SYBASE®

Installation and Configuration Guide

Sybase RAP - The Trading Edition™

DOCUMENT ID: DC00793-01-0100-04

LAST REVISED: September 2008

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Contents

About This Book

Audience

Sybase RAP - The Trading Edition Installation and Configuration Guide is intended for Sybase® Professional Services and other technical personnel who need to install and configure the components of Sybase RAP. Familiarity with Sybase Adaptive Server® Enterprise, Sybase IQ, data warehousing, and other related topics is assumed.

How to use this book

Before following the instructions in this book to install and configure Sybase RAP, refer to the *Sybase RAP - The Trading Edition Release Bulletin* for any last minute information regarding this product.

Related documents

Refer to the following documents for more information:

- *Sybase RAP - The Trading Edition Release Bulletin*
- *Sybase RAP - The Trading Edition Users Guide*
- *Sybase RAP - The Trading Edition Operations Console Users Guide*
- *Sybase RAP - The Trading Edition Developers Guide*
- Sybase IQ 12.7 product documentation
- Adaptive Server® Enterprise 15.0 product documentation
- OpenSwitch™ 15.1 product documentation
- PowerDesigner® 12.5 product documentation
- Open Client™ 15.0 product documentation
- White paper titled Using Sybase NonStopIQ and EMC CLARiiON for Backup/Restore, High Availability, and Disaster Recovery at http://www.sybase.com/detail?id=1054761
- White paper titled Time Series in finance: the array database approach at http://cs.nyu.edu/shasha/papers/jagtalk.html
- White paper titled FinTime --- a financial time series benchmark at http://www.cs.nyu.edu/cs/faculty/shasha/fintime.html

---

Note This product includes software developed by The Apache Software Foundation at http://www.apache.org/.

Other sources of information

Use the Sybase Getting Started CD, the Sybase Infocenter Web site, and the Sybase Product Manuals Web site to learn more about your product:

- The Getting Started CD contains the release bulletin, installation and configuration guide, developers guide, and users guides in PDF format. It is included with your software. To read or print documents on the Getting Started CD, you need Adobe Acrobat Reader, which you can download at no charge from the Adobe Web site using a link provided on the CD.
- The Sybase Infocenter Web site is an online version of the product manuals that you can access using a standard Web browser.

  To access the Infocenter Web site, go to Sybooks Online Help at http://infocenter.sybase.com/help/index.jsp
- The Sybase Product Manuals Web site is an online version of the SyBooks CD that you can access using a standard Web browser. In addition to product manuals, you will find links to EBFs/Maintenance, Technical Documents, Case Management, Solved Cases, newsgroups, and the Sybase Developer Network.

  To access the Sybase Product Manuals Web site, go to Product Manuals at http://www.sybase.com/support/manuals/.

Sybase certifications on the Web

Technical documentation at the Sybase Web site is updated frequently.

❖ **Finding the latest information on product certifications**

1. Point your Web browser to Technical Documents at http://www.sybase.com/support/techdocs/.
2. Click Certification Report.
3. In the Certification Report filter select a product, platform, and timeframe and then click Go.
4. Click a Certification Report title to display the report.

❖ **Finding the latest information on component certifications**

1. Point your Web browser to Availability and Certification Reports at http://certification.sybase.com/.
2. Either select the product family and product under Search by Base Product; or select the platform and product under Search by Platform.
3. Select Search to display the availability and certification report for the selection.

❖ **Creating a personalized view of the Sybase Web site (including support pages)**

Set up a MySybase profile. MySybase is a free service that allows you to create a personalized view of Sybase Web pages.

1 Point your Web browser to Technical Documents at http://www.sybase.com/support/techdocs/.

2 Click MySybase and create a MySybase profile.

Sybase EBFs and software maintenance

❖ **Finding the latest information on EBFs and software maintenance**

1 Point your Web browser to the Sybase Support Page at http://www.sybase.com/support.

2 Select EBFs/Maintenance. If prompted, enter your MySybase user name and password.

3 Select a product.

4 Specify a time frame and click Go. A list of EBF/Maintenance releases is displayed.

Padlock icons indicate that you do not have download authorization for certain EBF/Maintenance releases because you are not registered as a Technical Support Contact. If you have not registered, but have valid information provided by your Sybase representative or through your support contract, click Edit Roles to add the "Technical Support Contact" role to your MySybase profile.

5 Click the Info icon to display the EBF/Maintenance report, or click the product description to download the software.

Accessibility features

This document is available in an HTML version that is specialized for accessibility. You can navigate the HTML with an adaptive technology such as a screen reader, or view it with a screen enlarger.

Sybase RAP documentation complies with U.S. government Section 508 Accessibility requirements. Documents that comply with Section 508 generally also meet non-U.S. accessibility guidelines, such as the World Wide Web Consortium (W3C) guidelines for Web sites.

For information about accessibility support in the Sybase IQ plug-in for Sybase Central, see “Using accessibility features” in Chapter 1, “Introducing Sybase IQ” in *Introduction to Sybase IQ*. The online help for Sybase IQ, which you can navigate using a screen reader, also describes accessibility features, including Sybase Central keyboard shortcuts.

Note You might need to configure your accessibility tool for optimal use. Some screen readers pronounce text based on its case; for example, they pronounce ALL UPPERCASE TEXT as initials, and MixedCase Text as words. You might find it helpful to configure your tool to announce syntax conventions. Consult the documentation for your tool.

For information about how Sybase supports accessibility, see Sybase Accessibility at http://www.sybase.com/accessibility. The Sybase Accessibility site includes links to information on Section 508 and W3C standards.

If you need help

Each Sybase installation that has purchased a support contract has one or more designated people who are authorized to contact Sybase Technical Support. If you cannot resolve a problem using the manuals or online help, please have the designated person contact Sybase Technical Support or the Sybase subsidiary in your area.

Note If you need to contact Sybase regarding this product, use the internal version Sybase RAP 1.0 to identify this release.

CHAPTER 1

Sybase RAP - The Trading Edition

About this Chapter

This chapter describes Sybase RAP - The Trading Edition and provides some information that can help you plan your installation.

Contents

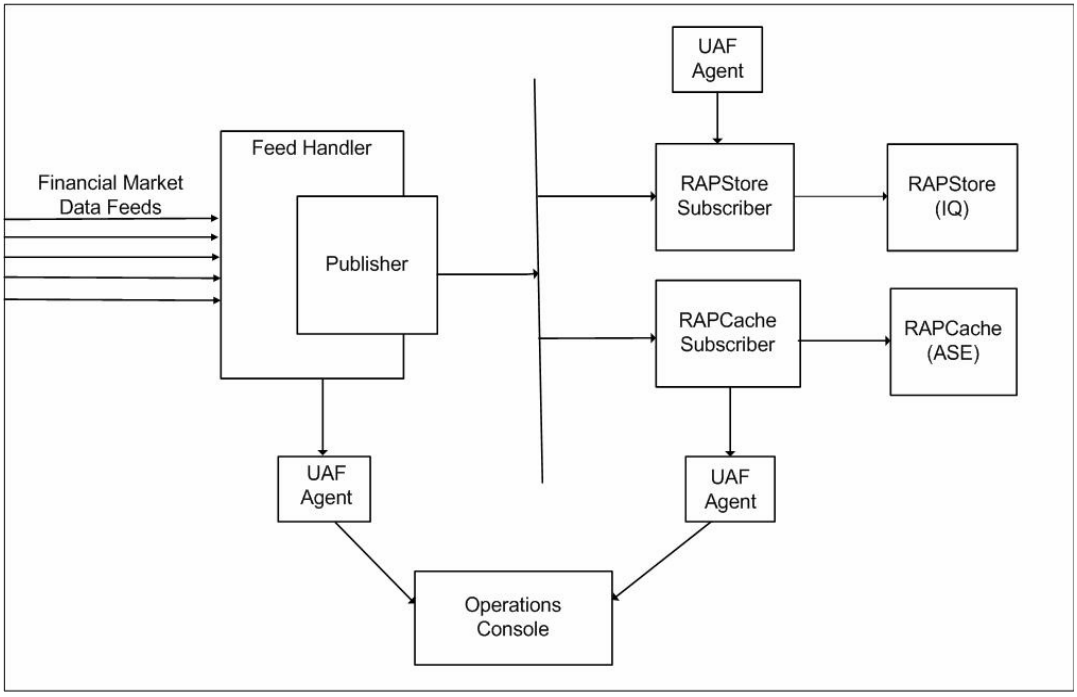
Overview

Sybase RAP - The Trading Edition (Sybase RAP) is a consolidated trading and risk data repository and data services platform for customers in the capital markets and investment management sectors.

Sybase RAP consolidates market data from vendor feeds, historical time series data, real-time trades and quotes (TAQ) data, and reference data in one repository, which eliminates or reduces intraday and overnight batch processing and supports model-driven quantitative trading and real-time portfolio decisions by presenting subsets of data to various applications.

Platform Architecture

Sybase RAP supports an architecture that is designed to intercept real-time market data feeds and publish the messages to subscribers that simultaneously load the data into repositories, which provide immediate data access and persistent data storage. This architecture includes Feed Handlers, subscribers, and data stores.



Feed Handlers

Feed Handlers are publishers that manage data that arrives directly from an exchange, like the NYSE, or a consolidated service provider like Reuters, and then send the data (using the Sybase RAP Publisher API) over the network to one or more subscribers.

Publisher API

Sybase RAP includes a Publisher API library which is used by feed handlers to format incoming market data and send the data to subscribers using UDP multicasting (for information on UDP multicasting, refer to the section entitled “What is UDP multicasting?” on page 6). Intended as a tool for developers, the Publisher API can be used to develop custom feed handlers.

FAST Feed Handler

FIX Adapted for Streaming (FAST) protocol is a binary encoding method for message oriented data streams. RAP includes a FAST Feed handler that reads FAST encoded messages directly from the wire, decodes the messages, and forwards the messages to the Publisher API.

Demo Feed Handler

Sybase RAP includes a Demo Feed Handler that is intended to help developers become familiar with the Publisher API and demonstrate network message flow. Demo Feed Handler reads comma delimited data from a demo file and transmits the data to subscribers using the Publisher API. The Demo Feed Handler is not required in a production environment.

Subscribers

Subscribers are configured to listen for message deliveries from one or more publishers. When a publisher sends a message, the subscribers intercept the message and load it into the appropriate data store.

There are two primary subscribers: the RAPCache Subscriber and the RAPStore subscriber. Because Sybase RAP is intended to provide immediate data access and persistent data storage, both subscribers should listen for message deliveries from the same publishers (though this is not a must). This ensures that the same data is loaded into the RAPCache database and the RAPStore database at the same time.

In addition to the two primary subscribers, there is also a Demo Subscriber.

RAPCache Subscriber

The RAPCache Subscriber receives messages from publishers and then loads them into the RAPCache database.

RAPStore Subscriber

The RAPStore Subscriber receives messages from publishers and then loads them into the RAPStore database.

Demo Subscriber

Demo Subscriber receives messages from publishers and either writes the message to a file or discards them (depending on configuration). Demo Subscriber is designed to demonstrate (in conjunction with the Demo Feed Handler) how messages flow through Sybase RAP. The Demo Subscriber is not required in a production environment.

Data Stores

RAP data stores include an in-memory cache database that provides immediate data access, and a historical repository that can manage massive historical and corporate reference data sets.

RAPCache database

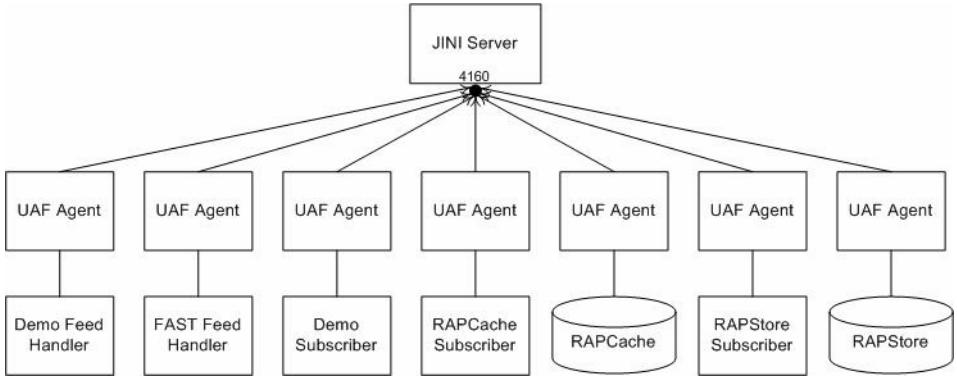
The RAPCache database is an ASE 15.0.2 database server, which provides near real-time access to inbound market data messages. Using this data store for current market data provides a separate memory space for the specialized use of traders who need up-to-the-second information, but may not need the full historical data held in the RAPStore database.

RAPStore Database

The RAPStore database is a Sybase IQ disk-based repository capable of storing massive amounts of historical data. The storage capacity of this repository can reach into the petabyte range, and is capable of scaling to a high number of concurrent user connections.

Operations Console

Operations Console is a Web-based console that lets you monitor system activities and perform routine administration tasks. Installing Operations Console also installs a JINI server that communicates with the Unified Agent Framework (UAF) agents that are installed with each component. The JINI server and UAF agents enable administration of components that are installed on different machines over the network. For more information about UAF agents and JINI, see the section entitled “Operations Console – RAP Components Communication” on page 10.



Network communications

User Datagram Protocol (UDP) provides the mechanism that lets publishers and subscribers exchange packets of data, called datagrams. A datagram is an independent, self-contained message sent over the network whose arrival, arrival time, and content are not guaranteed.

Although UDP does not guarantee delivery, RAP includes a mechanism to ensure that all messages reach their intended target. Publishers temporarily cache all message packets and subscribers track all messages deliveries. If a message is lost, a subscriber requests a new copy, and the publisher resends the message from the pool of cached messages.

What is UDP multicasting?

User Datagram Protocol (UDP) is one of the two main transport protocols utilized in IP networks. It is used by applications to transmit messages (also known as datagrams or packets) between networked hosts. UDP has better transmission rates compared to TCP. This is because UDP is a “best effort” protocol and does not implement some of the features offered by TCP. Unlike TCP, UDP does not guarantee reliability or ordering of packets. Avoiding the overhead of checking whether every packet arrives makes UDP faster and more efficient.

Multicasting is a technique developed to send packets from one host in the network to many other hosts. It differs from UDP broadcasting. With broadcasting, packets are sent to every possible receiver, while multicast packets can be sent to selected receivers. In multicasting, one packet is sent from a source and is replicated in the network to reach as many end-hosts as necessary. Multicasting is useful because it conserves bandwidth, which in many cases is the most expensive part of network operations. It does this by replicating packets as needed within the network, thereby not transmitting unnecessary packets. Multicasting uses UDP as its underlying transport protocol. Hence there is no acknowledgement of the success or failure of the transmission of any packet, and no retransmission, at the transport layer.

Multicast is based on the concept of a group. Every multicast requires a multicast group. An arbitrary group of receivers expresses an interest in receiving a particular data stream containing messages. Processes that are interested in receiving data flowing to a particular group must join the multicast group. The sender transmits to the group, and only members of the group can receive the multicast data. Multicast addresses are used to define the multicast group. The Internet Assigned Numbers Authority (IANA) controls the assignment of IP multicast addresses. It has assigned the Class D address space to be used for IP multicast. All IP multicast group addresses will fall in the range of 224.0.0.0 to 239.255.255.255.

In Sybase RAP, the publishers send messages over the network through the datastream channels using UDP multicasting. Each datastream channel can be configured to have a multicast address, an IP address of a local network interface and a local port (for more information, see the section entitled “What is a local interface?”). The subscribers can be configured to receive messages from one or more publishers by listening for messages on all of the publishers’ datastream channels. In order to ensure maximum performance, the publishers and subscribers should be set up on the same subnet. The multicast IP addresses reserved for “ad-hoc” usage fall in the range of 224.0.2.0 to 224.0.255.0. It is recommended that this range be used for the datastream multicast addresses. Since UDP is a “best effort” protocol, message reliability has been built into Sybase RAP by adding mechanisms in the subscribers for detecting transmission of lost messages over the network and requesting each lost message from the publisher over a TCP connection.

What is a Local Interface?

Every network interface card (NIC) on your computer has an IP address associated with it. This address is assigned when the computer is configured, or after the card is installed, by means of the utility ifconfig.

Typically, a computer will have just one network interface, and its IP address becomes the IP address of the host computer. If a second network interface card is added, its IP address must be assigned by a network administrator, and will be different than the host IP address.

Sybase RAP uses IPv4 multicasting, and requires IP addresses configured for the network interface to be 32-bit IPv4 addresses.

When using UDP multicasting, every multicast group must be associated, or bound to, a network interface, using the IP address of the interface. Sybase RAP documentation refers to this as the local interface. For systems with a single network interface, multicast groups will always be associated with the host IP address. In cases where more than one network interface exists, the multicast group may be associated with the IP address of the second interface. This is often the case when a network interface is dedicated to a subnet for very high speed linkage between hosts within the subnet.

When considering which local interface to use for configuring Sybase RAP systems, and there is more than one network interface card installed on your computer, it is recommended that you consult with your network administrator to determine which interface to use for Sybase RAP. Sybase RAP requires the use of an interface capable of transmitting at a minimum rate of 1 gigabit per second.

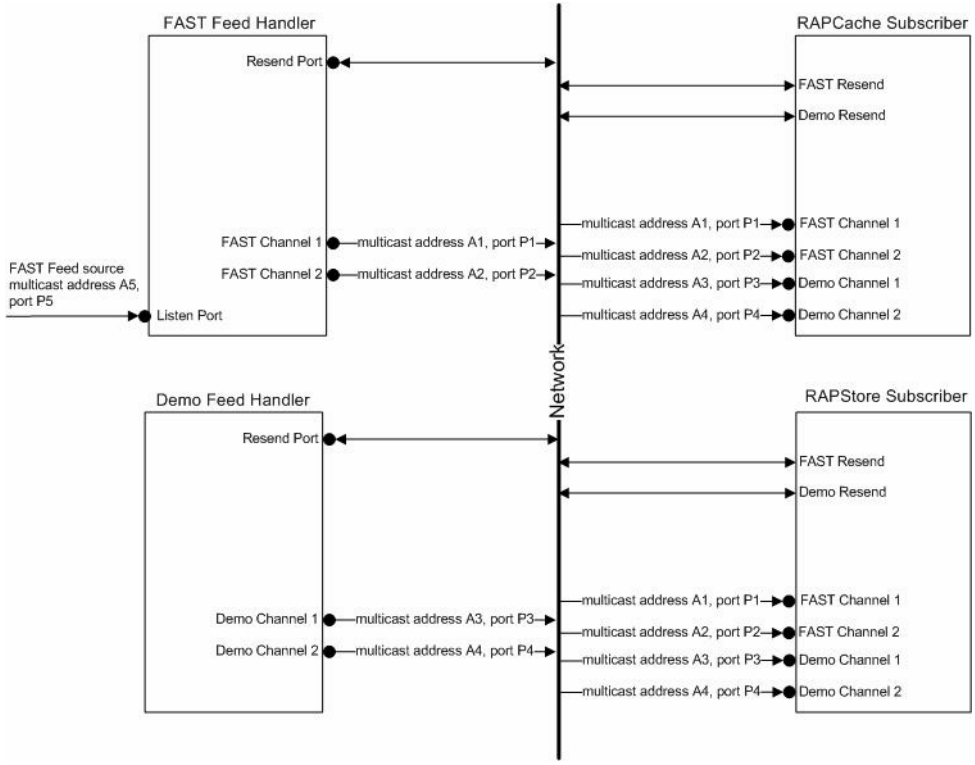
A demonstration installation of Sybase RAP that runs on a single machine may use the loopback interface as the local interface. The loopback interface (127.0.0.1 for IPv4) is not a physical device but a piece of software simulating a network interface. Any traffic that a computer program sends on the loopback interface is addressed to the same computer.

Publisher-Subscriber communication

Publishers send messages over the network through the datastream channels using UDP multicasting. Each datastream channel can be configured to have a multicast address, an IP address of a local network interface and a local port. The subscribers can be configured to receive messages from one or more publishers by listening for messages on the publishers' datastream channels. Before you install Sybase RAP, define the datastream channel ports, multicast IP addresses and local interface addresses that let components send and receive messages. Worksheets have been provided further on in this chapter to assist you with installation planning.

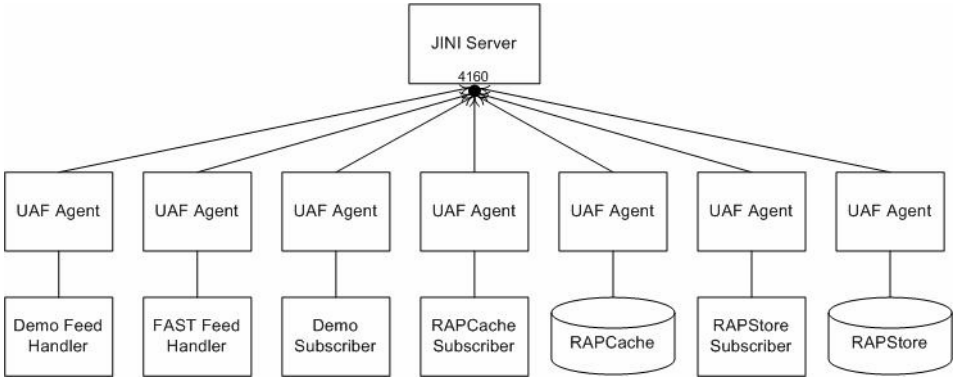
When a publisher sends a message to a multicast group, all subscribing recipients to that host and port receive the message. If a subscriber misses a message published by one of the publishers, the subscriber requests a resend of the message by communicating with the publisher through its resend port. You must define a unique resend port for each publisher.

The figure below shows the FAST Feed Handler and Demo Feed Handler configured with two datastream channels each. The RAPCache subscriber and RAPStore subscriber are configured to listen to both publishers and therefore each subscriber is listening on four datastream channels. The multicast addresses and port configurations of the subscribers' channels should be identical to the multicast addresses and port configurations of the publishers' data channels. The local interface address of the publisher/subscriber datastream channels should be the NIC address of the publisher/subscriber servers respectively. The resend port configurations of the subscribers must match the resend port configurations of the respective publishers. The resend IP addresses of the subscribers should be identical to the NIC addresses of the respective publisher servers.

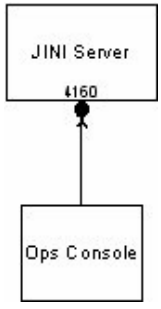


Operations Console – RAP Components Communication

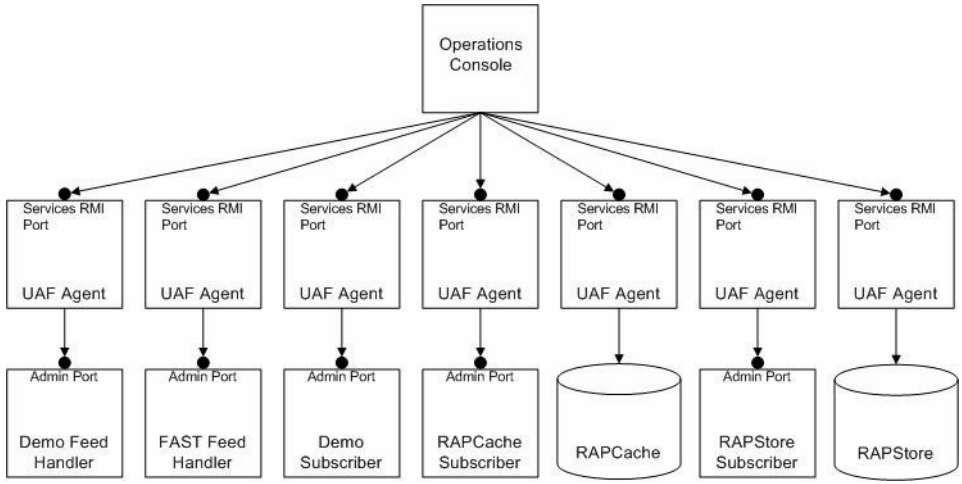
The installation of each Sybase RAP component (publisher, subscriber or data store) includes a UAF agent. The UAF agent handles communication between Operations Console and a Sybase RAP component. When a UAF agent starts, it registers with the JINI server as illustrated in the following figure.



The Operations Console queries the JINI server to discover where on the network agents are running and which RAP component the agent manages as illustrated in the following figure.



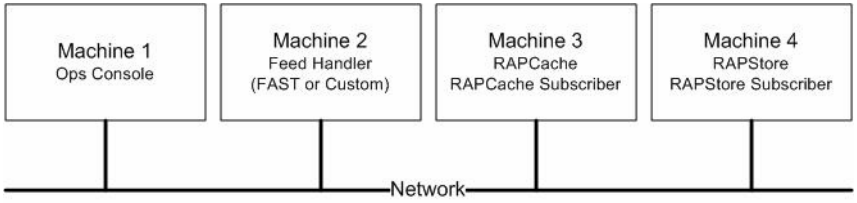
The Operations Console issues commands to UAF agents through the services RMI ports or the listening ports. You must provide a unique 'Listening Port' for each publisher, subscriber, RAPCache or RAPStore you install as illustrated in the following figure.



Each publisher and subscriber has an administration channel listening on a specific port. When the Operations Console issues commands to UAF agents through the services RMI ports, the UAF agent accepts the request and contacts its Sybase RAP component through the administration listening port. The Sybase RAP component sends a response through the UAF agent back to Operations Console. You must identify an Administration Listening Port for each publisher and subscriber. Each port number must be unique.

Deploying Sybase RAP in a production environment

Sybase recommends that you install components on separate servers. A typical production environment requires four servers.



Use the following table to help plan your installation requirements.

Server	Component	Minimum disk space	RAM	Comments
Machine 1	Operations Console	100MB	4GB	Includes the JINI server.
Machine 2	Feed Handlers	132MB	8GB	Actual space requirements depend the feed handlers you install. Feed handlers can reside on multiple machines.
Machine 3	RAPCache System	2GB	32GB	This represents the minimum requirements for the RAPCache Database Server and RAPCache Subscriber. You need additional disk space for the database.
Machine 4	RAPStore System	2GB	32GB	This represents only the minimum requirements for the RAPStore Database Server and RAPStore Subscriber. You need additional disk space for the database.

What installation type is right for you

Sybase RAP includes custom, express, and demonstration installation types. Use the following table to help decide what type of install is right for you:

Installation type	Description
Custom	The custom installation is designed for production environments. This installation type provides the most flexibility in configuration, but also requires the most planning. It is intended for users that want complete control over configuration of the system (ports, IP addresses, tuning parameters, etc.). To plan your custom installation, see the section entitled "Custom installation worksheets" on page 13.
Express	The express installation is designed for production environments. This installation type is intended for systems where default values for ports and tuning parameters are acceptable. To plan your express installation, see the section entitled "Express installation worksheets" on page 19.
Demo	The demonstration installation is designed to install a basic functional system on a single machine. This installation type requires no planning and is intended for demonstration systems. This installation type is not intended for production environments.

Using the worksheets to plan your installation

Decide what kind of installation you want to perform, then use these worksheets to plan the layout of your system. This section includes Custom installation worksheets and Express installation worksheets. Demo installations are non-interactive and do not require preplanning worksheets.

Custom installation worksheets

1 Decide which components you need to install. The following components are optional for a production system:

- Publisher API: Used by developers to write feed handlers
- Demo Feed Handler and Demo Subscribers: components used only for demonstrating the system
- FAST Feed Handler: required only if you intend to receive market data messages from a FAST encoded feed
- Samples: sample queries and sample data (this component does not require any configuration and so it does not appear on this worksheet)

2 Use this worksheet to plan the layout of your system. This worksheet lists the information for which the installer will prompt.

Descriptions of each of the fields are provided in “Custom Setup Options” on page 91. Recommended values are provided based on a system that handles 300,000 Stock Quote messages per second.

Note The installer installs components in a different order than is listed here.

Demo Feed Handler

Field	Your Value	Recommendation
Listening Port	1.	3001
Component Name (optional)	2.	
Channel Name	3.	Demo Channel 1
Local Interface	4.	127.0.0.1 for single-machine installation
Broadcast IP Address	5.	224.0.2.0
Broadcast Port	6.	3050
If the Demo Feed Handler will send data on multiple channels, record entries 3 through 6 for each channel. Local Interface and Broadcast IP Address can remain the same for each channel. Increase Broadcast Port by 1 for each channel. Four channels are recommended for a system that handles 300,000 messages per second.		
Administration Listening Port	7.	3002
Resend Listening Port	8.	3003
Number of packet buffers	9.	10000
JINI Server Host Name	10.	

