> ]
```

### **ssf::FindHandlers**

Finds handlers based on a set of query criteria.

```
Usage: ssf::FindHandlers transactionID <value> [ lock <value> ] [
queryAddresses <value> ] [ queryDeviceInstances <value> ] [
queryEvents <value> ] [ queryProtocols <value> ]
[ realmID <value> ]
```

### **ssf::RemoveHandlers**

Removes one or more handlers.

```
Usage: ssf::RemoveHandlers queryDeviceInstances <value>
transactionID <value> [ realmID <value> ]
```

### **ssf::CreateHandler**

Creates a new handler.

```
Usage: ssf::CreateHandler deviceInstanceID <value>
eventName <value> protocolName <value> transactionID <value>
url <value> realmID <value> ]
```

## **Statistics**

This section documents the statistics that the BPM system generates. Statistics are available using the `tash ShowStats` command.

### **TDS Statistics**

TDS statistics include:

- `TDS.start`: the number of TDS starts. Note that this increases with each execution of `StartServer` on a TDS, even without a corresponding `StopServer`. Because `StartServer` overwrites the configuration, initializes the set of directors, and initializes the IP pools, it may occasionally execute without intervening `StopServer` commands.
- `TDS.resiliency.initiateFailover.<IP_Address>`: the number of times the TDS has initiated a failover of a Resource Controller. `<IP_Address>` represents the IP address of the active Resource Controller.
- `TDS.resiliency.notifiedStartup.<IP_Address>`: the number of times the TDS has received notification of Resource Controller startup. `<IP_Address>` represents the IP address of the active Resource Controller.

### **Director Statistics**

Director statistics include:

- `SPDF.start`: the number of Director starts.
- `SPDF.resiliency.initiateFailover`: the number of times the Director has initiated a failover of a Resource Controller. Note that this exists for historical context. The product does not currently use it. The TDS initiates Resource Controller failover.
- `SPDF.resiliency.notifiedStartup`: the number of times the Director received notification of Resource Controller startup. Note that this exists for historical context. The product does not currently use it. Directors do not need notification of Resource Controller startup.



The QoS interface also has counters for each application function. `<Action>` is `QosReserve`, `QosModify`, `QosRefresh`, or `QosRelease`. `<Status>` is the Status code of the deny, for example, 10101 (deny on VC) or 10102 (deny on VP) in most cases. Counters include:

- `<ApplicationID>.qos.<Action>.accept.count`
- `<ApplicationID>.qos.<Action>.count`
- `<ApplicationID>.qos.<Action>.deny.<Status>.count`
- `<ApplicationID>.qos.<Action>.deny.count`
- `<ApplicationID>.qos.<Action>.error.count`

The QoS interface also has counters in aggregate of all application functions, including:

- `total.qos.<Action>.accept.count`
- `total.qos.<Action>.count`
- `total.qos.<Action>.deny.<Status>.count`
- `total.qos.<Action>.deny.count`
- `total.qos.<Action>.error.count`

## Resource Controller Statistics

Resource Controller statistics include:

- `ARACF.start`: The number of Resource Controller starts

The QoS interface has counters for each application function, including:

- `<ApplicationID>.qos.<Action>.accept.count`
- `<ApplicationID>.qos.<Action>.count`
- `<ApplicationID>.qos.<Action>.deny.<Status>.count`
- `<ApplicationID>.qos.<Action>.deny.count`
- `<ApplicationID>.qos.<Action>.error.count`

The QoS interface also has counters in aggregate of all application functions. `<Action>` is `QosReserve`, `QosModify`, `QosRefresh`, or `QosRelease`. `<Status>` is the Status code of the deny, for example, 10101 (deny on VC) or 10102 (deny on VP) in most cases. Counters include:

- `total.qos.<Action>.accept.count`
- `total.qos.<Action>.count`
- `total.qos.<Action>.deny.<Status>.count`
- `total.qos.<Action>.deny.count`
- `total.qos.<Action>.error.count`

## Session Manager and Network Manager Statistics

Number of times the specified event has occurred on the Resource Controller.

```
<ControlPoint>.<Protocol>.<Event>
```

Number of times the specified event has occurred and the system failed to process it on the Resource Controller.

```
<ControlPoint>.<Protocol>.<Event>.failures
```

The largest number of Network Sessions that have existed concurrently on the Resource Controller.

```
SSF.NetworkSessionHighWaterMark
```

The largest number of Device Sessions that have existed concurrently on the Resource Controller.

```
SSF.DeviceSessionHighWaterMark
```

The current number of devices loaded into the NSF on the Resource Controller.

```
NSF.Devices
```

## What's Next?

Besides day-to-day operations tasks, there are also maintenance tasks to perform as needed. [Chapter 4, \*Maintenance Tasks\*](#), describes these tasks.

# Maintenance Tasks

## Overview

Maintenance tasks include making changes to the initial setup of the Broadband Policy Manager (BPM) system. This may include adding, configuring, or removing components such as service engines, agents, or services. You can perform many maintenance tasks using the Broadband Policy Studio (BPS). The BPS gives a visible picture of the current BPM Configuration. The BPS uses standard windowing techniques and editing features.

This chapter discusses the following topics:

- [Log File Maintenance](#)
- [Realms](#)
- [Resource Controller Procedures](#)
- [Director Procedures](#)
- [Deploying Components](#)
- [Configuring Components](#)
- [Clusters and Failover](#)

## Log File Maintenance

Monitor the following files, because they can grow in number and size:

- accounting log files: You can change configuration settings to roll accounting logs over to a new file on the basis of time or size.
- tazz.log file
- .out files
- .err files
- .log files

# Realms

Topology, resources, active sessions, and active contexts exist in an information *realm*. The realm improves performance by restricting lookups and updates against smaller data sets, providing less lock contention and faster search times. It also allows a Resource Controller to comprehend the realms for which it is responsible. If a request involves a realm that the Resource Controller does not own, it can ignore the request (if that is its configured behavior). When a new Resource Controller is introduced, BRAS responsibility migrates from one Resource Controller to a new Resource Controller. The realm concept allows the information model to consider this a block movement of a realm. The migration affects only the realm that is moving. When scaling the BPM to accommodate more hardware and repartitioning the realms, only the realm/BRAS being moved is unavailable to quality of service (QoS) requests. Calls originating or terminating in other realms remain uninterrupted.

A BRAS defines a realm of self-contained information. The mapping of Resource Controllers to BRASs allows a single Resource Controller to handle multiple BRASs and their access. The state maintenance of various components is specified at the granularity of the Resource Controller. Thus, a single Topology Store Function (TSF) element handles more than one BRAS.

## Domain Realm

The Topology Database Server houses the domain realm, which maintains application-level information about the topology. The nodes in the topology are Director and Resource Controller systems. The Topology Database Server uses the Domain Realm to understand the system topology. Links represent connectivity between cluster pairs. This allows the Topology Database Server to act as the central contact point for the application portal and as the main owner on application-level configuration and management. Resources represent interfaces on the component systems, the health of each system, cluster information, and the configuration of each system.

## Director Realm

The Director realm stores information about what Resource Controller is responsible for a given device (such as a BRAS), and what IP address pools a given BRAS handles. The Topology Database Server maintains the realm, and distributes it to each Director when there are updates. Nodes in this realm represent Resource Realms. A Director uses this information to forward an incoming request to the correct Resource Controller.

## Resource Realm

A resource is any device or other item that can be used, such as a printer, disk drive, or memory. Resources, topologies, active sessions, and active contexts exist in a realm. A Resource Realm is a realm distributed to a Resource Controller. The Resource Controller Realm defines the topology of a given device (for example, a BRAS). It represents the ports, VPs, VCs, and assigned CPE devices for that BRAS. The realm improves performance by restricting lookups and updates against smaller data sets.

## Network Realm

The Network Realm stores specific network adaptation information, (such as the devices active on a particular Resource Controller), profiles, devices, and handlers. The Network Realm is centrally provisioned on the Topology Database Server, and distributed to all Resource Controllers.

## Session Realm

A Session Realm, unique to a Resource Controller, improves performance by restricting lookups and updates against smaller data sets.

## Resource Controller Procedures

There are several procedures you can perform with Resource Controllers.

### Adding Resource Controller

To add a Resource Controller, follow these steps:

1. Install a backend, as described in the [Cisco Broadband Policy Manager Installation and Configuration Guide](#).
2. Add a service engine for the Resource Controller, following this procedure:
  - a. Using the BPS, on the Network Administration window, right-click the Service Engines folder and choose **Add Service Engine** from the drop-down menu. The New Service Engine dialog appears.
  - b. Enter the information for the active service engine, including name, and the target system's hostname and port.



**Note:** The port offset is +10 from the base port you specified in the backend install for that target system.

- c. Click **Next** and **OK**.
3. If the Resource Controller is a cluster member, then follow this procedure for the standby Resource Controller:
  - a. Using the BPS, on the Network Administration window, right-click the Service Engines folder and choose **Add Service Engine** from the drop-down menu. The New Service Engine dialog appears.
  - b. Enter the information for the standby service engine, including name, and the target system's hostname and port.



**Note:** The port offset is +10 from the base port you specified in the backend install for that target system.

- c. Click **Next** and **OK**.
  - d. Cluster the active and standby Resource Controller service engines. Check **Hot Standby** on the Select Standby Engine dialog.
4. Right-click the service engine named `aracf-template` and choose **Copy Configuration** from the drop-down menu. The Copy Engine dialog appears.
5. Select the active (or standalone) Resource Controller, and click **Next**. This copies the agents and services from the template system to the active Resource Controller.

6. If installing SM or NM, right-click the service engine named `smm-rc-template` and choose **Copy Configuration** from the drop-down menu. The Copy Engine dialog appears. Select the active (or standalone) Resource Controller, and click **Next**. This copies the agents and services from the template system to the active Resource Controller.
7. If installing SM or NM, right-click the service engine named `smm-base-pif-daf-template` and choose **Copy Configuration** from the drop-down menu. The Copy Engine dialog appears. Select the active (or standalone) Resource Controller, and click **Next**. This copies the agents and services from the template system to the active Resource Controller.
8. Right-click the active cluster member and choose **Deploy All** from the drop-down menu.
9. After adding each Resource Controller, verify that it has been successfully added. For example, after adding the first Resource Controller, run commands on the TDS similar to the following. Note that after adding the second Resource Controller, the first command should return two entries, and so forth. Note also that the output of the `SN domain` command will show an entry for each system added already.

```

% SN domain
  id                               | active
-----+-----
192.168.111.52:10000 | t
% SRoN domain 192.168.111.52:10000
  node                               | rtype          | rid
-----+-----+-----
192.168.111.52:10000 | configuration  | 192.168.111.52:10000
192.168.111.52:10000 | health        | 192.168.111.52:10000
192.168.111.52:10000 | interface     | 192.168.111.52
192.168.111.52:10000 | role          | aracf
192.168.111.52:10000 | configuration  | aracf

```

For a Resource Controller, the `role` should be `aracf` and the `configuration` should be `aracf`.

During this procedure, the topology database server does the following:

- automatically configures the system as a Resource Controller (loading the configuration from the resource in the domain realm)
- starts the server on the target Resource Controller system, by invoking the `rc_operations/StartServer` service
- updates the health of the Resource Controller to 0, indicating proper operation of the Resource Controller

## Removing a Resource Controller

To remove a Resource Controller, follow these steps:

1. If the Resource Controller is a cluster active, in the BPS right-click the service engine and choose **Failover to standby**.
2. If the Resource Controller is a cluster standby, set the `_health` of it to `-1` (unavailable) by entering the command:

```
SetHealth <host> <port> s -1
```

where `<host>` is the host IP address, `<port>` is the port, and `s` represents standby.

3. Dissolve the cluster.
4. If the Resource Controller is a standalone, follow this procedure:
  - a. Stop SM and NM activity for that Resource Controller.
  - b. Remove all pool maps that reference Resource Realms on that Resource Controller, and then wait for all activity on that Resource Controller to stop.
  - c. Remove all Resource Realms from the Resource Controller. You can add those Resource Realms to other Resource Controllers.
5. Remove the service engine from the Resource Controller. This generates an event that the TDS uses to update its Domain and Director topology.

## **Starting Resource Controller**

If a Resource Controller is a former cluster active, you can reset the Resource Controller from the BPS.

## Director Procedures

There are several procedures you can perform with Directors.

### Adding a Director

To add a Director, follow these steps:

1. Install a backend, as described in the [Cisco Broadband Policy Manager Installation and Configuration Guide](#).
2. Add a service engine for the Director, following this procedure:
  - a. Using the BPS, on the Network Administration window, right-click the Service Engines folder and choose **Add Service Engine** from the drop-down menu. The New Service Engine dialog appears.
  - b. Enter the information for the service engine, including name, and the target system's hostname and port.



**Note:** *The port offset is +10 from the base port you specified in the backend install for that target system.*

- c. Click **Next** and **OK**.
3. Right-click the service engine named `spdf-template` and choose **Copy Configuration** from the drop-down menu. The Copy Engine dialog appears.
4. Select the Director, and click **Next**. This copies the agents and services from the template system to the Director.
5. Right-click the Director and choose **Deploy All** from the drop-down menu.
6. After adding each Director, verify that it has been successfully added. For example, after adding the first Director, run commands on the TDS similar to the following. Note that after adding the second Director, the first command should return two entries, and so forth. Note also that the output of the `SN domain` command will show an entry for each system added already.

```
% SN domain
  id                               | active
-----+-----
192.168.111.52:10000 | t
% SRoN domain 192.168.111.52:10000
  node                               | rtype           | rid
-----+-----+-----
192.168.111.52:10000 | configuration    | 192.168.111.52:10000
192.168.111.52:10000 | health          | 192.168.111.52:10000
192.168.111.52:10000 | interface       | 192.168.111.52
192.168.111.52:10000 | role            | spdf
192.168.111.52:10000 | configuration    | spdf
192.168.111.52:10000 | realm          | App1
```

For a Director, the `role` should be `spdf` and the `configuration` should be `spdf`.



- As you add each Director, make sure the topology on that Director looks accurate by running the following `show` command on the Director. If the output shows an `aracf` resource and a `pool` resource, the system has its initial topology correct.

```
% SR Appl
Resource Type: aracf
id | health | host | port | qos | saf | fof | _health | _host | _port | _qos | _saf | _fof
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
(0 rows)
Resource Type: pool
id | ip | mask | vpn | data
-----+-----+-----+-----+-----
(0 rows)
```

During this procedure, the topology database server does the following:

- automatically configures the system as a Director (loading the configuration from the resource in the domain realm)
- copies the Director realm to the target system, including the provisioned pool information and Resource Controller to BRAS mappings
- starts the server on the target Director system, by invoking the `dir_operations/StartServer` service
- updates the health of the Director to 0, indicating proper operation of the Director
- adds the Director's IP address to the in-memory "director set".

## Removing a Director

To manually remove a Director, follow these steps:

- Set the health of the Director to -1 (unavailable) by entering the command:

```
SetHealth <host> <port> a -1
```

where `<host>` is the host IP address, `<port>` is the port, and `a` represents active.

- Wait for about a minute, in case there are pending requests.
- Shut down the Director.

## Deploying Components

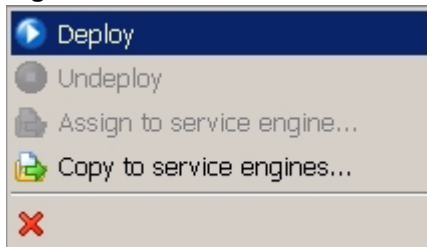
You can deploy all BPM components using the BPS, including service engines, agents, and services.

### Procedure: Deploying an Agent

You can deploy an agent using the following procedure:

1. In the Network Administration Tree pane, right-click an agent that is not deployed. A drop-down menu of commands appears (Figure 56).

**Figure 56. Tree Pane Commands for Agents**



2. Choose the **Deploy** command from the drop-down menu. The agent deploys.



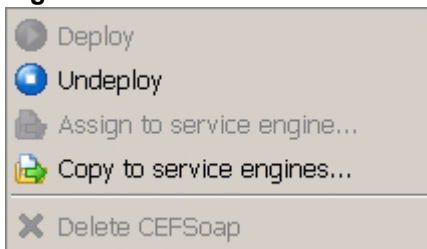
**Note:** To deploy all agents for a service engine, right-click the Agents folder under the service engine and click **Deploy all** from the drop-down menu.

### Procedure: Undeploying an Agent

You can undeploy an agent using the following procedure:

1. In the Network Administration Tree pane, right-click a deployed agent. A drop-down menu of commands appears (Figure 57).

**Figure 57. Tree Pane Commands for Agents**



**Note:** If you attempt to submit changes to an agent before you undeploy the agent, the system prompts you to undeploy the agent.

- Choose the **Undeploy** command from the drop-down menu. Alternatively, on the Configuration tab, click the **Undeploy** button.

The system displays a progress dialog box while undeploying the agent. When the system undeploys the agent, the running instance of the agent is deleted from the system, and the system displays an Agent object. The state field indicates that the agent is *Ready*. The **Deploy** button is enabled.



**Note:** *If the undeployment is unsuccessful, you receive a message explaining how to undeploy the agent.*



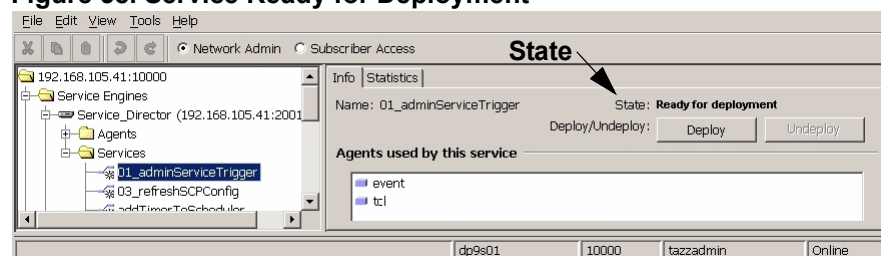
**Note:** *To undeploy all agents for a service engine, right-click the Agents folder under the service engine, and click **Undeploy all** from the drop-down menu.*

## Procedure: Deploying a Service

Follow the steps below to deploy a service:

- Click **Network Administration** and open the Services folder. Click the service to deploy. The State should be *Ready for deployment* (Figure 58).

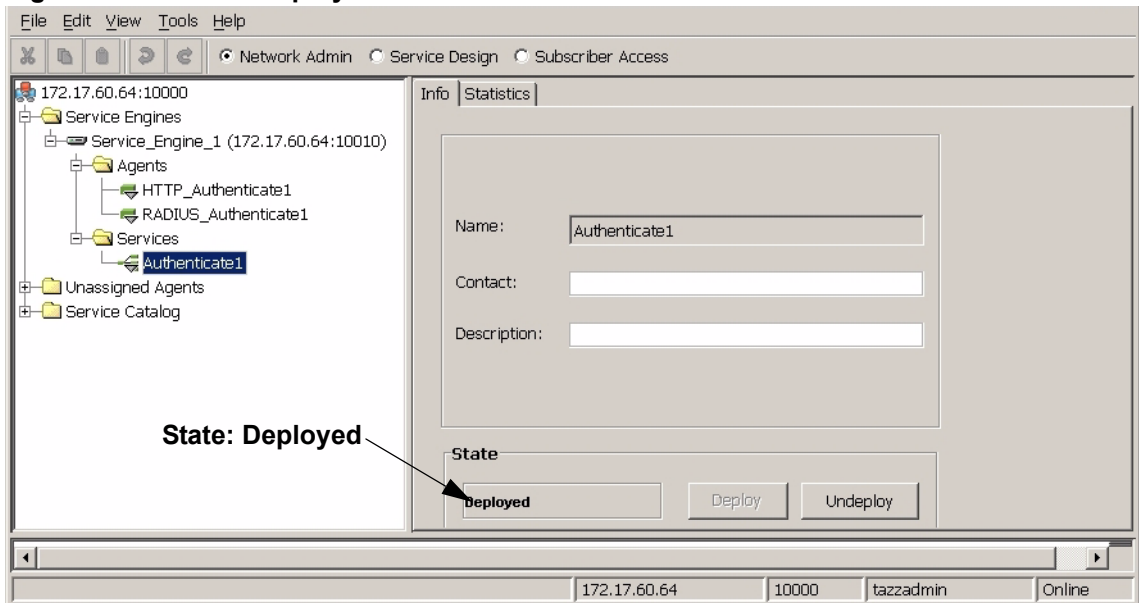
**Figure 58. Service Ready for Deployment**



- Click the **Deploy** button to deploy the service. You receive a message that the system is deploying the service.

When the State condition changes to *Deployed*, the service is working (Figure 59).

**Figure 59. Service - Deployed**



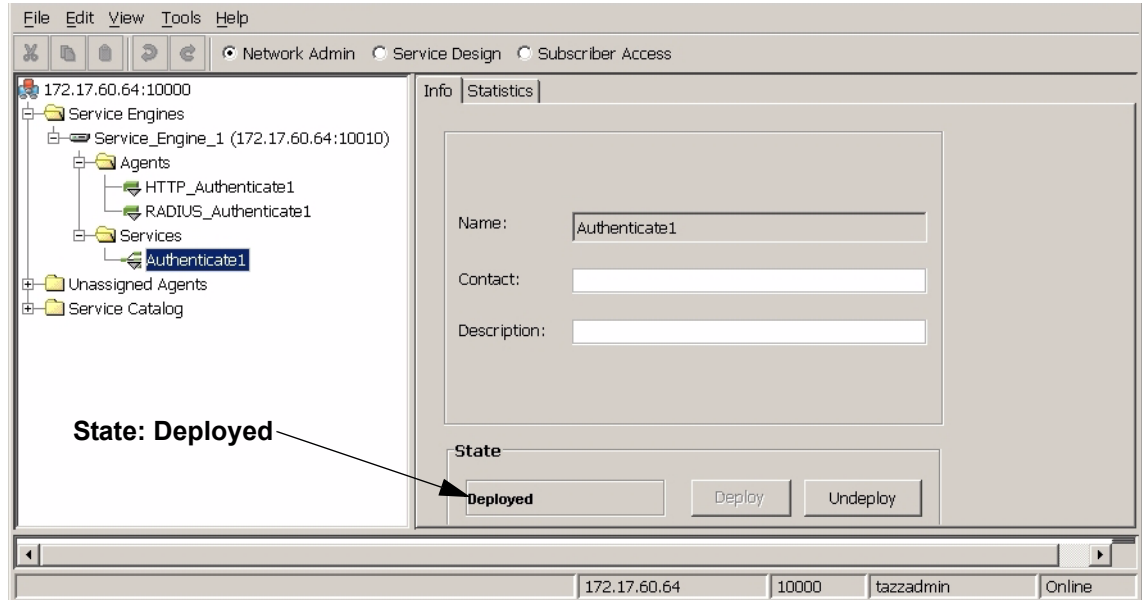
**Note:** To deploy all services for a service engine, right-click the Services folder under the service engine and click **Deploy all** from the drop-down menu.

## Procedure: Undeploying a Service

Follow the steps below to undeploy a service:

1. Click **Network Administration** and open the Services folder. Click the service to undeploy. The State should be *Deployed* (Figure 60).

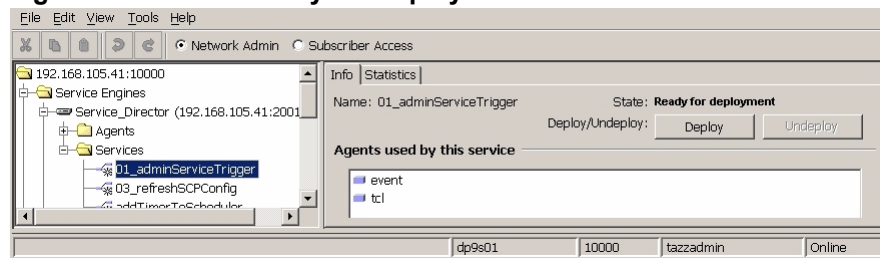
**Figure 60. Deployed Service**



2. Click the **Undeploy** button to undeploy the service. You receive a message that the system is undeploying the service.

The State condition changes to *Ready for deployment* (Figure 61). You can now make changes to the service, then deploy the service again.

**Figure 61. Service Ready for Deployment**



**Note:** To make a change to an agent, you must first undeploy the service, then undeploy the agent.



**Note:** *To undeploy all services for a service engine, right-click the Services folder under the service engine and click **Undeploy all** from the drop-down menu.*

## Configuring Components

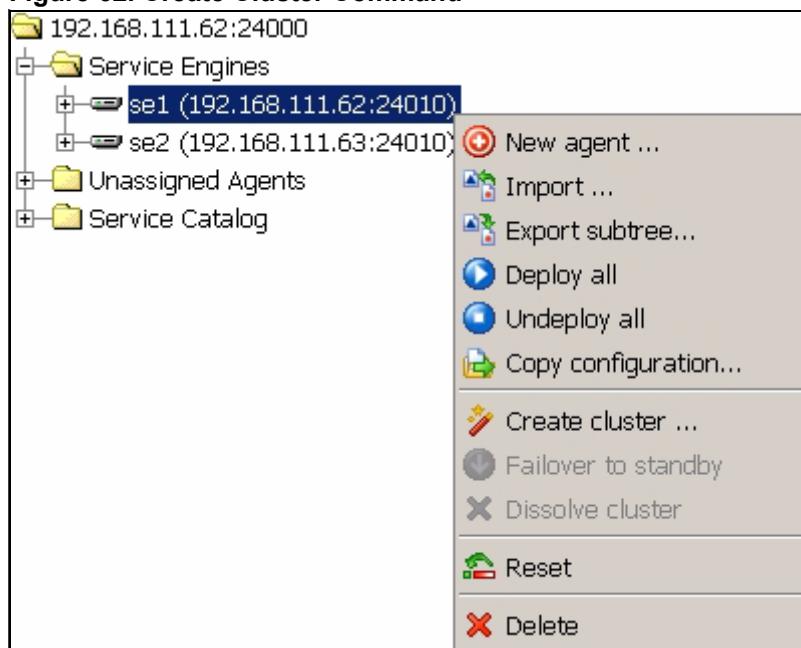
You can use the BPS to configure BPM components.

### Procedure: Configuring a Standby Service Engine

A standby service engine acts as a standby to an active service engine in case of failover. The TDS and Resource Controllers have standbys; Directors do not have standbys. You can configure an existing service engine to have a standby service engine using the following procedure:

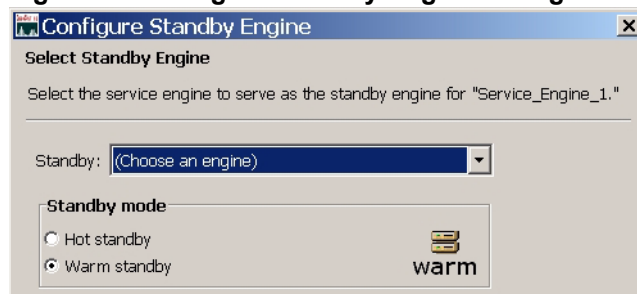
1. When performing a hot cluster (Resource Controller only), unassign any resources that are currently assigned to the standby Resource Controller before creating the cluster.
2. In the Network Administration Tree pane, right-click a service engine. This service engine becomes the active service engine. A drop-down menu of commands appears (Figure 62).

**Figure 62. Create Cluster Command**



3. Choose the **Create cluster** command from the drop-down menu. The Configure Standby Engine dialog box appears (Figure 63).

**Figure 63. Configure Standby Engine Dialog Box**



4. Click the down arrow, and choose the standby service engine from the drop-down list.

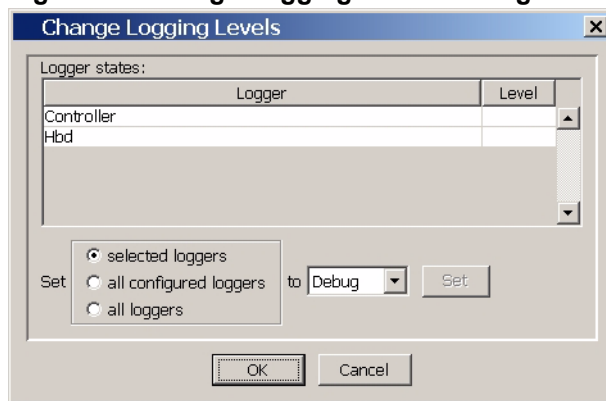
5. If clustering a Resource Controller, choose **Hot standby**. If clustering a Topology Database Server (TDS), choose **Warm standby**. Then click **Next**.
6. Click **Next**. Follow the prompts to continue the configuration.
7. If you selected **Warm standby** in step 5, delete the `/etc/hostname.<primary>` and `/etc/hostname.<secondary>` files on both service engines.

**Procedure: Changing Logging Level for Service Engine**

The Change Logging Levels dialog allows you to change the logging levels of loggers in a service engine:

1. Right-click the service engine in the tree pane.
2. Choose **Change Log Level** from the drop-down list. The Change Logging Levels dialog opens.

**Figure 64. Change Logging Levels Dialog**



3. To change selected loggers, perform these steps:
  - a. Click desired loggers in the list.
  - b. Choose **selected loggers**.
  - c. Click the down-arrow, then choose the logging level from the drop-down list.
  - d. Click **Set**.
4. To change configured loggers, perform these steps:
  - a. Choose **all configured loggers**.
  - b. Click the down-arrow, then choose the logging level from the drop-down list.
  - c. Click **Set**.
5. To change all loggers, perform these steps:
  - a. Choose **all loggers**.
  - b. Click the down-arrow, then choose the logging level from the drop-down list.
  - c. Click **Set**.
6. To change individual loggers, right-click the **Level** column beside the logger, then choose the logging level from the drop-down list.
7. When done changing logging levels, click **OK**.



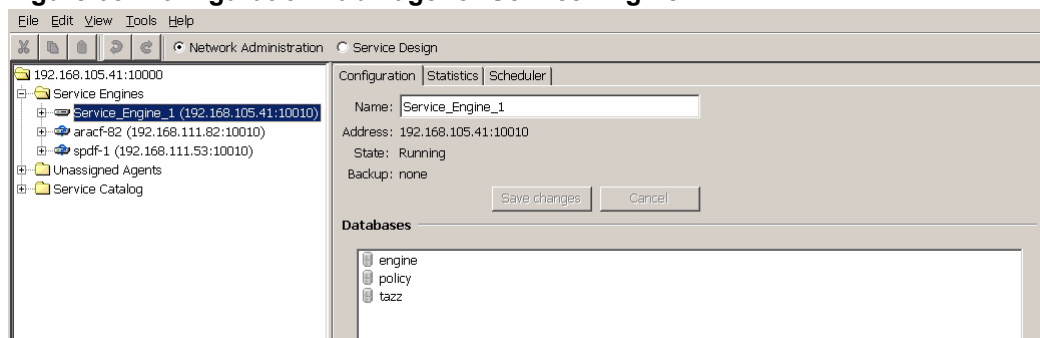
## Procedure: Changing the Name of a Component

You can change the name of a component in the Network Administration Tree pane by using the associated Configuration tab.

For example, you can change the name of a service engine using the following procedure:

1. In the Network Administration Tree pane, choose the service engine from the list.
2. Click the **Configuration** tab. The Configuration tab page for a service engine displays information about the selected service engine (Figure 65).

**Figure 65. Configuration Tab Page for Service Engine**



3. In the Name field, enter a name for the service engine. The **Save changes** button becomes available.
4. Click **Save changes**. The name of the service engine is changed.

## Procedure: Changing Agent Configuration

You can change the configuration of an agent using the following procedure:

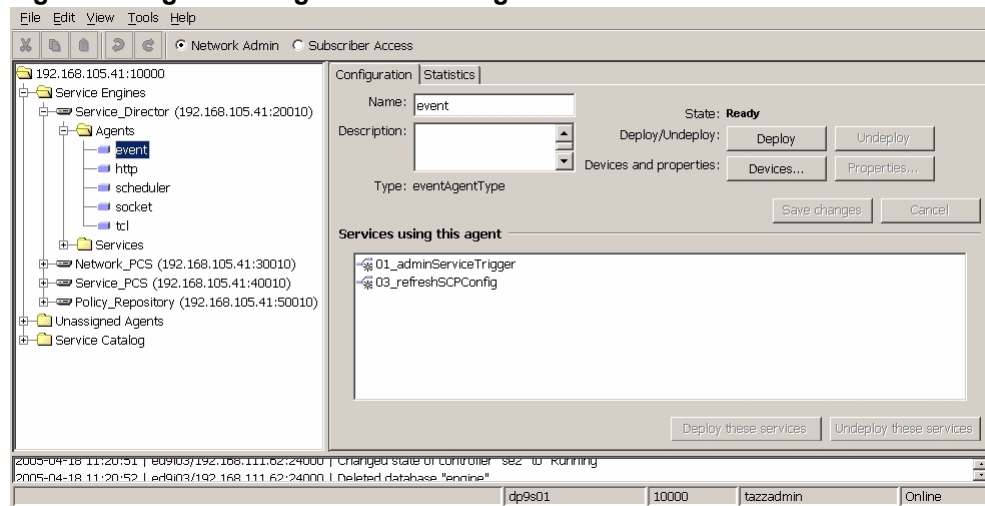
1. In the Network Administration Tree pane, choose the agent from the list.



**Note:** You must undeploy all services that depend on the agent before changing the selected agent.

2. Click the **Configuration** tab. The Agent Configuration tab page displays information about a specific agent. Use this pane to enter or change agent values, access its Properties dialog box or Devices dialog box, and submit those values to the system. [Figure 66](#) is an example of a configuration tab page for an agent.