Fast Feed Handler (optional)

Field	Your Value	Recommendation
Listening Port	11.	3101
Component Name (optional)	12.	
Listen Host IP Address (for FAST Feed)	13.	127.0.0.1 for single-machine installation
Listen Port (for FAST Feed)	14.	3170

Field	Your Value	Recommendation
Multicast Host IP Address	15.	224.0.255.0
Channel Name	16.	FAST Channel 1
Local Interface	17.	127.0.0.1 for single-machine installation
Broadcast IP Address	Enter the value from line 5.	
Broadcast Port	18.	3150
If the FAST Feed Handler will send data on multiple channels, record Channel Name, Local Interface, Broadcast IP Address and Broadcast Port for each channel. Local Interface and Broadcast IP Address can remain the same for each channel. Increase Broadcast Port by 1 for each channel. Four channels are recommended for a system that handles 300,000 messages per second.		
Administration Listening Port	19.	3102
Resend Listening Port	20.	3103
Number of packet buffers	21.	10000
JINI Server Host Name	Enter the value from line 10.	

Publisher API (optional)

Field	Your Value	Recommendation
Listening Port	22.	3201
Component Name (optional)	23.	
Channel Name	24.	Custom Channel 1
Local Interface	25.	127.0.0.1 for single-machine installation
Broadcast IP Address	Enter the value from line 5.	
Broadcast Port	26.	3250
If the feed handler that is being developed will send data on multiple channels, record Channel Name, Local Interface, Broadcast IP Address and Broadcast Port for each channel. Local Interface and Broadcast IP Address can remain the same for each channel. Increase Broadcast Port by 1 for each channel. Four channels are recommended for a system that handles 300,000 messages per second.		
Administration Listening Port	27.	3202
Resend Listening Port	28.	3203
Number of packet buffers	29.	10000
Number of message buffers	30.	Number of threads simultaneously sending messages in your feed handler.
JINI Server Host Name	Enter the value from line 10.	

RAPCache Database Server

Prompt	Your Value	Recommendation
Listening Port	31.	3301
RAPCache Server Name	32.	RAPCache
RAPCache Port	33.	5000
Server Page Size	34.	8k
Master Device Size (MB)	35.	500
Master Database Size (MB)	36.	100
RAPCache Backup Port	37.	5001
JINI Server Host Name	Enter the value from line 10.	

RAPCache Subscriber

Field	Your Value	Recommendation
Listening Port	38.	3401
Component Name (optional)	39.	
Host Name (for RAPCache)	40.	The installer will fill in this value for you.
Port (for RAPCache)	41.	The installer will fill in this value for you.
Channel Name	Enter the Channel Name, Broadcast IP Address, and Broadcast Port for each channel you entered for the Demo Feed Handler. Enter Demo Feed Handler as the Publisher Name. Enter the Channel Name, Broadcast IP Address, and Broadcast Port for each channel you entered for the FAST Feed Handler. Enter FAST Feed Handler as the Publisher Name. Enter the Channel Name, Broadcast IP Address, and Broadcast Port for each channel you entered for the Publisher API. Enter Custom Feed Handler as the Publisher Name.	
Publisher Name		
Local Interface		
Broadcast IP Address		
Broadcast Port		
Administration Listening Port	42.	3402
Number of Packet Buffers	43.	5000
Number of Packets Tracked	44.	5000
Publisher Name	45.	Demo Feed Handler, FAST Feed Handler, or Custom Feed Handler
IP Address	46.	

Field	Your Value	Recommendation
Resend Listener Port	Enter the value from line 8, 21 or 30, depending on which publisher you are registering.	
If the RAPCache Subscriber will receive data from multiple publishers, record Publisher Name, IP Address and Resend Listener Port for each publisher.		
Location for RAPCache Installation	This field does not appear if the RAPCache and RAPCache Subscriber are installed at the same time.	
JINI Server Host Name	Enter the value from line 10.	

RAPStore Database Server

Field	Your Value	Recommendation
Listening Port	47.	3501
Server Name	48.	RAPStore
Port	49.	2638
JINI Server Host Name	Enter the value from line 10.	

RAPStore Subscriber

Field	Your Value	Recommendation
Listening Port	50.	3601
Component Name (optional)	51.	
Host Name (for RAPCache)	52.	The installer will fill in this value for you.
Port (for RAPCache)	53.	The installer will fill in this value for you.
Number of File Buffers	Enter the value 3.	
Primary File Location	54.	
Overflow File Location	55.	
Channel Name	Enter the same information you entered for the RAPCache Subscriber.	
Publisher Name		
Local Interface		
Broadcast IP Address		
Broadcast Port		
Administration Listening Port	56.	3602
Number of Packet Buffers	Enter the value from line 43.	
Number of Packets Tracked	Enter the value from line 44.	

Field	Your Value	Recommendation
Publisher Name	Enter the same information you entered for the RAPCache Subscriber.	
IP Address		
Resend Listener Port		
Location for RAPStore Installation	This field does not appear if the RAPCache and RAPCache Subscriber are installed at the same time.	
JINI Server Host Name	Enter the value from line 10.	

Demo Subscriber (optional)

Field	Your Value	Recommendation
Listening Port	57.	3701
Component Name (optional)	58.	
Channel Name	Enter the same information you entered for the RAPCache Subscriber.	
Publisher Name		
Local Interface		
Broadcast IP Address		
Broadcast Port		
Administration Listening Port	59.	3702
Number of Packet Buffers	Enter the value from line 43.	
Number of Packets Tracked	Enter the value from line 44.	
Publisher Name	Enter the same information you entered for the RAPCache Subscriber	
IP Address		
Resend Listener Port		
JINI Server Host Name	Enter the value from line 10.	

Operations Console

Field	Your Value	Recommendation
RMI Port	60.	9092
Class Server Port	61.	9095
Host Name (for RAPCache)	Enter the value from line 40.	
Port (for RAPCache)	Enter the value from line 41.	
Host Name (for RAPStore)	Enter the value from line 52.	

Field	Your Value	Recommendation
Port (for RAPStore)	Enter the value from line 53.	
HTTP Port	62.	8080
Shutdown Port	63.	8005
AJP Port	64.	8009

Express installation worksheets

1 Decide which components you need to install. The following components are optional for a production system:

- Publisher API: Used by developers to write feed handlers
- Demo Feed Handler and Demo Subscribers: components used only for demonstrating the system
- FAST Feed Handler: required only if you intend to receive market data messages from a FAST encoded feed
- Samples: sample queries and sample data (this component does not require any configuration and so it does not appear on this worksheet)

2 Use this worksheet to plan the layout of your system. This worksheet lists the information for which the installer will prompt.

Descriptions of each of the fields are provided in “Express Setup Options” on page 149. Recommended values are provided based on a system that handles 300,000 Stock Quote messages per second.

Note The installer installs components in a different order than is listed here.

Demo Feed Handler (optional)

Field	Your Value	Recommendation
Channel Name	1.	Demo Channel 1
Local Interface	2.	127.0.0.1 for single-machine installation
Broadcast IP Address	3.	224.0.2.0
Broadcast Port	4.	3050

Field	Your Value	Recommendation
If the Demo Feed Handler will send data on multiple channels, record entries 1 through 4 for each channel. Local Interface and Broadcast IP Address can remain the same for each channel. Increase Broadcast Port by 1 for each channel. Four channels are recommended for a system that handles 300,000 messages per second.		
JINI Server Host Name	5.	

Fast Feed Handler (optional)

Field	Your Value	Recommendation
Listen Host IP Address (for FAST Feed)	6.	127.0.0.1 for single-machine installation
Listen Port (for FAST Feed)	7.	3170
Multicast Host IP Address	8.	224.0.255.0
Channel Name	9.	FAST Channel 1
Local Interface	10.	127.0.0.1 for single-machine installation
Broadcast IP Address	Enter the value from line 3.	
Broadcast Port	11.	3150
If the FAST Feed Handler will send data on multiple channels, record Channel Name, Local Interface, Broadcast IP Address and Broadcast Port for each channel. Local Interface and Broadcast IP Address can remain the same for each channel. Increase Broadcast Port by 1 for each channel. Four channels are recommended for a system that handles 300,000 messages per second.		
JINI Server Host Name	Enter the value from line 5.	

Publisher API (optional)

Field	Your Value	Recommendation
Channel Name	12.	Custom Channel 1
Local Interface	13.	127.0.0.1 for single-machine installation
Broadcast IP Address	Enter the value from line 3.	
Broadcast Port	14.	3250
If the feed handler that is being developed will send data on multiple channels, record Channel Name, Local Interface, Broadcast IP Address and Broadcast Port for each channel. Local Interface and Broadcast IP Address can remain the same for each channel. Increase Broadcast Port by 1 for each channel. Four channels are recommended for a system that handles 300,000 messages per second.		
Number of message buffers	15.	Number of threads simultaneously sending messages in your feed handler.

Field	Your Value	Recommendation
JINI Server Host Name	Enter the value from line 5.	

RAPCache Subscriber

Field	Your Value	Recommendation
Channel Name	Enter the Channel Name, Broadcast IP Address, and Broadcast Port for each channel you entered for the Demo Feed Handler. Enter Demo Feed Handler as the Publisher Name. Enter the Channel Name, Broadcast IP Address, and Broadcast Port for each channel you entered for the FAST Feed Handler. Enter FAST Feed Handler as the Publisher Name. Enter the Channel Name, Broadcast IP Address, and Broadcast Port for each channel you entered for the Publisher API. Enter Custom Feed Handler as the Publisher Name.	
Publisher Name		
Local Interface		
Broadcast IP Address		
Broadcast Port		
Publisher Name	16.	Demo Feed Handler, FAST Feed Handler, or Custom Feed Handler
IP Address	17.	
Resend Listener Port	Enter the value 3003 for the Demo Feed handler, 3103 for the FAST Feed Handler, and 3203 for a custom feed handler.	
If the RAPCache Subscriber will receive data from multiple publishers, record Publisher Name, IP Address and Resend Listener Port for each publisher.		
JINI Server Host Name	Enter the value from line 5.	

RAPStore Subscriber

Field	Your Value	Recommendation
Primary File Location	18.	
Overflow File Location	19.	
Channel Name	Enter the same information you entered for the RAPCache Subscriber.	
Publisher Name		
Local Interface		
Broadcast IP Address		
Broadcast Port		
Publisher Name	Enter the same information you entered for the RAPCache Subscriber.	
IP Address		
Resend Listener Port		

Field	Your Value	Recommendation
JINI Server Host Name	Enter the value from line 5.	

Demo Subscriber (optional)

Field	Your Value	Recommendation
Channel Name	Enter the same information you entered for the RAPCache Subscriber.	
Publisher Name		
Local Interface		
Broadcast IP Address		
Broadcast Port		
Publisher Name	Enter the same information you entered for the RAPCache Subscriber.	
IP Address		
Resend Listener Port		
JINI Server Host Name	Enter the value from line 5.	

Operations Console

Field	Your Value	Recommendation
Host Name (for RAPCache)	20.	
Host Name (for RAPStore)	21.	

CHAPTER 2

Installing Sybase RAP

About this Chapter

This chapter describes installation and configuration procedures for Sybase RAP core components.

Contents

Overview

This chapter tells you how to install and configure Sybase RAP. Instructions apply to both UNIX and Linux platforms, except where noted.

Platforms and operating systems

Available platform a operating system configurations that support Sybase RAP are listed in the table below.

Hardware	Supported O/S	Runtime databases
AMD Opteron 2.2 GHz or greater	Red Hat Linux x64 - RedHat Enterprise Linux 4.0 (AMD) update: 0	Sybase IQ 12.7 ESD #3 and ASE 15.0.2
UltraSPARC 1.2GHz or greater	Sun Solaris 10 (SPARC) for 64-bit systems	Sybase IQ 12.7 ESD #3 and ASE 15.0.2
AMD Opteron 2.2 GHz or greater	Sun Solaris 10 x64/x86	Sybase IQ 12.7 ESD #3 and ASE 15.0.2
IBM P6 4.7 GHz	IBM AIX 5.3	Sybase IQ 12.7 ESD #3 and ASE 15.0.2

Planning your installation

- Determine what kind of installation best suits your needs and environment. There are Custom, Express, and Demonstration installs available. See “What installation type is right for you” on page 12 to choose the appropriate installation type.
- Plan your installation, complete all appropriate worksheets, and keep the worksheets close at hand when you install the software. If you have not done so, see “Using the worksheets to plan your installation” on page 13 about using the worksheets.
- Review all “Preinstallation tasks” on page 25 and “Installation options” on page 28 before you begin the install.
- Read the release bulletins for the latest information on the products (Adaptive Server Enterprise, Sybase IQ, and so on) that you are installing. See the special installation instructions section in the release bulletin.
- Install operating system patches, if required. If you are having trouble starting the installer, make sure you have the required operating system patches for the Java Runtime Environment (JRE).
- Use the Adaptive Server Configuration Guide to review SySAM procedures and plan your client/server configuration.
- Make sure that there is sufficient available disk space and the path name of the directory does not contain any spaces.
- Check to ensure that the operating system meets the version level, RAM, and network protocol requirements for your platform.

- Verify that your network software is configured. Sybase software uses network software even if Adaptive Server and Sybase IQ client applications are installed on a machine that is not connected to a network.

  If you are having connection problems or need to verify your network configuration, ping the host.

- Allocate a shared memory segment that is at least as large as the Adaptive Server total logical memory configuration parameter.

  The default total logical memory parameter on Sun is 47,104 2K pages (92MB) on the 64-bit operating system. To adjust the shared memory value of the operating system on Sun Solaris 10, add the following lines to the operating system configuration file */etc/project*:

```
project-sybase:200:For use by Sybase:sybase:sybase:
project.max-shm-memory=\(<threshold_level>,<threshold_value>,deny)
```

  For more details, see "Pre-installation tasks for Adaptive Server" in Chapter 2, "Installing Adaptive Server" of the *Installation Guide for Adaptive Server Enterprise* for version 15.0.2.

  Alternatively, the older method of adding an entry for shminfo_shmmax in the Solaris operating system configuration file */etc/system* will still work:

```
set64 shmsys:shminfo_shmmax = <new_value_in_bytes>
```

  To adjust the shared memory value of the operating system on Linux, add the new value in bytes to the file */proc/sys/kernel/shmmax*, so that the new value takes effect immediately. Also add a line to the file */etc/sysctl.conf*, so that the new value is retained the next time the machine is rebooted.

Preinstallation tasks

Perform the following tasks before you install Sybase RAP.

Obtain the Local Interface value

Every UDP multicast group must be bound to a network IP interface address. The RAP installer refers to this value as the **local interface**.

Custom or Express install

If you plan to perform a Custom or Express installation, ask your Network Administrator to determine which network interface to use, and record this information on the installation worksheet. Use this value for all **local interface** fields during installation.

Demo install

If you intend to perform a Demo install, you can skip this step: the installer assigns `127.0.0.1` as local interface value.

Determine the multicast IP address

Publishers multicast messages to subscribers on a class D (multicast) IP address. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0.*

Custom or Express install

If you plan to perform a Custom or Express setup, review the default multicast channel assignment with your Network Administrator. The installer assigns `224.0.2.0` as the default multicast IP Address for all publishers. If this channel is in use or not appropriate for your site, ask your administrator to assign an appropriate multicast IP address for Sybase RAP message traffic

Demo install

If you intend to perform a Demo install, you can skip this step. The installer assigns `224.0.2.0` as the multicast IP Address for all publishers and subscribers.

Check for operating system patches

Apply any operating system patches before you install Sybase RAP. Refer to the *Sybase RAP - The Trading Edition Release Bulletin* for information on operating system levels and required operating system patches.

See the table below for commands to identify the patches on your system.

OS	Command
Sun Solaris	showrev -p
Linux	uname -a
IBM AIX	oslevel -r

Note Do not use a patch that is earlier than the version suggested for your operating system. Use the patch recommended by the operating system vendor, even if it supersedes the patch listed here.

Change the Linux host file mappings

On UNIX and Linux based systems, settings in the */etc/host* file map hostnames to IP addresses. Linux maps the host machine's hostname to `127.0.0.1` by default. If you are installing on Linux, you must remove this setting from all of the servers running Sybase RAP components. This change allows the JINI server to locate remote components on the various machines in the network.

Example

On a Linux system called *MyComputer*, change the host and domain name entry in the */etc/hosts* file that looks like this:

```
127.0.0.1       MyComputer MyComputer.mydomain.com localhost
```

To this:

```
127.0.0.1       localhost
```

Configure AIX for Large Files

For each machine running IBM AIX, configure AIX to allow large file sizes and set the memory requirements to their hard limits. To make these changes, modify the following lines in the */etc/security/limits* file as follows:

`fsize = -1` (indicates unlimited file size)
`rss = -1` (indicates unlimited memory)
`data = -1` (indicates unlimited data segment)

The changes take effect upon your next log in. Users logged in at the time of the change must log out and back in again for the change to take effect.

Installing platform components

Assemble the installation media and review the "Installation options" on page 28. Know what items you want to install and where you want to install them.

Installation media

The Sybase RAP Platform Product DVD and several related client CDs are required for installation.

Media	Description
Sybase RAP Product DVD	Installs Sybase RAP components and other support tools.
ASE 15.0.2 PC-Client CD	Installs several additional Sybase products. Before installing any product, review the *readme.txt file*, which contains a brief description of each of the products, lists dependencies, and contains any last-minute instructions or changes
Sybase IQ 12.7 Network Client CD	Installs the components required for client connections to a Sybase IQ network server.
OpenSwitch 15.1 Product CD	Installs OpenSwitch 15.1, which can provide near-continuous database access during planned downtime and unplanned outages.
RAP PowerDesigner 12.5 CD	Installs PowerDesigner 12.5, which you use to view and modify the Sybase RAP data models on Windows.

Installation options

The Sybase RAP Product DVD installs all of the components you need to deploy a fully configured Sybase RAP environment. There are three installation types: Demo installations as well as Custom and Express installations. Review each installation type to determine which installation best suits you environment.

Demo installations

Demonstration installations are designed to install a basic functioning system on a single machine. Demo installations are performed from the command line, require little planning, and are designed to demonstrate system functionality.

Before you decide which install to perform, review the sections Plan your disk space and memory requirements, Preinstallation tasks, and the sections on Custom installer actions and Express installer actions to help you decide which installation type is most suitable for your environment.

Plan your disk space and memory requirements

Use the following table to plan your disk space and memory requirements on the target machine.

Component		Disk Space	RAM
Demo Feed Handler		50MB	8GB
FAST Feed Handler		50MB	8GB
Publisher API		32MB	N/A
RAPCache System	RAPCache Database Server	2GB	32GB
	RAPCache Subscriber		
RAPStore System	RAPStore Database Server	2GB	32GB
	RAPStore Subscriber		
Demo Subscriber		50MB	8GB
Operations Console		100MB	4 GB
Samples		280MB	N/A

Demo installer actions

Demo installs create a test environment on a single machine that you can use before you deploy a production system. During installation, the installer:

- Assigns default port values in the `3000 - 4000` range to all components. Port values are validated during installation, and if a defined port is already in use, the install will try the range of 3800 - 4000. If one of these is in use, the install can be run and the port range for **port in use** issues can be changed.

  For example, the worksheet has 3001 for the Listening Port for the Demo Feed Handler. If this is free, the demo install will use this value. If that port is in use, the install will try the range of 3800 - 4000.

- Assigns `224.0.2.0` as the default class D multicast IP addresses to all publishers, and `127.0.0.1` as the default local host address.

- Assigns two data channels for each publisher. Default channels names depend on the component. for Channels and Publishers will be entered. For FASTFeedHandler, the channel names are `FAST Channel 1` and `FAST Channel 2`; for Demo Feed Handle, the channel names are `Demo Feed 1` and `Demo Feed 2`; and for the Publisher API, the channel names are `Custom Channel 1` and `Custom Channel 2`.

- Assigns default settings for various configuration options, such as the Number of Packet Buffers.

- Installs the RAPCache database and RAPCache Subscriber on the target machine. RAPCache is set to ASE 15.0.2 defaults. RAPCache Subscriber is configured to communicate with the RAPCache database and all publishers.
- Installs the RAPStore database and RAPStore subscriber on the target machine. RAPStore is set to Sybase IQ 12.7 ESD #3 defaults. RAPStore Subscriber is configured to communicate with the RAPCache database and all publishers.

❖ **Running a demo install**

1 Insert the Sybase RAP Product DVD into the drive.

```
<CD_mount_directory>/rap_cd1/setup<Platform>.bin -V
InstallType=demo
```

where <*Platform*> is the name of your operating system. For example, use the command `/cdrom/rap_cd1/setupSolaris.bin -V InstallType=demo` on Sun Solaris.

Note If you are installing on IBM AIX, omit `.bin` from the command. For example: `/cdrom/rap_cd1/setupaix -V InstallType=demo`

This command starts the InstallShield Wizard. When the installer prompts you to continue, choose Next.

2 When the InstallShield Wizard starts, follow the directions on the screen.

Notes

- After you run the Demo install, refer to the appropriate “Postinstallation tasks” on page 36
- The installer assigns default port values to all components. If the default port assignments are currently in use, run the installer with this command to generate a pool of secondary set of port assignments:

```
setup<platform>.bin –V START_PORT=<startvalue> -V
END_PORT = <endvalue>
```

where <*Platform*> is the name of your operating system. For example, use the command `/cdrom/rap_cd1/setupSolaris.bin -V InstallType=demo START_PORT=3800 -V END_PORT=4000` to perform a Demo install on Sun Solaris with a range of port assignments in the 3800 – 4000 range.

Custom and Express installations

Custom and Express installations are suitable for production environments. Although all components are selected by default, you can choose the components you want to install. You can select group packages, individual items, or a combination of the two. Clicking the box next to a component selects or deselects that component.

Before you decide which install to perform, review the section on Planning your disk space and memory requirements, and the sections on Custom installer actionsand Express installer actions to help you decide which installation type is most suitable for your environment.

Planning your disk space and memory requirements

A typical production environment requires four servers. The following tables identify installation options, recommended memory requirements and installation locations. Disk space requirements for each server depends on the components you install.

Machine 1	Component	Required	Optional	Disk Space	RAM
	Demo Feed Handler		•	50MB	8GB
	FAST Feed Handler		•	50MB	8GB
	Publisher API		•	32MB	N/A

Machine 2	Component	Required	Optional	Disk Space	RAM
	RAPCache Database Server	•		2GB	32GB
	RAPCache Subscriber	•			

Machine 3	Component	Required	Optional	Disk Space	RAM
	RAPStore Database Server	•		2GB	32GB
	RAPStore Subscriber	•			

Machine 4	Component	Required	Optional	Disk Space	RAM
	Operations Console	•		100MB	4 GB

Notes

- All publishers must be installed on the same machine. This is machine 1 in the above tables.
- Disk space requirements for the RAPCache system and RAPStore System represent a combined minimum requirement for the database server and subscriber. Additional disk space is required for each database.

Custom installer actions

Custom installs provide the most flexibility for production environments. During installation, the installer:

- Prompts you for the name of the JINI server host. The JINI server is installed on the same machine where you install Operations Console. If you install multiple components at the same time, the installer only prompts you for the JINI server host name once. This value is inherited by the other components you are installing during that installation sequence. When you complete an installation sequence on one machine and restart the installer to install components on another, you may be prompted to specify the location of the JINI server.
- Assigns default port values in the `3000 – 4000` range to all publisher components. Port values are validated during installation, and if a defined port is already in use, the installer displays two popup message boxes. The first message box tells you that the port number is already in use and an alternative port will be selected. The second message box then provides the new port assignment. If your Network Administrator chooses a different set of port values, records these items on your worksheet, and replace the default value with these items.
- Assigns *224.0.2.0* as the default class D multicast IP addresses to all publishers. This value appears in the Broadcast IP Address field and identifies the multicast channel that the publisher uses to multicast messages to subscribers. If your Network Administrator chooses another multicast address for your site, record the address on your worksheet, and replace the default with that value.

- Assigns default Channel names to Feed Handlers and Publishers. Default channels names depend on the component. For FAST Feed Handler, the first default channel name is `FAST Channel 1`; the first default channel name for Demo Feed Handler is `Demo Feed 1`; and the first default channel name for the Publisher API is `Custom Channel 1`. If you want to use different names to identify your data channels, replace the default value with the one you want to use.
- Lets you install the RAPCache Database Server and RAPCache Subscriber separately. Running this kind of install installs the RAPCache Database Server on the target machine with ASE 15.0.2 defaults, and sets the host name to the install machine. If you install the RAPCache Subscriber at the same time, this component inherits all appropriate server connection settings, and prompts you for the appropriate data channel(s) and publisher configuration. If you install the RAPCache Subscriber separately you must configure the data channel(s) and publisher and specify the location of the RAPCache Database Server.
- Lets you install the RAPStore Database Server and RAPStore Subscriber separately. Running this kind of install installs the RAPStore Database Server on the target machine with Sybase IQ 12.7 ESD #3 defaults, and set the host name to the install machine. If you install the RAPStore Subscriber at the same time, this component inherits all appropriate server connection settings, but prompts you for the appropriate data channel(s) and publisher configuration as well as the location of the RAPStore Database Server.
- Assigns all appropriate port values for Operations Console. If you install Operations Console at the same time you install RAPCache and RAPStore, the installer will populate the appropriate Host and port values for these items. If not, the Host field(s) will be left empty and the port will be the appropriate default for the database (5000 for RAPCache and 2638 for RAPStore).

Note You may want to review “Custom Setup Options” on page 91 before you start. This appendix describes all of the configuration options you may see during installation.

Express installer actions

Express installs involves fewer configuration options, but are suitable for some production environments. During installation, the installer:

- Prompts you for the name of the JINI server host. The JINI server is installed on the same machine where you install Operations Console. If you install multiple components at the same time, the installer only prompts you for the JINI server host name once. This value is inherited by the other components you are installing during that installation sequence. When you complete an installation sequence on one machine and restart the installer to install components on another, you may be prompted to specify the location of the JINI server.
- Assigns default port values in the `3000 – 4000` range to all feed handler and publisher components.Port values are validated during installation, and if a defined port is already in use, the installer displays two popup message boxes. The first message box tells you that the port number is already in use and an alternative port will be selected. The second message box then provides the new port assignment. If your Network Administrator chooses a different set of port values, records these items on your worksheet, and replace the default value with these items.
- Assigns *224.0.2.0* as the default class D multicast IP addresses to all publishers. This value appears in the Broadcast IP Address field and identifies the multicast channel that the publisher uses to multicast messages to subscribers. If your Network Administrator chooses another multicast address for your site, record the address on your worksheet, and replace the default with that value.
- Assigns default settings for various configuration options, such as the Number of Packet Buffers.
- Assigns default Channel names to Feed Handlers and Publishers. Default channels names depend on the component. For FAST Feed Handler, the first default channel name is `FAST Channel 1`; the first default channel name for Demo Feed Handler is `Demo Feed 1`; and the first default channel name for the Publisher API is `Custom Channel 1`. If you want to use different names to identify your data channels, replace the default value with the one you want to use.
- Assigns default settings for various configuration options, such as the Number of Packet Buffers.
- Installs the RAPCache database and RAPCache Subscriber on the same target machine at the same time. You cannot install these items separately. The RAPCache Database Server is set to ASE 15.0.2 defaults, and the host name is set to the install machine. The RAPCache Subscriber inherits all appropriate server connection settings, but prompts you for the appropriate data channels to monitor and publisher settings.

- Installs the RAPStore database and RAPStore subscriber on the same target machine at the same time. You cannot install these items separately. The RAPStore Database Server is set to Sybase IQ 12.7 ESD #3 defaults, and the host name is set to the install machine. The RAPStore Subscriber inherits all appropriate server connection settings, but prompts you for the appropriate data channels to monitor and publisher settings.
- Assigns all appropriate port values for Operations Console. If you install Operations Console at the same time you install RAPCache and RAPStore, the installer will populate the appropriate Host and port values for these items. If not, the Host field(s) will be left empty and the port will be the appropriate default for the database (5000 for RAPCache and 2638 for RAPStore).

Note You may want to review Appendix B, "Express Setup Options" before you start. This appendix describes all of the configuration options you may see during installation.

❖ **Running the Sybase RAP installer**

1 Log in as root.

2 Insert the Sybase RAP Product DVD into the drive.

If Sybase RAP Setup does not start automatically, mount the drive, then use this command to start the installer:

```
<CD_mount_directory>/rap_cd1/setup<Platform>.bin
```

where <*Platform*> is the name of your operating system. For example, use the command `/cdrom/rap_cd1/setupSolaris.bin` on Sun Solaris.