**Figure 66. Agent Configuration Tab Page**



[Table 8](#) describes the agent object components.

**Table 8. Agent Object Pane Components and Functions**

Component	Function
Category*	Indicates the agent category.
Type*	Indicates the agent type within the category. *This can vary, depending on the agent.
Assigned to	Denotes that the agent is complete but not yet deployed.
State	Denotes that the agent is deployed in the BPM system, and you can use the agent in a running service.
Local	If selected, agent runs on the current BPM system (local).
Remote	If selected, agent runs on another BPM system (remote).
IP Address	The IP address of the remote agent.
Subnet Mask	The subnet mask of the remote agent.

**Table 8. Agent Object Pane Components and Functions**

Component	Function
Gateway	The IP gateway the remote agent uses.
Port	Port number through which remote agent communicates with network.

Depending on the agent you select, the system also presents buttons to select either an Agent Properties dialog box or a Devices dialog box from this page. For some agents, the system presents buttons for both dialog boxes. If the system presents a dialog box for the agent, complete it before you submit the agent values to the system. (See *Agent Properties Dialog Box* and *Agent Devices Dialog Box* below.)

3. Click **Submit**. The configuration of the agent is changed.

### Agent Properties Dialog Box

If the system presents an Agent Properties dialog box, complete it before you submit the agent values to the system. The Agent Properties dialog box (Figure 67) displays information about the selected agent. Use this box to enter or change values for the agent and submit those values to the system. Required properties have an asterisk (\*) beside them.

**Figure 67. Sample Agent Properties Dialog Box**

Property	Value
servletAddress *	192.168.3.124
servletPort *	5555
templatePrefix	http://test/tazz
urlPrefix	http://test:8080/tazz-1.4.1.4/servletRequestHandler/

Table 9 describes the agent properties.

**Table 9. Agent Properties, Definitions, and Examples for HTTP Agent**

Property	Definition	Example
servletAddress	Servlet IP address.	10.68.3.124
servletPort	Port number associated with the servlet address.	5555
templatePrefix	Prefix of the template.	http://traveler/tazz/
urlPrefix	Prefix of the URL.	http://lanai:8080/tazz-1.0.1.5/servlet/RequestHandler

## Agent Devices Dialog Box

If the system presents an Agent Devices dialog box, complete it before you submit the agent values to the system. The Agent Devices dialog box (Figure 68) displays device information about the selected agent. It lists the devices that the agent uses, and the name and value that describe each device. Required properties have an asterisk (\*) beside them. Use this box to enter or change device information for the agent and submit those values to the system.

**Figure 68. Sample Agent Devices Dialog Box**

Property	Value
IPAddress	10.168.3.3
expirationPeriod	
port	1812
portAccounting	1813
proxyPort	
proxyPortAccounting	
proxySharedSecret	
proxySharedSecretAccounting	
retransmissionAttempts	
serverPortAccounting	
serverSharedSecretAccounting	
sharedSecret	*****
sharedSecretAccounting	**
timeoutPeriod	

## Procedure: Patching

A patch is a small piece of code or data that corrects the original code or data. You can import patches for rules or binary files. To import patches:

1. Choose **Tools --> Patch**.
2. Click **Binary files** or **Rules**.
3. Browse to the patch file to import. The selected patch is imported and installed.

## Clusters and Failover

The system offers manual and automatic failover to support system resiliency. Failover is implemented by clustering service engines so that one service engine acts as the standby to another service engine. This section discusses that process. You only cluster Resource Controllers and the Topology Database Server.

The system supports two types of resiliency: warm standby (for Topology Database Servers) and hot standby (for Resource Controllers). There are two major differences between the two schemes.

In warm standby (for Topology Database Servers), agents and services are present on the standby, but not started. Additionally, data-related network interfaces on the standby have identical configurations to those on the active, but remain disabled. When a failover event occurs, the standby enables its interfaces, and sends gratuitous Address Resolution Protocols (ARPs) to inform other network equipment of the new IP to MAC address mapping. Once the standby enables its interfaces, the standby starts the deployed agents and services.

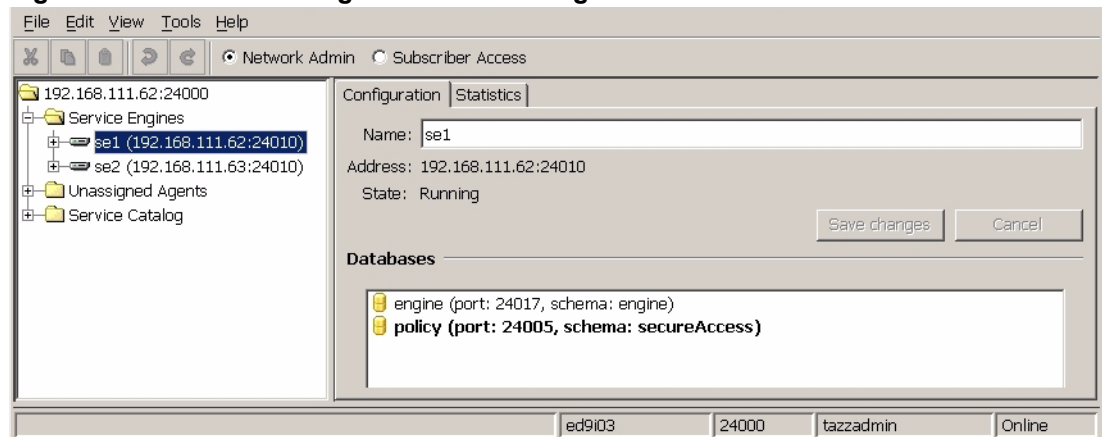
In hot standby (Resource Controller only), agents and services are running, but remain idle. The data interfaces have unique IP addresses and are enabled. A request director forwards requests to the active node in the pair. When a failover event occurs, the director is notified and forwards subsequent requests to the new active (former standby).

### Procedure: Using BPS to Create a Cluster

To join two service engines together into a cluster, first determine which engine serves as the active engine and which is the standby. Both engines must be running on machines with more than one network interface card (NIC), and they both must run on the same port. Both Service Engines must not be part of another cluster already. Both Service Engines must be running.

For example, you can cluster the two Service Engines in [Figure 69](#) together:

**Figure 69. Two Service Engines for Clustering**

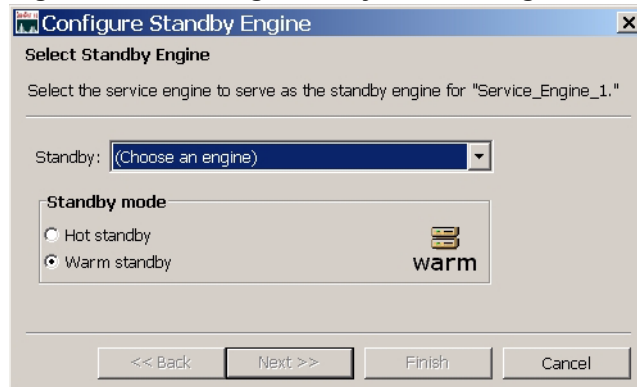


To create a cluster using the BPS:

1. When performing a hot cluster (Resource Controller only), unassign any resources that are currently assigned to the standby Resource Controller before creating the cluster.
2. Right-click the engine that you have chosen to be the “active” member of the cluster.
3. From the drop-down menu, choose **Create cluster**. A software wizard appears.

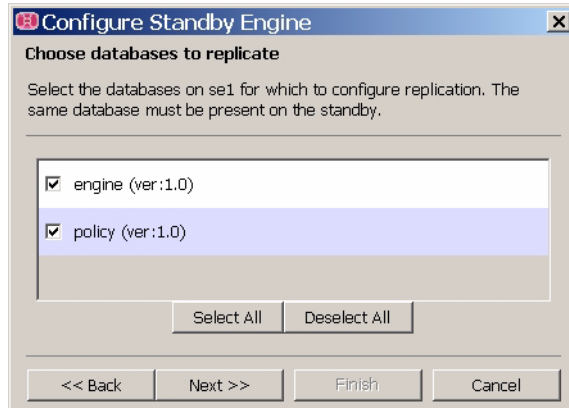
- Choose the engine that you have chosen to be the *standby* engine.

**Figure 70. Choosing Standby Service Engine**



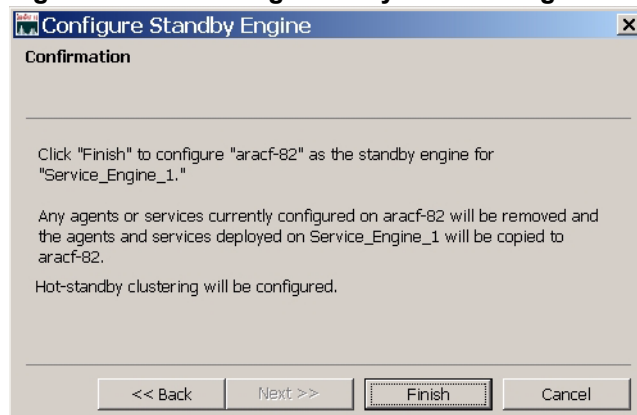
- If clustering a Resource Controller, choose **Hot standby**. If clustering a Topology Database Server (TDS), choose **Warm standby**. Then click **Next**.
- Choose all databases on the *active* service engine to replicate. Click **Next**.

**Figure 71. Confirming Standby Service Engine**



- Confirm your choice, and click **Finish**.

**Figure 72. Confirming Standby Service Engine**

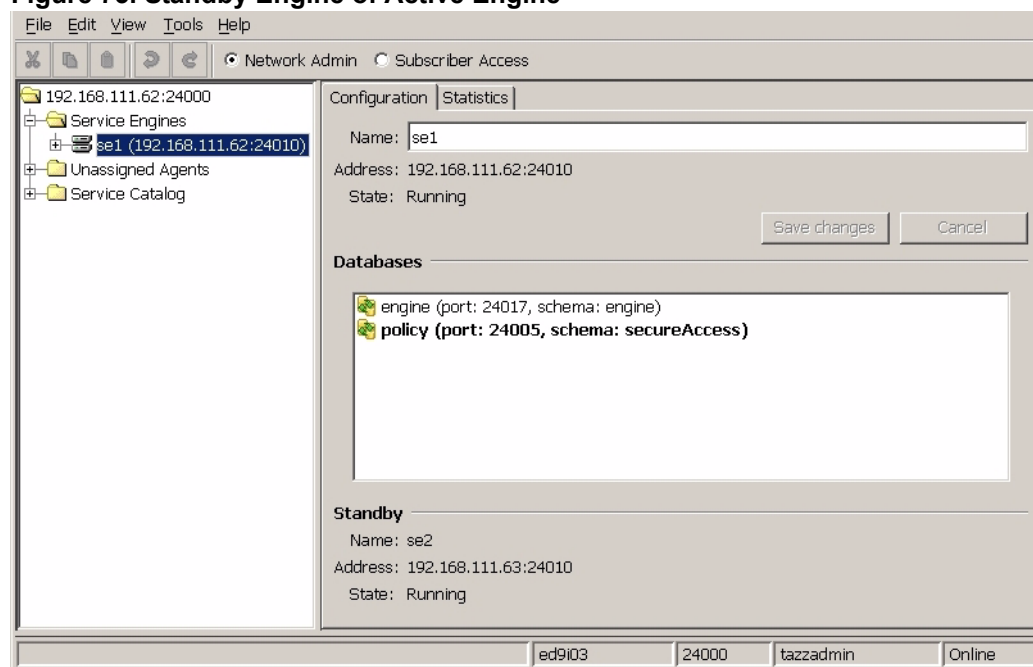


When you click **Finish**, the software clusters the engines together.

8. If you selected **Warm standby** in step 5, delete the `/etc/hostname.<primary>` and `/etc/hostname.<secondary>` files on both service engines.

To visually represent the cluster, the standby engine does not appear independently in the tree list. Instead, the active service engine has a double icon, and the standby service engine appears as the standby in the right pane when the active engine is selected. In addition, the replicated databases also have double icons.

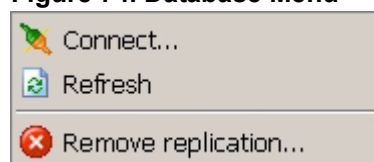
**Figure 73. Standby Engine of Active Engine**



### **Procedure: Removing Database from Replication**

If you right-click one of the databases for a clustered service engine, a drop-down menu appears. From this menu, you can remove the database from replication.

**Figure 74. Database Menu**



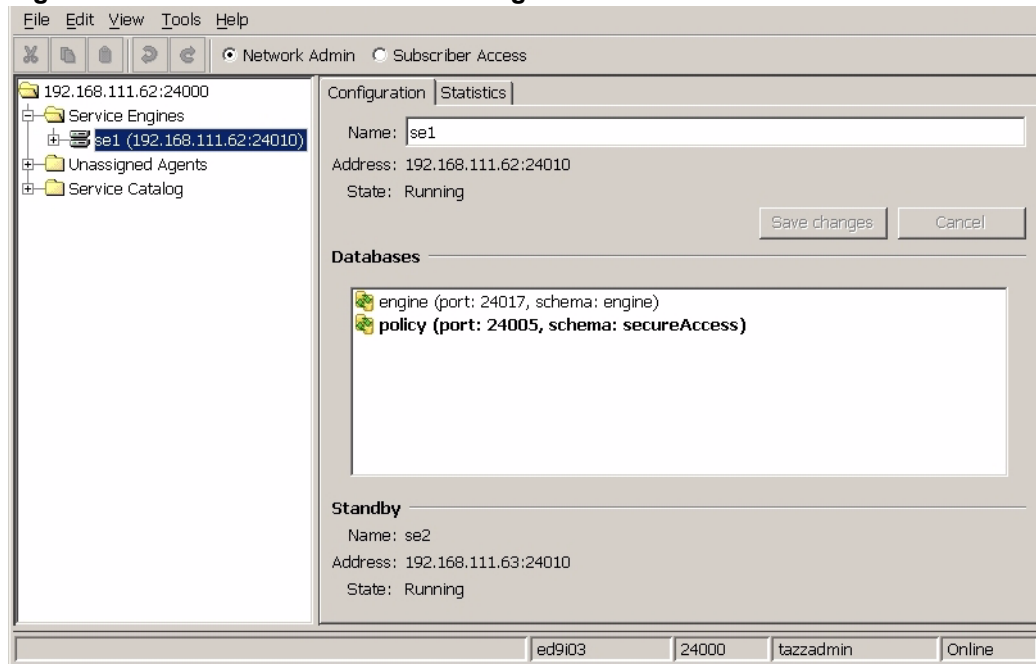
## Procedure: Verifying Cluster Health

You can use the BPS to verify various aspects of the health of a cluster.

### Viable Cluster

A viable cluster shows both engines in the Running state.

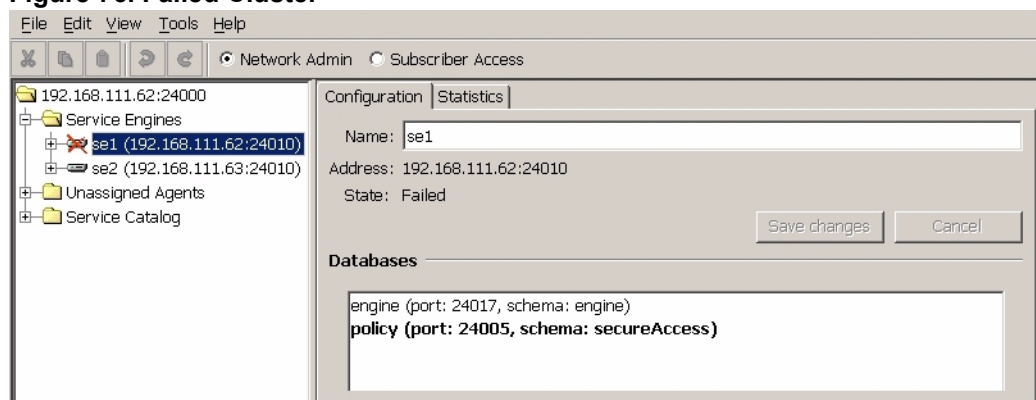
**Figure 75. Viable Clusters: Both Running**



### Failed Cluster

If something goes wrong in the cluster, you can view the states of the cluster components in the user interface. The former *active* service engine appears as Failed.

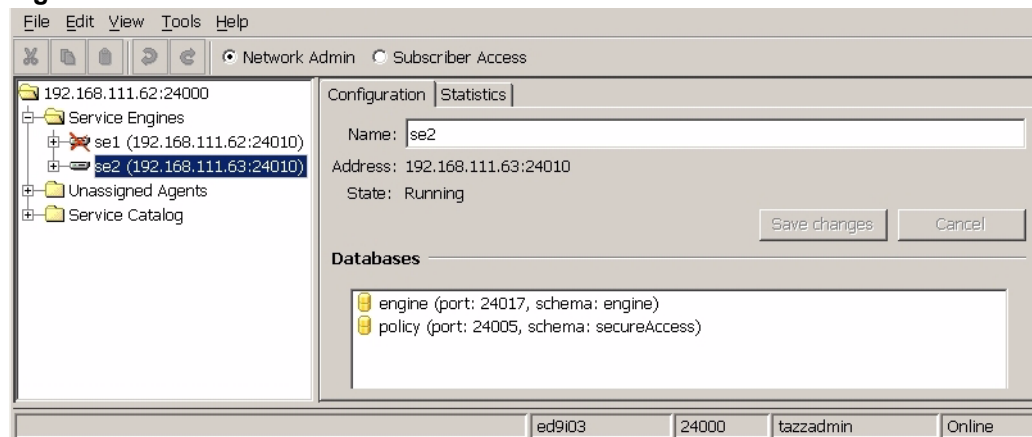
**Figure 76. Failed Cluster**





If the active member of the cluster fails, the cluster is dissolved. The former *standby* service engine is now the active service engine.

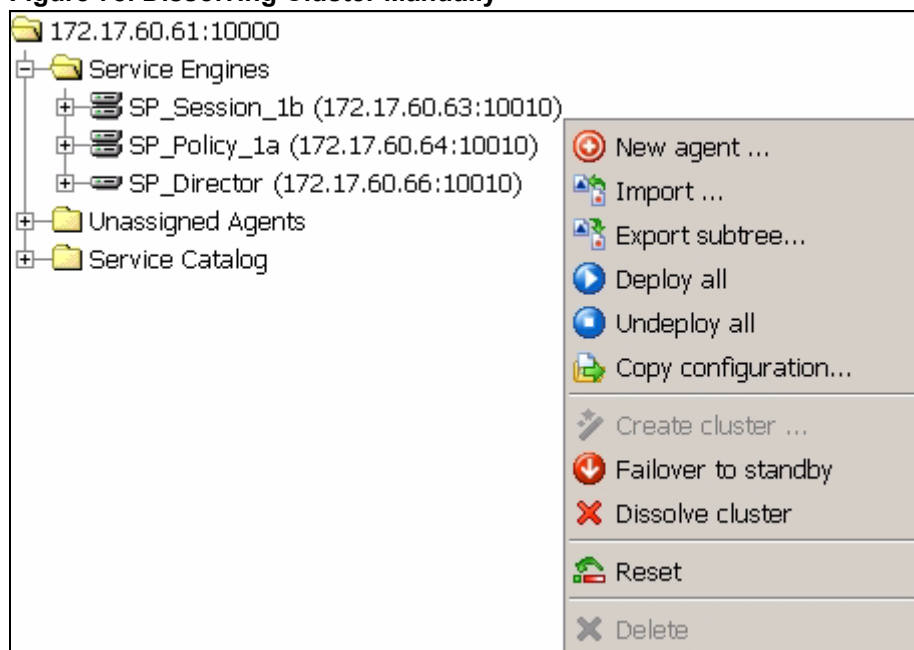
**Figure 77. Active Cluster Fails: Cluster Dissolved**



## Procedure: Dissolving a Cluster

You can dissolve a cluster manually by right-clicking the active service engine and choosing **Dissolve cluster**.

**Figure 78. Dissolving Cluster Manually**



For a warm standby (for Topology Database Servers), if you need to uncluster the service engines, you must re-create the host files for the primary and secondary interfaces. As root on all clustered systems, create the following files:

```
touch /etc/hostname.e1000g2
touch /etc/hostname.e1000g3
```

## Failover

There are two types of failover: manual and automatic.

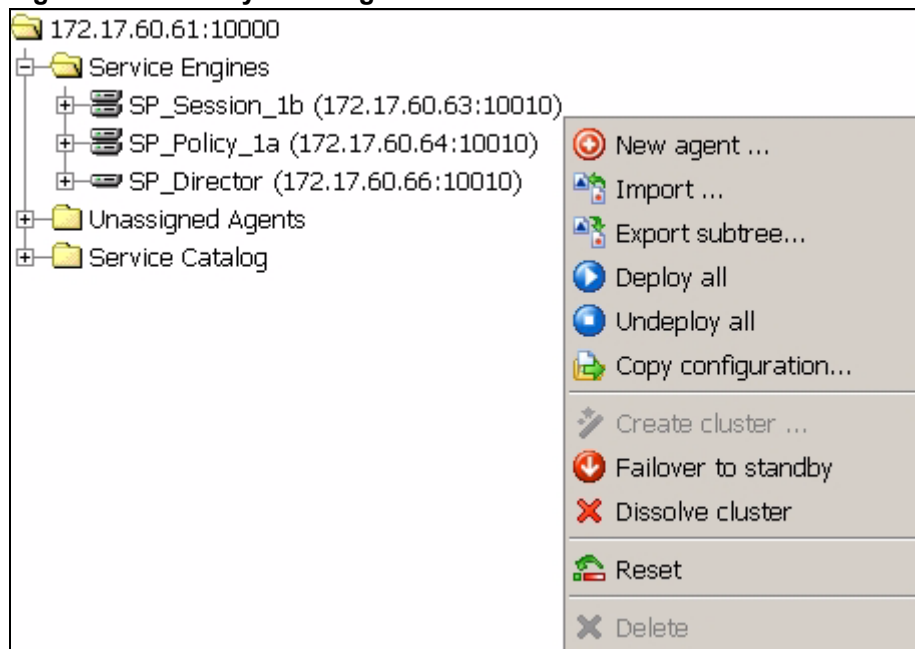
### Manual Failover

The manual failover procedure provides the BPS user a means of initiating a controlled failover where both the active and standby service engines coordinate the scenario in a synchronized fashion. This synchronized sequence of events allows the active scheduler a time window to complete any current jobs it might be in the middle of executing before the interfaces shut off on the active and subsequently start on the standby service engine.

#### Procedure: Manually Initiating Failover

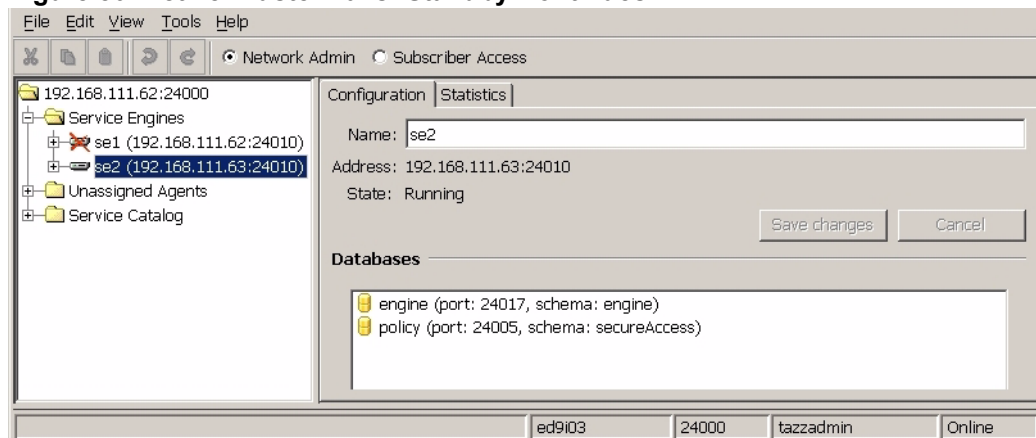
You can manually initiate a failover of the active to the standby. Right-click the active Service Engine and choose **Failover to standby**.

**Figure 79. Manually Initiating Failover**



The active *fails*. All of the agents and services move to the standby.

**Figure 80. Active Cluster Fails: Standby Continues**



During warm failover (for Topology Database Servers), the standby service engine takes over the configured primary and secondary IP addresses. It does this by enabling these interfaces and sending out a gratuitous ARP to force attached hosts to update their respective ARP caches with the new MAC addresses from the standby service engine. The active service engine, in turn, disables its primary and secondary interfaces and changes them over to acquire the old IP addresses from the former standby service engine, disables those interfaces, and proceeds with a reset and shutdown of the system.

## Automatic Failover

For warm clusters (for Topology Database Servers), the TDS initiates automatic failover based on QoS errors reported by Directors, based on the inability of the TDS to connect with a Resource Controller.

## Failed Node Recovery

Recovering a failed node depends on which node failed and how the failure occurred. Recovery usually involves failed node recovery, action on the current active node, BPS actions, and ramdisk considerations.

## Resolving Failover

After a failover event, the former standby of a clustered pair becomes the active, and the former active is considered non-functional. The Director Realm is automatically updated to reflect this change.

The topology Realm on each Director and the TDS has two sets of fields for the active and standby of a given Resource Controller pair. The fields for the active do not start with underscores, those for the standby do. Note also that the system swaps these values when needed, for example, after a failover event. Therefore, the fields without underscores always refer to the currently running active, not necessarily the system originally provisioned as the active.

One of the fields for the active and standby is "Health" (or "\_Health".) This represents the current state of that system with the following values:

- 0 means that the system is running and healthy. In the case of the standby, this means it is available to become active, if needed.
- -1 means that the system is not healthy. This usually means that it has failed, and the operator has not yet intervened to correct the situation.
- -2 means that the system is in failover transition.
- A value greater than 0 means that the system is running and healthy, but that the interface has failed (for example, a reservation has failed) the number of times represented by the number. After a certain (specifiable) number of failures, a failover will occur.

### **Procedure: Resolving Failover of Active Resource Controller**

If an active cluster member fails, its standby should become the active cluster member. Perform the following steps to confirm that this has happened, and to add the former active back into the clustered pair, as the new standby system:

1. Using the portal, check the Director realm on the TDS and verify that the `health` field for the former standby is 0 and that the `_health` field for the former active is -1. This means that the standby has taken over, and that the former active is not available for service. Also, the `host` and `port` fields should match the new standalone, and the `_host` and `_port` fields should match the former active.
2. On the former active, run this command:

```
start_tazz -reset
```

3. Rebuild the cluster by following these steps:
  - a. Using the BPDS, right-click the new active, then choose **Create Cluster**.
  - b. Select both policy and engine databases for replication, when prompted.

### **Procedure: Resolving Failure of Standby Resource Controller**

If a standby cluster member fails, its active remains the active cluster member. Perform the following steps to make sure the Director does not try to failover to the failed standby, and to add the failed standby back into the clustered pair:

1. On the standby, run this command:

```
start_tazz -reset
```

2. Rebuild the cluster by following these steps:
  - a. Using the BPDS, right-click the new active, then choose **Create Cluster**.
  - b. Select both policy and engine databases for replication, when prompted.

### **Procedure: Resolving Failure of Active or Standby TDS**

TDS failover is initiated either as a manual operation, or by the standby heartbeat daemon if it has problems communicating with the active.

Because TDS systems use warm clustering, when a TDS failover occurs, the Directors and Resource Controllers see the new active exactly as the former active (for example, with the same IP address).

For a soft failure (for example, initiated from the BPDS), execute this command on the failed system:

```
start_tazz -reset
```

For hard failures (such as power outage, cable pulls, reboot, and execution of `stop_tazz -f`), perform the following steps:

1. Reboot the failed system.
2. Upon boot, run the following command to plumb and configure the interfaces:

```
/usr/sbin/ifconfig.tazz-setuid <interface> plumb <ip-address>/  
<num-mask-bits> broadcast <broadcast-addr> [up]
```

For example, to configure `e1000g2` with an IP address of `172.1.2.3` and a subnet mask of `255.255.255.0`, one would use:

```
/usr/sbin/ifconfig.tazz-setuid e1000g2 plumb 172.1.2.3/24  
broadcast 172.1.2.2555 up
```

3. To persist IP address changes to the configuration files, run the `tazzifconfig` command with the `-pe[rsist]` option as follows:

```
tazzifconfig -p -c <primary-ip-address> -pe  
tazzifconfig -s -c <secondary-ip-address> -pe
```

where `primary-ip-address` is the former standby IP primary address, and `secondary-ip-address` is the former standby IP secondary address.

4. Execute the following:

```
start_tazz -reset
```

When `start_tazz` prompts "Reset database replication configuration?", type `yes`.

5. If a standby TDS system fails, reset and recluster the standby TDS with the active, using **Warm Standby**.

## What's Next?

In addition to maintenance tasks, problems can arise. [Chapter 5, \*Troubleshooting Tasks\*](#), suggests how to handle certain problems.



# Troubleshooting Tasks

## Overview

Occasionally issues may arise with the Broadband Policy Manager (BPM). There are several ways of identifying and solving problems using the Broadband Policy Studio (BPS).

## Logging in

If you cannot log in to the BPS, check to make sure that you have the correct address and port for the Domain Controller BPM and the application database, and that you are using the correct username and password. If all those items are correct, the Domain Controller BPM or application database may not be running, or there may be an interruption in the connection to them.

## System Information

Some issues are evidence of a more widespread problem. You can examine several aspects of system information using the BPS.

## Network Administration Tree Pane

Here are some of the components you can check using the tree pane of the Network Administration view.

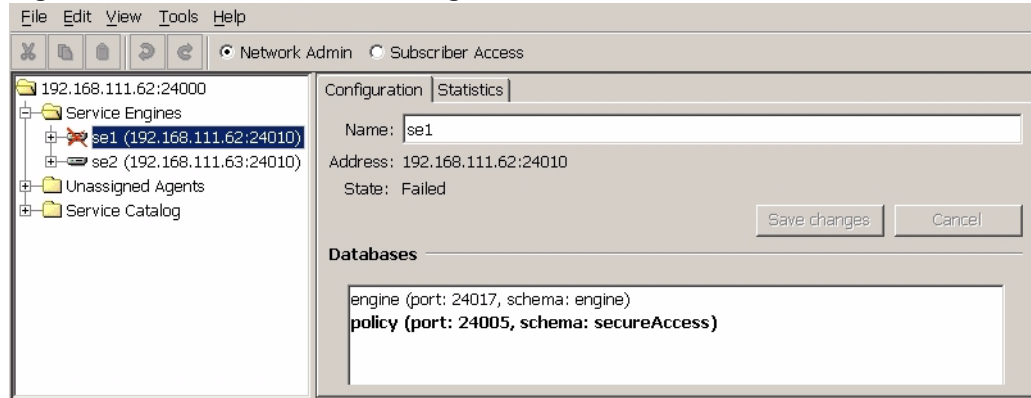
## Service Engine Status

Examining the service engine status can help resolve questions such as:

- Are there any issues with service engines?
- Do clusters of service engines still have their standby?
- Has there been a failover?

Open the Service Engine folder ([Figure 81](#)). A list of Service Engines appears. If there is a problem with a service engine, a red X symbol appears over the Service Engine icon. If this was an active Service Engine, this indicates that a failover has occurred. The problem may mean that the platform is not running, that the BPM system is not running, or that the platform is not reachable on the network.

**Figure 81. Unreachable Service Engine in Network Administration View**



A blue question-mark symbol on a Service Engine icon indicates a new Service Engine that has not yet been activated.

You can tell if a Service Engine is part of a cluster by choosing that Service Engine and clicking the **Configuration** tab. Clustered Service Engines have a standby Service Engine listed. To dissolve the cluster, right-click the Service Engine, then choose **Dissolve cluster**.

You can see which databases a search engine has by choosing the search engine and clicking the **Configuration** tab. The databases appear on the Configuration tab, as in [Figure 81](#).

### Agent Status

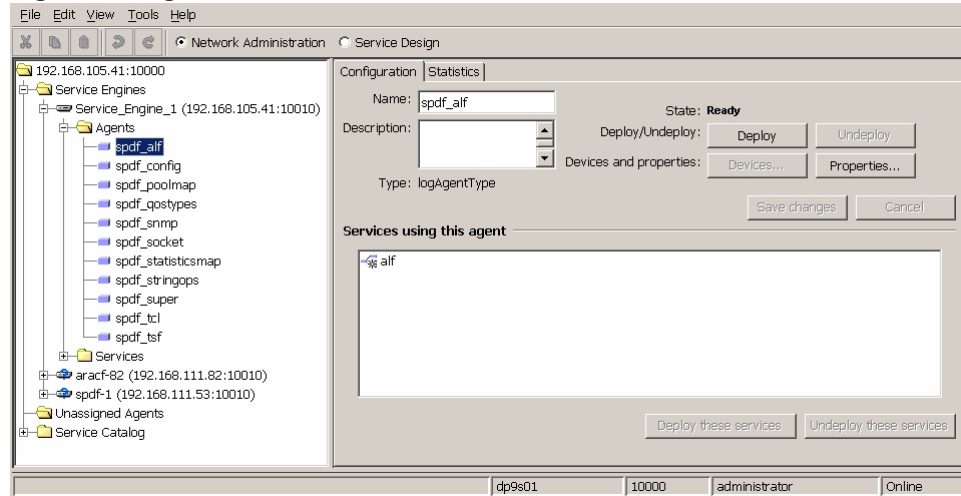
Examining the agent status can help resolve questions such as:

- Are all agents started?
- Are agents deployed?
- Are agents running?
- Are agents configured properly?



Open the Agents folder under each Service Engine folder (Figure 82). A list of agents appears. An agent with a blue icon is not deployed. A blue icon with a star is a new agent. An agent with an X symbol on its blue icon is Invalid. Configure it correctly. An agent with a green icon is running. An agent with a triangle symbol on a green icon is Deployed.

**Figure 82. Agent Status in Network Administration View**



You can obtain statistics on agents. Click the agent, then click the **Statistics** tab. Available statistics on that agent appear. A list of statistics is in [Appendix B - Statistics](#). Some of the more important statistics for troubleshooting purposes include:

**Table 10. Sample Agent Statistics**

Statistic	Detail
agent-restarts	Number of agent restarts.
agent-starts	Number of agent starts.
dispatch-failed	Number of messages switch failed to dispatch. Does not include message-failed.
message-failed	Total number of messages processed unsuccessfully.
message-failed-percent	Percentage of messages processed unsuccessfully.
switch-restarts	Number of switch restarts.

## Service Status

Examining the status of services can help resolve questions such as:

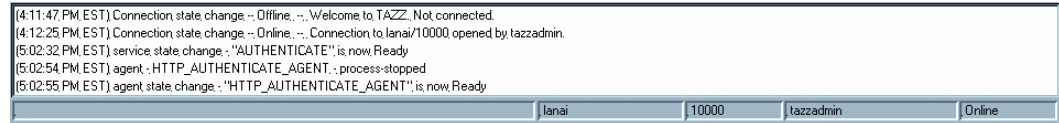
- Are all services started?
- Are services deployed?
- Are services running?
- Are services configured properly?

Open the Services folder under each Service Engine folder. A list of services appears. A service with a blue icon is new and you must configure it. A service with a green icon and a triangle is Deployed.

### Log Pane

The Log pane (Figure 83) presents informational and error messages. This information can be useful for reviewing operations and for investigating problems. For example, the log pane indicates changes to agent state, as well as processes that have stopped or started.

**Figure 83. Sample Log Pane Content**



The Log pane only presents BPM log messages that have occurred since the BPS began running. Thus, it is a good idea to keep the BPS running continuously.

### Isolating the Problem

The first step in troubleshooting is to isolate the problem. Before attempting to identify a specific cause, try to define concisely what is happening. The following questions can help you determine the scope of the problem:

- What is the symptom?
- Is the problem isolated to one service type or does it affect many types of service?
- When did the problem start?
- Has the configuration of the BPM been changed?
- Has any hardware been replaced or upgraded?
- Has the configuration, software, or hardware of network devices changed?

Whether or not you are able to pinpoint the trouble spot, review the general troubleshooting steps outlined in this chapter.

## Investigating Hardware Issues

Hardware may be the problem. A necessary hardware unit may be powered off, configured incorrectly, connected incorrectly, or malfunctioning. The hardware setup is specific to your arrangement, so no specific instructions appear here.

Types of hardware to investigate include:

- PCs, Sun devices, and other computer platforms
- Modems
- Broadband Remote Access Servers (BRAS)
- Routers
- Other network hardware

To investigate hardware issues, one or more of the following steps may be necessary:

- Visual examination of on/off switches
- Visual examination of indicator lights
- Internal diagnostic tests
- External diagnostic tests
- Hardware utilities

Perform these tasks according to the directions of your hardware vendor.

## Investigating Software Issues

Software other than the BPM may be responsible for the problem. Software may be installed improperly, configured incorrectly, failing to interface with other software, or malfunctioning. As with hardware, the software environment is specific to your arrangement.

Types of software to investigate include:

- Operating systems
- Support software, such as security, monitoring, or maintenance software
- Network software

To investigate software issues, one or more of the following steps may be necessary:

- Ensuring that software is running
- Diagnostic tests
- Checking interfaces

## Troubleshooting Considerations

After examining all the available sources of BPM information, you probably have a good idea of the source of the issue and how best to solve it. Here are some further considerations when investigating problems.

### Is the BPM Running?

1. If the BPM is not running, this may affect in many different ways. One way to determine if the BPM is running is to try to log in to the BPM using the BPS.

2. If you can successfully log in to the BPM, that means that the BPM is running. If you cannot successfully log in to the BPM, the BPM may not be running. You may need to start the BPM. However, it is also possible that the BPM is running, but the BPS cannot reach the BPM. This may be due to network problems between the two, or to incorrect login parameters.

### **Is the Service Engine Running?**

If a service engine is not running, none of its associated services are running. The BPS indicates whether a service engine is running or not.

### **Are Services and Agents Deployed?**

If necessary services and agents are not deployed, this may affect in many different ways. The BPS indicates whether services and agents are deployed.

### **When Did Problem Occur?**

The timing of an issue is a significant indication to its cause. What was happening as the problem occurred? Were any component configurations changing? Identifying associated activities and events can help resolve an issue.

# Appendix A - Glossary

This appendix contains abbreviations, acronyms, terms, and their definitions.

**Table A-1. Terms and Definitions.**

Term	Definition
<b>A</b>	
Accounting Log Function	ALF. The Accounting Log Function records entrance parameters, internal decisions, and exit responses.
ACF	Admission Control Function. The ACF provides the core logic for performing admission control. It is programmed with a set of policies that define admission control behavior.
Action	An action is an operational category for changing, or inquiring about, a network element.
Active BPM	In a pair of BPMs, the active BPM processes requests. A standby BPM constantly monitors the health of the active BPM. If the active BPM is not viable, the standby BPM becomes the active BPM.
Admission Control Function	ACF. The ACF provides the core logic for performing admission control. It is programmed with a set of policies that define admission control behavior.
Agent	An internal BPM component that interacts with a device. The designer creates the agent and configures it to interact with a specific device by indicating the device type, IP address, and port number. The designer then assigns the agent to perform service functions.
Agent Configuration	Agent information that comprises a specific agent type instance. For example, a RADIUS agent configuration contains appropriate IP address, port, and shared secret values for a RADIUS agent type.
Agent Function	The service designer uses the BPDS to drag and drop an agent function into a flow in the BPDS. An agent, interacting with a device, performs the actual operation.
Agent Instance	A running instance of an agent type.
Agent Package	Software that allows agents to interact with a particular device type. For example, a RADIUS agent package contains software that allows the creation of agents that interact with specific RADIUS devices.
Agent Type	The agent type describes a particular type of agent that you can load onto the system. You select the agent type when you create the agent instance.
AI	Application Interface. The underlying frameworks use Application Interfaces to notify the Application of network events. The Cisco framework provides these interfaces.
Alarm Notification Function	ANF. The Alarm Notification Function issues SNMP traps to alert external systems of aberrant behavior in the BPM.
ALF	Accounting Log Function. The Accounting Log Function records entrance parameters, internal decisions, and exit responses.

Table A-1. Terms and Definitions.

Term	Definition
ANF	Alarm Notification Function. The Alarm Notification Function (ANF) issues SNMP traps to alert external systems of aberrant behavior in the BPM.
API	Application Program Interface. An API is a set of routines, protocols, and tools for building software applications. An API makes it easier to develop a program by providing the required building blocks. A programmer puts the blocks together.
Application	A service that maps business models and operational procedures directly into IP services, executable by their customers, for example, video on demand or automatic backup. See also Service.
Application Interface	AI. The underlying frameworks use Application Interfaces to notify the Application of network events. The Cisco framework provides these interfaces.
Application Program Interface	API. An API is a set of routines, protocols, and tools for building software applications. An API makes it easier to develop a program by providing the required building blocks. A programmer puts the blocks together.
Application Service Provider	ASP. An ASP is a business that provides computer-based services to customers over a network.
ASP	Application Service Provider. An ASP is a business that provides computer-based services to customers over a network.
Asynchronous Transfer Mode	ATM. Asynchronous Transfer Mode is a network technology based on transferring data in cells or packets of a fixed size.
ATM	Asynchronous Transfer Mode. ATM is a network technology based on transferring data in cells or packets of a fixed size.
Attribute	An attribute is a datum about a network session or a device session. Attributes contain a name and value and a distinguishing namespace. In the BPDS Object Manager tool, a simple type with a default value. An object can have several attributes.
<b>B</b>	
Backend	Software that runs on the BPM. It comprises the controller, engine, agent host, activation daemon, and scheduler processes; synonymous with BPM.
BGP	Border Gateway Protocol. An exterior gateway routing protocol that enables groups of routers to share routing information to establish efficient, loop-free routes. BGP is commonly used within and between ISPs.
Border Gateway Protocol	BGP. An exterior gateway routing protocol that enables groups of routers to share routing information to establish efficient, loop-free routes. BGP is commonly used within and between ISPs.
BPDS	Broadband Policy Design Studio. The BPDS is a graphical user interface to the BPM. The BPDS includes a service design feature.
BPM	Broadband Policy Manager. The BPM is a product suite used by service providers to create and deploy advanced services on broadband networks. A BPM system can be configured as a Director, Domain Controller, Resource Controller, or Topology Database Server.
BPS	Broadband Policy Studio. The BPS is a graphical user interface, similar to the BPDS. The BPS does not include the service design feature.

Table A-1. Terms and Definitions.

Term	Definition
BRAS	Broadband Remote Access Server. A BRAS device routes traffic to and from the digital subscriber line access multiplexers on an ISP network.
Broadband Policy Design Studio	BPDS. The BPDS is a graphical user interface, similar to the BPS. The BPDS includes a service design feature.
Broadband Policy Manager	BPM. The BPM is a product suite used by service providers to create and deploy advanced services on broadband networks.
Broadband Policy Studio	BPS. The BPS is a graphical user interface, similar to the BPDS. The BPS does not include the service design feature.
Broadband Remote Access Server	BRAS. A BRAS device routes traffic to and from the digital subscriber line access multiplexers (DSLAM) on an ISP network.
<b>C</b>	
CAC	Capacity Admission Control. CAC monitors, controls, and enforces the use of network resources and services with policy-based management over broadband access and MPLS core networks.
Capacity Admission Control	CAC. CAC monitors, controls, and enforces the use of network resources and services with policy-based management over broadband access and MPLS core networks.
Cisco Network Registrar	CNR. The CNR is a full-featured DNS/DHCP system that provides scalable naming and addressing services for service provider and enterprise networks.
Class of Service	CoS. This is a traffic prioritization scheme that enables more predictable traffic delivery, based on application requirements.
Classless Inter-Domain Routing	CIDR. This IP addressing scheme addresses the size of routing tables and makes more IP addresses available within organizations. CIDR is also called supernetting.
CIDR	Classless Inter-Domain Routing. This IP addressing scheme addresses the size of routing tables and makes more IP addresses available within organizations. CIDR is also called supernetting.
Client	This is a generic term that denotes the BPM BPDS application.
CoS	Class of Service. This is a traffic prioritization scheme that enables more predictable traffic delivery, based on application requirements.
CPE	customer premises equipment. This is communications equipment that resides on the customer premises. It is owned or leased by the customer.
CLI	command line interface. This is a user interface common to computers. The user enters a command. The computer acts on the command.
Cluster	A pair of cooperating and redundant BPMs.
CNR	Cisco Network Registrar. The CNR is a full-featured DNS/DHCP system that provides scalable naming and addressing services for service provider and enterprise networks.
Command Line Interface	CLI. This is a user interface common to computers. The user enters a command. The computer acts on the command.
Component	An object comprising data and code. A component provides a well-specified set of publicly available services. All devices, services, and applications on a network are components.

Table A-1. Terms and Definitions.

Term	Definition
Configuration	Information necessary to construct an instance of a type (agent, service).
Controller	A software element that runs on the BPM and controls various elements of the backend. Usually only one controller exists per backend; therefore, from the BPDS perspective, the controller is the backend.
Customer Premises Equipment	CPE. This is communications equipment that resides on the customer premises. It is owned or leased by the customer.
<b>D</b>	
DAF	Device Adapter Function. A DAF translates between the protocol and device type-specific events of at the PIF layer and the abstract application events at the Application layer. A DAF can be assigned to multiple device types and multiple DAFs can be assigned to one device.
Deep Packet Inspection Protocol	DPI. This is network packet filtering that examines packet <i>data</i> , searching for nonprotocol compliance or predefined criteria, to decide if the packet can pass. This is in contrast to shallow packet inspection (called packet inspection), which checks only the packet <i>header</i> .
Device	Any piece of software or hardware connected to a network. RADIUS servers, routers, billing systems, accounting systems, and video servers are devices. An agent communicates with a device.
Device Access	A device access is data about accessing a device instance. Most devices require authentication before any device action can occur. The device access contains this authentication data and other related data. Each device instance has one device access per management protocol.
Device Action	A device action is the implementation of an action for a given device type. That is, it is the actual set of instructions necessary to change the functioning of the device instance.
Device Adapter Function	DAF. A DAF translates between the protocol and device type-specific events of at the PIF layer and the abstract application events at the Application layer. A DAF can be assigned to multiple device types and multiple DAFs can be assigned to one device.
Device Adapter Function Flow	A Flow that handles a protocol event for a specific device type.
Device Handler Dispatch Service	DHDS. DHDS provides routing services for PIFs and Applications requesting invocation of DAF operations.
Device Instance	A device instance is a device type in use in the network. For example, a Cisco 10K device at IP address 128.148.176.10. Device instances are grouped according to roles.
Device Rule	A device rule is a provisioned list of steps that apply a policy to a device. A device rule consists of a set of instructions that the BPM sends to the device to apply the given policy. Device rules can retrieve information from connected devices. Preconfigured device rules are useful for configuring a new BPM system. See also Device Type and Policy Rule.



Table A-1. Terms and Definitions.

Term	Definition
Device Session	A device session contains data about a device instance used by a network session. For example, information about the bras would be encoded in a device session.
Device Type	A device type is a vendor's network element hardware. Device types are grouped according to roles and are based on device attributes, such as vendor, model, hardware version, and software version. See also Device Rule.