Notes

If you are installing on IBM AIX, omit `.bin` from the command. For example: `/cdrom/rap_cd1/setupaix`

When you run the setup command, your current directory can be any directory that is *not* on the DVD.

3 Click Next on the Welcome screen.

4 Choose a location, then read and accept the license agreement. Click Next.

If you do not find a license agreement that matches your location, or if the license agreement is unreadable on your system, you can read all available license agreements on the Sybase Web site at http://www.sybase.com/softwarelicenses.

5 Choose an installation directory and click Next.

You can accept the default target directory or choose another installation directory. The default installation directory is */opt/sybase/RAP*.

6 Choose the components you want to install.

7 Follow the instructions on the screen to complete your installation.

Note

- You must run the install for each RAPCache server you want to install. If you want to install multiple servers or deploy components on different machines, you must run the install multiple times.
- If you need to find an installation item, see Appendix C, "Installation and Configuration File Locations."

Postinstallation tasks

The RAPStore and RAPCache servers are running when the installation completes. After you install the Sybase RAP core components, shut down the RAPCache server, obtain and activate your SySAM license, restart the RAPCache server, check the status of your servers, and load the sample data.

Obtain a valid SySAM license

Sybase Software Asset Management (SySAM) performs license administration and asset management tasks for Sybase products. Sybase RAP requires a valid SySAM license.

The Sybase Product Download Center (SPDC) enables authorized Revenue Order Contacts and Technical Support Contacts with current support contracts with download access to Sybase product releases. You must use SPDC to dynamically generate license keys for Sybase RAP.

Accessing SPDC requires a separate login and password; authorized contacts automatically receive SPDC login and password information through e-mail.

Refer to the SySAM online documentation at http://infocenter.sybase.com/help/index.jsp or the *User's Guide Sybase Software Asset Management 2.0* on the Getting Started CD for more information on the use of SySAM.

❖ **Obtaining your SySAM license**

1 Start your browser and log in to the Sybase Product Download Center (SPDC) at https://sybase.subscribenet.com.

2 On the Product List page, choose RAP - The Trading Edition.

3 On the Product Information Page, choose the appropriate version:

- RAP - The Trading Edition for Linux x86-64 - 64bit
- RAP - The Trading Edition for Sun Solaris SPARC - 64bit
- RAP - The Trading Edition for Sun Solaris x64/x86
- RAP - The Trading Edition for IBM AIX - 64bit

4 On the Sybase Software License Agreement page, select a country or region. Click I agree, if necessary.

5 Agree to the software terms and conditions.

6 On the Product Download page, click the License Keys link.

7 On the License Information page, choose the applicable license, scroll to the bottom of the page, click Select to Generate.

8 On the Generate Licenses page in Step 1 - License Model, choose the appropriate license model, click Next.

Questions about your license type? Click the more information link to learn more about the licenses Sybase offers.

9 On the Generate Licenses page in Step 2 - License Quantity, specify the number of licenses you want to generate. Click on the links for more information.

10 On the Generate Licenses page in Step 3 - Host Information, add the Server Host ID, Host Name, and (optional) Port Number. Click Generate.

Click on "What's My Host ID," if you need help determining your host ID.

The View Licenses page displays your license information.

11 Click Download License File to save your license file to disk.

See also

For more information on the use of SySAM, refer to:

- "SySAM Configuration" in the *Adaptive Server Enterprise Version 15.0 Configuration Guide* for your platform
- SySAM online documentation at http://infocenter.sybase.com/help/index.jsp
- *User's Guide Sybase Software Asset Management 2.0* on the Getting Started CD
- "Troubleshooting SySAM Issues" in the *Adaptive Server Enterprise 15.0 Installation Guide* for your platform

Activate your SySAM license

Activate your SySAM license after you install Sybase RAP on your host machine and obtain the license file.

❖ **Activating your SySAM license**

1 Change directory to *$RAP*/RAPCache/, where *$RAP* is your Sybase RAP installation directory. The default installation directory is */opt/sybase/RAP*.

2 Source the environment variables:

```
source SYBASE.sh
```

3 Login to the RAPCache server, check the SySAM license type (the default is CP), and change the type, if necessary.

- To check the license type, run the stored procedure sp_lmconfig in Interactive SQL (isql) on your RAPCache server:

  ```
  sp_lmconfig
  go
  ```

  The valid license types for Sybase RAP are:

  - CP (CPU license for Production)
  - DT (Development and Test, not CPU dependent)
  - SF (CPU license for Stand By Production)

- To change the license type to SF or DT to match your license file, run the appropriate command:

```
sp_lmconfig 'license type', 'SF'
go
```

– Or –

```
sp_lmconfig 'license type', 'DT'
go
```

4 Shut down the RAPCache server:

```
shutdown with nowait
go
```

5 If you are running in unserved mode, copy the license file saved from SPDC to the *$RAP/*RAPCache*/SYSAM-2_0/licenses* directory on the host machine. (See "Obtain a valid SySAM license" on page 36.)

If you use the ftp utility to transfer the license file, be sure to specify ASCII mode.

6 If you are running in served mode, copy the license file to the directory *<license_server_install_dir>/SYSAM-2_0/licenses* on the license server.

Change directory to *$RAP/*RAPCache*/SYSAM-2_0/bin*. If the license server is not running, run the following command to start the license server:

```
sysam start
```

Run the following command to load all the licenses in the license server folder, including the new Sybase RAP license:

```
sysam reread
```

You must have at least one served license in the *licenses* directory *$RAP/*RAPCache*/SYSAM-2_0/licenses*, or the error `"License Manager: Can't initialize..."` is reported when you attempt to start the license server.

7 Update the license server options file.

- Change directory to *<license_server_install_dir>/SYSAM-2_0/licenses* on the license server.
- Using a text editor, add the following entry to the file *SYBASE.opt*:

```
INCLUDE SY_RAP PROJECT RAP
```

Note If the options file *SYBASE.opt* does not exist, then the sysam script (sysam start) in the directory *$RAP/*RAPCache*/SYSAM-2_0/bin* creates the options file. You can change the name and location of the options file, if you explicitly set the name and location in the license file. For full details on the options file, see the *FLEXnet Licensing End User Guide* in the SySAM documentation set.

8 Create a *sybase.lic* file in the directory *$RAP/*RAPCache*/SYSAM-2_0/licenses*. The *sybase.lic* file specifies the port number of the license server, so SySAM can retrieve the appropriate license. The contents of the *sybase.lic* file is:

```
SERVER <license_server_host_name> ANY 27000
USE_SERVER
```

where 27000 is the default port. You can use a different port number.

9 Restart the RAPCache server. See "Start the RAPCache server" on page 40 for more information.

Note To verify your license, you can run the sp_lmconfig stored procedure in Interactive SQL (isql) or check the ASE error log.

In the ASE error log, a line beginning with `"Checked out license..."` indicates that the license configuration is successful. If you see a `"Sysam: FLEXnet Licensing error:"` message, check with your Sybase representative to resolve the issue.

Start the RAPCache server

After you install the core components and activate your SySAM license, restart the RAPCache server. To start RAPCache, you execute a *RUN_<server_name>* file that contains the information required to start or restart the server.

❖ **Starting the RAPCache server**

1 Open a command window.

2 Change directory to *$RAP/*RAPCache, where *$RAP* is your Sybase RAP installation directory.

3 Use the appropriate shell command to source the environment variables:

```
source SYBASE.csh
```

– Or –

```
. ./SYBASE.sh
```

4 Change directory to *$RAP/*RAPCache*/ASE-15.0/install*.

5 Use this command to start the RAPCache server:

```
./startserver -f RUN_<server_name> &
```

where <*server_name*> is the name of the server. The & starts a background process that continues to run after you exit the terminal window.

For example, the *RUN_*<*server_name*> file name for an Adaptive Server named rapserver is *RUN_rapserver*.

Verify that all servers are running

After you start RAPCache, check to see that your servers are running and that you can connect to the servers.

❖ **Verifying RAPCache**

Use the showserver command to determine whether the RAPCache server is running.

1 Open a command window.

2 Change to *$RAP/*RAPCache, where *$RAP* is your Sybase RAP installation directory.

3 Use the appropriate shell command to source the environment variables:

```
source SYBASE.csh
```

– Or –

```
. ./SYBASE.sh
```

4 Run this command:

```
$RAP/RAPCache/ASE-15.0/install/showserver
```

5 Check the message for references to: `...$RAP/`RAPCache`/ASE-15.0/bin/dataserver...`

❖ **Verifying RAPStore**

Use the stop_asiq utility to verify that the RAPStore Server is up and running.

1 Open a command window.

2 Change directory to *$RAP/*RAPStore*/ASIQ-12_7*.

3 Use one of these commands to source the environment variables:

```
source ASIQ-12_7.csh
```

– Or –

```
. ./ASIQ-12_7.sh
```

4 Run this command:

```
$RAP/RAPStore/ASIQ-12_7/bin/stop_asiq
```

RAPStoreServer displays a message that indicates the current status of your server, and prompts you to decide whether you want to shut down the server.

5 Type `N` to respond to the prompt.

6 If the RAPStoreServer is not running, change directory to *$RAP/*RAPStore*/ASIQ-12_7/RAPDB* and enter:

```
start_asiq @<RAPStore_server>.cfg <RAPDB>.db
```

where <RAPStore_*server*> is the name of your RAPStore server and <*RABDB*> is the name of your RAPStore database.

Load the sample data into RAPCache

Run the following script to load the sample data into the RAPCache database: *$RAP/*RAPCache*/datamodel/scripts/load_unix_cache_TAQ_tables.sh*

For more details, see “Loading sample data” in Chapter 3, “Sample Queries” of the *Sybase RAP - The Trading Edition Users Guide*.

Adjust the default named cache size in the RAPCache

Adjusting the default named cache size increases RAPCache performance as an in-memory database. The amount of shared memory available to ASE determines the following configuration variables:

```
max memory
```

```
default data cache
```

The value for `max memory` should be the maximum amount of shared memory available on your system. The amount of shared memory available is generally controlled by operating system configuration, and will likely need to be modified before a usable RAPCache can be properly configured.

The amount of memory to reserve for shared memory is dependent on total memory available on your system. The maximum amount of memory should be allocated for shared memory, but enough memory should be retained for use by the various processes that will be running on the system. For example, if a system is dedicated to the RAPCache and RAPCache Subscriber, and has a total of 32 Gigabytes of memory (RAM), then approximately 28-30GB of operating system shared memory should be configured.

Once the operating system shared memory has been configured, then the RAPCache can be configured to use it. This is done by first configuring 'max memory' to the total amount of shared memory available on your system, and then by configuring the default data cache.

Data cache size will be dependent on the max memory configuration value. Use as much memory as can be spared, given the system on which ASE is running. This depends on value of max memory. If max memory is 16GB, allocate 15GB for data cache; the rest is to be used for proc cache, RAPCache system memory usage, etc.

When configuring default data cache size, it is also best to partition the cache, so that access to the cache from multiple engines does not cause contention. The partition count should equate to the number of engines, but has to be specified in powers of two. For example, if you have configured 6 engines, then the cache partition count should be set to 8.

Here is an example of a SQL script used to configure `max memory` to 16GB, and to configure `default data cache` with 15 gigabytes of memory, and 8 cache partitions:

```
sp_configure 'max memory', 8000000 -- 8 million 2k pages = 16GB
go
sp_cacheconfig 'default data cache', '15G', 'cache_partition=8'
go
```

Adjust the number of locks in the RAPCache

By default, the RAPCache is configured with 5000 locks. System administrators can use `sp_configure` to change this limit. For example:

```
sp_configure 'number of locks', 200000
```

You may also need to adjust the `sp_configure` parameter `max memory` since each lock uses memory.

At a minimum, the number of locks needs to be greater than the number of locks required by the RAPCache Subscriber. RAPCache may also need locks for other applications which may be querying the data. The RAPCache Subscriber requires locks when inserting data into the RAPCache. The amount of locks required is affected by the bulk batch size parameter in the *rapcache.xml* configuration file. This parameter tells the RAPCache Subscriber how many rows to do in a bulk insert before committing the transaction.

The bulk batch size parameter is a value which is used per message type, so if the bulk batch size is 10,000 and the RAPCache Subscriber handles 20 message types; the number of locks required will be 200,000. In addition, if the table is partitioned, then the number of locks required just for this program is (number of partitions) * (bulk batch size).

The reference application contains 14 message types. Two of these message types (stock quote and stock trade) have tables, which are partitioned with 3 data partitions. The remaining 12 message types do not have partitioned tables. With a bulk batch size of 10,000, the RAPCache Subscriber would need a minimum of (12 * 10000) + 2 * (3 * 10000) = 180,000 locks.

Clear the cache

Clear the data from the RAPCache on a regular schedule, as the data is permanently stored in the RAPStore and is redundant in the cache. For details, see the section “Clearing the cache” in Chapter 3, “Sample Queries” of the *Sybase RAP - The Trading Edition Users Guide*.

Use the ftp utility to move the data models to Windows

The Sybase RAP data models are stored in separate subdirectories (*$RAP/*RAPCache*/datamodel* and *$RAP/*RAPStore*/datamodel*). If you want to view the data models with PowerDesigner, use the ftp utility to move the data models from UNIX to Windows.

Create the primary and overflow file directories

During installation, the installer prompts you to identify the primary and overflow file directories that RAPStore uses to write market data messages before it loads the database. The location of these directories depends on the type of install you performed.

Demo installs

If you ran a Demo install, you must create these directories at the following location:

```
$RAP/RAPStoreSubscriber/files/primary
$RAP/RAPStoreSubscriber/files/overflow
```

Custom and Express installs

If you ran a Custom or Express install, you can create the primary or overflow directories at any location. The location of these directories should match the values on your worksheet, and must match the location you indentifed during the install.

Postinstallation task for Linux users

As you tune the RAPCache server by adding named caches or increasing the sizes of existing caches, the server may stop rebooting at some point. The memory allocated to the server is large enough to accommodate the additional cache size, and the operating system level parameter SHMMAX (the maximum size in bytes for a shared memory segment) is also large enough, but the server does not restart. The problem may be that the operating system level parameter SHMALL, the system-wide maximum of shared memory pages, is not large enough. To resolve this issue, increase the value of SHMALL in the file */proc/sys/kernel/shmall*. Modifying */proc/sys/kernel/shmall* requires root permission.

Managing servers with Sybase Central

Sybase Central is a management console for Sybase products. Sybase Central lets you manage multiple servers from a single location, and perform complex administration tasks without the need to remember the Transact-SQL commands or system stored procedures.

❖ **Starting Sybase Central**

1 Open a terminal window, and change directory to *$RAP/*RAPCache.

2 Use one of these commands to source the environment variables:

```
source SYBASE.csh
```

– Or –

```
. ./SYBASE.sh
```

3 Use this command to start Sybase Central:

```
$RAP/RAPCache/ASEP/bin/aseplugin
```

4 Select Tools | Connect, then choose a server from the drop-down box.

5 On the Connect dialog, enter the user name and password of the System Administrator in the appropriate text boxes.

Warning! The first time you log in to Adaptive Server Enterprise, use the default sa user name and leave the password blank. After you log in for the first time, change the password of the System Administrator. See "Setting the System Administrator password" on page 46.

6 Choose the name of your RAPCache server as the Server Name.

7 Click OK.

❖ **Registering the RAPStore Database Server with Sybase Central**

To access the RAPStore Server from Sybase Central, you must register the Sybase IQ plug-in.

1 Click Tools | Plug-ins.

2 On the Plug-ins panel, click Register.

3 On the plug-in registration dialog, click Browse.

4 Change the path to *$RAP/*RAPStore*/ASIQ-12_7/java*.

5 Choose *iq.jpr* and click OK.

6 Click Finish to close the plug-in registration dialog.

7 Close the Plug-ins panel.

Setting the System Administrator password

RAPCache and RAPStoreServer include different administrator default logins. Sybase recommends that you change these logins to protect your system.

❖ **Adding the RAPCache administrator**

The RAPCache installation creates an administrator called sa. There is no initial password set for this account. Because an sa user can use any database on Adaptive Server with full privileges, Sybase recommends that you create a password to protect your production environment.

1 In the left pane, click Adaptive Server Enterprise | Default | <RAPCache_*server*>, where <RAPCache_*server*> is the name of your RAPCache server.

2 Click the Logins folder.

3 In the right pane, right-click sa and choose Properties.

4 On the Login Properties dialog, click the Parameters tab, then click Change Password.

The Parameters tab also includes options that allow you to set the minimum password length, maximum login attempts, and expiration period.

5 On the Change Password dialog:

- Type your password in New Password box.
- Type your password in New Password box, then retype the password in the Confirm New Password box.

6 Click OK.

Notes

- Sybase recommends that you create passwords with at least six characters that include both letters and numbers.
- If you change the RAPCache password, you must also change and encrypt the passwords for various agents. See

---

❖ **Adding the RAPStoreServer administrator**

The RAPStore Server installation creates an administrator account with a default user ID (DBA) and password (SQL). The DBA user ID identifies a user with full administration and resource creation rights. The first time you log in to the RAPStoreServer, change the DBA password to protect your production environment.

1 In the left pane, right-click the Sybase IQ icon and choose Connect.

2 On the connection dialog, click the Database tab.

3 From the Server name drop-down box, enter *host_name*:*port_number*, where *host_name* is the name of the machine that hosts the RAPStore server, and *port_number* is the server port number.

4 Click the Browse button next to the Database file drop-down box, choose <RAPStore_*database_name*>.*db*, where <RAPStore_*database_name*> is the name of your RAPStore database. Click OK

5 Click the Identification tab.

6 Type `DBA` in the User id box, type `SQL` in the Password box, click OK.

You are now connected to the RAPStore database on the RAPStore server. Expand Sybase IQ in the left panel to see the server name. Expand the server name to see the database name.

7 Expand the database folder, then expand the Users & Groups folder.

8 Right click DBA and select Properties in the pop-up menu.

9 In the DBA User Properties dialog, enter and confirm the new DBA password.

10 Click OK to save and exit the dialog. The new password takes effect immediately for new DBA connections.

Encrypting Sybase RAP passwords

If you change the server administrator and RAPCache administrator passwords, you must update and encrypt the password(s) in the agent plugins.

❖ **Using the ASE password encryption tool to encrypt the DBA password**

This procedure uses the ASE password encryption utility, passencrypt, to update and encrypt passwords for the RAPCache *agent-plugin.xml* only. Updating other agent plugins requires the RAP encryption tool.

1 Stop the RAPCache UAF agent, if necessary.

2 Open a command window, change to *$RAP/*RAPCache*//UAF-2_0/bin.*

3 Run the passencrypt encryption tool. The syntax is:

```
passencrypt [-help]
-[algorithm <encryption_algorithm>]
[-provider <cipher_provider_class>]
-text <clear_text>
```

where:

- **help** – print help
- **algorithm** – is the encryption algorithm, and is optional. The default value is DES. The supported algorithms are DES and RSA.
- **provider** – is the cipher provider class, optional. The default DES provider Sun's JCE. Default RSA provider is *org.bouncycastle.jce.provider.BouncyCastleProvider*.
- **text** – clear text, required. The <*clear_text*> value should be the new ASE password.

4 Update the *ase.password* property in the file *$RAP/*RAPCache*/UAF-2_0/plugins/com.sybase.ase/agent-plugin.xml* with the encrypted value retrieved from the utility in step 2.

5 Restart the RAPCache UAF agent.

Using the rapencrypt tool to encrypt component passwords

If you change passwords in RAPCache and RAPStore, you must also update and encrypt some component passwords. You must use the Sybase RAP encryption tool, rapencrypt, to encrypt component passwords. The syntax is:

```
./rapencrypt "password"
```

Passwords must have at least one character. Only alphanumeric and special characters are allowed. Avoid characters like `! "` and `` ` `` which may not be permissible in certain shells.

Changing the RAPCache RAP_USER password

If you change the `RAP_USER` password in the RAPCache, use the rapencrypt utility to encrypt the new password, then refer to the following table to update the appropriate values in the configuration files.

Go to this directory...	Open this file...	Update this...
$RAP/RAPCacheSubscriber/config	*rapcache.xml*	LoginPassword
$RAP/OpsConsole/WebMonitoringConsole/bin	*opsonconsole.properties*	DatabasePassword

Changing the RAPStore RAP_USER password

If you change the `RAP_USER` password in the RAPStore, use the rapencrypt utility to encrypt the new password, then refer to the following table to update the appropriate values in the configuration files.

Go to this directory...	Open this file...	Update this...
$RAP/RAPStoreSubscriber/config	*rapstore.xml*	WorkDBPassword StoreDBPassword
$RAP/OpsConsole/WebMonitoringConsole/bin	*opsonconsole.propertie*	RapStoreDatabasePassword

Creating Sybase RAP users

You must create separate user and group accounts for RAPCache and RAPStoreServer for users to query the databases. Before you create these accounts, be aware that user and group models in Adaptive Server Enterprise and Sybase IQ differ.

How users connect

- All ASE users require a server login ID and password and a user ID for each database they want to access on that server.
- Sybase IQ users do not require a server login ID. All Sybase IQ users receive a user ID and password for a database.

User groups

ASE and IQ groups differ:

- ASE allows each user to be a member of only one group.
- In IQ, users and groups are database objects. Groups can contain users and other groups. Users can belong to multiple groups, and group hierarchies are allowed.

Database object permissions

GRANT and REVOKE statements for granting permissions on individual database objects are very similar.

- Both ASE and Sybase IQ allow SELECT, INSERT, DELETE, UPDATE, and REFERENCES permissions on database tables and views, and UPDATE permissions on selected columns of database tables.

  For example, the following statement is valid in both products:

  ```
  GRANT INSERT, DELETE
  ON TITLES
  TO MARY, SALES
  ```

  This statement grants permission to use the INSERT and DELETE statements on the TITLES table to user MARY and to the SALES group.
- Both products allow EXECUTE permissions to be granted on stored procedures.
- ASE also supports GRANT and REVOKE on additional items:

    - Objects: columns within tables, columns within views, and stored procedures
    - User abilities: Create Database, Create Default, Create Procedure, Create Rule, Create Table, Create View
- Sybase IQ requires a user to have RESOURCE authority to create database objects. (A closely corresponding Adaptive Server Enterprise permission is GRANT ALL, used by a Database Owner.)
- Both products support the WITH GRANT OPTION clause, allowing the recipient of permissions to grant them in turn, although IQ does not permit WITH GRANT OPTION to be used on a GRANT EXECUTE statement.

Database-wide permissions

Adaptive Server Enterprise and Sybase IQ use different models for database-wide permissions.

- Sybase IQ employs DBA permissions to allow a user full authority within a database.
- ASE System Administrators enjoy database-wide user permission for all databases on the server, but the database owner must use a SETUSER statement to gain permissions on objects owned by other users.

Adding users

ASE and Sybase IQ use different models to create users.

- ASE requires separate server and database accounts. sp_addlogin creates a server login account for a new user. sp_adduser gives database and group privileges.
- Sybase IQ adds users in a single step, but does not assign permissions beyond connecting to the database and viewing the system tables. Permissions to access database tables must be explicitly assigned.
- Sybase IQ does not require stored procedures to add or drop users, but does allow DBAs to add or drop user accounts. When a DBA enables IQ User Administration, these accounts let DBAs control user connections and password expirations.

Where to find additional information

For details about managing Adaptive Server login accounts and database users, see “Managing Adaptive Server Logins, Database Users, and Client Connections” in the *Adaptive Server Enterprise System Administration Guide, Volume 1*.

For details about Sybase IQ User Administration, see “Managing User IDs and Permissions” and “Sybase IQ as a Data Server,” in the *Sybase IQ System Administration Guide*.

Installing client tools

You must install the appropriate tools on each client machine to connect to the Sybase RAP databases. Adaptive Server client tools are required for a client to connect to RAPCache. Sybase IQ client tools are required for a client to connect to the RAPStoreServer.

Adaptive Server Client tools

The ASE 15 PC-Client CD contains several additional Sybase products. Before installing any product, review the *readme.txt file*, which contains a brief description of each of the products, lists dependencies, and contains any last-minute instructions or changes.

❖ **Installing the client tools on Windows**

Because some portion of the PC-Client components are required by the other products, install the PC-Client components first.

1 Use an account with Administrator privileges to log in to Windows.

2 Exit any Windows programs running on your machine.

3 Insert the PC-Client CD into the CD drive.

If PC-Client Setup does not start automatically:

- Click Start, choose Run.
- Browse to select your CD drive letter, choose *setup.exe*.

4 Follow the instructions on your screen to install the PC-Client tools.

Note To configure client connections to Adaptive Server, see *Configuring Adaptive Server Enterprise* for your platform.

Configuring network connections for client products

Adaptive Server communicates with other Adaptive Servers, Open Server applications (such as Backup Server), and client software on your network. Clients can talk to one or more servers, and servers can communicate with other servers by remote procedure calls.

For Sybase products to interact with one another, each product must know where the others reside on the network. This information is stored in the interfaces file (*sql.ini* on Windows), or in a Lightweight Directory Access Protocol (LDAP) server.

For additional information, see "Installing Sybase PC-Client Products" in the *Adaptive Server Enterprise Installation Guide* for your platform. This document includes detailed installation instructions for Adaptive Server client products like the Adaptive Server plug-in to Sybase Central.

Sybase IQ Network Client tools

This section includes procedures that tell you how to install the Sybase IQ Network Client on Windows and Linux.

You can also install the IQ Network Client on many end-user machines remotely, without displaying dialog boxes or requiring interactive responses. For directions, see "Installing without human interaction" in the *Sybase IQ Installation and Configuration Guide* for your platform.

❖ **Installing client tools on Windows**

1 Use an account with Administrator privileges to log in to Windows.

2 Exit any Windows programs running on your machine.

3 Insert the IQ Network Client CD into the CD drive.

If Network Client Setup does not start automatically:

- Click Start and choose Run.
- Browse to select your CD drive letter and choose *setup.exe*.

4 Follow the instructions on your screen to install the IQ Network Client tools.

Note If you installed Sybase Central, see the *Sybase IQ System Administration Guide* for instructions on configuring and running the IQ Agent. You must run the IQ Agent in order to use Sybase Central.

❖ **Installing client tools on Linux**

Sybase IQ Network Client for Linux contains the components required for connection to a network server, and is compatible with IQ servers on all supported server platforms. Do not install the Sybase IQ Network Client in the same directory as an IQ server.

1 Use this command to change to the installation directory:

```
% cd $RAP/RAPStore
```

2 Start the install utility sybinstall.

You can run this utility as a series of menus with prompts or bypass menus using the sybinstall command line parameters:

Parameter	Function
-add_agent	Install standalone 12.7 IQ Agent only
-autoinstall	Install all defaults
-help	Display all parameters and usage
-I_accept_sybase_license	Bypass license agreement
-info|version	Display information about this product
-y	Assume yes for all questions, warnings, and errors

3 Do one of the following:

- Run sybinstall with all menus and prompts. This allows you to choose the components you want to install:

```
% /cdrom/sybinstall
```

- Run sybinstall to install the default products:

```
% /cdrom/sybinstall -autoinstall
-I_accept_sybase_license
```

4 When the Welcome screen appears, press Return.

5 Type the number that corresponds to your location.

If you cannot read the license agreement or find a license agreement that matches your location, you can read all available license agreements on the Sybase Web site at http://www.sybase.com/softwarelicenses and rerun sybinstall, passing it the parameter `-I_accept_sybase_license`. For example:

```
% /cdrom/sybinstall -I_accept_sybase_license
```

The first screen of the Software Test and Evaluation License Agreement appears.

6 Read and accept the license agreement. As you read, hold down the Return key until you reach the end of the agreement.

The script next lists the amount of free space available in your *$RAP/*RAPStore directory and the amount of space required for the three components it installs.

7 Do one of the following:

- To install all four products, type `s`.
- To deselect or change any of the installed products, type the option number at the prompt.

8 Type `s` to start the installation.

Note An installation procedure log is created in *$RAP/*RAPStore*/sybinstall.log*. If the log file cannot be created in the *$RAP/*RAPStore directory, the path defaults to */tmp/sybinstall.log*.

❖ **Setting Linux environment variables**

After you install Sybase IQ Network Client for Linux, use these instructions to set the environment variables.

1 Open a command window.

2 Change directory to *$RAP/*RAPStore*/ASIQ-12_7* and use one of the following commands to source the environment variables:

```
% source ASIQ-12_7.csh
```

– Or –

```
% . ./ASIQ-12_7.sh
```

Managing client connections

Clients and servers communicate through a connection. For a client application to connect to a server application, the server application must be listening for the client connection request.

Adaptive Server

Clients, such as Sybase Central and Open Client, are used to access Adaptive Server. Some ASE client applications use ODBC drivers to connect to Adaptive Server. For example, PowerDesigner connects through an ODBC driver. Other third-party applications may also require an ODBC driver.

You can also install the driver separately on other client workstations on which you will be running third-party or developed products. For more information about the ODBC driver, see the *Adaptive Server Configuration Guide* for your platform and the *ODBC Driver Reference Guide* in Technical Library.

Sybase IQ

Sybase IQ requires an ODBC data source on the client computer for each database you want to access using ODBC. A data source describes how to get to data on a network. Configuration requirements are platform specific.

Windows

On Windows, use a system DSN, which offers faster performance than a file DSN. You can also share system DSNs with all users on the network.

For information on creating ODBC data sources, see the chapter "Configuring Sybase IQ" in the *Sybase IQ Installation and Configuration Guide for Windows*.

UNIX and Linux

On UNIX or Linux operating systems, ODBC data sources are held in a file named *odbc.ini*. When you create an *odbc.ini* file, you must use the long form of each identifier, for example:

```
[My Data Source]
EngineName=myserver
CommLinks=tcpip(port=1870)
Userid=DBA
Password=SQL
```

For information about creating ODBC data sources, see the chapter "Configuring Sybase IQ" in the *Sybase IQ Installation and Configuration Guide* for your server platform.

Distribute the connection information to users

If you created File DSNs, you can make them available to users on the network or distribute the files. Otherwise, make the connection information available to users, so they can create the data sources on their machines.

Starting and stopping UAF agents

UAF plug-ins running on each server handle communication between Operations Console and Sybase RAP components. The UAF agent for each Sybase RAP component must be started manually, in order for the component to be visible in and controlled by Operations Console.

When a UAF agent starts on a server, the agent starts all of its plug-ins and registers with the JINI server. Operations Console queries the JINI server to learn about all agents and their plug-ins.

Sybase recommends starting the UAF agents and the JINI server at system start-up to ensure that components can be monitored in the event of a hardware reboot.

❖ **Starting the UAF agents and their plug-ins**

1 Start the JINI server:

- Change directory to *$RAP/OpsConsole/ua/bin*
- Run the command

```
./startJini
```

2 Start the Operations Console:

- Change directory to *$RAP/OpsConsole*
- Run the command

```
startOpsConsole.sh
```

The RAPCache and RAPStore servers should already be running. See “Verify that all servers are running” on page 41.

3 Start the agent for each component that you want to see on the Operations Console. Repeat this step for each component:

- Change directory to *$RAP/<component_dir>/ua/bin*
  where *<component_dir>* is the name of the component.

For example, <*component_dir*> can be DemoFeedHandler, DemoSubscriber, FASTFeedHandler, RAPCacheSubscriber, or RAPStoreSubscriber.

- Run the command

```
./agent &
```

The & starts a background process that continues to run after you exit the terminal window.

4 Start the RAPStore agent:

- Change directory to *$RAP/*RAPStore*/ASIQ-12_7*
- Source the environment variables:

```
source ASIQ-12_7.csh
```

– *Or* –

```
../ASIQ-12_7.sh
```

- Change directory to *$RAP/*RAPStore*/ua/bin*
- Run the command

```
./agent &
```

5 Start the RAPCache agent:

- Change directory to *$RAP/*RAPCache*/*
- Source the environment variables:

```
source SYBASE.sh
```

- Change directory to *$RAP/*RAPCache*/UAF-2_0/bin*
- Run the command

```
./agent &
```

When you log into Operations Console, the Sybase RAP components whose agents are running are visible.

❖ **Stopping UAF agents**

1 Stopping a UAF agent:

- Change directory to *$RAP/*<*component_dir*>*/ua/bin*
  where <*component_dir*> is the name of the component.

For example, <*component_dir*> can be DemoFeedHandler, DemoSubscriber, FASTFeedHandler, RAPCacheSubscriber, or RAPStoreSubscriber.

- Run the command:

```
./agentd stop
```

2 Stopping Operations Console

- Change directory to *$RAP/OpsConsole/WebMonitoringConsole*.
- Run this command:

```
./stopOpsConsole.sh
```

3 Stopping JINI Server

- Find the JINI server processes

```
ps –ef | grep "ua/thirdparty"
ps –ef | grep rmid
```

- Use kill command to stop three JINI processes

```
kill <pid>
```

Installing PowerDesigner

You install PowerDesigner 12.5 on Windows from the Sybase RAP PowerDesigner 12.5 product CD.

PowerDesigner 12.5 requires a valid SySAM license.

- You must download a SySAM license file from the Sybase Product Download Center (SPDC) in order to install and run PowerDesigner.
- The SySAM license type is SS (Standalone seat)
- There are two possible license models:
  - Served: requires a license server
  - Unserved: license file is on client machine

- A license key is required for installation

Note The Sybase RAP Product DVD contains an older version of PowerDesigner. For compatibility with Sybase RAP, install PowerDesigner 12.5 from the separate RAP PowerDesigner 12.5 product CD.

For installation details, refer to the PowerDesigner 12.5 *Installation Guide*. For information on PowerDesigner licenses, see the chapter "Managing PowerDesigner Licenses" in the PowerDesigner 12.5 *General Features Guide*.

Refer to the SySAM online documentation at http://infocenter.sybase.com/help/index.jsp or the *User's Guide Sybase Software Asset Management 2.0* on the Getting Started CD for more information on the use of SySAM.

Uninstalling Sybase RAP components

Before uninstalling the RAP components, shut down all RAPCache and VLDB servers. See "Verify that all servers are running" on page 41 for information on determining the current status of your servers.

To uninstall RAP, run *$RAP/_uninst/uninstall.bin*.

Note If the *uninstall* command does not remove all Sybase RAP component directories, use the `rm -r` command to remove any remaining directories and files recursively.

CHAPTER 3

Sybase RAP in a High Availability Environment

About this Chapter

This chapter describes the roles specific components play when you deploy Sybase RAP in a High Availability environment.

Contents

Introduction

Continuous availability of financial market data is essential. High Availability and Disaster Recovery are provided in Sybase RAP by a combination of server configurations and procedures. For example, in the event the in-memory cache (ASE RAPCache) becomes unusable, ASE is synchronized with a second copy of the RAPCache to ensure automatic, timely recovery and failover of the primary cache. Another example is the redundant data streams that ensure fault tolerance, if a feed handler stream is interrupted.

For Disaster Recovery of the IQ RAPStore, Sybase IQ multiplex is used for automatic failover of the writer node from a designated reader node.

Ensuring continuous market data feed

The approach to highly available financial market data in the Sybase RAP - The Trading Edition system is the configuration of redundant components within a given Sybase RAP installation.

The incoming market data feed(s) from SIAC, NYSE, NASDAQ, and other sources have the ability to support redundant feeds, such that two or more processes can subscribe to these feeds, each of which receives identical data when all systems are running normally.

Feed handlers provided by third parties leverage this ability to support redundant feeds, and stress the necessity of providing both feed fault tolerance as well as feed handler fault tolerance.

Feed fault tolerance is provided by configuring the feed handler to subscribe to a pair of fault tolerant multicast data streams sent by the data providers (for example, SIAC and NASDAQ). If one of the data streams is interrupted, the feed handler continues to receive all of the data via the remaining data stream.

Feed handler fault tolerance is provided by implementing two feed handlers running in a hot standby mode using a heartbeat mechanism. A primary instance and a secondary instance process exactly the same data and maintain exactly the same internal state, but only the primary instance of a feed handler pair forwards market data to subscribing applications. In the event of a feed, feed handler, or feed handler hardware failure, the heartbeats cease and the secondary instance takes over as the primary instance. The new primary instance sends full record or order book images for all items that were modified just before the last heartbeat was received.

Ensuring market data consistency

The components most affected by the need to provide feed fault tolerance are the Publishers and Subscribers.

The Publisher is designed to send messages over multiple multicast channels and the Subscriber is designed to receive messages over the same channels.

By leveraging capabilities of the feed itself, or those provided by various feed handlers, Sybase RAP handles market data messages at a very high rate with a high probability of delivery. A set of multicast channels is designated as the primary channels to be used by one or more Publishers. Another set of multicast channels is designated as the secondary channels. When the primary feed handler is operational, the primary and secondary subscribers get the messages from the primary channels. If the primary feed handler fails, the secondary feed handler begins sending messages over the secondary channels to the primary and secondary subscribers. On the Subscriber side, all subscribers are configured to receive data over both primary and secondary channels.

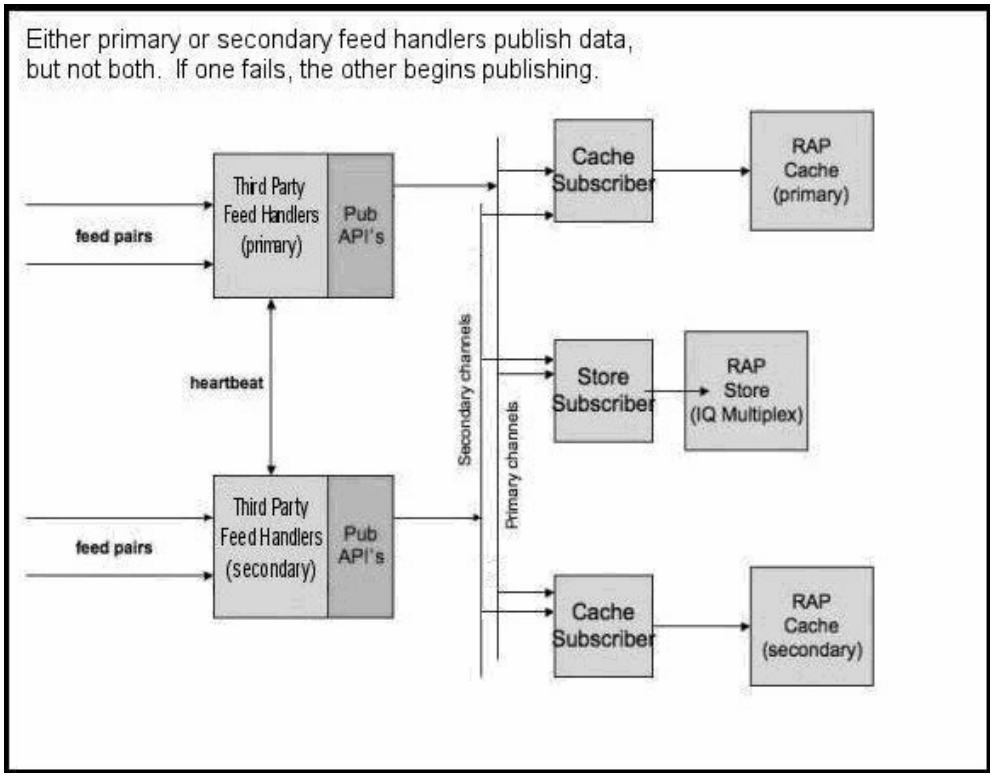
If high availability is required, you must ensure that there is no overlap between the primary and secondary channels used for multicasting.

The primary RAPCache must also have a secondary RAPCache configured, to go along with the secondary Subscriber for the RAPCache. This way, both RAPCaches contain the same data during normal operations.

The primary RAPStore (an instance of an IQ multiplex server) does not usually have a secondary RAPStore configured. The RAPStore potentially contains several years' worth of market data, making a redundant configuration impractical. Therefore, if the data in the RAPStore becomes stale due to loss of the data stream or Subscriber, the RAPStore must be loaded from data in the RAPCache. Sybase RAP provides no automatic mechanism to load the RAPStore from the RAPCache. A method to manually load the RAPStore is discussed in the section "Loading the RAPStore from the RAPCache" on page 71.

The diagram below illustrates the overall configuration which can be used to substantially enhance availability of market data to trading applications. Similar configurations are possible using the software of third party vendors.

Figure 3-1: Sybase RAP High Availability configuration



Individual High Availability component features are described at the component level in the section “Roles of Sybase RAP components” on page 64.

Roles of Sybase RAP components

All high availability requirements are met through the configuration of installed Sybase RAP components. A specific software feature set to provide a high availability mechanism is not necessary.

High availability in Sybase RAP involves a **primary data stream** and a **secondary data stream**.

The primary data stream is defined as all of the software processes, networking, and hardware necessary to capture and store financial market data obtained from one or more feeds into the primary RAPCache (ASE) and the RAPStore (Sybase IQ).

The secondary data stream is defined as all of the software processes, networking, and hardware necessary to capture and store the same financial market data into a secondary RAPCache.

All primary data stream software components configured should be installed and run on their own hardware and network; all secondary data stream software components configured should be installed and run on their own hardware and network, which is different than that of the primary data stream.

This discussion does not detail the implementation of high availability by any potential third-party providers of feed handlers. This discussion focuses on the Sybase RAP components that have a role in capturing and delivering the market data through the Sybase RAP system. These components are:

- Sybase RAP Publisher API
- RAPCache Subscriber
- FAST Feed Handler
- RAPCache
- RAPStore and RAPStore Subscriber

Sybase RAP Publisher API

The Sybase RAP Publisher API uses multicast channels to route market data messages to subscribers, who deliver the data to the final destinations in the RAPCache and RAPStore. Publishing Sybase RAP data in a high availability environment requires a primary and secondary collection of publishers. Each collection includes one or more publishers that accept data from a feed handler and broadcast messages across a set of multicast channels specific to that collection. The primary collection of Publishers broadcasts on one set of multicast channels; the secondary collection broadcasts on another set of multicast channels.

RAPCache Subscriber

The role of the RAPCache Subscriber in high availability is to ensure that no data is lost in the Sybase RAP system. On the Subscriber side, the primary channels used by the Publisher can be configured for one or more Subscribers, and a second set of Subscribers can be configured to receive the same data over the secondary channels. In this way, two identical streams of data are being published from the feed handlers into the Sybase RAP system via redundant subscribers.

RAPCache

RAPCache failover strategy requires a secondary installation of ASE to complement the secondary configuration of the RAPCache Subscriber.

A RAPCache is always associated with a Subscriber. The primary Subscriber for the RAPCache has a corresponding RAPCache installed and configured. If high data availability is required for the RAPCache, a secondary RAPCache is installed and configured. During normal operations, both RAPCaches contain the same data

The Subscriber/RAPCache pair for the primary data stream is installed and run on a single host, while the Subscriber/RAPCache pair for the secondary data stream is installed and run on a separate host, preferably using a separate network.

RAPStore and RAPStore Subscriber

Because the RAPStore may store and maintain several years' worth of market data, a redundant configuration is not practical. If the data in the RAPStore becomes stale due to loss of the data stream or Subscriber, the RAPStore data must be manually loaded from the RAPCache.

You may choose to install a complete secondary system for the RAPStore, as in the case of the RAPCache (the secondary data stream does not preclude this configuration), but the expense of a secondary RAPStore could be prohibitive, as there may be many terabytes of data.

If you configure a secondary system for the RAPStore, the RAPStore is loaded only by a primary Subscriber.

If there is a failure anywhere in the primary data stream, data in the RAPStore becomes stale until manual intervention occurs. In this case, the RAPStore data is loaded from either the primary RAPCache or the secondary RAPCache, whichever is current.

A redundant RAPStore (an instance of an IQ multiplex server) and RAPStore Subscriber should be installed on a separate machine. The RAPStore Subscriber is configured to listen on the primary data stream. The redundant RAPStore and RAPStore Subscriber do not run, unless there is a hardware failure on the machine running the RAPStore. In the event of hardware failure of the RAPStore machine, you must, at the end of the trading day, launch the redundant RAPStore, load data from the current RAPCache into the RAPStore, and use the redundant RAPStore Subscriber for the operations of the following day, until the original hardware is repaired.

FAST Feed Handler

Most providers of market data feeds, including NYSE, NASDAQ, and SIAC allow delivery of market data over separate lines, and depend on the feed handler to eliminate duplicates, when both lines are delivering data without interruption of service.

To enable high availability of market data, separate FAST feed handlers, or collections of feed handlers, are configured. One collection of FAST Feed Handlers is configured to receive data from the primary line provided by the feed, and another collection of FAST Feed handlers is configured to receive data from the secondary line provided by the feed, as described earlier in this chapter.

The primary FAST Feed Handler collection (one or more feed handlers) uses a Publisher API configured to publish data over the Sybase RAP primary multicast channels; the secondary FAST Feed Handler collection uses a Publisher API configured to publish data over the Sybase RAP secondary multicast channels.

Where to find more information

- Adaptive Server Enterprise 15.0 documentation
- Sybase IQ 12.7 documentation

Preinstallation tasks

Refer to “Installation options” on page 28 for information on minimum hardware requirements for Sybase RAP components and to the *Sybase RAP - The Trading Edition Release Bulletin* for information on operating system levels and required operating system patches.

Application connection failover

Applications using the primary RAPCache do not necessarily failover to the secondary RAPCache, if the primary data stream fails. The failure may not be with the RAPCache itself; upstream components may have failed.

For this reason, the normal Open Client (and ODBC, JDBC) connection failover mechanisms that are available to Sybase client applications cannot be relied upon, unless the primary RAPCache is shut down. Otherwise, when a failure in the primary data stream is detected, software using the primary RAPCache must be redirected to failover by other, non-automatic means.

There are several possible scenarios for minimizing the complications of connection failover:

- If the primary RAPCache is running, shut down the RAPCache server. The shutdown forces automatic client-side connection failover, if failover is configured for client applications.
- Use Sybase OpenSwitch. OpenSwitch is designed for handling this situation, but must be instructed when to make the failover, as failure of the primary data stream may not be obvious.
- Change applications to be aware of both the primary and secondary ASE installations. For existing application software, this may be impractical for the user.
- Change the *interfaces* or *sql.ini* file of the application. For hundreds of applications running on PCs or workstations, however, this may not be a practical solution.

Preinstallation tasks for OpenSwitch

Sybase OpenSwitch can provide near-continuous database access during planned downtime and unplanned outages. OpenSwitch is placed between client connections and two or more Sybase RAPCache servers. During a failover or switchover, OpenSwitch transfers access from the primary RAPCache server to the next available server in the pool.

If you use OpenSwitch as a component of your High Availability configuration, read this section to learn about tasks you perform before you install OpenSwitch.

Minimum requirements for OpenSwitch

Deploying Sybase RAP in a High Availability environment with OpenSwitch requires at least two nodes to provide an Active/Active standby capability. Sybase RAP read/write applications apply inbound transactions in two parallel streams to both the Primary and Standby RAPCache databases. After failover, data flow to the Standby database meets the same latency conditions that existed before failover.

To ensure high availability in a production environment, Sybase strongly recommends that you install OpenSwitch on a host other than the host on which Adaptive Server Enterprise and Replication Server® are installed.

Implementation options

Several configurations are possible, depending on whether you use Replication Server, whether the Replication Server is configured for warm-standby, and whether your Adaptive Servers are configured for failover.

Name	Software requirements
Simple	• 1 OpenSwitch • 1 Optional Coordination Module (CM) • 2 Adaptive Servers

Name	Software requirements
High availability, warm-standby	• 1 OpenSwitch server and 1 Replication Coordination Module (RCM) configured to coordinate failover through the OpenSwitch server OpenSwitch servers in this configuration do not communicate with each other, unlike a mutually aware configuration. • 2 Adaptive Servers configured for high availability • 1 Replication Server configured for warm-standby
Redundant high availability, warm-standby	• 2 OpenSwitch servers (1 primary and 1 secondary) • 2 RCMs configured to coordinate failover through the OpenSwitch server • 2 Adaptive Servers configured for high availability • 1 Replication Server configured for warm-standby
Mutually aware	• 2 mutually aware, companion OpenSwitch servers. Both mutually aware OpenSwitch servers within the same cluster regard each other as companions and both are aware of each other's state and the state of the other servers. • 2 CMs or 2 RCMs (optional) • 2 Adaptive Servers, which may be configured for high availability

Where to find more information

- See the *OpenSwitch Administration Guide* and the *OpenSwitch Coordination Module Reference Manual* for details about possible OpenSwitch configurations.
- See the Replication Server documentation for information about using warm-standby. See the Adaptive Server Enterprise documentation for information about high availability failover.

Install Sybase RAP core components

First use the Sybase RAP installer to install RAPCache for each server in your cluster, then install OpenSwitch.

Check the operating system requirements

OpenSwitch requires a minimum of 256MB RAM and 450MB hard disk space. The OpenSwitch installer requires 100MB free space in your temporary directory. Refer to the OpenSwitch documentation for other hardware and software requirements, including operating system versions and patch levels.

Apply any operating system patches before you install OpenSwitch. See the table in the section “Check for operating system patches” on page 26 to identify the patches on your system.

Read the OpenSwitch documentation

Deploying OpenSwitch in a High Availability environment is not difficult, but does require planning. Read the OpenSwitch *Administration Guide* to learn about OpenSwitch functionality, features, and deployment issues. Refer to the *Installation Guide* and the *OpenSwitch Coordination Module Reference Manual* for specific installation and configuration details and issues.

The *OpenSwitch Installation Guide* contains instructions for installing and configuring OpenSwitch:

- Installing OpenSwitch from the installation CD
- Configuring OpenSwitch using the configuration wizard tool or using a text editor
- Performing postinstallation tasks such as updating environment variables and verifying client connections

See the Adaptive Server Enterprise documentation for information about high availability failover.

Loading the RAPStore from the RAPCache

The primary RAPStore contains such a large amount of historical market data, that configuring a secondary RAPStore is impractical. If the data in the RAPStore becomes stale due to the loss of the data stream or Subscriber, the RAPStore must be loaded from data in the RAPCache using a manual procedure. This procedure to reload the RAPStore data from the RAPCache should be performed in the event of a failure that prevents the RAPStore from being loaded by the RAPStore Subscriber.

Perform the following steps for each table you are loading:

1 Determine the maximum key value currently in the RAPStore. The maximum key value is determined by date and time. Each market data message is stored in its corresponding database table with a date and timestamp value. Corresponding values for these key fields are a) the date today, and b) maximum timestamp value less 10 minutes. Since the messages may arrive out of order, Sybase recommends allowing a grace period of at least 10 minutes.

2 Create a view in the current RAPCache to select from the corresponding ASE table where key values are greater than the key values from RAPStore. For example, if you are transferring the STOCK_QUOTE table, you can create a view called STOCK_QUOTE_XFER using the following SQL statement:

```
create view RAP_USER.STOCK_QUOTE_XFER (
INSTRUMENT_ID, QUOTE_DATE, QUOTE_SEQ_NBR,
TRADING_SYMBOL, QUOTE_TIME, ASK_PRICE, ASK_SIZE,
BID_PRICE, BID_SIZE ) as select from STOCK_QUOTE
where QUOTE_TIME > 'datetime key value from previous
step'
```

3 Bulk copy out to a file from the view in current RAPCache, using the ASE bcp utility. Ensure that you use the following flags: -c -t , -r \\n

- The -c flag tells bcp to output character data instead of binary data.
- The -t , flag tells bcp to use the comma as a field terminator.
- The -r \\n flag tells bcp to use a newline as a row terminator.

Here is the bcp command, using the example view STOCK_QUOTE_XFER:

```
bcp RAP_USER.STOCK_QUOTE_XFER out
stock_quote_xfer.csv -c -t , -r \\n
-URAP_USER -Prap_user
-S<RAPCache Database Server Name>
```

For more information on the bcp utility, refer to the ASE 15.0 *Utility Guide* Chapter 4, “Using bcp to Transfer Data to and from Adaptive Server.”

4 Transfer the bulk copy files from the ASE RAPCache server to the Sybase IQ server hosting the RAPStore database.

5 On the Sybase IQ server, load the data into the RAPStore. Use dbisql to submit a LOAD TABLE command. Use the IGNORE CONSTRAINT UNIQUE 0 option to enable IQ to throw away any duplicate key values without terminating the load operation. For information on the LOAD TABLE command, refer to the Sybase IQ 12.7 *System Administration Guide* Chapter 7, "Moving Data In and Out of Databases."

Continuing the example, the LOAD TABLE command looks like the following:

```
load table RAP_USER.STOCK_QUOTE
(
INSTRUMENT_ID    null (blanks,'NULL') ,
QUOTE_DATE    null (blanks,'NULL') ,
QUOTE_SEQ_NBR    null (blanks,'NULL') ,
TRADING_SYMBOL null (blanks,'NULL') ,
QUOTE_TIME    null (blanks,'NULL') ,
ASK_PRICE    null (blanks,'NULL') ,
ASK_SIZE    null (blanks,'NULL') ,
BID_PRICE    null (blanks,'NULL') ,
BID_SIZE  '\x0a'
)
from '<path>/stock_quote_xfer.csv'
quotes off
escapes off
preview on
ignore constraint unique 0
;
commit
;
```

Once the LOAD TABLE command completes for every table that must be reloaded, clean up the RAPStore before restarting the RAPStore Subscriber. The RAPStore loads data using files written by the RAPStore Subscriber.

Perform the following steps to clean up the RAPStore:

1 Go to the directory where the files are located. The files are located in the directory set as the PrimaryFileLocation in the RAPStore Subscriber *rapstore.xml* configuration file.

2 For each file in the directory, issue the following command using dbisql. This command removes the file from the RAP_WORK_FILE table.

```
delete from RAP_WORK_FILE where FILE_NAME='filename'
```

3 Delete all of the files in the directory.

CHAPTER 4

Sybase RAP Configuration Files

About this Chapter

This chapter describes the elements of the Sybase RAP feed handler and publisher configuration files

Contents

Demo Feed Handler

This is the element description for *publisher.xml*.

Demo Feed Handler Elements

Element	Description
Publisher	Root element for the configuration file.
Logger	Contains settings for logging activities
LogLevel	The level of logging. Valid values are: **error:** only log errors **warning:** log warnings in addition to errors **info:** log informational messages in addition to messages logged at the warning level **debug:** log debugging messages in addition to messages logged at the info level
LogFile	The name and location of the log file. The file name can be relative or a full path.
NumMessageBuffers	The number of message buffers. One message buffer is required for each message that is being simultaneously built. This setting can have a value from 1 to 65535, though the machine must have enough memory to hold the number of buffers specified.
NumPacketBuffers	The maximum number of packets to cache in order to satisfy requests by a subscriber to resend a packet. This number of packets is cached per data stream channel. This setting can have a value from 1 to 4294967296, though the machine must have enough memory to hold the number of packets specified. This number of buffers is allocated on initialization of the Publisher.
MessageFlushInterval	This is the interval in milliseconds during which a partially filled message buffer must be idle before being sent on the network. This setting can have a value from 1 to 65535
LatencyCheckInterval	The number of seconds after which to do a latency check on a message. This setting can have a value from 1 to 65535.
AdminChannel	Information about the administration channel. This channel accepts requests for version information, statistics, and shutdown.

Element	Description
AdminPort	This is the port number used by the UAF agent to communicate with the Publisher. The Publisher will listen for incoming administration requests on this port.
ResendChannel	Information about the resend channel. This channel listens for connections from subscribers. Subscribers will open a connection to a publisher and issue requests to resend packets.
ResendPort	This is the port number used by the Subscribers to request resends of dropped network packets, and to time network latency between publisher and subscriber.
DataStreamChannelList	A list of data stream channel definitions. There can be up to 255 data stream channels.
DataStreamChannel	Contains information for one data stream channel. Each message sent by the publisher will be sent in a network packet buffer over exactly one of the defined channels. The publisher attempts to balance sending over all the channels while the system is under load.
ChannelName	A descriptive name for the channel. This name is used to identify the channel when logging.
LocalInterface	This is the IP address of a network interface on the local machine which should be used for sending data
IPAddress	This is the UDP multicast address for sending the messages over the network.
Port	The port over which messages will be sent using UDP multicasting.

FAST Feed Handler

The FAST Feed Handler contains two configuration files namely *fastfeedhandler.xml* and *publisher.xml*.