DHDS	Device Handler Dispatch Service. DHDS provides routing services for PIFs and applications requesting invocation of DAF operations.
Digital Subscriber Line	DSL. DSL technologies use sophisticated modulation schemes to pack data onto copper wires.
Digital Subscriber Line Access Multiplexer	DSLAM. This mechanism links customer DSL connections to a single high-speed ATM line.
Director	A Director is one or more stateless installations that takes requests and routes them to appropriate Resource Controllers, to handle the specific incoming requests.
Director Realm	The Director Realm stores information required by Director systems, including information about network devices (such as BRAS devices). The information specifies the Resource Controller responsible for each device and the IP address pools each device handles. A Director uses this information to forward an incoming request to the correct Resource Controller. The Topology Database Server maintains the Director Realm, and the server distributes its updates to each Director when updates occur.
Domain	One or more cooperating Broadband Policy Managers (BPMs) managed by a single domain repository.
Domain Controller	The Domain Controller is a standalone system responsible for domain management, including application deployment, configuration, and health for all systems in the domain. Only one Domain Controller exists per domain.
Domain Data	Data maintained about the elements in a domain; for example, controller host and port configuration, database host and port information, agent and service configuration and deployment information.
Domain Realm	The Domain Realm maintains application level information about the physical network topology. The nodes in the topology represent Director and Resource Controller systems. The Topology Database Server uses the Domain Realm to understand the system topology. Links represent connectivity between cluster pairs. Resources represent interfaces on the component systems, system health, cluster information, and system configuration.
Domain Repository	The master database that contains configuration information for each domain element.
DPI	Deep Packet Inspection Protocol. This is network packet filtering that examines packet <i>data</i> , searching for nonprotocol compliance or predefined criteria, to decide if the packet can pass. This is in contrast to shallow packet inspection (called packet inspection), which checks only the packet <i>header</i> .

Table A-1. Terms and Definitions.

Term	Definition
DSL	Digital Subscriber Line. DSL technologies use sophisticated modulation schemes to pack data onto copper wires.
DSLAM	Digital Subscriber Line Access Multiplexer. This mechanism links customer DSL connections to a single high-speed ATM line.
<b>E</b>	
Element	An object with the BPM: package; agent configuration; service instance; shared object.
Enumeration	In the BPDS Object manager tool, enumeration is contained within a simple type.
Ethernet	The Ethernet is a large and diverse family of frame-based computer networking technologies for local area networks (LANs). It defines a number of wiring and signaling standards for the physical layer, two means of network access at the Media Access Control (MAC)/Data Link Layer, and a common addressing format. Ethernet has been standardized as IEEE 802.3.
ETSI	European Telecommunications Standards Institute. ETSI is an independent, non-profit organization, whose mission is to produce telecommunications standards for today and for the future.
European Telecommunications Standards Institute	ETSI. ETSI is an independent, non-profit organization, whose mission is to produce telecommunications standards for today and for the future.
<b>F</b>	
Field Replaceable Unit	FRU. An FRU represents an element (e.g., entire system, BPDS client software, agent) within the Broadband Policy Managers (BPM) that has a version associated with it. A FRU is a subset of an element.
Flow	The movement of data or control between agents. It is a collection of one or more operators and zero or more routes. The designer uses flows to define services and applications.
FRU	Field Replaceable Unit. An FRU represents an element (e.g., entire system, BPDS client software, agent) within the Broadband Policy Managers (BPM) that has a version associated with it. A FRU is a subset of an element.
Function	The element that performs an operation, based on inputs and returns the results of the operation via its outputs. The designer drags and drops a function into a flow in the BPDS. An agent, interacting with a device, performs the actual operation.
<b>G</b>	
Graphical User Interface	GUI. A program interface that takes advantage of the computer's graphics capabilities to make the program easier to use. For the BPM, the GUI is the BPDS.
GUI	Graphical User Interface. A program interface that takes advantage of the computer's graphics capabilities to make the program easier to use. For the BPM, the GUI is the BPDS.
<b>H</b>	

Table A-1. Terms and Definitions.

Term	Definition
Handler	A handler enables flow of control between the PIF, DAF, and SMF interfaces. It includes details about the appropriate service flow to call under specific conditions.
Handler Flow	A Handler Flow normalizes protocol-specific parameters before forwarding them to an application. An application can indirectly invoke a Handler Flow using the DHDS.
Head Version	The latest version of an element.
Hypertext Preprocessor	PHP. PHP is an open source, server-side, HTML embedded scripting language used to create dynamic Web pages.
<b>I</b>	
Implementation	An instruction set for executing a specification.
Instance	An executing type (agent, service), created from a specification, implementation, and configuration. An agent instance is a specific implementation of that agent type.
Interface	A collection of functions.
Internet Service Provider	ISP. An ISP is a company that provides access to the Internet. For a monthly fee, the company provides a software package, username, password and access phone number. In addition to serving individuals, ISPs also serve large companies, providing a direct connection from the company network to the Internet.
IP address	The address that identifies a computer. The IP address format is a 32-bit numeric address written as four numbers (0 to 255) separated by periods.
ISP	Internet Service Provider. An ISP is a company that provides access to the Internet. For a monthly fee, the company provides a software package, username, password and access phone number. In addition to serving individuals, ISPs also serve large companies, providing a direct connection from the company network to the Internet.
<b>J</b>	
<b>K</b>	
Key	A key is an identifier used in conjunction with network sessions.
<b>L</b>	
LAN	Local Area Network. A LAN is computer network that spans a relatively small area. Most LANs are confined to a single building or group of buildings. A LAN connect workstations and personal computers. This allows users to share devices and data and communicate via email.
L2TP	Layer Two Tunneling Protocol. L2TP is an extension to the PPP protocol that enables ISPs to operate VPNs.
Layer Two Tunneling Protocol	L2TP. L2TP is an extension to the PPP protocol that enables ISPs to operate VPNs.
Link	A link is a line or channel over which data is transmitted.

Table A-1. Terms and Definitions.

Term	Definition
Local Area Network	LAN. A LAN is computer network that spans a relatively small area. Most LANs are confined to a single building or group of buildings. A LAN connect workstations and personal computers. This allows users to share devices and data and communicate via email.
<b>M</b>	
Management Protocol	A management protocol is the mechanism for managing a network element. Common management protocols are RADIUS and SNMP.
Metadata	In the BPDS, this is the data structure. A customer can import metadata to invoke a structure for his or her database.
MPLS	Multiprotocol Label Switching. MPLS integrates Layer 2 network link information into Layer 3 within an autonomous system or ISP. It improves IP-packet exchange and allows operators to divert and route traffic around link failures, congestion, and bottlenecks.
Multiprotocol Label Switching	MPLS. MPLS integrates Layer 2 network link information into Layer 3 within an autonomous system or ISP. It improves IP-packet exchange and allows operators to divert and route traffic around link failures, congestion, and bottlenecks.
<b>N</b>	
N + 1 Redundancy	The ability for service engines to use one service engine as a backup.
NAF	Network Adaptation Function. NAF. The NAF dynamically resizes network links and queue sizes, based on the ability of the underlying network to adapt after a request from the ACF.
Namespace	A namespace helps distinguish two or more values that otherwise would conflict with each other.
NAS	Network Attached Storage. A NAS device is a server dedicated to file sharing, allowing more hard disk storage space to be added to a network that already utilizes servers without shutting them down for maintenance and upgrades. A NAS device can exist anywhere in a LAN and can be made up of multiple networked NAS devices.
NAV	Network Admin view. In the BPS and BPDS graphical user interfaces to the BPM, this is the network view where you can perform administration tasks.
Network	A network is a group of two or more computer systems linked together. Local-area networks (LANs), wide-area networks (WANs), and metropolitan-area networks MANs are typical networks.
Network Adaptation Function	NAF. The NA) dynamically resizes network links and queue sizes, based on the ability of the underlying network to adapt after a request from the ACF.
Network Admin View	NAV. In the BPS and BPDS graphical user interfaces to the BPM, this is the network view where you can perform administration tasks.
Network Attached Storage	NAS. A NAS device is a server dedicated to file sharing, allowing more hard disk storage space to be added to a network that already utilizes servers without shutting them down for maintenance and upgrades. A NAS device can exist anywhere in a LAN and can be made up of multiple networked NAS devices.
Network Event	A network event is a set of install and uninstall rules, contained within a profile, that are performed in sequence.

Table A-1. Terms and Definitions.

Term	Definition
Network Manager	NM. The NM product provides a framework for controlling and querying the element configurations in the broadband network.
Network Policy	A network policy is a device rule entry. The device rule contains commands to configure a network device to apply a network policy. See also Device Rule, Policy Rule.
Network Realm	The Network Realm stores specific network adaptation information, such as the devices active on a particular Resource Controller, profiles, and handlers. The Network Realm is centrally provisioned on the Topology Database Server, and it is distributed to all Resource Controllers.
Network Session	A network session represents a single point-to-point connection in the network, for example, a VoIP call.
Network Storage Function	NSF. The Network Storage Function provides access to the Network Information Model.
NM	Network Manager. The NM product provides a framework for controlling and querying the element configurations in the broadband network.
Node	In networks, a processing location. A node can be a computer or some other device, such as a printer. Every node has a unique network address, sometimes called a Data Link Control (DLC) address or Media Access Control (MAC) address.
NSF	Network Storage Function. The NSF provides access to the Network Information Model.
<b>O</b>	
Object	An agent, controller, function, service, switch, or service within the Broadband Policy Manager (BPM).
Object Dependency	An exact object type, for example a Cisco 2500 router agent, that a service depends on. The service designer adds the object type to the dependency list of the service. All Interfaces supported by the object type are then available for use with the service.
Object Type	In the BPDS, an object type is defined with attributes. It can own contain, and associate with other object types.
OC	Orchestration Controller. That portion of the Broadband Policy Managers (BPM) that controls processes such as username and password authentication.
Operation and Support System	OSS. OSS refers to a suite of programs that enable an enterprise to monitor, analyze, and manage a network system. The term originally referred to a management system that controlled telephone and computer networks. It now applies to the business world to mean a system that supports network operations.
Operator	A representation of actions to be undertaken on a system networked to a Broadband Policy Managers (BPM).
Orchestration Controller	OC. That portion of the Broadband Policy Managers (BPM) that controls processes such as username and password authentication.
Orchestration Network	The process for handling service calls over a network. It defines the flow of control and information between work units.

Table A-1. Terms and Definitions.

Term	Definition
OSS	Operation and Support Systems. OSS refers to a suite of programs that enable an enterprise to monitor, analyze, and manage a network system.
<b>P</b>	
Pad	A collection of pins on an operator. This appears as a box along the edge of an operator.
Path Computation Function	PCF. The PCF determines the path through the topology for any given end-to-end session, as requested by the ACF.
PCF	Path Computation Function. The PCF determines the path through the topology for any given end-to-end session, as requested by the ACF.
PDP	Policy Decision Point. The PDP is a component of policy-based management. When a user tries to access a file or other resource on a system using policy-based access management, the PDP decides whether or not to authorize the user based on user attributes.
PE	Policy Engine. The software that stores and manages user profile information, subscriber access records, policy rules; also known as the policy database.
PEP	Policy Enforcement Point. The PEP is the logical entity or place on a server that makes admission control and policy decisions in response to a request from a user wanting to access a resource on a computer or network server.
PHP	Hypertext Preprocessor (PHP). PHP is an open source, server-side, HTML embedded scripting language used to create dynamic Web pages.
PIF	Protocol Interface Function. A PIF service encapsulates an interface with an external device or service
PIF Agent	An Agent that acts an adaptor between the system and an external device or service.
Pin	An input or output from an operator. The pin serves as a route endpoint and holds a single input or output value. For example, an operator that needs a username and password as input has two input pins; one for the username; the other, the password.
PMF	Profile Management Function. The Profile Management Function (PMF) activates and deactivates network profiles on subscriber sessions.
Point-to-Point Protocol Over ATM	PPPoA. PPPoA relies on two widely accepted standards: PPP and ATM. It is an end-to-end asymmetric digital subscriber line (ADSL) architecture.
Point-to-Point Termination Aggregation	PTA. This is a method of aggregating IP traffic by terminating PPP sessions and amassing the IP traffic into a single routing domain.
Policy	A flow comprising a rule or set of rules that take a specific action provided by an ISP for its subscribers. For example, a policy for subscriber access directs how the system identifies a subscriber via user id, access type, and log in location. A policy performs an operation, based on input and returns the results of its action as output.

Table A-1. Terms and Definitions.

Term	Definition
Policy Database	The database of policy objects that services access to make policy decisions.
Policy Decision Point	PDP. The PDP is a component of policy-based management. When a user tries to access a file or other resource on a system using policy-based access management, the PDP decides whether or not to authorize the user based on user attributes.
Policy Enforcement Point	PEP. The PEP is the logical entity or place on a server that makes admission control and policy decisions in response to a request from a user wanting to access a resource on a computer or network server.
Policy Engine	PE. The software that stores and manages user profile information, subscriber access records, policy rules; also known as the policy database.
Policy Function	Policy rules encapsulated in a TCL agent <i>execute</i> function.
Policy Repository	The Policy Repository BPM stores all persistent data associated with customers and services. It utilizes industry-standard database technology that allows any of the underlying system elements to interrogate it.
Pool	A pool represents a range of IP addresses. A BRAS handles one or more address ranges. A Resource Controller potentially handles multiple BRASs. So a typical Resource Controller can handle multiple ranges of IP addresses (multiple pools).
PPPoA	Point-to-Point Protocol Over Asynchronous Transfer Mode. PPPoA relies on two widely accepted standards: PPP and ATM. It is an end-to-end asymmetric digital subscriber line (ADSL) architecture.
Presence Director	The Presence Director is an optional, modified, Director service that handles receives session requests and distributes them to the appropriate Resource Controllers.
Profile	A profile is a procedure for changing a set of related network elements for a given purpose, for example, increasing the bandwidth associated with a network session.
Profile Management Function	PMF. The Profile Management Function (PMF) activates and deactivates network profiles on subscriber sessions.
Property	The parameter or characteristic of an agent or device.
Protocol Interface Function	PIF. A PIF service encapsulates an interface with an external device or service.
PTA	Point-to-Point Termination Aggregation. This is a method of aggregating IP traffic by terminating PPP sessions and amassing the IP traffic into a single routing domain.
<b>Q</b>	
QoS	Quality of Service. QoS specifies a guaranteed throughput level that allows providers to guarantee to their customers that end-to-end latency will not exceed a specified level.
Quality of Service	QoS. QoS specifies a guaranteed throughput level that allows service providers to guarantee to their customers that end-to-end latency will not exceed a specified level.

Table A-1. Terms and Definitions.

Term	Definition
<b>R</b>	
RACS	Resource and Admission Control Subsystem. RACS consists of the Policy Decision Function (PDF) and Access-RAC Function (A-RACF), which controls QoS within the access network.
RADIUS	Remote Authentication Dial-In User Service. RADIUS is a client/server protocol enabling remote access server communication with a central server to authenticate dial-in users and authorize their access to the requested system or service. RADIUS allows a company to maintain user profiles in a central database and set up a policy that can be applied at a single administered network point.
Realm	A realm represents a collection of information, stored in the database, that should be transferred, as a unit, between BPM systems. The realm defines a unit for intersystem communication and improves performance by restricting lookups and updates against smaller data sets.
Remote Authentication Dial-in User Service	RADIUS. RADIUS is a client/server protocol enabling remote access server communication with a central server to authenticate dial-in users and authorize their access to the requested system or service. RADIUS allows a company to maintain user profiles in a central database and set up a policy that can be applied at a single administered network point.
Remote Method Invocation	RMI. RMI is the basis of distributed object computing in the Java environment. It defines how Java components can interoperate in a Java environment.