Elements for fastfeedhandler.xml

Element	Description
FASTFeedHandler	This is the root element of the configuration file.
Logger	Contains settings for logging activities

Element	Description
LogLevel	The level of logging. Valid values are **error:** only log errors **warning:** log warnings in addition to errors **info:** log informational messages in addition to messages logged at the warning level **debug:** log debugging messages in addition to messages logged at the info level
LogFile	The name and location of the log file. The file name can be relative or a full path.
FASTTemplateLibrary	This defines a FAST template library. The library must contain at least one TemplateLibraryFile tag. If duplicate FAST templates are present within the definition of the library, the definition which appears last will be used.
TemplateLibraryFile	This tag specifies an absolute or relative location of a file which is in the FAST template XML format, or the absolute or relative location of a directory containing files which are in the FAST template XML format. The tag may be repeated 1 or more times in a template library.
SessionManager	This contains information about the FAST session established between the FAST Feed handler and the source producing FAST encoded messages.
FASTDataMode	This determines whether the FAST data being received is using stream mode or block mode. Valid values are stream and block.
MessageHandler	This contains configuration parameters related to the shared library message handler plug-in.
SharedLibrary	The name of the shared library to load for the message handler.
InitFunctionName	The name of the initialization function to look up in the shared library.
FiniFunctionName	The name of the finalization function to look up in the shared library.
ProcessFunctionName	The name of the process function to look up in the shared library.
InitParams	A list of initialization parameters to be passed to the shared librarys initialization function.
InitParam	The initialization parameters name and value which will be passed to the shared librarys initialization function.
SessionConnection	This describes the FIX session connection information between the FAST Feed Handler and the source producing FAST encoded messages.

Element	Description
LocalInterface	This is the IP address of a network interface on the local machine which should be used for receiving data from the source producing FAST encoded messages.
MulticastHost	This is the UDP multicast address of the source producing FAST encoded messages.
ListenPort	This is the UDP multicast port of the source producing FAST encoded messages.

The contents of *publisher.xml* are identical to the DemoFeedHandler's *publisher.xml* file (for further details please refer to the *publisher.xml* settings in the Demo Feed Handler section) except that the logging details in the *fastfeedhandler.xml* file supersede the logging details specified in the *publisher.xml* file.

Publisher API

The contents of *publisher.xml* are identical to the DemoFeedHandler's *publisher.xml* file (for further details please refer to the *publisher.xml* settings in the Demo Feed Handler section).

Demo Subscriber

Demo Subscriber contains three configuration files located in DemoSubscriber/config folder called *opensubscriber.xml*, *messagefilter.xml* and *demodsh.xml*.

The *opensubscriber.xml* file contains publisher-subscriber communication information.

Elements for opensubscriber.xml

Element	Description
OpenSubscriber	Root element for the configuration file.
Logger	Contains settings for logging activities

Element	Description
LogLevel	The level of logging. Valid values are **error:** only log errors **warning:** log warnings in addition to errors **info:** log informational messages in addition to messages logged at the warning level **debug:** log debugging messages in addition to messages logged at the info level
LogFile	The name and location of the log file. The file name can be relative or a full path.
TemplateDir	This is the directory where the message templates are located.
DataStreamHandlerConfig	This is the file which should be used for the data stream handler configuration.
NumPacketBuffers	This is the number of network packet buffers to create per data channel. The size of one buffer is about the size of the MTU of the network interfaces being used. A larger number of packet buffers means that a larger burst of traffic can be handled without requiring resends. This setting can have a value from 1 to 4294967296.
PacketWindowSize	This determines how many packets are tracked for resend purposes. This setting can have a value from 1 to 4294967296. The value should be 50-75% of the Publishers NumPacketBuffers.
LatencyCheckInterval	This determines how often (in seconds) the network latency between Publisher and Subscriber is checked. This setting can have a value from 1 to 65535.
FlushInterval	This indicates how long a data channel can sit idle before a check is made for dropped packets. This setting can have a value from 1 to 65535
AdminChannel	Information about the administration channel. This channel accepts requests for version information, statistics, and shutdown.
AdminPort	This is the port number used by the UAF agent to communicate with the subscriber. The subscriber will listen for incoming administration requests on this port.
DataStreamChannelList	This is a list of data stream channels from which the subscriber will receive messages. Either all or none of the channels for a given publisher should be present. Channels from multiple publishers may be present. There can be up to 255 data stream channels.

Element	Description
DataStreamChannel	Contains information for one publisher data stream channel.
LocalInterface	This is the IP address of a network interface on the local machine which will receive the messages.
PublisherName	This contains a publisher name for each unique publisher. This is used to identify the publisher.
ChannelName IPAddress Port	The ChannelName, IPAddress and Port should have the same values as in the corresponding channel in the publisher. ChannelName is a descriptive name for the publisher data stream channel. IPAddress is the UDP multicast address used by the publishers data stream channel for sending the messages over the network. Port is the local port used by the publishers data stream channel to send messages over the network using UDP multicasting.
PublisherList	This contains one entry for each unique publisher being used by the subscriber. The list can contain publisher entries which are unused in the DataStreamChannel definitions, however this will consume resources.
Publisher	This contains information about each unique publisher being used by the subscriber.
PublisherName	This should correspond to the PublisherName used in the DataStreamChannel definitions
IPAddress	This is the IP address or host name of the machine hosting the publisher.
ResendPort	The ResendPort must have the same value as the publishers ResendPort definition in the publisher.xml.

Using the *messagefilter.xml* file, a subscriber can determine which messages are included or excluded. Filter rules allow exclusion of messages based on message type, or inclusion or exclusion of messages based on values within fields of the message. Only simple equality checks are performed. Exclusions based on message types are faster than checks which require field-level comparisons. The rules specified must adhere to the *messagefilter.xsd* schema. In a default out of the box installation, all the message types are included and no message filtering is performed.

Elements for messagefilter.xml

Element	Description
MessageFilter	Root element for the configuration file.
RuleList	This contains the list of message filtering rules.
Rule	This contains information about a message filtering rule. This element contains one MessageType element and zero or one FieldRule elements.
MessageType	The type of the message to include/exclude.
RuleType	This determines the message filtering rule type. Valid values are include: if the rule holds true, include the message exclude: if the rule holds true, exclude the message
FieldRule	This contains the conditions for rules on a particular message field.
FieldName	The name of the field to include/exclude when the value of the field matches FieldValue.
FieldValue	The value of the field to match. If the value is a time, the format of the value must be hh:mm:ss or hh:mm:ss.sss. If the value is a date, the format must be YYYY-MM-DD. If the value is a DateTime, then the format must be YYYY-MM-DDThh:mm:ss or YYYY-MM-DDThh:mm:ss.sss.

The *demodsh.xml* file contains configuration information specific to the Demo Subscriber.

Elements of demodsh.xml

Element	Description
DemoDSH	Root element for the configuration file. This element can either contain a FileOperationMode element or a DiscardOperationMode element but not both.
FileOperationMode	In FileOperationMode a file is created for each publisher channel in the directory specified by DirName.
DirName	Contains the name of the directory where the file is created for each publisher channel. As the messages arrive from a publisher channel, the messages are written to a file created in this directory.
DiscardOperationMode	In DiscardOperationMode the messages received from the publishers are simply ignored.

RAPCache Subscriber

The RAPCache Subscriber contains three configuration files located in RAPCacheSubscriber/config folder called *opensubscriber.xml*, *messagefilter.xml* and *rapcache.xml*.

The contents of *opensubscriber.xml* and *messagefilter.xml* are identical to the DemoSubscriber's configuration files (for further details please refer to the *opensubscriber.xml* and *messagefilter.xml* settings in the Demo Subscriber section).

The *rapache.xml* file contains configuration information specific to the RAPCache Subscriber.

Elements of rapcache.xml

Element	Description
CacheDSH	Root element for the configuration file.
CacheDB	This contains information for the RAPCache database.
LoginName	This contains the user name for connecting to the RAPCache database.
LoginPassword	This contains the password for connecting to the RAPCache database.
ServerHostName	This contains the name of the machine hosting the RAPCache database.
ServerPort	This contains the port number for the RAPCache database.
TDSPacketSize	This parameter allows the configuration of the packets sent between the RAPCache Subscriber and the RAPCache database. Valid values for TDSPacketSize are multiples of 512 between 1024 and 32K.Up to a point, larger packet sizes may result in marginally better load performance.
BulkLoadingAtrributes	This specifies the bulk loading attributes used to bulk load the messages into the RAPCache database.
BulkBatchSize	This parameter determines how many rows may be inserted in a table partition before a commit occurs.

Element	Description
BulkInsertArraySize	This parameter determines how many rows are inserted into a table partition at one time. It must be a divisor of the batch size. Larger values generally yield better performance. The memory for the bulk insert array is pre-allocated for each partition of each database table reference in the message templates.
IdleBufferWriteDelayMSec	This parameter gives the number of milliseconds that a table partition may sit idle with uncommitted data available for insert.

RAPStore Subscriber

The RAPStore Subscriber contains three configuration files located in RAPStoreSubscriber/config folder called *opensubscriber.xml, messagefilter.xml* and *rapstore.xml*.

The contents of *opensubscriber.xml* and *messagefilter.xml* are identical to the DemoSubscriber's configuration files (for further details please refer to the *opensubscriber.xml* and *messagefilter.xml* settings in the Demo Subscriber section).

The *rapstore.xml* file contains configuration information specific to the RAPStore Subscriber.

Elements of rapstore.xml

Element	Description
StoreDSH	Root element for the configuration file.
WorkDB	The WorkDB section gives the connection information for the RAPStore database and is used to persist information about files awaiting loading into the RAPStore Database.
WorkDBServer	This parameter contains the name of the machine hosting the WorkDB database.
WorkDBPort	This contains the port number for WorkDB database.
WorkDBUser	This contains the user name for connecting to the WorkDB database.

Element	Description
WorkDBPassword	This contains the password for connecting to the WorkDB database.
StoreDB	The StoreDB section gives the connection information for the RAPStore database and is the database which used to store the messages processed by Sybase RAP.
StoreDBServer	This parameter contains the name of the machine hosting the StoreDB database.
StoreDBPort	This contains the port number for StoreDB database.
StoreDBUser	This contains the user name for connecting to the StoreDB database.
StoreDBPassword	This contains the password for connecting to the StoreDB database.
PrimaryFileLocation	This parameter should point to a directory where the generated binary files (containing messages received from the publishers) can be written so that the messages can be loaded into the RAPStore database. The directory should be on a separate disk/controller from the RAPStore database devices. It should have enough free space to hold any generated binary files that have not yet been loaded during the course of the day.
OverflowFileLocation	The OverflowFileLocation is used only used when the PrimaryFileLocation is full. It should be on a separate disk/controller from both the RAPStore database devices as well as the PrimaryFileLocation.
IOBufferSizeMB	This parameter specifies the size in megabytes of each internal IO Buffer which is used to store messages received from the publishers before the messages are written out to file. Each table referenced in the message templates has NumIOBuffers buffers created. Larger buffers can result in better throughput for writing the messages to disk, but also consumes more memory.
NumIOBuffers	This parameter determines the number of IO Buffers created per table. This should have the value 3.
TargetFileSizeMB	This specifies the maximum size of a generated binary file in the file system. It should be a multiple of IOBufferSizeMB. Larger binary files require more memory in the RAPStore database and take longer to load.
IdleBufferWriteDelaySec	This parameter specifies the number of seconds data can be present in an opened and idle generated binary file before it is written out and made available for loading into the RAPStore Database.

Operations Console

The Operations Console contains a configuration file called *opsconsole.properties* located in the OpsConsole/WebMonitoringConsole/bin folder. The configurable elements of *opsconsole.properties* are as follows:

Configurable elements of ops console

Element	Description
JINIServerHost	Contains the name of the machine hosting the JINI server.
JINIServerPort	Contains the JINI server port.
MonitorStatsRefreshRate	Contains the number of seconds after which the statistics displayed on the operations console monitoring page should be periodically refreshed.
MonitorLatencyRefreshRate	Contains the number of seconds after which the latency information displayed on the operations console monitoring page should be periodically refreshed.
DatabaseDriver	Contains information about the JDBC driver used to connect to the RAPCache database.
DatabaseUrl	Contains the RAPCache database JDBC URL used to connect to the RAPCache database.
DatabaseUser	Contains the user name used to connect to the RAPCache database.
DatabasePassword	Contains the password used to connect to the RAPCache database.
RapStoreDatabaseDriver	Contains information about the JDBC driver used to connect to the RAPStore database.
RapStoreDatabaseUrl	Contains the RAPStore database JDBC URL used to connect to the RAPStore database.
RapStoreDatabaseUser	Contains the user name used to connect to the RAPStore database.
RapStoreDatabasePassword	Contains the password used to connect to the RAPStore database.

Element	Description
RapStoreCompletedFileDisplayTime	This parameter specifies the amount of time (in minutes) that a binary file (containing messages) that has been successfully loaded into the RAPStore database will be displayed on the operations console RAPStore subscriber file activity page.
RapStoreNonCompletedFileDisplayTime	This parameter specifies the amount of time (in minutes) that a binary file (containing messages) that has not been successfully loaded into the RAPStore database will be displayed on the operations console RAPStore subscriber file activity page.

Publisher/Subscriber UAF Agents

The UAF agents contain three configuration files namely *service-config.xm*l located in the ua/services/Discovery folder and *service-config.xml* located in the ua/services/RMI folder and *agent-plugin.xml* located in the ua/plugins/com.sybase.rapservice/ folder.

The *service-config.xml* file located in the ua/services/Discovery folder contains UAF agent-Operations Console communication information. The configurable elements of *service-config.xml* are as follows:

UAF-OpsConsole elements of service-config.xml

Element	Description
host	Contains the name of the machine hosting the JINI server
port	Contains the JINI server port

The *service-config.xml* file located in the ua/services/RMI folder contains the UAF agent configuration information. The configurable elements of *service-config.xml* are as follows:

UAF elements of service-config.xml

Element	Description
port	Contains the UAF agent port. This is the port used by the operations console to send administrative commands to the UAF agent.

The *agent-plugin.xml* file contains information about the UAF agent-component (publisher or subscriber) communication. The configurable elements of *agent-plugin.xml* are as follows:

UAF-Component elements of agent-plugin.xml

Element	Description
rap.service_host	Contains the name of the machine hosting the component (publisher or subscriber) communicating with the UAF agent.
rap.service_port	Contains the component administration port. This should match the AdminPort value of the publisher specified in *publisher.xml* or the subscriber specified in *opensubscriber.xml*.
rap.service_startCommand	Contains the command used to start the component
rap.service_plugindisplayname	Contains the display name for the component e.g. a value of My Component in the *agent-plugin.xml* file for the Demo Subscriber would be displayed in the operations console as Demo Subscriber (My Component)

RAPCache Database UAF Agent

The UAF agent contain three configuration files namely *service-config.xml* located in the RAPCache/UAF-2_0/services/Discovery folder and *service-config.xml* located in the RAPCache/UAF-2_0/services /RMI folder and *agent-plugin.xml* located in the RAPCache/UAF-2_0/plugins/com.sybase.ase folder.

The configurable elements of the two *service-config.xml* files are identical to the values of the two *service-config.xml* files specified in the Publisher/Subscriber UAF agents section.

The *agent-plugin.xml* file contains information about the UAF agent-RAPCache database communication. The configurable elements of *agent-plugin.xml* are as follows:

UAF-RAPCache database communication

Element	Description
ase.server.name	Contains the name of the machine hosting the RAPCache database
ase.home	Contains the location of the RAPCache installation
ase.start.command	Contains the command used to start the RAPCache database.
ase.port	Contains the RAPCache database port.
ase.user	Contains the user name used to connect to the RAPCache database.
ase.password	Contains the password used to connect to the RAPCache database.

RAPStore Database UAF Agent

The UAF agent contain three configuration files namely *service-config.xml* located in the RAPStore/ua/services/Discovery folder and *service-config.xml* located in the RAPStore/ua/services /RMI folder and *agent-plugin.xml* located in the RAPStore/ua/plugins/com.sybase.rap.iq folder.

The configurable elements of the two *service-config.xml* files are identical to the values of the two *service-config.xml* files specified in the Publisher/Subscriber UAF agents section.

The *agent-plugin.xml* file contains information about the UAF agent-RAPStore database communication. The configurable elements of *agent-plugin.xml* are as follows:

UAF-RAPStore database communication

Element	Description
rap.iq_host	Contains the name of the machine hosting the RAPStore database.
rap.iq_port	Contains the RAPStore database port.
rap.iq_username	Contains the user name used to connect to the RAPStore database.
rap.iq_password	Contains the password used to connect to the RAPStore database.
rap.iq_startCommand	Contains the command used to start the RAPStore database.
rap.iq_stopCommand	Contains the command used to stop the RAPStore database.

APPENDIX A

Custom Setup Options

Refer to this appendix to review the platform component configuration options you may encounter when you run a Custom setup.

Overview

This appendix is intended as a reference for Custom installs and displays all configuration dialogs and configuration options. Custom installs provide the most flexibility for production environments.

Custom installer actions

During installation, the installer:

- Prompts you for the name of the JINI server host. The JINI server is installed on the same machine where you install Operations Console. If you install multiple components at the same time, the installer only prompts you for the JINI server host name once. This value is inherited by the other components you are installing during that installation sequence. When you complete an installation sequence on one machine and restart the installer to install components on another, you may be prompted to specify the location of the JINI server.
- Assigns default port values in the `3000 – 4000` range to all publisher components. Port values are validated during installation, and if a defined port is already in use, the installer displays two popup message boxes. The first message box tells you that the port number is already in use and an alternative port will be selected. The second message box then provides the new port assignment. If your Network Administrator chooses a different set of port values, records these items on your worksheet, and replace the default value with these items.
- Assigns *224.0.2.0* as the default class D multicast IP addresses to all publishers. This value appears in the Broadcast IP Address field and identifies the multicast channel that the publisher uses to multicast messages to subscribers. If your Network Administrator chooses another multicast address for your site, record the address on your worksheet, and replace the default with that value.
- Assigns default Channel names to Feed Handlers and Publishers. Default channels names depend on the component. For FAST Feed Handler, the first default channel name is `FAST Channel 1`; the first default channel name for Demo Feed Handler is `Demo Feed 1`; and the first default channel name for the Publisher API is `Custom Channel 1`. If you want to use different names to identify your data channels, replace the default value with the one you want to use.
- Lets you install the RAPCache Database Server and RAPCache Subscriber separately. Running this kind of install installs the RAPCache Database Server on the target machine with ASE 15.0.2 defaults, and sets the host name to the install machine. If you install the RAPCache Subscriber at the same time, this component inherits all appropriate server connection settings, and prompts you for the appropriate data channel(s) and publisher configuration. If you install the RAPCache Subscriber separately you must configure the data channel(s) and publisher and specify the location of the RAPCache Database Server.

- Lets you install the RAPStore Database Server and RAPStore Subscriber separately. Running this kind of install installs the RAPStore Database Server on the target machine with Sybase IQ 12.7 ESD #3 defaults, and set the host name to the install machine. If you install the RAPStore Subscriber at the same time, this component inherits all appropriate server connection settings, but prompts you for the appropriate data channel(s) and publisher configuration as well as the location of the RAPStore Database Server.
- Assigns all appropriate port values for Operations Console. If you install Operations Console at the same time you install RAPCache and RAPStore, the installer will populate the appropriate Host and port values for these items. If not, the Host field(s) will be left empty and the port will be the appropriate default for the database (5000 for RAPCache and 2638 for RAPStore).

Demo Feed Handler

Demo Feed Handler is a Sybase feed handler that reads comma-delimited market data from a demo file (*STOCK_QUOTE.csv)*. This component is intended to help developers become familiar with the Publisher API and demonstrate network message flow.

Architecture

Demo Feed Handler is an executable that reads messages from a demo file, and sends those messages to a subscriber.

Syntax

To run Demo Feed Handler from the command line:

```
demofeedhandler -m <messageFile> -n <numMessages> -s
<maximum messages per second>
```

- Include the full path and file name of the demo data file as the `messageFile` argument.
- Use a positive integer (any number from *1* to over *4 billion*) as the numMessages argument to process a specific number of messages.

- `-s <maximum messages per second>` is an optional switch that allows the user to simulate various scenarios. If absent, messages will be sent at the maximum possible speed.
- Specify *-1* to continuously loop through the file, until the user chooses to stop it.

Note Although Sybase RAP includes *STOCK_QUOTE.csv* that you can use as a test message file for Demo Feed Handler, you can use *.csv* file that contains Stock Quote messages will work.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- Demo Feed Handler configuration dialog 1 sets the UAF agent listening port that Demo Feed Handler uses to communicate with Operations Console. This dialog also lets you set the component name.
- Demo Feed Handler configuration dialog 2 prompts you to identify the data channels the Demo Feed Handler uses to send information to subscribers.
- Demo Feed Handler configuration dialog 3 sets different listening ports that Demo Feed Handler uses to listen for Operations Console administrative requests and resend requests for subscribers. This dialog also lets you set some performance options.
- Demo Feed Handler configuration dialog 4 prompts you for the JINI server name that Operations Console uses to track RAP system components.

Demo Feed Handler configuration dialog 1

Operations Console communicates with Demo Feed Handler through an intermediary agent. Use this dialog to set the UAF agent listening port and optional UAF agent component name.

Table A-1: Demo Feed Handler dialog 1 configuration options

Field name	Default value	Description
Listening Port	3001	Identifies the listening port that the UAF agent uses to communicate with Operations Console. You can use any free port on your system.
Component Name	None	Identifies this component. This name appears in the component's panel on the Operations Console Monitor. Although the Component Name is optional, naming the component provides an identity that distinguishes this component from other components of the same type. Projects may include several subscribers or publishers. Assigning unique component names can help you track each item more easily.

Demo Feed Handler configuration dialog 2

Demo Feed Handler multicasts messages to subscribers over one or more data channels. Four channels are recommended for a system that handles 300,000 messages per second. Use this dialog to identify the data channels.

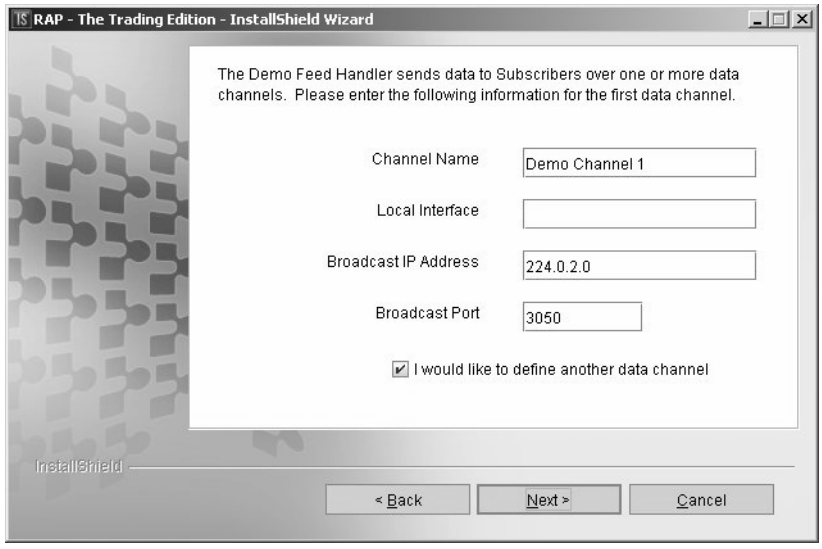


Table A-2: Demo Feed Handler dialog 2 configuration options

Field name	Default value	Description
Channel Name	Demo Channel 1	Identifies the data channel that Demo Feed Handler uses to send messages to a subscriber. Although the installer assigns Demo Channel 1 to the first data channel by default, you can use any value as the channel name. You must supply a Channel Name to any subsequent data channels you define. Assigning unique names to channels can help you track each one more easily in the event that an error is logged.
Local Interface	None	Identifies the IP address of the network interface to which this channel will be bound. This allows a you to bind various data channels to separate network interfaces to enhance performance. • If you plan to install all publishers and subscribers on the same server, you must configure the Local Interface to use the loopback interface (127.0.0.1). • If you install publishers and subscribers on different servers, use the local machine address of the secondary Ethernet network card installed by the system administrator.

Field name	Default value	Description
Broadcast IP Address	224.0.2.0	Identifies the Class D multicast IP address that Demo Feed Handler uses to multicast messages to subscribers. This value should match the multicast address that the subscribers use to receive messages. Although the installer assigns 224.0.2.0 as a default multicast IP address to all Broadcast IP address fields, your Network Administrator can choose an alternate multicast address. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	3050	Identifies the local port on which this data channel will operate. Publishers can multicast messages to a subscriber on multiple data channels. Data channels can share the same local interface and Broadcast IP Address values, but the Broadcast Port must be unique for each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to display a blank data channel dialog.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

Demo Feed Handler configuration dialog 3

Data Feed Handler listens for administrative requests from Operations Console and resend requests from subscribers. Use this dialog to set different listening ports and performance options.

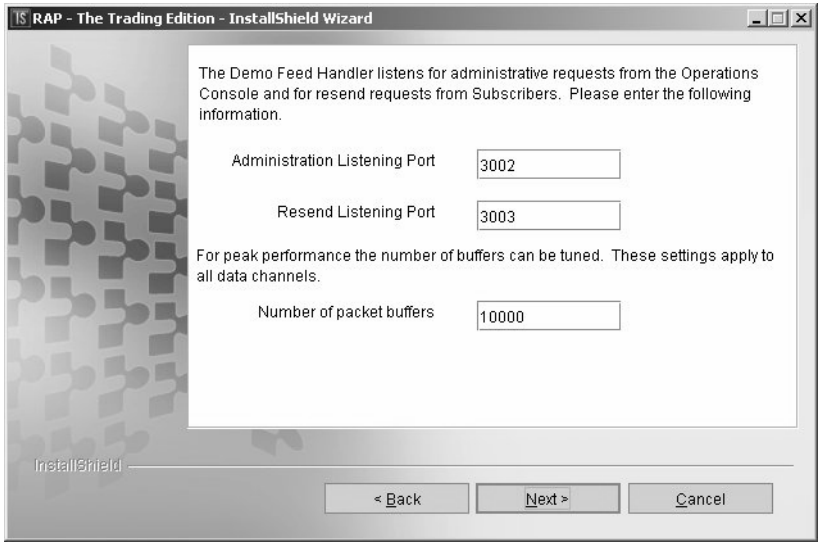


Table A-3: Demo Feed Handler dialog 3 configuration options

Field name	Default value	Description
Administration Listening Port	3002	Identifies the listening port that the publisher uses to monitor administrative requests from Operations Console.
Resend Listening Port	3003	Identifies the port that the publisher uses to resend cached message packets to the subscriber. Publishers maintain a cache of recently sent message packets to ensure that the subscriber processes all message deliveries.
Number of packet buffers	10000	Identifies the number of packet buffers. The publisher holds onto arriving packets of messages until the packets are processed. The number of packet buffers is per data stream. This setting can have a value between 1 and 4 billion, but the machine must have enough memory to hold the number of packets specified. This setting should be adjusted if resend requests from publishers are failing, because the packet no longer exists at the publisher. For publishers, a value of 10000 will usually provide sufficient packet history to maintain data integrity under moderate loads.

Demo Feed Handler configuration dialog 4

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog prompts you for the JINI server host name, which is the same host machine where you install Operations Console.You must identify the JINI Server Host Name to register Demo Feed Handler with the JINI Server.

Table A-4: Demo Feed Handler dialog 4 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed. You must identify the JINI Server Host Name to register this component with the JINI Server.

FAST Feed Handler

FIX Adapted for Streaming (FAST) protocol is a binary encoding method for message oriented data streams. The FAST Feed Handler reads FAST encoded data directly from the wire, decodes the data, and forwards the messages to the Publisher API.

Note The Sybase RAP FAST Feed Handler supports FAST SCP 1.1

Configuration

If you choose this option, the installer displays several configuration dialogs.

- FAST Feed Handler configuration dialog 1 sets the UAF agent listening port that FAST Feed Handler uses to communicate with Operations Console. This dialog also lets you set the component name.
- FAST Feed Handler configuration dialog 2 prompts you for the FAST parameters.
- FAST Feed Handler configuration dialog 3 prompts you for the data channels that FAST Feed Handler uses to send messages to subscribers.
- FAST Feed Handler configuration dialog 4 sets different listening ports that FAST Feed Handler uses to listen for Operations Console administrative requests. This dialog also lets you set some performance options.
- FAST Feed Handler configuration dialog 5 prompts you for the host name of the JINI server that FAST Feed Handler uses to track RAP system components.

FAST Feed Handler configuration dialog 1

Operations Console communicates with the FAST Feed Handler through an intermediary UAF agent. Use this dialog to set the UAF agent listening port and optional UAF agent component name.

Table A-5: FAST Feed Handler dialog 1 configuration options

Field name	Default value	Description
Listening Port	3101	Identifies the listening port that the UAF agent uses to communicate with Operation Console. You can use any free port on your system.
Component Name	None	Identifies this component. This name appears in the component's panel on the Operations Console Monitor. Although the Component Name is optional, naming the component provides an identity that distinguishes this component from other components of the same type. Projects may include several subscribers or publishers. Assigning unique component names can help you track each item more easily.