Resource	A resource is any device or other item that can be used. Devices such as printers and disk drives are resources. Memory is also a resource. In many operating systems, a resource is specifically data or routines that are available to programs. These are also called system resources.
Resource and Admission Control Subsystem	RACS. RACS consists of the Policy Decision Function (PDF) and Access-RAC Function (A-RACF), which controls QoS within the access network.
Resource Controller	A Resource Controller is a stateful installation that tracks resource utilization for the system.
Resource Realm	A Resource Realm represents a BRAS device and its connected CPE equipment. The Resource Realm is provisioned on the Topology Database Server and distributed to the Resource Controller that coordinates activity for that BRAs. At runtime, the Resource Realm stores capacity and usage information required to perform CAC decisions.
RMI	Remote Method Invocation. RMI is the basis of distributed object computing in the Java environment. It defines how Java components can interoperate in a Java environment.
Role	A role is as a functional category for device types and device instances. For example, <i>bras</i> and <i>dpi</i> are roles.



Table A-1. Terms and Definitions.

Term	Definition
Role-based Dependency	A dependency in which a service designer indicates that multiple service elements support the same interface. The designer defines different roles and assigns the required service interfaces to each. The different roles are added to the dependency list for the service and operators are clearly marked to indicate their assigned role.
Route	A route is a path between operators.
Rule	Criteria applied to the objects and methods of a business system to determine how objects and methods are used by, or for, a given system subscriber. A flow comprises a rule or set of rules. Rules prescribe terms and conditions for a specific action provided by an ISP for its subscribers. One rule can call another rule.
<b>S</b>	
S-VLAN	Stacked VLAN. An S-VLAN provides a two-level S-VLAN tag structure that extends the VLAN ID space to more than 16 million VLANs.
SAV	Service Admin view. In the BPS and BPDS graphical user interfaces to the BPM, this is the view where you can perform service tasks.
Schema	A set of rules and syntax for storing data.
SDV	Service Design view. In the BPS and BPDS graphical user interfaces to the BPM, this is the view where you can design services.
SE	Service Engine. SE is an unassigned and unconfigured system. It is also known as the backend.
Service	An application, created by the BPM designer, that maps business models and operational procedures directly into IP services, executable by their customers, for example, video on demand or automatic backup. A service comprises objects (agent, controller, function, switch, or other service) and can comprise one or more flows.
Service Admin View	SAV. In the BPS and BPDS graphical user interfaces to the BPM, this is the view where you can perform service tasks.
Service Configuration	The information needed to construct a service. The service configuration specifies agent configurations for each function in the service type. The BPM designer creates the service configuration.
Service Dependency	The dependencies of a service, created by the service designer. The designer builds a service by defining data-flows that use operators from multiple objects, including agents and other services. The designer builds a service upon a concrete set of agents and services.  If a service is portable across different agents and services, the designer specifies any constraints on the concrete instances and specifies the interfaces that those concrete instances must support.
Service Design View	SDV. In the BPS and BPDS graphical user interfaces to the BPM, this is the view where you can design services.
Service Engine	SE. SE is the generic term for an unassigned and unconfigured system. It is also known as the backend.
Service Interface Dependency	If a service uses a particular service interface, but does not require that a specific object provide the service interface, the service designer can add the service interface as a dependency. Here, the service interface operators are available for use in the current service, but the object that provides the interface is determined later.

**Table A-1. Terms and Definitions.**

<b>Term</b>	<b>Definition</b>
Service Instance	The running of a service type created by the subscriber.
Service Level Agreement	SLA. An SLA is a contract between an ASP and the end user that stipulates the required level of service and its fee.
Service Palette	The agent types available to a service.
Service Profile	A collection of services and information about service execution.
Service Provider	SP. This is the provider of Internet connectivity services.
Service Type	The definition of what agent types are required for a service; the defined flow of data between functions of agent types. The service designer creates the service type.
Servlet	An applet that runs on a server. Usually refers to a Java applet that runs within a Web server environment. Analogous to a Java applet that runs within a Web browser environment.
Session Management Application	SMA. Within the Session Manager, the SMA encapsulates customer-specific business logic for managing network sessions.
Session Management Function	SMF. The SMF encapsulates customer-specific business logic applied to network sessions. Abstracted from specific protocols and devices used in the network through the DAF and PIF layers, the SMF notifies applications of session state changes.
Session Manager	SM. The SM provides a framework for tracking user sessions connecting to the network.
Session Realm	A Session Realm stores Session Manager contexts and assists in the decision-making process during network adaptation.
Session Storage Function	SSF. The SSF provides access to the Session Information Model.
SF	Statistics Function. The SF records and queries system statistics and provides a location for various components to store runtime state statistics.
Shared Secret	An authentication string that ensures security between devices. KERBEROS is an instance of a shared-secret authentication protocol.
SIF	Signaling Interface Function (SIF): The SIF sends QoS requests from an application to the Director ACF. If more than one Director exists, an external Load Balancer selects a Director. The SIF receives replies from Director ACFs and forwards them to the application.
Signaling Interface Function	SIF. The SIF sends QoS requests an application to the Director ACF. If more than one Director exists, an external Load Balancer selects a Director. The SIF receives replies from Director ACFs and forwards them to the application.
Simple Object Access Protocol	SOAP. This is a lightweight XML-based messaging protocol that encodes the information in Web service request and response messages before sending them over a network. SOAP messages are independent of any operating system or protocol and may be transported using a variety of Internet protocols, including SMTP, MIME, and HTTP.
Simple Type	In the BPDS Object manager tool, a simple type is similar to data type, except it can express with enumerations.

Table A-1. Terms and Definitions.

Term	Definition
Simple Network Management Protocol	SNMP. A protocol by which networked devices are periodically polled for information as part of a network management system.
SLA	Service Level Agreement. An SLA is a contract between an ASP and the end user that stipulates a required level of service and its fee.
SM	Session Manager. The SM provides a framework for tracking user sessions connecting to the network.
SMA	Session Management Application. Within the Session Manager, the SMA encapsulates customer-specific business logic for managing network sessions.
SMF	Session Management Function. The SMF encapsulates customer-specific business logic applied to network sessions. Abstracted from specific protocols and devices used in the network through the DAF and PIF layers, the SMF notifies applications of session state changes.
SNMP	Simple Network Management Protocol. A protocol by which networked devices are periodically polled for information as part of a network management system.
SOAP	Simple Object Access Protocol. This is a lightweight XML-based messaging protocol that encodes the information in Web service request and response messages before sending them over a network. SOAP messages are independent of any operating system or protocol and may be transported using a variety of Internet protocols, including SMTP, MIME, and HTTP.
SP	Service Provider. This is the provider of Internet connectivity services.
Specification	A type definition that includes interface definitions, configuration schemas, and binding information.
SQL	Structured Query Language. SQL is a standardized query language for requesting information from a database. SQL enables several users on a local-area network to access the same database simultaneously.
SSF	Session Storage Function. The SSF provides access to the Session Information Model.
Stacked VLAN	S-VLAN. An S-VLAN provides a two-level S-VLAN tag structure that extends the VLAN ID space to more than 16 million VLANs.
Standby BPM	In a pair of BPMs, the standby BPM constantly monitors the health of the active BPM to assess its ability to process requests. If the active BPM is not viable, the standby BPM becomes the active.
Statistics Function	SF. The SF records and queries system statistics and provides a location for various components to store runtime state statistics.
Status	A status is a condition used in conjunction with network sessions.
Structured Query Language	SQL. SQL is a standardized query language for requesting information from a database. SQL enables several users on a local-area network to access the same database simultaneously.
Subscriber	A customer of a service provider. The service provider delivers a variety of online services, including e-mail, stock quotes, news, and online forums.
Subscriber Profile	A table entry containing information, such as authentication, authorization, and location on a specific subscriber.

Table A-1. Terms and Definitions.

Term	Definition
Super Operator	A reusable flow that other flows can call. To the other flows, the super operator appears as an operator that they can call and insert on any route.
Switch	A device that filters and forwards packets between LAN segments. Switches operate at the data link layer and the network layer of the OSI Reference Model.
Super User	The term denotes the highest level of user privilege. It allows unlimited access to a system. Usually, super user is the highest level of privilege for applications, as opposed to operating or network systems.
<b>T</b>	
TAF	Topology Awareness Function. The TAF extracts and reacts to changes in the underlying network. The information can be read from provisioning files or received from the TDS.
TISPAN	Telecommunications and Internet Services and Protocol for Advanced Networking. TISPAN is the ETSI core competence center for fixed networks and for migration from switched circuit networks to packet-based networks with an architecture that can serve in both. TISPAN is responsible for all aspects of standardization for present and future converged networks.
Telecommunications and Internet Services and Protocol for Advanced Networking	TISPAN. TISPAN is the ETSI core competence center for fixed networks and for migration from switched circuit networks to packet-based networks with an architecture that can serve in both. TISPAN is responsible for all aspects of standardization for present and future converged networks.
Topology Awareness Function	TAF. The TAF extracts and reacts to changes in the underlying network. The information can be read from provisioning files or received from the TDS.
Topology Database Server	In resilient pairs, Topology Database Servers maintain the global topology database for the system as a whole. The Director detects delayed response times or dropped requests and notifies the Topology Database Server. The Topology Database Server initiates Resource Controller failover when necessary.
Topology Store Function	TSF. The TSF maintains the TIM for a given BP Resource Controller system component.
Transaction Remote Procedure Call	TRPC. The TRPC protocol is the interface between Cisco BPM components.
TRPC	Transaction Remote Procedure Call. The TRPC protocol is the interface between Cisco BPM components.
TSF	Topology Store Function. The TSF maintains the TIM for a given BP Resource Controller system component.
Type	A BPM component group that has a unique specification. It may have an implementation, and it may have one or more configurations and instances.
<b>U</b>	
<b>V</b>	

Table A-1. Terms and Definitions.

Term	Definition
VC	Virtual Circuit. A connection between two devices that acts as though it's a direct connection even though it may physically be circuitous.
Virtual Circuit	VC. A VC is a connection between two devices that acts as though it's a direct connection even though it may physically be circuitous.
Virtual LAN	VLAN. A network of computers that behave as if connected to the same wire even though they can be physically located on different segments of a LAN. VLANs are configured through software rather than hardware and extremely flexible.
Virtual Path	VP. A VC is a set of links across an ATM network between two specified end points.
Virtual Private Network	VPN. A VPN is constructed using public wires to connect nodes. A number of systems exist that enable the creation of networks using the Internet as the medium for transporting data. They use security mechanisms to ensure that only authorized users can access the network and data cannot be intercepted.
VLAN	Virtual LAN. A network of computers that behave as if they are connected to the same wire even though they may be physically located on different segments of a LAN. VLANs are configured through software rather than hardware and are extremely flexible.
Voice-over-IP	VoIP. Voice delivered using the Internet Protocol.
VoIP	Voice-over-IP. Voice delivered using the Internet Protocol.
VP	Virtual Path. VP. A VP is a set of link across an ATM network between two specified end points.
VPN	Virtual Private Network. A VPN is constructed using public wires to connect nodes. A number of systems exist that enable the creation of networks using the Internet as the medium for transporting data. They use security mechanisms to ensure that only authorized users can access the network and data cannot be intercepted.
<b>W</b>	
WDSL	Wireless Digital Subscriber Line. WDSL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. It is extensible to allow description of endpoints and their messages regardless of what message formats or network protocols are used to communicate.
Wireless Digital Subscriber Line	WDSL. WDSL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. It is extensible to allow description of endpoints and their messages regardless of what message formats or network protocols are used to communicate.
Workspace	The BPDS area where the designer visually programs services.



# Appendix B - Statistics

## Overview

The system generates a variety of statistics that may be useful in analyzing performance, creating reports, or planning changes. [Table 11](#) lists statistics available for viewing.

**Table 11. Statistics**

Type	Name	Description.
HTTP agent	STAT_PERCENT_VALID	Percentage of valid requests handled by servlet since loaded.
HTTP agent	STAT_REGISTERED_URLS	Number of URLs registered with servlet.
HTTP agent	STAT_SERVLET_BUILD_DATE	Date servlet built.
HTTP agent	STAT_SERVLET_ENGINE_INFO	Describes Servlet Engine where servlet is loaded.
HTTP agent	STAT_SERVLET_RTT	Round trip time to contact servlet in milliseconds.
HTTP agent	STAT_SERVLET_STATUS	Describes whether or not agent can contact servlet.
HTTP agent	STAT_SERVLET_VERSION	Version number of servlet at build time.
HTTP agent	STAT_TOTAL_REQUESTS	Total number of requests handled by servlet since loaded.
HTTP agent	STAT_VALID_REQUEST	Number of valid requests handled by servlet since loaded.
Netconf agent	average-run-time	Average running time in milliseconds of all requests that have run; excludes wait time due to serialization lock.
Netconf agent	average-time	Average time in milliseconds of all requests that have run; including time waiting to run because of serialization lock.
Netconf agent	maximum-running- requests	Largest number of requests that ran simultaneously.
Netconf agent	maximum-run-time	Time in milliseconds of longest time for a request to run so far; excludes wait time due to serialization lock.
Netconf agent	maximum-time	Time in milliseconds of longest request so far; including time waiting to run because of serialization lock.
Netconf agent	maximum-waiting-requests	Largest number of requests waiting to run at one time.

Table 11. Statistics

Netconf agent	minimum-run-time	Time in milliseconds of shortest time for a request to run so far, excluding wait time due to a serialization lock.
Netconf agent	minimum-time	Time in milliseconds of shortest request so far, including time waiting to run because of a serialization lock.
Netconf agent	number-of-cmds-at-max-run-time	Number of CLI commands executed for request that took longest to run.
Netconf agent	number-of-cmds-at-min-run-time	Number of CLI commands executed for request that took longest to run.
Netconf agent	number-of-failures	Total number of netconf requests that have succeeded.
Netconf agent	number-of-ping-attempts	Total number of individual 'ping' attempts to devices.
Netconf agent	number-of-ping-failed-attempts	Number of individual 'ping' attempts that failed.
Netconf agent	number-of-ping-failures	Number of ping requests that failed.
Netconf agent	number-of-ping-requests	Total number of requests processed by the ping operator.
Netconf agent	number-of-ping-successes	Number of ping requests that succeeded.
Netconf agent	number-of-requests	Total number of netconf operator requests serviced.
Netconf agent	number-of-running-requests	Number of requests currently running.
Netconf agent	number-of-successes	Total number of netconf requests that have succeeded.
Netconf agent	number-of-waiting-requests	Number of requests currently waiting to run, due to a serialization lock.
RADIUS agent	client-accounting-invalid-start-responses	Number of unsuccessful Client "Start" accounting requests. Includes all requests for which no corresponding Accounting-Response was received. Only includes original requests that timed out on every [re]transmission attempt.
RADIUS agent	client-accounting-invalid-stop-responses	Number of unsuccessful Client "Stop" accounting requests. Includes all requests for which no corresponding Accounting-Response was received. Only includes original requests that timed out on every [re]transmission attempt.
RADIUS agent	client-accounting-total-requests	Total number of accounting requests made by agent's radius Client. Does not include retransmissions of original requests for which no response was received within timeoutPeriod.
RADIUS agent	client-accounting-valid-start-responses	Number of successful Client "Start" accounting requests. Includes all requests for which a corresponding Accounting-Response was received.



Table 11. Statistics

RADIUS agent	client-accounting-valid-stop-responses	Number of successful Client "Stop" accounting requests. Includes all requests for which a corresponding Accounting-Response was received.
RADIUS agent	client-authentication-invalid-responses	Number of rejected Client authentication requests. Includes all requests for which a corresponding Access-Reject or Access-Challenge was received.
RADIUS agent	client-authentication-password-errors	Number of Client authentication requests containing an invalid or malformed password. Maximum password length is 253 characters.
RADIUS agent	client-authentication-timeouts	Number of times the radius Client timed out waiting for an authentication response. Only includes original requests that timed out on every [re]transmission attempt.