FAST Feed Handler configuration dialog 2

FAST Feed Handler listens for FAST messages sent by a FAST feed. This dialog prompts you for the FAST feed parameters.

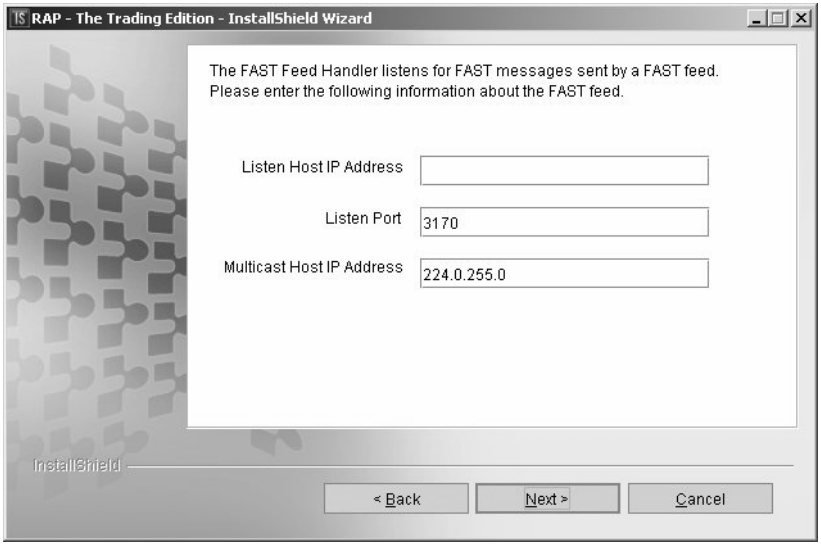


Table A-6: FAST Feed Handler dialog 2 configuration options

Field Name	Default value	Description
Listen Host IP Address	None	Identifies the IP address for a network interface to which the incoming feed's UDP channel will be bound. This allows optimization of network resources, by allowing various channel loads to be spread out among multiple physical network adapters.
Listen Port	3170	Identifies the UDP Local listening port for incoming messages.
Multicast Host IP Address	224.0.255.0	Identifies the class D (multicast) IP address on which this data channel will operate. A publisher can multicast messages to a subscriber on multiple channels, and the combination of the Broadcast IP Address and Broadcast Port must be unique for each channel. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.

FAST Feed Handler configuration dialog 3

FAST Feed Handler multicasts messages to subscribers over one or more data channels. Four channels are recommended for a system that handles 300,000 messages per second. Use this dialog to identify the data channels.

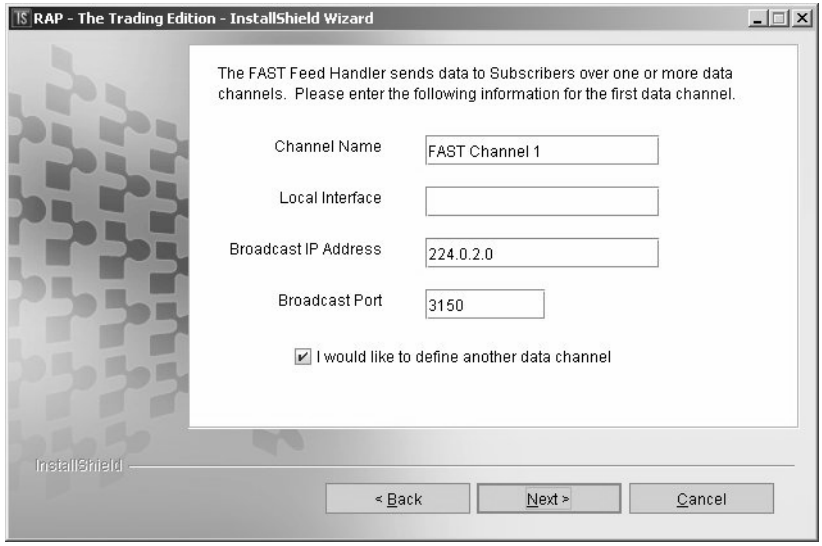


Table A-7: FAST Feed Handler dialog 3 configuration options

Field name	Default value	Description
Channel Name	FAST Channel 1	Identifies the data channel that FAST Feed Handler uses to send messages to a subscriber. Although the installer assigns FAST Channel 1 to the first data channel by default, you can use any value as the channel name. You must supply a Channel Name to any subsequent data channels you define. Assigning unique names to channels can help you track each one more easily in the event that an error is logged.
Local Interface	None	Identifies the IP address of the network interface to which this channel will be bound. This allows a you to bind various data channels to separate network interfaces to enhance performance. • If you plan to install all publishers and subscribers on the same server, you must configure the Local Interface to use the loopback interface (127.0.0.1). • If you install publishers and subscribers on different servers, use the local machine address of the secondary Ethernet network card installed by the system administrator.

Field name	Default value	Description
Broadcast IP Address	224.0.2.0	Identifies the Class D multicast IP address that FAST Feed Handler uses to multicast messages to subscribers. This value should match the multicast address that the subscriber uses to receive messages. Although the installer assigns 224.0.2.0 as a default multicast IP address to all Broadcast IP address fields, your Network Administrator can choose an alternate multicast address. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	3150	Identifies the local port on which this data channel will operate. Publishers can multicast messages to a subscriber on multiple data channels. Data channels can share the same local interface and Broadcast IP Address values, but the Broadcast Port must be unique for each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to define another data channel.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

FAST Feed Handler configuration dialog 4

FAST Feed Handler listens for administrative requests from Operations Console and resend requests from subscribers. Use this dialog to set different listening ports and performance options.

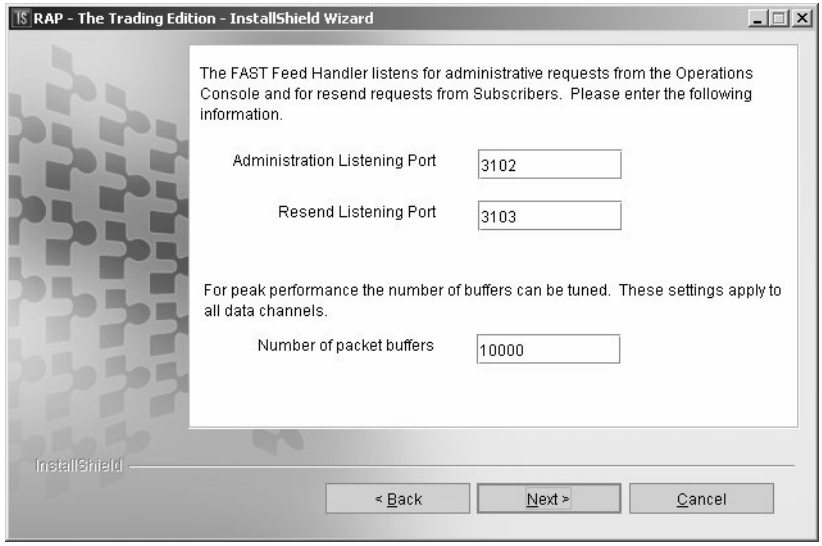


Table A-8: FAST Feed Handler dialog 4 configuration options

Field name	Default value	Description
Administration Listening Port	3102	Identifies the listening port that the publisher uses to monitor administrative requests from Operations Console. These requests can include shutdown, restart, and refresh commands.
Resend Listening Port	3103	Identifies the port that the publisher uses to resend cached message packets to the subscriber. Publishers maintain a cache of recently sent message packets to ensure that the subscriber processes all message deliveries.
Number of packet buffers	10000	Identifies the number of packet buffers. The publisher holds onto arriving packets of messages until the packets are processed. The number of packet buffers is per data stream. This setting can have a value between 1 and 4 billion, but the machine must have enough memory to hold the number of packets specified. This setting should be adjusted if resend requests from publishers are failing, because the packet no longer exists at the publisher. For the publisher, a value of 10000 will usually provide sufficient packet history to maintain data integrity under moderate loads.

FAST Feed Handler configuration dialog 5

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register the Publisher API with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

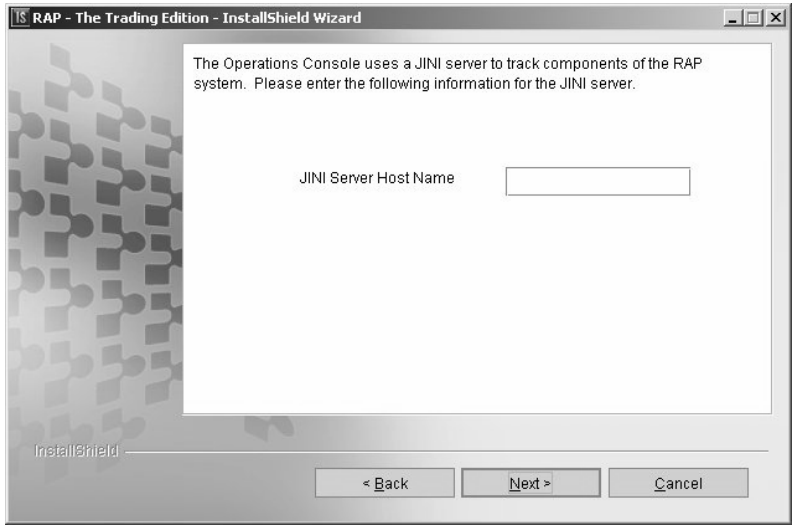


Table A-9: FAST Feed Handler dialog 5 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

Publisher API

The Publisher API formats incoming market data messages, and forwards those messages to a subscriber. This API is used by feed handlers (Demo Feed Handler and FAST Feed Handler), and developers who want to map proprietary market data into Sybase RAP.

Installation options
An initial installation dialog gives you the option of configuring the Publisher API during the install or later. If you configure later, the installer installs the files, but leaves the configuration settings at their default values. You must manually configure these settings. See Publisher API in the *Sybase RAP - The Trading Edition Developers Guide* for more information.

Configuration

If you choose this option, the installer displays several configuration dialogs:

- Publisher API configuration dialog 1 sets the UAF agent listening port that the Publisher API uses to communicate with Operations Console. This dialog also lets you set the component name.
- Publisher API configuration dialog 2 identifies the data channels that the Publisher API uses to send messages to subscribers.
- Publisher API configuration dialog 3 sets different listening ports that the Publisher API uses to listen to Operation Console administrative requests and resend requests from subscribers. This dialog also lets you set some performance options.
- Publisher API configuration dialog 4 prompts you for the host name of the JINI server that Operations Console uses to track RAP system components.

Publisher API configuration dialog 1

Operations Console communicates with the Publisher API through an intermediary UAF agent. Use this dialog to set the UAF agent listening port and optional UAF component name.

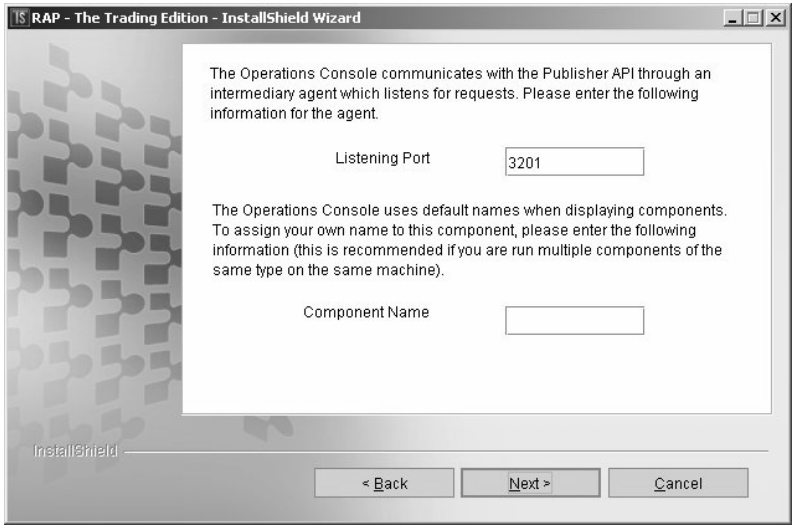


Table A-10: Publisher API dialog 1 configuration options

Field name	Default value	Description
Listening Port	3201	Identifies the listening port that the UAF agent uses to communicate with Operation Console. You can use any free port on your system.
Component Name	None	Identifies this component. This name appears in the component's panel on the Operations Console Monitor. Although the Component Name is optional, naming the component provides an identity that distinguishes this component from other components of the same type. Projects may include several subscribers or publishers. Assigning unique component names can help you track each item more easily.

Publisher API configuration dialog 2

The Publisher API multicasts messages to subscribers over one or more data channels. Four channels are recommended for a system that handles 300,000 messages per second. Use this dialog to identify the data channels.

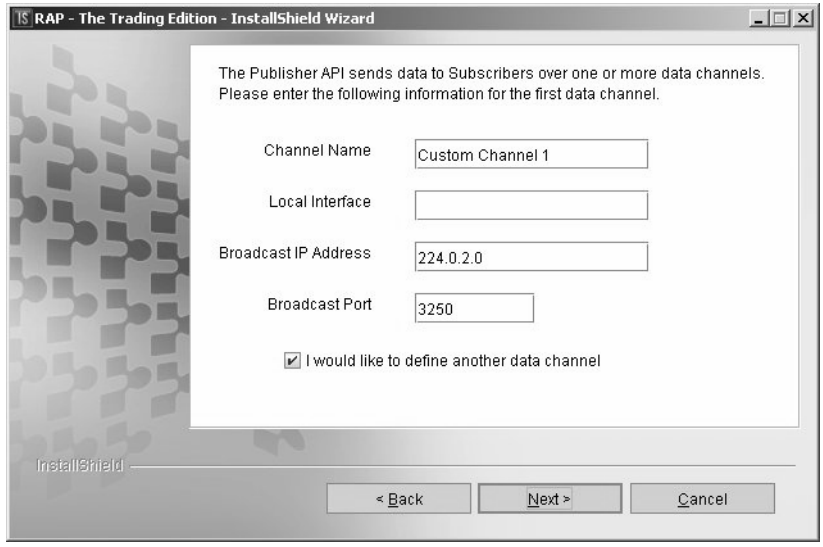


Table A-11: Publisher API dialog 2 configuration options

Field name	Default value	Description
Channel Name	Custom Channel 1	Identifies the data channel that Publisher API uses to send messages to a subscriber. Although the installer assigns Custom Channel 1 to the first data channel by default, you can use any value as the channel name. You must supply a Channel Name to any subsequent data channels you define. Assigning unique names to channels can help you track each one more easily in the event that an error is logged.
Local Interface	None	Identifies the IP address of the network interface to which this channel will be bound. This allows a user to bind various data channels to separate network interfaces to enhance performance. • If you plan to install all publishers and subscribers on the same server, you must configure the Local Interface to use the loopback interface (127.0.0.1). • If you install publishers and subscribers on different servers, use the local machine address of the secondary Ethernet network card installed by the system administrator.

Field name	Default value	Description
Broadcast IP Address	224.0.2.0	Identifies the Class D multicast IP address that the Publisher API uses to multicast messages to subscribers. This value should match the multicast address that the subscribers use to receive messages. Although the installer assigns 224.0.2.0 as a default multicast IP address to all Broadcast IP address fields, your Network Administrator can choose an alternate multicast address. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	3250	Identifies the local port on which this data channel will operate. Publishers can multicast messages to a subscriber on multiple data channels that share the same local interface and Broadcast IP Address, but the Broadcast Port must be unique to that data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to display a blank data channel dialog.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

Publisher API configuration dialog 3

The Publisher API listens for administrative requests from Operations Console and resend requests. Use this dialog to set different listening ports and performance options.

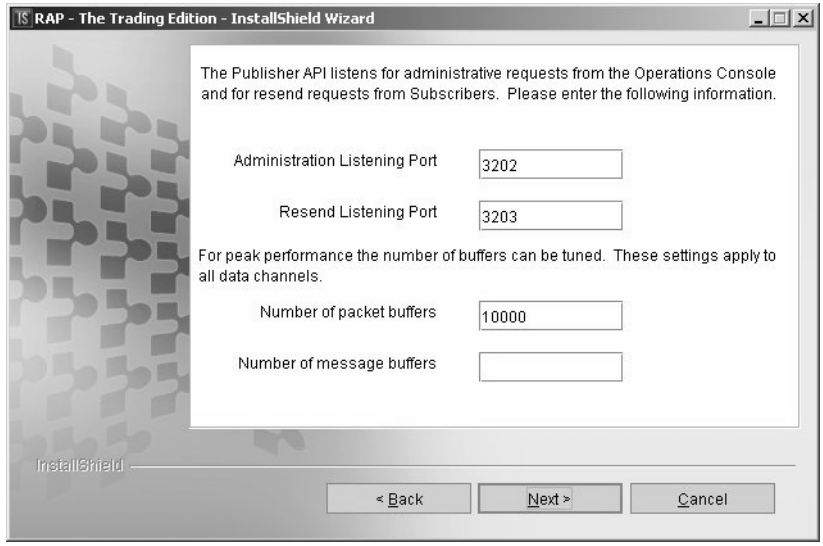


Table A-12: Publisher API dialog 3 configuration options

Field name	Default value	Description
Administration Listening Port	3202	Identifies the listening port that the publisher uses to monitor administrative requests from Operations Console. These requests can include shutdown, restart, and refresh commands.
Resend Listening Port	3203	Identifies the port that the publisher uses to resend cached message packets to the subscriber. Publishers maintain a cache of recently sent message packets to ensure that the subscriber processes all message deliveries.
Number of packet buffers	10000	Identifies the number of packet buffers. The publisher holds onto arriving packets of messages until the packets are processed. The number of packet buffers is per data stream. This setting can have a value between 1 and 4 billion, but the machine must have enough memory to hold the number of packets specified. This setting should be adjusted if resend requests from publishers are failing, because the packet no longer exists at the publisher. For publishers, a value of 10000 will usually provide sufficient packet history to maintain data integrity under moderate loads.

Field name	Default value	Description
Number of message buffers	None	Identifies the number of message buffers the publisher temporarily stores to track data deliveries. One message buffer is required for each message that is being simultaneously built. This setting can have a value from 1 to 65535, though the machine must have enough memory to hold the number of buffers specified

Publisher API configuration dialog 4

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register the Publisher API with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

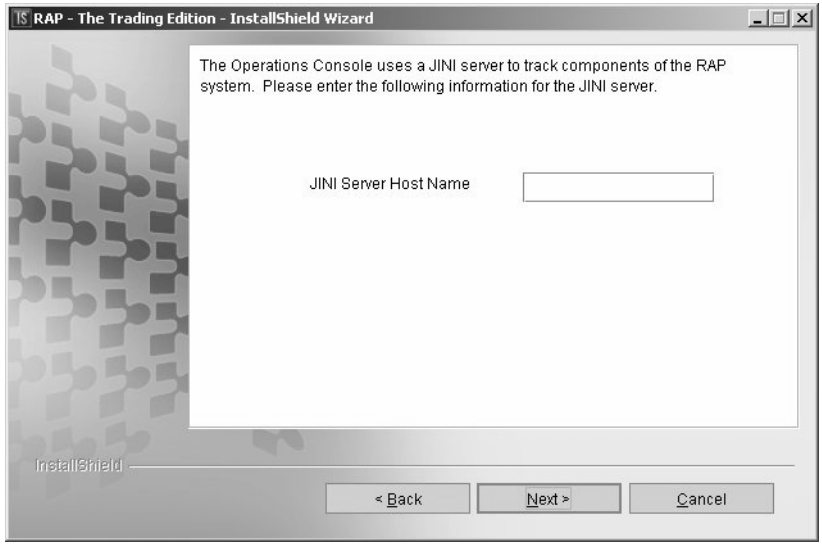


Table A-13: Publisher API dialog 4 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

RAPCache System

The RAPCache System includes two options: RAPCache Database Server and RAPCache Subscriber.

RAPCache Database Server

Choosing this option silently installs ASE 15.0.2, which creates the cache database objects, and loads the sample data. This option also creates the tables and default user accounts for Operations Console.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- RAPCache Database Server configuration dialog 1 sets the UAF agent listening port that the RAPCache Database Server uses to communicate with Operations Console.
- RAPCache Database Server configuration dialog 2 prompts you for the RAPCache Server settings.
- RAPCache Database Server configuration dialog 3 prompts you for the host name of the JINI server that Operations Console uses to track RAP system components.

RAPCache Database Server configuration dialog 1

Operations Console communicates with the RAPCache Database through an intermediary UAF agent. Use this dialog to set the UAF agent listening port.

Table A-14: RAPCache Database Server dialog 1 configuration options

Field Name	Default value	Description
Listening Port	3301	Identifies the listening port that the UAF agent uses to communicate with Operations Console. You can use any free port on your system.

RAPCache Database Server configuration dialog 2

This dialog prompts you for the RAPCache Server settings. Any initial values reflect ASE 15.0.2 defaults.

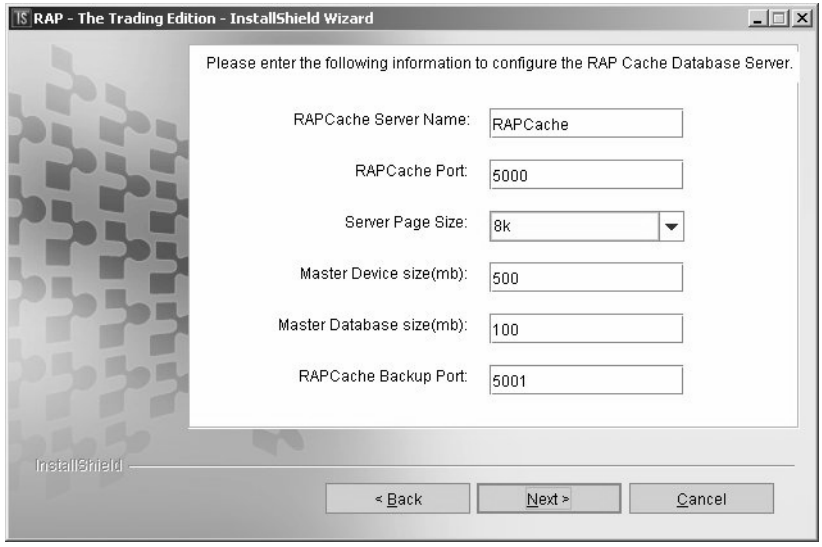


Table A-15: RAPCache Database Server dialog 2 configuration options

Field name	Default value	Description
RAPCache Server Name	RAPCache	Identifies the RAPCache server name.
RAPCache Port	5000	Identifies the port that the RAPCache Server uses to service connection requests. Although the default port is 5000, you can specify any free port.
Server Page Size	8K	Identifies the page size that Adaptive Server assigns to all database and database objects on the server.
Master Device Size	500MB	Identifies the master device size in megabytes (MB).
Master Database Size	100MB	Identifies the master database size in megabytes (MB). The master database contains Operations Console database tables and objects.
RAPCache Backup Port	5001	Identifies the port number that the backup server uses to handle network connection requests. The default value is 5001

RAPCache Database Server configuration dialog 3

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register the RAPCache Database Server with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

Table A-16: RAPCache Database Server dialog 3 configuration options

Field Name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

RAPCache Subscriber

RAPCache Subscriber processes messages from publishers that arrive over multiple data channels, then loads the RAPCache database.

Note If you install the RAPCache Subscriber at a different time than the RAPCache Database Server, the installer will prompt you for the server's location. The RAPCache Subscriber must be installed on the same machine as the RAPCache Database Server.

Architecture

RAPCache Subscriber queues up messages that are consumed by a data stream handler, which routes the messages to their final destination.

Subscribers receive message packets from publishers on multiple data streams. When a packet arrives, the subscriber records that the packet arrived. This record is used to identify any missing packets. If a packet is missing, the subscriber requests a resend from the publisher for that specific packet.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- RAPCache Subscriber configuration dialog 1 sets the UAF agent listening port that the RAPCache Subscriber uses to communicate with Operations Console. This dialog also lets you set the component name.
- RAPCache Subscriber configuration dialog 2 is an optional dialog that only appears if you install the RAPCache Subscriber separately from the RAPCache Database Server. Use this dialog to set the connection parameters to the RAPCache Database Server.
- RAPCache Subscriber configuration dialog 3 prompts you for the data channels that RAPCache Subscriber monitors for message deliveries from publishers.
- RAPCache Subscriber configuration dialog 4 sets the listening ports that RAPCache Subscribers uses to listen for Operations Console administrative requests. This dialog also lets you set some performance options.
- RAPCache Subscriber configuration dialog 5 prompts you for the publishers that communicate with RAPCache Subscriber.

- RAPCache Subscriber configuration dialog 6 is an optional dialog that prompts you for the server's location if you install the RAPCache Subscriber at a different time than the RAPCache Database Server.
- RAPCache Subscriber configuration dialog 7 prompts you for the host name of the JINI server that tracks RAP system components.

RAPCache Subscriber configuration dialog 1

Operations Console communicates with the RAPCache Subscriber through an intermediary UAF agent. Use this dialog to set the UAF agent listening port and optional UAF component name.

Table A-17: RAPCache Subscriber dialog 1 configuration options

Field name	Default value	Description
Listening Port	3401	Identifies the listening port that the UAF agent uses to communicate with Operations Console. You can use any free port on your system.
Component Name	None	Identifies this component. This name appears in the component's panel on the Operations Console Monitor. Although the Component Name is optional, naming the component provides an identity that distinguishes this component from other components of the same type. Projects may include several subscribers or publishers. Assigning unique component names can help you track each item more easily.

RAPCache Subscriber configuration dialog 2

This is an optional dialog that only appears if you install the RAPCache Subscriber separately from the RAPCache Database Server. Use this dialog to set the connection parameters to the RAPCache Database Server.

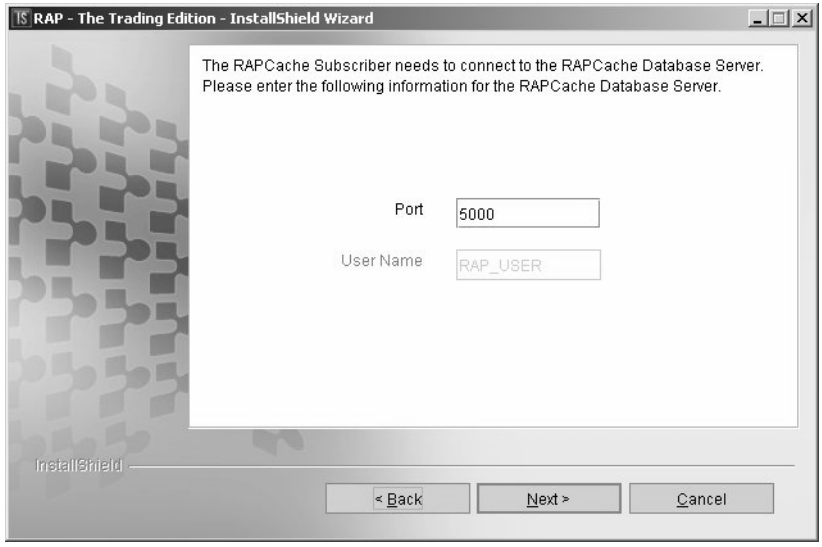


Table A-18: RAPCache Subscriber dialog 2 configuration options

Field name	Default value	Description
Port	5000	Identifies the port that RAPCache uses to service connection requests. This value must match the port you defined when you installed the RAPCache Server.
User Name	RAP_USER	Identifies the default Sybase RAP user, RAP_USER. You may not change this value.

Note If you install the RAPCache server and RAPCache subscriber as a group, this dialog inherits the server settings you defined.

RAPCache Subscriber configuration dialog 3

RAPCache Subscriber listens to one or more data channels for data sent by publishers. Use this dialog to identify each channel that you want RAPCache Subscriber to monitor.