RADIUS agent	client-authentication-total-requests	Total number of authentication requests made by the agent's radius Client. Does not include retransmissions of original requests for which no response was received within timeoutPeriod.
RADIUS agent	client-authentication-valid-responses	Number of accepted Client authentication requests. Includes all requests for which a corresponding Access-Accept was received.
RADIUS agent	proxy-accounting-forward-requests	Number of accounting requests successfully forwarded by the agent's radius Proxy to the external server.
RADIUS agent	proxy-accounting-forward-responses	Number of accounting responses successfully forwarded by the agent's radius Proxy to original client.
RADIUS agent	proxy-accounting-silent-discards	Number of radius messages discarded by the agent's accounting Proxy. Includes any message received on proxy accounting port that is not an Accounting-Request or contains an invalid request Authenticator or invalid value for Acct-Status-Type attribute, as well as any response message received from the external server containing an invalid Authenticator. Also includes messages that could not be either forwarded to external server or returned to original client. Additionally, it includes requests Rejected or Ignored by proxyStartAccounting and proxyStopAccounting TAZZ flows, since no response message is sent from the agent in such cases.
RADIUS agent	proxy-accounting-total-requests	Total number of radius messages received on Proxy accounting socket. Includes all messages received on socket, regardless of whether or not they are valid.
RADIUS agent	proxy-accounting-valid-start-responses	Number of successful Proxied "Start" accounting requests. Includes all Accounting-Response messages received from the external server that correspond to a "Start" request sent by the Proxy, since there is no response for an unsuccessful accounting request.

**Table 11. Statistics**

RADIUS agent	proxy-accounting-valid-stop-responses	Number of successful Proxied "Stop" accounting requests. Includes all Accounting-Response messages received from the external server that correspond to a "Stop" request sent by the Proxy, since there is no response for an unsuccessful accounting request.
RADIUS agent	proxy-authentication-forward-requests	Number of authentication requests successfully forwarded by agent's radius Proxy to external server.
RADIUS agent	proxy-authentication-forward-responses	Number of authentication responses successfully forwarded by agent's radius Proxy to original client.
RADIUS agent	proxy-authentication-invalid-responses	Number of unsuccessful Proxied authentication requests. Includes all Access-Reject messages received from the external server that correspond to a request sent by the Proxy. Proxy authentication requests that receive an Access-Challenge response are not included in this statistic.
RADIUS agent	proxy-authentication-silent-discards	Number of radius messages discarded by agent's authentication Proxy. Includes any message received on the proxy authentication port that is not an Access-Request, as well as any response message received from the external server containing an invalid Authenticator. Also includes messages that could not be either forwarded to the external server or returned to original client. Additionally, it includes requests ignored by proxyAuthenticate TAZZ flow, since no response message is sent from the agent in such cases.
RADIUS agent	proxy-authentication-total-requests	Total number of radius messages received on Proxy authentication socket. Includes all messages received on the socket, regardless of whether or not they are valid.
RADIUS agent	proxy-authentication-valid-responses	Number of successful Proxied authentication requests. Includes all Access-Accept messages received from the external server that correspond to a request sent by the Proxy.
RADIUS agent	radius-accounting-silent-discards	Number of messages received on Radius accounting socket that were ignored. Includes any message that is not an Accounting-Response, as well as any response message for which there is no pending Session.
RADIUS agent	radius-authentication-silent-discards	Number of messages received on Radius authentication socket that were ignored. Includes any message that is not an Access-Accept, Access-Reject, or Access-Challenge, as well as any response message for which there is no pending Session.

Table 11. Statistics

RADIUS agent	server-accounting-silent-discards	Number of radius messages discarded by agent's accounting Server. Includes any message received on server accounting port that is not an Accounting-Request or contains an invalid request Authenticator or invalid value for Acct-Status-Type attribute. Also includes messages that could not be returned to original client. Additionally, it includes requests Rejected or Ignored by proxyStartAccounting and proxyStopAccounting TAZZ flows, since no response message is sent from the agent in such cases.
RADIUS agent	server-accounting-total-requests	Total number of radius messages received on Server accounting socket. Includes all messages received on the socket, regardless of whether or not they are valid.
RADIUS agent	server-accounting-valid-start-requests	Number of "Start" accounting requests accepted by agent's accounting Server. Includes all valid Accounting-Request messages received by the internal Server that have an Acct-Status-Type attribute value of 1 (Start), 3 (Interim-Update), or 7 (Accounting-On).
RADIUS agent	server-accounting-valid-start-responses	Number of "Start" accounting responses sent by agent's accounting Server. This only includes valid "Start" accounting requests that were also Accepted by proxyStartAccounting TAZZ flow.
RADIUS agent	server-accounting-valid-stop-requests	Number of "Stop" accounting requests accepted by agent's accounting Server. Includes all valid Accounting-Request messages received by internal Server that have an Acct-Status-Type attribute value of 2 (Stop) or 8 (Accounting-Off).
RADIUS agent	server-accounting-valid-stop-responses	Number of "Stop" accounting responses sent by agent's accounting Server. This only includes valid "Stop" accounting requests that were also Accepted by proxyStopAccounting TAZZ flow.
RADIUS agent	tazzflow-accounting-invalid-start-responses	Number of successful "Start" accounting attempts that were then rejected by the TAZZ flow. Includes all invocations of the proxyStartAccounting flow for which the value of the responseType input to the proxyResponse operator was "Reject".
RADIUS agent	tazzflow-accounting-invalid-stop-responses	Number of successful "Stop" accounting attempts that were then rejected by the TAZZ flow. Includes all invocations of the proxyStopAccounting flow for which the value of the responseType input to the proxyResponse operator was "Reject".
RADIUS agent	tazzflow-accounting-valid-start-responses	Number of successful "Start" accounting attempts also accepted by the TAZZ flow. Includes all invocations of the proxyStartAccounting flow for which the value of the responseType input to the proxyResponse operator was "Accept" or "Ignore".
RADIUS agent	tazzflow-accounting-valid-stop-responses	Number of successful "Stop" accounting attempts also accepted by the TAZZ flow. Includes all invocations of the proxyStopAccounting flow for which the value of the responseType input to the proxyResponse operator was "Accept" or "Ignore".

Table 11. Statistics

RADIUS agent	tazzflow-authorization-invalid-responses	Number of successful authentication attempts that were then rejected by the TAZZ flow. Includes all invocations of the proxyAuthenticate flow for which the value of the responseType input to the proxyResponse operator was "Reject".
RADIUS agent	tazzflow-authorization-valid-responses	Number of successful authentication attempts also accepted by the TAZZ flow. Includes all invocations of the proxyAuthenticate flow for which the value of the responseType input to the proxyResponse operator was "Accept" or "Ignore".
SNMP agent	notification-port	Notification port currently in use.
SNMP agent	number-of-bad-messages	Number of malformed messages.
SNMP agent	number-of-unknown-notifications	Number of received notifications whose type is not well-known.
SNMP agent	total-notifications	Total number of notifications received.
Socket agent	current-connections	Current number of connections established by all clients and monitored by Agent.
Socket agent	discarded-requests	Message rejected, but the Agent couldn't send the error back to the client (possibly because the connection was closed).
Socket agent	discarded-responses	Responses that the Agent couldn't send back to the Client due to a problem (specifically, the connection being closed).
Socket agent	invalid-connection-attempts	Number of times an unauthorized client tried to connect.
Socket agent	invalid-requests	Number of messages returned because they were incorrect. May be broken up into more meaningful categories.
Socket agent	invalid-too-large-requests	Messages sent to Agent that exceed allowed limit for size and were therefore rejected.
Socket agent	pending-connections	Number of connections in the process of being accepted, pending successful authentication, authorization, etc.
Socket agent	successful-requests	Number of valid Requests that Agent processed.
Socket agent	successful-responses	Messages sent back to client from Agent.
Socket agent	total-connections	Cumulative total of all successful connections for deployment lifetime of Agent.
Socket agent	unauthenticated-client-requests	Requests sent before a client successfully authenticated and therefore rejected.
Socket agent	unauthorized-connection-attempts	Number of connections refused for lack of resources (no more sockets, threads, etc.).
Socket agent	unsupported-protocol-requests	Requests sent with a Protocol revision number other than the one Agent supports and therefore rejected.

Table 11. Statistics

Tcl agent	auth-interval-20-to-30	Number of Authentication requests received between 20 and 30 milliseconds after previous request.
Tcl agent	auth-interval-30-to-40	Number of Authentication requests received between 30 and 40 milliseconds after previous request.
Tcl agent	auth-interval-over-40	Number of Authentication requests received more than 40 milliseconds after previous request.
Tcl agent	auth-interval-under-20	Number of Authentication requests received less than 20 milliseconds after previous request.
Tcl agent	latency-avg	Average latency between when a request for the named TAZZ flow is received and when the invocation completes, in microseconds.
Tcl agent	latency-max	Longest latency between when a request for the named TAZZ flow was received and when the invocation completed, in microseconds.
Tcl agent	latency-min	Shortest latency between when a request for the named TAZZ flow was received and when the invocation completed, in microseconds.
Tcl agent	queue-max	Largest number of invocation requests ever pending in the queue for the named TAZZ flow.
Tcl agent	queue-max-accounting-start	Largest number of successful Start Accounting requests ever pending in the proxyStartAccounting queue.
Tcl agent	queue-max-accounting-stop	Largest number of successful Stop Accounting requests ever pending in the proxyStopAccounting queue.
Tcl agent	queue-max-authentication	Largest number of successful Authentication requests ever pending in the proxyAuthenticate queue.
Tcl agent	request-latency-average	Average latency between when a request is received and when its response is returned, in microseconds.
Tcl agent	request-latency-max	Shortest latency between when any request was received and when its response was returned, in microseconds.
Tcl agent	request-latency-min	Longest latency between when any request was received and when its response was returned, in microseconds.
Tcl agent	start-interval-20-to-30	Number of Start Accounting requests received between 20 and 30 milliseconds after previous request.
Tcl agent	start-interval-30-to-40	Number of Start Accounting requests received between 30 and 40 milliseconds after previous request.
Tcl agent	start-interval-over-40	Number of Start Accounting requests received more than 40 milliseconds after previous request.
Tcl agent	start-interval-under-20	Number of Start Accounting requests received less than 20 milliseconds after previous request.
Tcl agent	stop-interval-20-to-30	Number of Stop Accounting requests received between 20 and 30 milliseconds after previous request.

Table 11. Statistics

Tcl agent	stop-interval-30-to-40	Number of Stop Accounting requests received between 30 and 40 milliseconds after previous request.
Tcl agent	stop-interval-over-40	Number of Stop Accounting requests received more than 40 milliseconds after previous request.
Tcl agent	stop-interval-under-20	Number of Stop Accounting requests received less than 20 milliseconds after previous request.
other	agent-restarts	Number of agent restarts.
other	agent-starts	Number of agent starts.
other	build-date	Date/time built.
other	controller-startup	Date/time controller started.
other	dispatch-failed	Number of messages switch failed to dispatch. Does not include message-failed.
other	dispatch-queue-size	Number of messages currently being dispatched by switch.
other	free-memory	Available free memory.
other	max-memory	Maximum memory in use.
other	message-failed	Total number of messages processed unsuccessfully.
other	message-failed-percent	Percentage of messages processed unsuccessfully.
other	message-pending	Number of messages currently processing.
other	message-total	Total number of messages processed.
other	num-agents	Number of active agents.
other	num-logged-on	Number of users logged on.
other	object-creation-date	Time at which object's constructor was called.
other	object-name	Name.
other	object-version	Version.
other	percent-valid	The percentage of valid requests handled by the servlet since it was loaded.
other	pid	Process id.
other	reference-count	Incubus component reference count.
other	registered-urls	Number of URLs registered with servlet.
other	repository-name	Name of object in engine repository.
other	servlet-engine-info	Describes the Servlet Engine where servlet is loaded.
other	servlet-roundtrip-time	Round trip time to contact servlet, in milliseconds.
other	servlet-status	Describes whether or not the agent is able to contact the servlet.
other	switch-restarts	Number of switch restarts.
other	switch-startup	Date/time switch started.
other	total-memory	Total memory in use.
other	total-requests	Total number of requests handled by the servlet since it was loaded.

**Table 11. Statistics**

other	valid-requests	Number of valid requests handled by the servlet since it was loaded.
other	version	Code revision.





# Index

## A

- abbreviations 113
- accounting
  - log 6
- Accounting Log Function 6
- ACF 5
- acronyms 113
- active service engine 97, 99, 101, 102, 103
- Admission Control Function 5
- agent configuration 21, 94
- agent devices 96
- agent properties 95
- agent statistics 25
- agent status 108
- ALF 6
- application database 17
- Author 23, 25
- automatic failover 97, 103

## B

- backup 59
- bar
  - Status 40
- BPDS client 15
- BPS
  - uses 13

## C

- CAC 2
- change connection to database 20
- change password 56
- cluster 97, 99, 100, 101, 102
  - dissolving 101
- clustering 97
- command line interface 2
  - tash 62
- configuration 119
  - agent 21, 94
- configuring standby engine 91
- Context Timer Function 6
- controller
  - domain 17
- conventions
  - iconic xi
  - text ix
  - used in guide ix
- Copy configuration 59
- creating roles 43

- creating users 49
- CTF 6

## D

- DAF 8, 10
- database
  - application 17
  - change connection 20
  - refresh 20
  - remove from replication 20, 99
- databases 20, 108
- definitions 95, 113
- deleting role 48
- deleting user 57
- deny permission 44, 52, 55
- dependencies 23
- dependencies (rules) 25
- dependents (rules) 25
- deployed 112
- deploying service 87
- Device Adapter Function 8, 10
- Device Handler Dispatcher Service 8, 10
- DHDS 8, 10
- Director 11, 12, 84
- dissolving cluster 101
- domain controller 17
- domains 42

## E

- e-mail address 50
- engine
  - standby 91
- errors 38, 110
- ETSI 6
- export 59
- exporting RBAC data 57

## F

- failover 102, 103, 107
  - automatic 97, 103
  - manual 97, 101, 102
- fatal 38

## G

- glossary 113
- grant permission 44, 52, 55
- guide
  - organization of xii

- I
  - implementation 119
  - import 59
  - importing RBAC data 58
  - Index 141
  - ISP 119
- L
  - log file 38
  - Log in screen 17, 18
  - log messages 38
  - Log pane 38, 110
  - logging levels 39, 92
  - login
    - system 17, 59, 78, 105
- M
  - manual failover 97, 101, 102
  - messages 38
  - metadata 59
  - MPLS 120
  - Multiprotocol Label Switching (MPLS) 120
- N
  - name 23, 25
  - name, role 45
  - name, user 50
  - network 120
  - network BPM host 107
  - network BPM port 107
  - network BPM URL 107
  - Network Manager 9
  - Network Storage Function 10
  - new role 43
  - new role window 43
  - new user 49
  - new user window 49
  - NM 9
  - NSF 10
- O
  - object 121
- P
  - pane
    - Log 38, 110
  - parameters 21, 23
  - parameters (policy function) 23
  - parameters (rule) 25
  - password 17
  - password, change 56
  - patch 96
  - Patching 96
  - Path Computation Function 5
  - PCF 4, 5
  - permission, deny 44, 52, 55
  - permission, grant 44, 52, 55
  - permissions 47
  - PIF 8, 9
  - PMF 10
  - PMI 10
  - policy function 21, 22
  - Presence Director 11
  - problems 110
  - procedures
    - logging into the BPM system 17
  - Profile Management Function 10
  - Profile Management Interface 10
  - Protocol Interface Function 8, 9
  - protocols
    - MPLS 120
    - Multiprotocol Label Switching (MPLS) 120
    - RADIUS 113, 124
    - Remote Access Dial-In Service (RADIUS) 113, 124
    - Remote Method Invocation (RMI) 124
    - RMI 124
    - Simple Network Management Protocol (SNMP) 127
    - SNMP 127
    - TRPC 128
- R
  - RADIUS 113, 124
    - agent 113
  - RBAC 41
  - RBAC Administration 42
  - RBAC data, exporting 57
  - RBAC data, importing 58
  - RBAC, starting 42
  - refresh 20
  - refresh database 20
  - Remote Access Dial-In Service (RADIUS) 113, 124
  - Remote Authentication Dial-In User Service (RADIUS) 124
  - Remote Method Invocation (RMI) 124
  - remove database from replication 20, 99
  - renaming 93
  - replication
    - remove database 20, 99
  - Resource Controller 11, 12, 83
  - restore 59
  - RMI 124
  - role assignments 51, 54
  - role name 45
  - role, deleting 48
  - role, new 43

- role-based access control 41
- Roles 43
- roles 42, 43
- roles, creating 43
- Rule folder 24
- rules 23, 24
- running 111
- S
- SAF 6
- screen
  - Log in 17, 18
- script 23, 25
- service
  - configuration 126
  - deploying 87
  - type 126
  - undeploying 89
- service engine 107
  - active 97, 99, 101, 102, 103
  - standby 97, 98, 99, 101, 102, 103
- Session Awareness Function 6
- Session Management Function 9
- Session Manager 8
- Session Storage Function 8
- Session Store Function 6
- severity 39
- SF 7
- shutting down BPM 14
- SIF 5
- Signaling Interface Function 5
- Simple Network Management Protocol (SNMP) 127
- SM 8
- SMF 9
- SNMP 127
- source rule 21, 23
- source script 21, 23
- SSF 6, 8
- standby 107
- standby engine
  - configuring 91
- standby service engine 91, 97, 98, 99, 101, 102, 103
- starting BPM 14
- statistics 26
- Statistics Function 7
- status 19
- Status bar 40
- stopping BPM 14
- subnet mask 94
- symptom 110
- system
  - login 17, 59, 78, 105
- T
- TAF 4
- tash 62
  - command line interface 62
- TCP port 17
- terms 113
- text conventions ix
- TIM 4
- TISPAN 6
- Topology Awareness Function 4
- topology database 11, 12
- Topology Information Model 4
- Topology Store Function 4
- Transaction Remote Procedure Call (TRPC) 128
- troubleshooting 110, 111
- TRPC (Transaction Remote Procedure Call) 128
- TSF 4
- U
- uncluster 101
- undeploying service 89
- user assignments 46
- user name 17, 50
- user, deleting 57
- user, new 49
- users 42, 48, 49
- users, creating 49