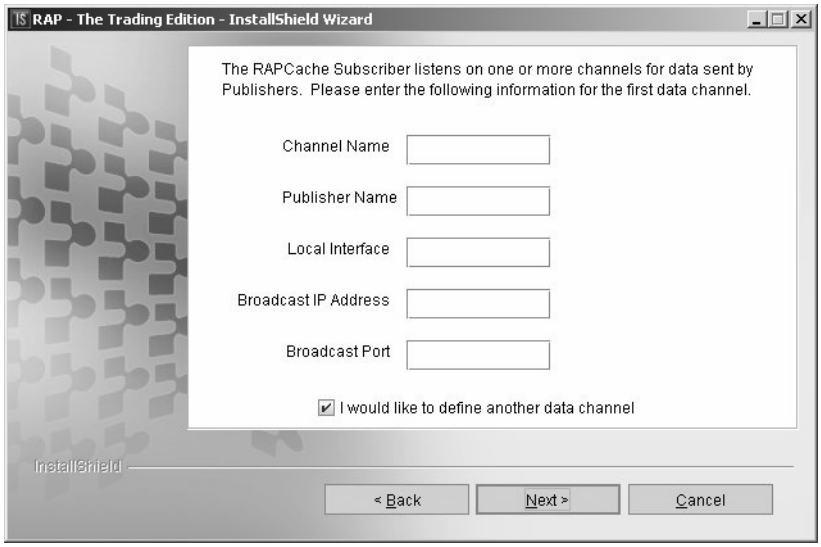


Table A-19: RAPCache Subscriber dialog 3 configuration options

Field name	Description
Channel Name	Identifies a specific data channel that a particular publisher uses to multicast messages to RAPCache Subscriber. The Channel Name depends on the data channel you want RAPCache Subscriber to monitor. If, for example, you want to RAPCache Subscriber to monitor FAST Feed Handler, and kept all the defaults when you defined the FAST Feed data channels, you would use FAST Channel 1 as the first Channel Name. Subsequent channel names would match those of other data channel names.
Publisher Name	Identifies the publisher that sends messages to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.

Field name	Description
Local Interface	Identifies the IP address of the network interface that the publisher uses to multicast messages to RAPCache Subscriber. This value must match the local interface value that the publisher uses to multicast messages. If you want RAPCache Subscriber to monitor a particular FAST Feed Handler data channel, for example, the local interface value you use here must match the local interface value for the data channel you defined for FAST Feed Handler.
Broadcast IP Address	Identifies the Class D multicast IP address that the publisher uses to multicast messages to RAPCache Subscriber. This value must match the multicast address that the publisher uses to multicast messages to RAPCache Subscriber. If you want RAPCache Subscriber to monitor a FAST Feed Handler data channel that multicasts messages on 224.0.2.0, for example, you would use that address for this field. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	Identifies the port from which the publisher multicasts messages to RAPCache Subscriber. Publishers can multicast messages to a RAPCache Subscriber on multiple data channels that share the same local interface and Broadcast IP Address, but the Broadcast Port must be unique to each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to define another data channel.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

RAPCache Subscriber configuration dialog 4

RAPCache Subscriber listens for administrative requests from Operations Console. Use this dialog to set the administrative listening port and some performance options.

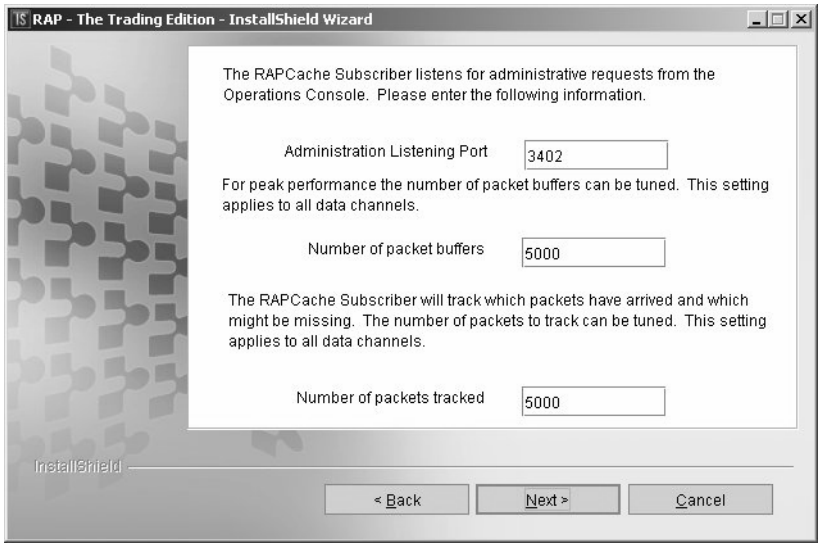


Table A-20: RAPCache Subscriber dialog 4 configuration options

Field name	Default value	Description
Administration Listening Port	3402	Identifies the listening port that the subscriber uses to monitor administrative requests from Operations Console.
Number of packet buffers	5000	Identifies the number of packet buffers. The subscriber holds onto arriving packets of messages until the packets are processed. The number of packet buffers is per data stream. This setting can have a value between 1 and 4 billion, but the machine must have enough memory to hold the number of packets specified. This setting should be adjusted if resend requests from publishers are failing, because the packet no longer exists at the publisher. For subscribers, a value of 500 will usually provide sufficient buffering to handle small bursts of high-volume data.
Number of packets tracked	5000	Identifies the number of packets the subscriber tracks. Subscribers track the packets delivered by a publisher. If a subscriber suspects that a packet may be missing, it issues a resend request to the publisher to retrieve the missing data. The number of packets tracked is for each data stream.

RAPCache Subscriber configuration dialog 5

RAPCache Subscriber communicates with one or more publishers. Use this dialog identify the publisher(s) you want RAPCache Subscriber to monitor.

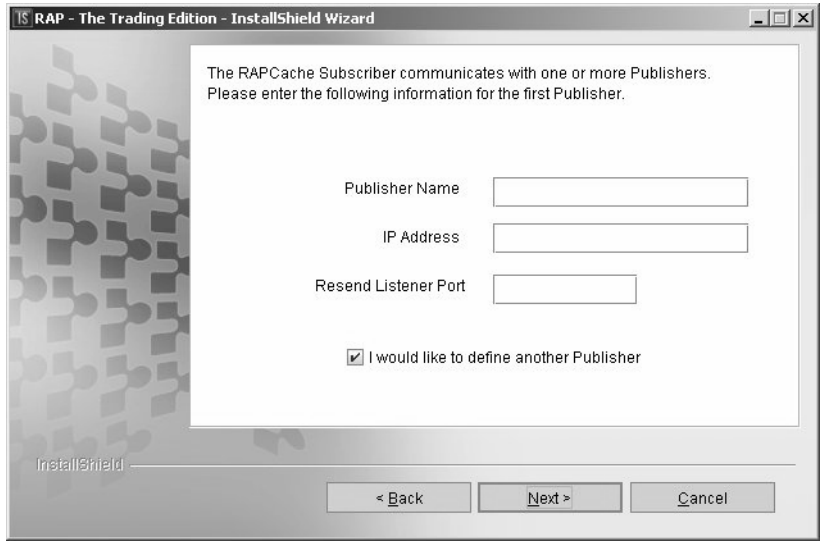


Table A-21: RAPCache Subscriber dialog 5 configuration options

Field name	Description
Publisher Name	Identifies the publisher that publishes to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.
IP Address	Identifies the IP address of the machine on which the publisher resides. This is used to connect to the publisher to request resends of packets.
Resend Listener Port	Identifies the port that the publisher uses to resend cached message packets to the subscriber. Publishers maintain a cache of recently sent message packets to ensure that the subscriber processes all message deliveries.

Note After you define the first publisher, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another publisher** box to define another publisher.
- Remove the checkmark in the **I would like to define another publisher** box, and click **Next** to display the next set of configuration options.

RAPCache Subscriber configuration dialog 6

If you install the RAPCache Subscriber at a different time than the RAPCache Database Server, this optional dialog appears and prompts you for the server's location.

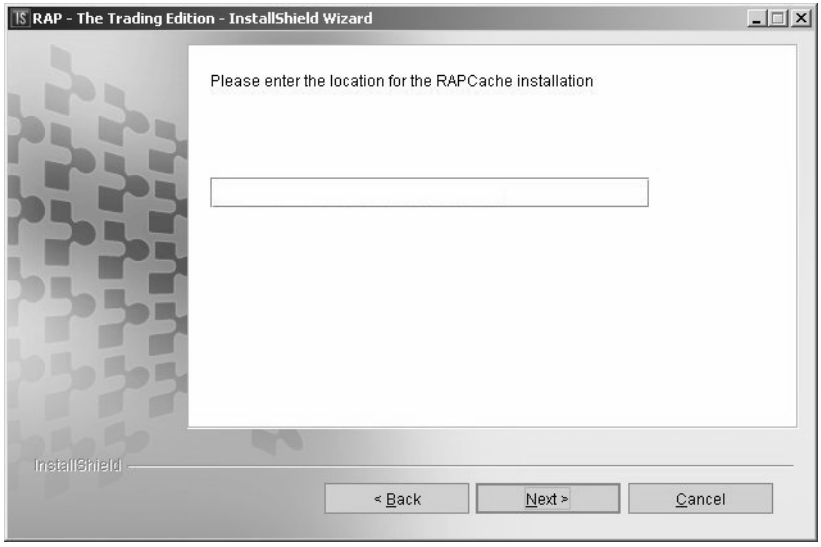


Table A-22: RAPCache Subscriber dialog 6 configuration options

Field Name	Default value	Description
RAPCache Server installation location	Defaults to the install machine name	Identifies the host name of the server where you installed the RAPCache Server.

RAPCache Subscriber configuration dialog 7

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register the RAPCache Subscriber with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

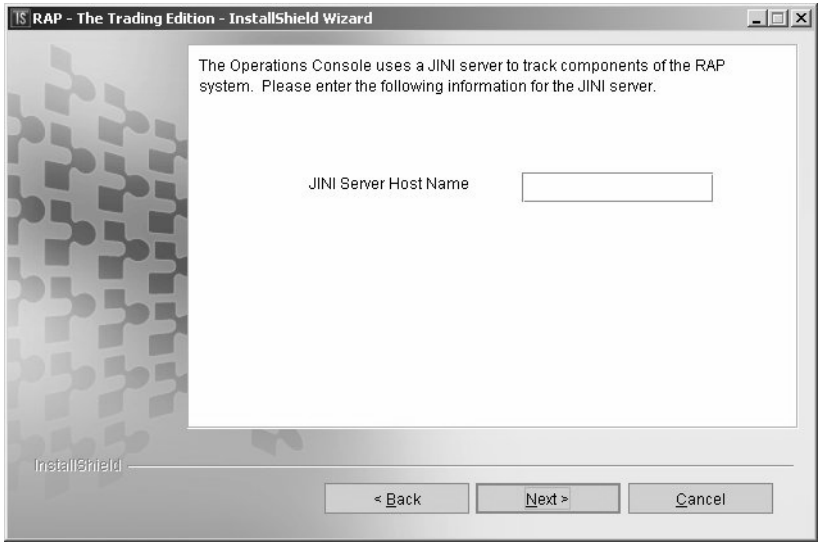


Table A-23: RAPCache Subscriber dialog 7 configuration options

Field Name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

RAPStore System

The RAPStore System includes two options: RAPStore Database Server and RAPStore Subscriber.

RAPStore Database Server

Installing the RAPStore Database Server silently installs Sybase IQ 12.7 ESD #3, creates the RAPStore database objects, and loads the sample data. During installation, the installer sets all configuration parameters to IQ defaults.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- RAPStore Database Server configuration dialog 1 sets the listening port for the UAF agent that Operations Console uses to communicate with the RAPStore Database Server.
- RAPStore Database Server configuration dialog 2 prompts you for the server's name and port number.
- RAPStore Database Server configuration dialog 3 prompts you for the host name of the JINI server that Operations Console uses to track RAP system components.

RAPStore Database Server configuration dialog 1

Operations Console communicates with the RAPStore Database through an intermediary UAF agent which listens for requests. This dialog sets the UAF agent listening port.

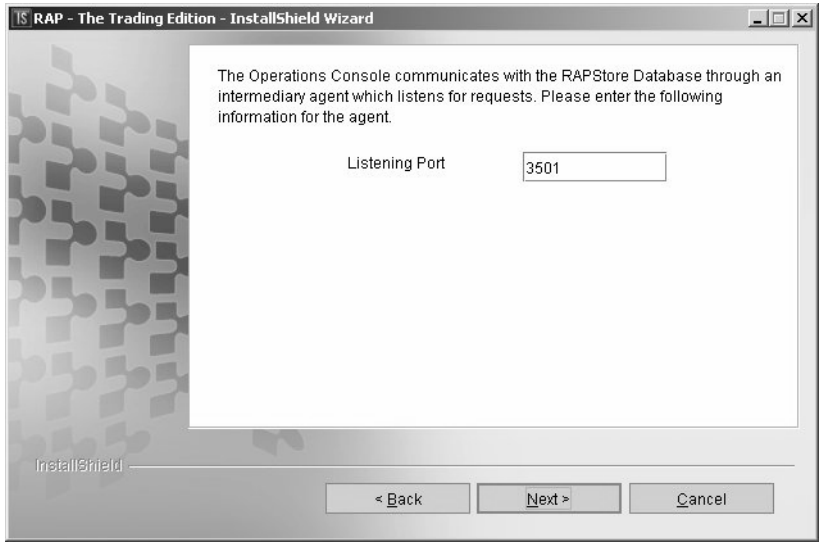


Table A-24: RAPStore Database Server dialog 1 configuration options

Field name	Default value	Description
Listening Port	3501	Identifies the listening port that the UAF agent uses to communicate with Operations Console. You can use any free port on your system.

RAPStore Database Server configuration dialog 2

This dialog identifies the RAPStore Database Server's name and port number.

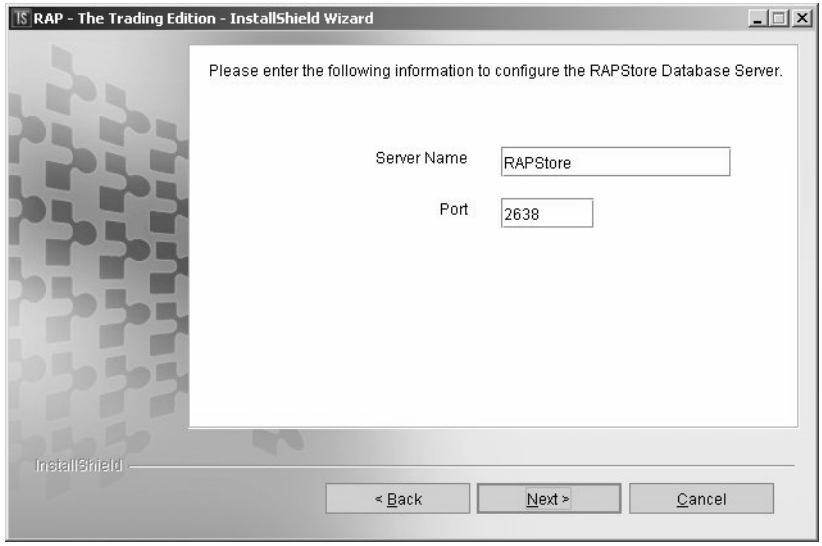


Table A-25: RAPStore Database Server dialog 2 configuration options

Field Name	Default value	Description
Server Name	RAPStore	Identifies the server where the RAPStore database resides.
Port	2638	Identifies the RAPStore communications port.

RAPStore Database Server configuration dialog 3

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register RAPStore Database Server with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

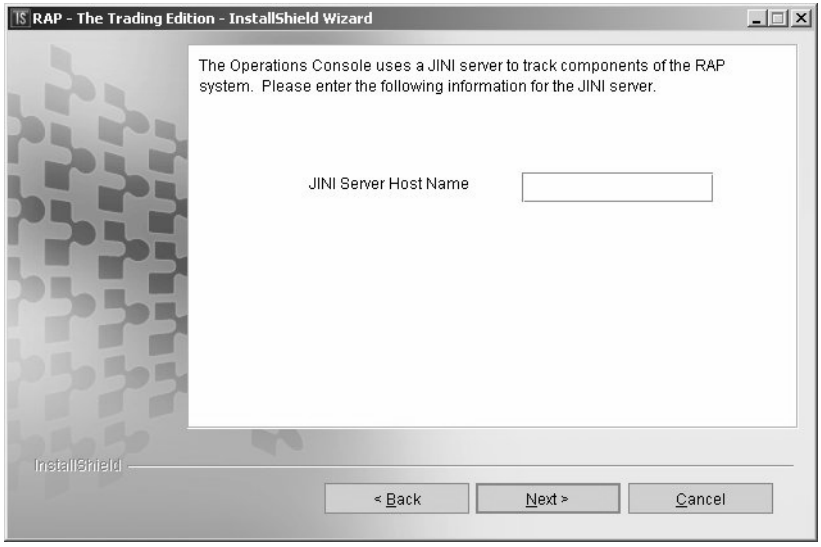


Table A-26: RAPStore Database Server dialog 3 configuration options

Field Name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

RAPStore Subscriber

RAPStore Subscriber processes message packets that arrive on multiple data streams and loads the RAPStore database.

Note If you install the RAPStore Subscriber at a different time than the RAPStore Database Server, the installer will prompt you for the server's location. You must install the RAPStore Subscriber on the same machine as the RAPStore Database Server.

Architecture

RAPStore Subscriber queues messages that are consumed by a data stream handler, which routes the messages to their final destination.

Subscribers receive message packets from publishers on multiple data streams. If a packet is missing, the subscriber requests a resend from the publisher for that specific packet.

Configuration

If you choose this option, the installer displays several configuration dialogs:

- RAPStore Subscriber configuration dialog 1 sets the UAF agent listening port that Operations Console uses to communicate with RAPStore Subscriber. This dialog also lets you set the component name.
- RAPStore Subscriber configuration dialog 2 lets you to decide where to store temporary market data files.
- RAPStore Subscriber configuration dialog 3 identifies the data channels that RAPStore Subscriber monitors for message deliveries from publishers.
- RAPStore Subscriber configuration dialog 4 identifies the listening port that RAPStore Subscriber uses to listen for Operations Console administrative requests. This dialog also sets some performance options.
- RAPStore Subscriber configuration dialog 5 prompts you for the publishers that send messages to RAPStore Subscriber.
- RAPStore Subscriber configuration dialog 6 is an optional dialog that appears if you install the RAPCache Subscriber separately from the RAPStore Database Server.
- RAPStore Subscriber configuration dialog 7 prompts you for the host name of the JINI server that tracks RAP system components.

RAPStore Subscriber configuration dialog 1

Operations Console communicates with the RAPStore Subscriber through an intermediary UAF agent which listens for requests. This dialog sets the UAF agent listening port and optional UAF component name.

Table A-27: RAPStore Subscriber dialog 1 configuration options

Field name	Default value	Description
Listening Port	3601	Identifies the listening port that the UAF agent uses to communicate with Operations Console. You can use any free port on your system.
Component Name	None	Identifies this component. This name appears in the component's panel on the Operations Console Monitor. Although the Component Name is optional, naming the component provides an identity that distinguishes this component from other components of the same type. Projects may include several subscribers or publishers. Assigning unique component names can help you track each item more easily.

RAPStore Subscriber configuration dialog 2

The RAPStore Subscriber temporarily writes market data messages to files. The location of these directories should match the values on your worksheet. You must manually create these directories after you complete the install. See “Create the primary and overflow file directories” on page 45 for more information.

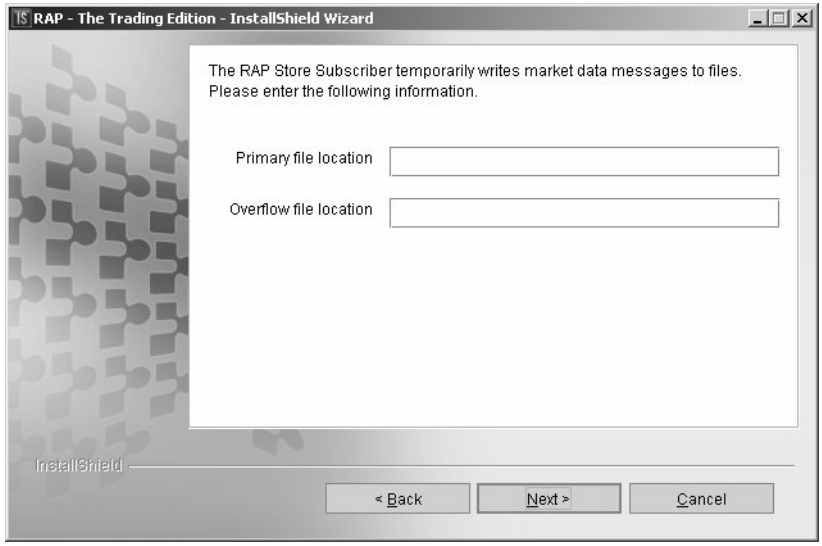


Table A-28: RAPStore Subscriber dialog 2 configuration options

Field name	Description
Primary file location	Identifies the directory where the subscriber temporarily writes market data messages before it loads the database.
Overflow file location	Identifies the directory where the subscriber stores temporary market data message files when the primary directory is full.

RAPStore Subscriber configuration dialog 3

RAPStore Subscriber listens on one or more data channels for messages sent by publishers. Use this dialog to identify the data channels you want RAPStore Subscriber to monitor.

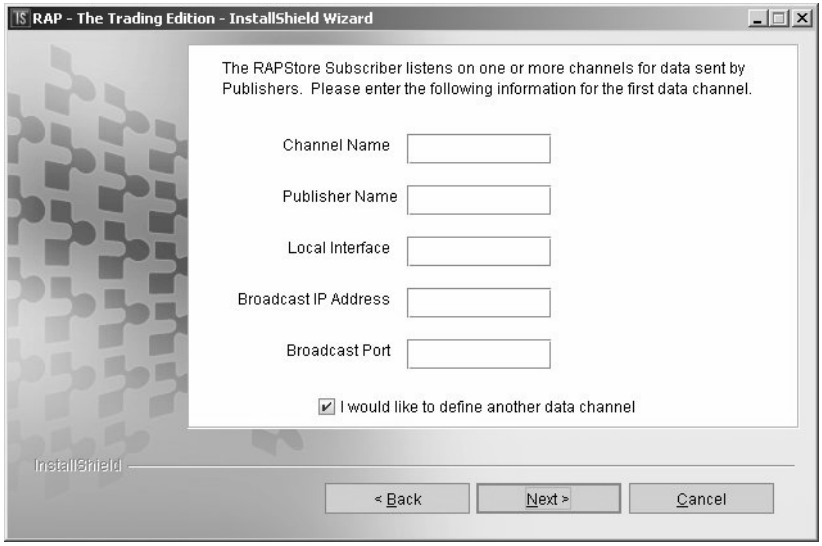


Table A-29: RAPStore Subscriber dialog 3 configuration options

Field name	Description
Channel Name	Identifies a specific data channel that a particular publisher uses to multicast messages to RAPStore Subscriber. The Channel Name depends on the data channel you want RAPStore Subscriber to monitor. If, for example, you want to RAPStore Subscriber to monitor FAST Feed Handler, and kept all the defaults when you defined the FAST Feed data channels, you would use FAST Channel 1 as the first Channel Name. Subsequent channel names would match those of other data channel names.
Publisher Name	Identifies the publisher that sends messages to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.

Field name	Description
Local Interface	Identifies the IP address of the network interface that the publisher uses to multicast messages to RAPStore Subscriber. This value must match the local interface value that the publisher uses to multicast messages. If you want RAPStore Subscriber to monitor a particular FAST Feed Handler data channel, for example, the local interface value you use here must match the local interface value for the data channel you defined for FAST Feed Handler.
Broadcast IP Address	Identifies the Class D multicast IP address that the publisher uses to multicast messages to RAPStore Subscriber. This value must match the multicast address that the publisher uses to multicast messages to RAPStore Subscriber. If you want RAPCache Subscriber to monitor a FAST Feed Handler data channel that multicasts messages on 224.0.2.0, for example, you would use that address for this field. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	Identifies the port from which the publisher multicasts messages to RAPStore Subscriber. Publishers can multicast messages to a RAPStore Subscriber on multiple data channels that share the same local interface and Broadcast IP Address, but the Broadcast Port must be unique to each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to define another data channel.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

RAPStore Subscriber configuration dialog 4

The RAPStore Subscriber listens to administrative requests from Operations Console. Use this dialog to identify the administration listening port and set some performance options.

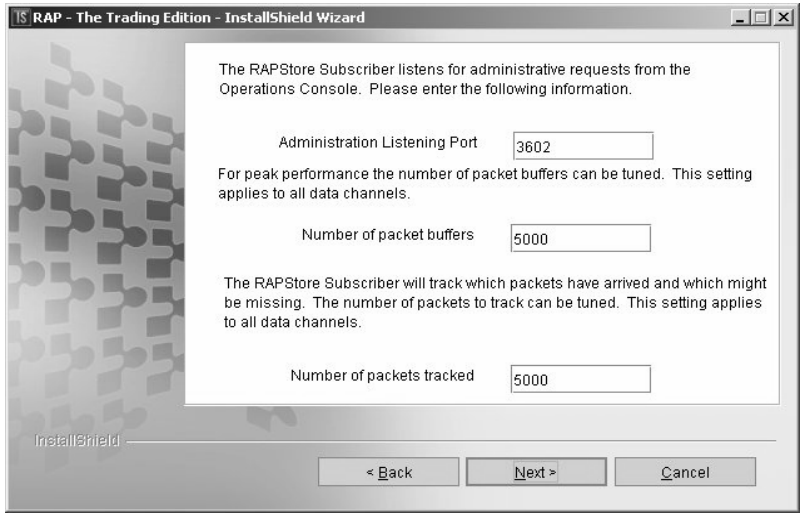


Table A-30: RAPStore Subscriber dialog 4 configuration options

Field name	Default value	Description
Administration Listening Port	3602	Identifies the listening port that the subscriber uses to monitor administrative requests from Operations Console.
Number of packet buffers	5000	Identifies the number of packet buffers. The subscriber holds onto arriving packets of messages until the packets are processed. The number of packet buffers is per data stream. This setting can have a value between 1 and 4 billion, but the machine must have enough memory to hold the number of packets specified. This setting should be adjusted if resend requests from publishers are failing, because the packet no longer exists at the publisher. For subscribers, a value of 500 will usually provide sufficient buffering to handle small bursts of high-volume data.
Number of packets tracked	5000	Identifies the number of packets the subscriber tracks. Subscribers track the packets delivered by a publisher. If a subscriber suspects that a packet may be missing, it issues a resend request to the publisher to retrieve the missing data. The number of packets tracked is for each data stream.

RAPStore Subscriber configuration dialog 5

The RAPStore Subscriber communicates with one or more publishers. Use this dialog to identify the publishers you want the RAPStore Subscriber to monitor.

Table A-31: RAPStore Subscriber dialog 5 configuration options

Field name	Description
Publisher Name	Identifies the publisher that publishes to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.
IP Address	Identifies the IP address of the machine on which the publisher resides. This is used to connect to the publisher to request resends of packets.
Resend Listener Port	Identifies the port that this subscriber uses to request missing packets from the publisher.

Note After you define the first publisher, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another publisher** box to display a blank publisher dialog.
- Remove the checkmark in the **I would like to define another publisher** box, and click **Next** to display the next set of configuration options.

RAPStore Subscriber configuration dialog 6

This is an optional dialog that appears if you install the RAPCache Subscriber separately from the RAPStore Database Server.

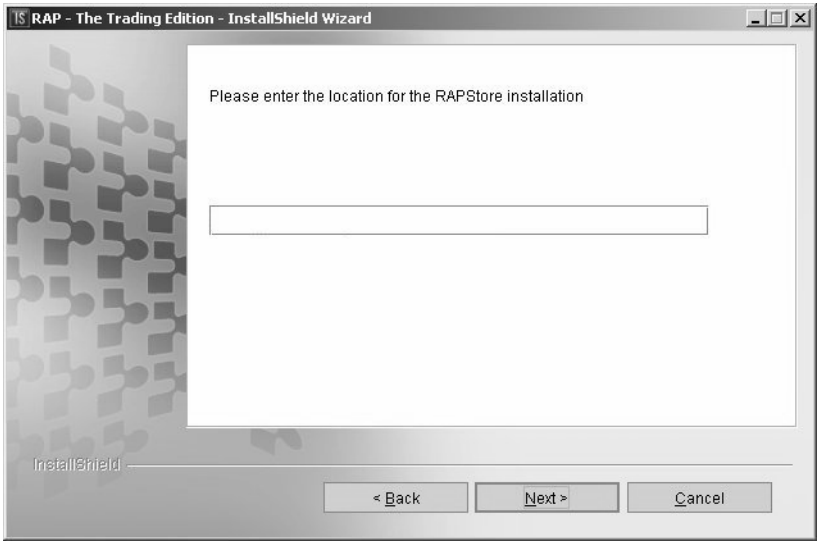


Table A-32: RAPStore Subscriber dialog 6 configuration options

Field name	Default value	Description
RAPStore Server location	Defaults to the install machine name	Identifies the location of the RAPStore Database Server. You must specify this location to allow the subscriber to communicate with the database. This value defaults to the name of the machine where you are running the install.

RAPStore Subscriber configuration dialog 7

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register RAPStore Subscriber with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

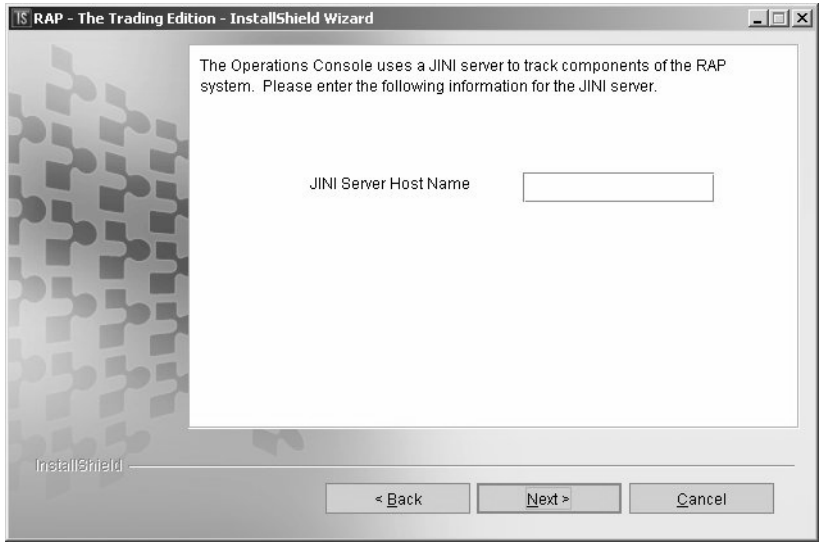


Table A-33: RAPStore Subscriber dialog 7 configuration options

Field Name	Default value	Description
JINI Server Host Name	None	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed. You must identify the JINI Server Host Name to register this component with the JINI Server.

Demo Subscriber

Demo Subscriber processes message packets that arrive over data streams from a publisher. The purpose of the Demo Subscriber is to demonstrate, in conjunction with the Demo Feed Handler, message flow through Sybase RAP.

Architecture

The Demo Subscriber receives messages from publishers on multiple data streams. Each data stream listens for packets on one UDP broadcast channel. When a packet arrives, the Demo Subscriber will either write the messages to a file or discard the message, depending on how the Demo Subscriber is configured.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- Demo Subscriber configuration dialog 1 sets the UAF agent listening port that Demo Subscriber uses to communicate with Operations Console. This dialog also lets you set the component name.
- Demo Subscriber configuration dialog 2 identifies the data channels that Demo Subscriber monitors for message deliveries from publishers.
- Demo Subscriber configuration dialog 3 sets the listening port that Demo Subscriber uses to listen for Operations Console administrative requests. This dialog also lets you set some performance options.
- Demo Subscriber configuration dialog 4 prompts you for the publishers that send messages to Demo Subscriber.
- Demo Subscriber Configuration dialog 5 prompts you for the host name of the JINI server that Operations Console uses to track RAP system components.

Demo Subscriber configuration dialog 1

Operations Console communicates with the Demo Subscriber through an intermediary UAF agent. Use this dialog to set the UAF agent listening port and optional UAF component name.

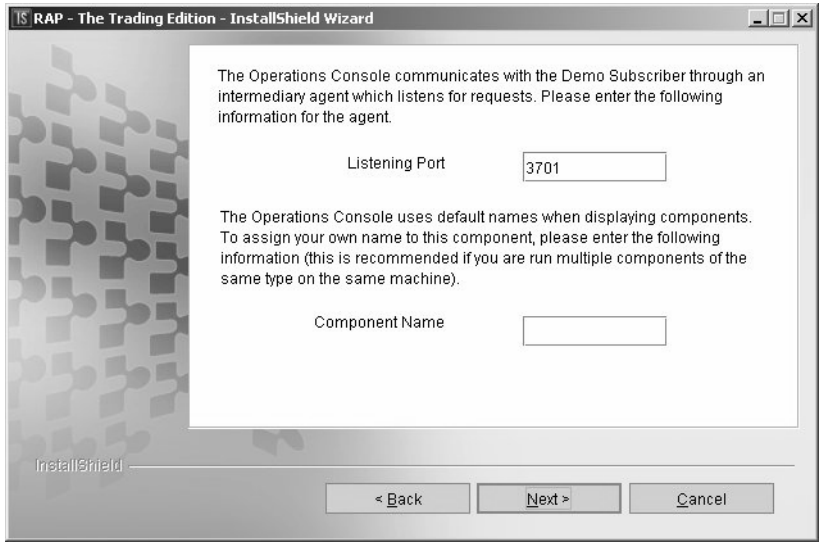


Table A-34: Demo Subscriber dialog 1 configuration options

Field name	Default value	Description
Listening Port	3701	Identifies the listening port that the UAF agent uses to communicate with Operations Console. You can use any free port on your system.
Component Name	None	Identifies this component. This name appears in the component's panel on the Operations Console Monitor. Although the Component Name is optional, naming the component provides an identity that distinguishes this component from other components of the same type. Projects may include several subscribers or publishers. Assigning unique component names can help you track each item more easily.

Demo Subscriber configuration dialog 2

Demo Subscriber listens to one or more data channels for messages sent by publishers. Use this dialog to identify each channel that you want Demo Subscriber to monitor.

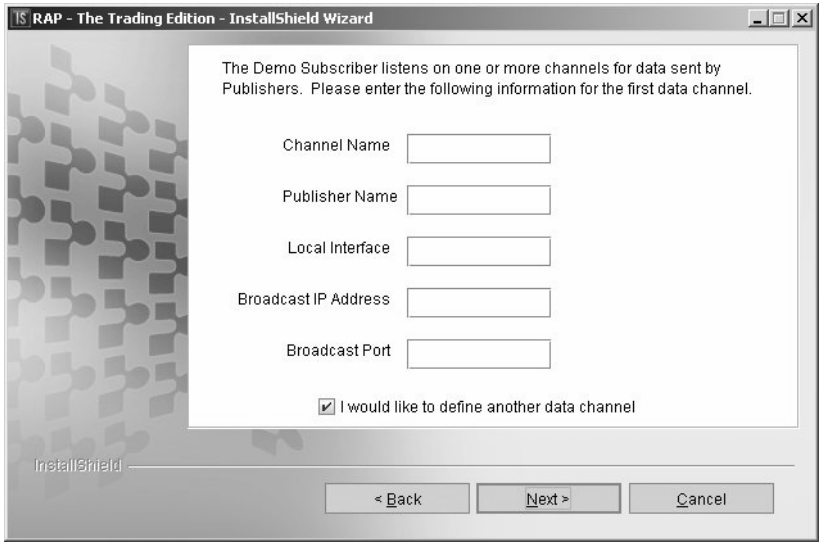


Table A-35: Demo Subscriber dialog 2 configuration options

Field name	Description
Channel Name	Identifies a specific data channel that a particular publisher uses to multicast messages to Demo Subscriber. The Channel Name depends on the data channel you want Demo Subscriber to monitor. If, for example, you want to Demo Subscriber to monitor FAST Feed Handler, and kept all the defaults when you defined the FAST Feed data channels, you would use FAST Channel 1 as the first Channel Name. Subsequent channel names would match those of other data channel names.
Publisher Name	Identifies the publisher that sends messages to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.

Field name	Description
Local Interface	Identifies the IP address of the network interface that the publisher uses to multicast messages to Demo Subscriber. This value must match the local interface value that the publisher uses to multicast messages. If you want Demo Subscriber to monitor a particular FAST Feed Handler data channel, for example, the local interface value you use here must match the local interface value for the data channel you defined for FAST Feed Handler.
Broadcast IP Address	Identifies the Class D multicast IP address that the publisher uses to multicast messages to Demo Subscriber. This value must match the multicast address that the publisher uses to multicast messages to Demo Subscriber. If you want Demo Subscriber to monitor a FAST Feed Handler data channel that multicasts messages on 224.0.2.0, for example, you would use that address for this field. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	Identifies the port from which the publisher multicasts messages to Demo Subscriber. Publishers can multicast messages to Demo Subscriber on multiple data channels that share the same local interface and Broadcast IP Address, but the Broadcast Port must be unique to each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to display a blank data channel dialog.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

Demo Subscriber configuration dialog 3

Demo Subscriber listens for administrative requests from Operations Console. Use this dialog to identify the administrative listening port and sets some performance options.

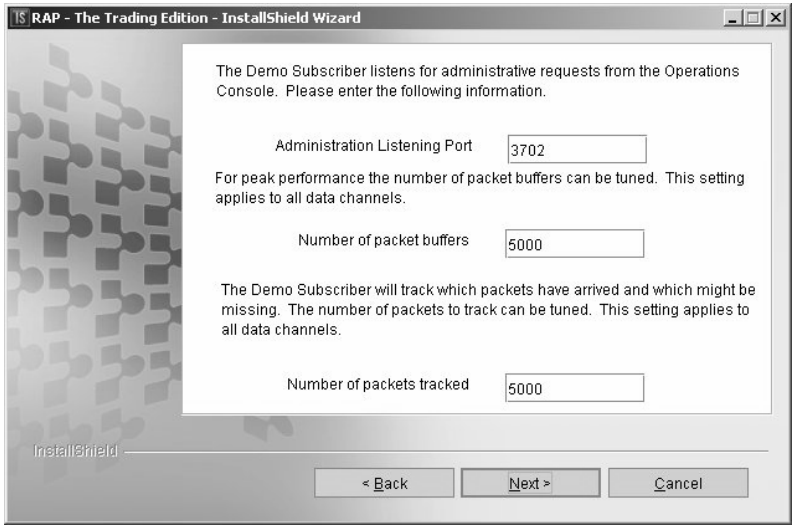


Table A-36: Demo Subscriber dialog 3 configuration options

Field name	Default value	Description
Administration Listening Port	3702	Identifies the listening port that the subscriber uses to monitor administrative requests from Operations Console.
Number of packet buffers	5000	Identifies the number of packet buffers. The subscriber holds onto arriving packets of messages until the packets are processed. The number of packet buffers is per data stream. This setting can have a value between 1 and 4 billion, but the machine must have enough memory to hold the number of packets specified. This setting should be adjusted if resend requests from publishers are failing, because the packet no longer exists at the publisher. For subscribers, a value of 500 will usually provide sufficient buffering to handle small bursts of high-volume data.
Number of packets tracked	5000	Identifies the number of packets the subscriber tracks. Subscribers track the packets delivered by a publisher. If a subscriber suspects that a packet may be missing, it issues a resend request to the publisher to retrieve the missing data. The number of packets tracked is for each data stream.

Demo Subscriber configuration dialog 4

Demo Subscriber communicates with one or more publishers. Use this dialog to identify the publisher(s) you want Demo Subscriber to monitor.

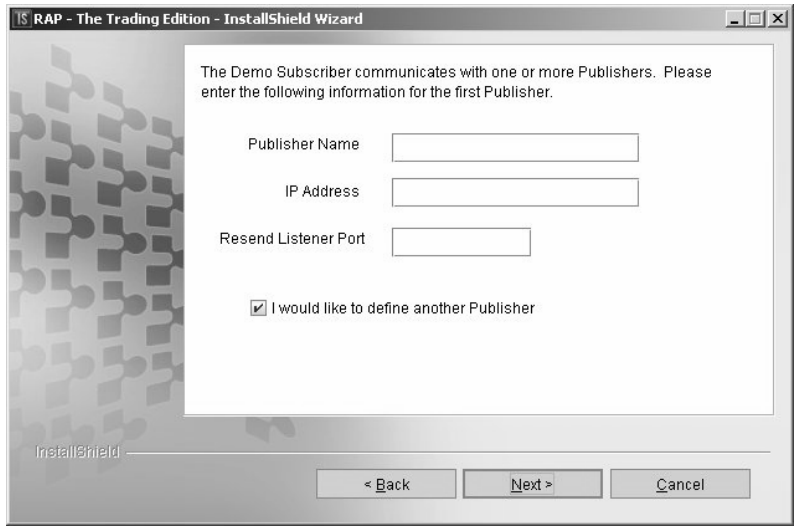


Table A-37: Demo Subscriber dialog 4 configuration options

Field name	Description
Publisher Name	Identifies the publisher that publishes to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.
IP Address	Identifies the IP address of the machine on which the publisher resides. This is used to connect to the publisher to request resends of packets.
Resend Listener Port	Identifies the port that the publisher uses to resend cached message packets to the subscriber. Publishers maintain a cache of recently sent message packets to ensure that the subscriber processes all message deliveries.

Note After you define the first publisher, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another publisher** box to display a blank publisher dialog.
- Remove the checkmark in the **I would like to define another publisher** box, and click **Next** to display the next set of configuration options.

Demo Subscriber Configuration dialog 5

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register Demo Subscriber with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

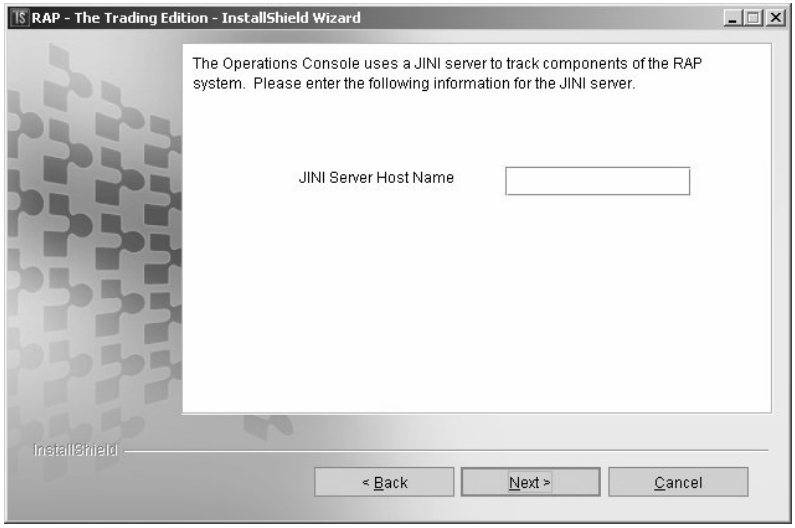


Table A-38: Demo Subscriber dialog 5 configuration options

Field name	Default value	Description
JINI Server Host Name	None	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

Operations Console

Operations Console is a web-based interface that provides Sybase RAP administration and monitoring functions.

Architecture

Operations Console interacts with different system components through the Unified Agent Framework (UAF). UAF plug-ins running on each server handle communication between Operations Console and Sybase RAP components. When an agent starts on a server, the agent starts all of its plug-ins and registers with the JINI server. Operations Console queries the JINI server to learn about all agents (and their plug-ins).

Interaction between Operations Console and the plug-ins occurs via RMI (remote method invocation). Publisher and subscriber plug-ins communicate with publishers and subscribers components via TCP over a local socket. Each publisher and subscriber has an administration channel listening on a specific port. When Operations Console issues a command, the agent intercepts the request, and contacts the component. The component sends a response through the agent back to Operations Console.

Configuration

If you install this option, the installer displays several configuration dialogs.

- Operations Console configuration dialog 1 prompts you for the JINI port numbers and cache server settings.
- Operations Console configuration dialog 2 prompts you for the RAPStore Database Server settings.
- Operations Console configuration dialog 3 prompts you for the Web server settings.

Operations Console configuration dialog 1

Operations Console uses a JINI server to track RAP system components. This dialog prompts you for the JINI port numbers and cache server settings

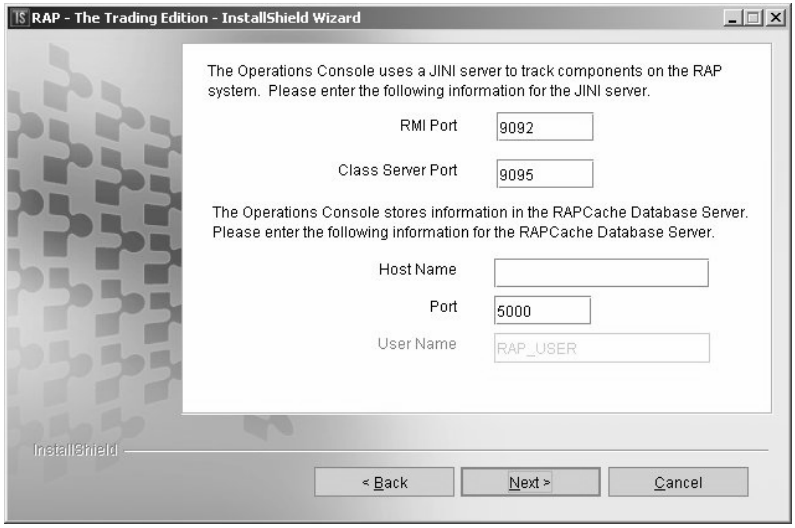


Table A-39: Operation Console dialog 1 configuration options

Field name	Default value	Description
RMI Port	9092	Identifies the RMI port that the JINI server uses internally in its operation. This value is set to port number 9092 by default, but you can set the port number to any free port on the JINI server.
Class Server Port	9095	Identifies the port that the JINI server uses internally in its operation. This value is set to port number 9095 by default, but you can set the port number to any free port on the JINI server.
Host Name	Host machine	Identifies the machine that hosts the RAPCache Database server. This value defaults to the name of the machine where you run the install.
Port	5000	Identifies the port that the RAPCache Database Server listens for connections.
User Name	RAP_USER	Identifies the default Sybase RAP user. This value is set to RAP_USER by default. You cannot change this value.

Operations Console configuration dialog 2

Operations Console retrieves status information from the RAPStore Server. This dialog prompts you for the RAPStore Database Server settings.

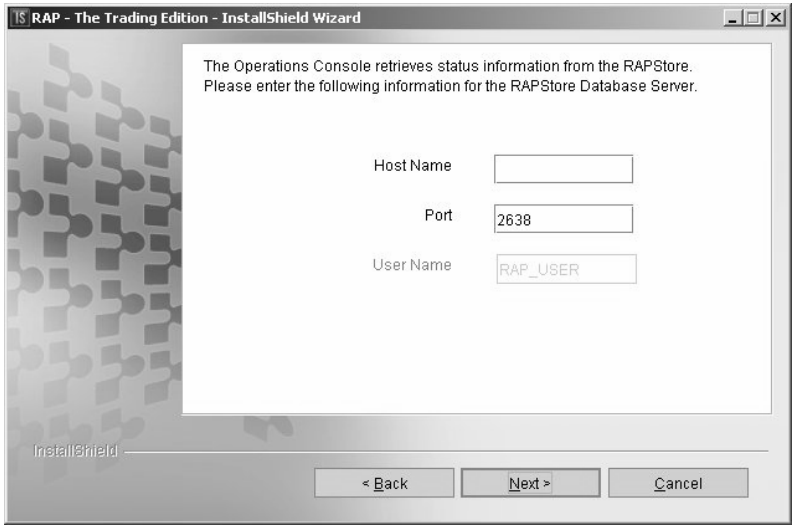


Table A-40: Operation Console dialog 2 configuration options

Field name	Default value	Description
Host Name	Default to install machine name	Identifies the machine that hosts the RAPStore Database server. This value defaults to the name of the machine where you run the install.
Port	2638	Identifies the port that the RAPStore Database Server uses to listen for connections.
User Name	RAP_USER	Identifies the default Sybase RAP user. This value is set to RAP_USER by default. You cannot change this value.

Operations Console configuration dialog 3

This dialog prompts you for the Web server settings.

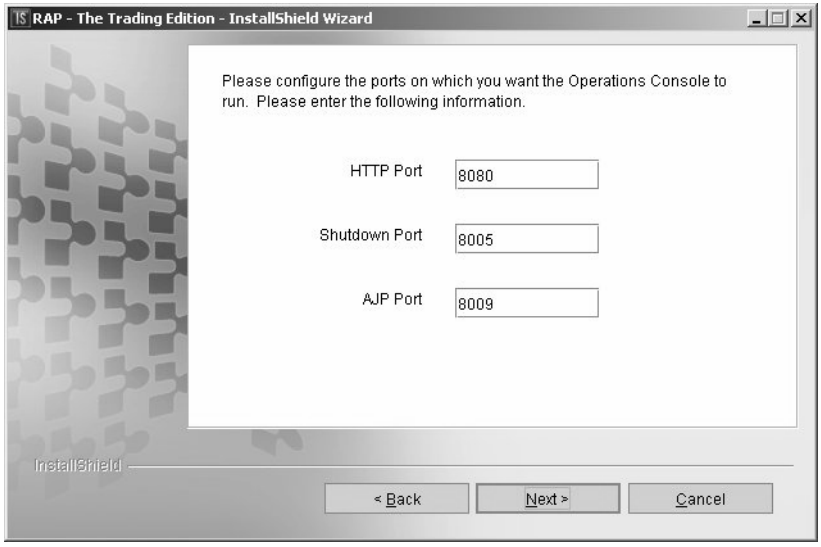


Table A-41: Operation Console dialog 3 configuration options

Field name	Default value	Description
HTTP Port	8080	Identifies the port that Tomcat uses to listen for incoming connections from web browsers. This value is set to 8080 by default.
Shutdown Port	8085	Identifies the port that Tomcat uses to close all Web applications, and shutdown all services. The shutdown port is set to 8005 by default
AJP Port	8009	Identifies the AJP (TCP) port that Tomcat uses to create a server socket and wait for incoming connections. AJP protocol invisibly integrates Tomcat into existing or new Apache installations. The AJP port is set to 8009 by default.

Samples

Installs the sample queries and sample data. See “Samples” on page 192 for a list of items installed with this option.

APPENDIX B

Express Setup Options

Refer to this appendix to review the platform component configuration options you may encounter during an Express installation.

Overview

This appendix is intended as a reference for Express installs and displays all configuration dialogs and configuration options. An Express installs involves fewer configuration options, and is suitable for some production environments.

Express installer actions

During installation, the installer:

- Prompts you for the name of the JINI server host. The JINI server is installed on the same machine where you install Operations Console. If you install multiple components at the same time, the installer only prompts you for the JINI server host name once. This value is inherited by the other components you are installing during that installation sequence. When you complete an installation sequence on one machine and restart the installer to install components on another, you may be prompted to specify the location of the JINI server.
- Assigns default port values in the `3000 – 4000` range to all feed handler and publisher components.Port values are validated during installation, and if a defined port is already in use, the installer displays two popup message boxes. The first message box tells you that the port number is already in use and an alternative port will be selected. The second message box then provides the new port assignment. If your Network Administrator chooses a different set of port values, records these items on your worksheet, and replace the default value with these items.
- Assigns *224.0.2.0* as the default class D multicast IP addresses to all publishers. This value appears in the Broadcast IP Address field and identifies the multicast channel that the publisher uses to multicast messages to subscribers. If your Network Administrator chooses another multicast address for your site, record the address on your worksheet, and replace the default with that value.
- Assigns default settings for various configuration options, such as the Number of Packet Buffers.
- Assigns default Channel names to Feed Handlers and Publishers. Default channels names depend on the component. For FAST Feed Handler, the first default channel name is `FAST Channel 1`; the first default channel name for Demo Feed Handler is `Demo Feed 1`; and the first default channel name for the Publisher API is `Custom Channel 1`. If you want to use different names to identify your data channels, replace the default value with the one you want to use.
- Assigns default settings for various configuration options, such as the Number of Packet Buffers.
- Installs the RAPCache database and RAPCache Subscriber on the same target machine at the same time. You cannot install these items separately. The RAPCache Database Server is set to ASE 15.0.2 defaults, and the host name is set to the install machine. The RAPCache Subscriber inherits all appropriate server connection settings, but prompts you for the appropriate data channels to monitor and publisher settings.

- Installs the RAPStore database and RAPStore subscriber on the same target machine at the same time. You cannot install these items separately. The RAPStore Database Server is set to Sybase IQ 12.7 ESD #3 defaults, and the host name is set to the install machine. The RAPStore Subscriber inherits all appropriate server connection settings, but prompts you for the appropriate data channels to monitor and publisher settings.
- Assigns all appropriate port values for Operations Console. If you install Operations Console at the same time you install RAPCache and RAPStore, the installer will populate the appropriate Host and port values for these items. If not, the Host field(s) will be left empty and the port will be the appropriate default for the database (5000 for RAPCache and 2638 for RAPStore).

Demo Feed Handler

Demo Feed Handler is a Sybase feed handler that reads comma-delimited market data from a demo file (*STOCK_QUOTE.csv).* This component is intended to help developers become familiar with the Publisher API and demonstrate network message flow.

Architecture

Demo Feed Handler is an executable that reads messages from a demo data file, and sends those messages to a subscriber.

Syntax

To run Demo Feed Handler from the command line:

```
demofeedhandler -m <messageFile> -n <numMessages> -s
<maximum messages per second>
```

- Include the full path and file name of the demo data file as the `messageFile` argument.
- Use a positive integer (any number from *1* to over *4 billion*) as the numMessages argument to process a specific number of messages.
- Specify *-1* to continuously loop through the file, until the user chooses to stop it.

- `-s <maximum messages per second>` is an optional switch that allows the user to simulate various scenarios. If absent, messages will be sent at the maximum possible speed.

Note Although Sybase RAP includes *STOCK_QUOTE.csv* that you can use as a test message file for Demo Feed Handler, you can use *.csv* file that contains Stock Quote messages will work.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- Demo Feed Handler configuration dialog 1 prompts you for the data channels the Demo Feed Handler uses to send information to subscribers.
- Demo Feed Handler configuration dialog 2 prompts you for the JINI server name This is the name of the machine where you Operations Console is installed.

Demo Feed Handler configuration dialog 1

Demo Feed Handler multicasts messages to subscribers over one or more data channels. Four channels are recommended for a system that handles 300,000 messages per second. Use this dialog to identify the data channels.

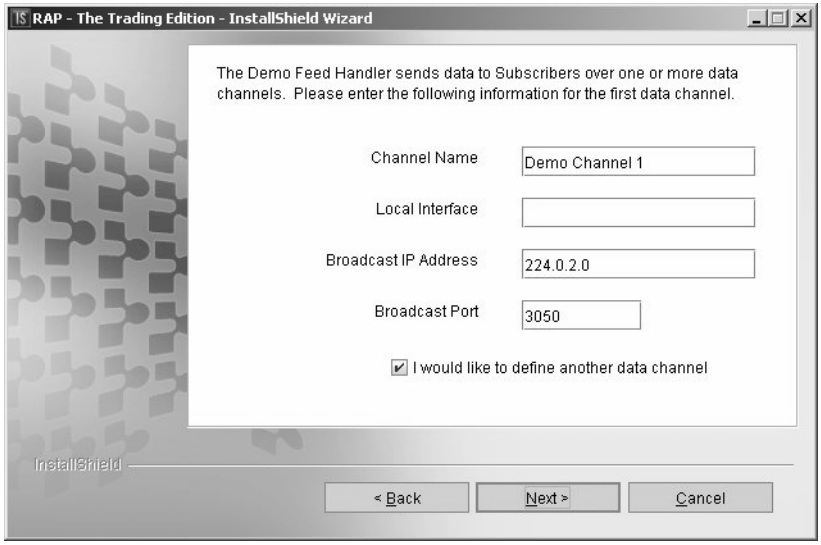


Table B-1: Demo Feed Handler dialog 1 configuration options

Field name	Default value	Description
Channel Name	Demo Channel 1	Identifies the data channel that Demo Feed Handler uses to send messages to a subscriber. Although the installer assigns Demo Channel 1 to the first data channel by default, you can use any value as the channel name. You must supply a Channel Name to any subsequent data channels you define. Assigning unique names to channels can help you track each one more easily in the event that an error is logged.
Local Interface	None	Identifies the IP address of the network interface to which this channel will be bound. This allows a you to bind various data channels to separate network interfaces to enhance performance. • If you plan to install all publishers and subscribers on the same server, you must configure the Local Interface to use the loopback interface (127.0.0.1). • If you install publishers and subscribers on different servers, use the local machine address of the secondary Ethernet network card installed by the system administrator.

Field name	Default value	Description
Broadcast IP Address	224.0.2.0	Identifies the Class D multicast IP address that Demo Feed Handler uses to multicast messages to subscribers. This value should match the multicast address that the subscribers use to receive messages. Although the installer assigns 224.0.2.0 as a default multicast IP address to all Broadcast IP address fields, your Network Administrator can choose an alternate multicast address. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	3050	Identifies the local port on which this data channel will operate. Publishers can multicast messages to a subscriber on multiple data channels. Data channels can share the same local interface and Broadcast IP Address values, but the Broadcast Port must be unique for each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to display a blank data channel dialog.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

Demo Feed Handler configuration dialog 2

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog prompts you for the JINI server host name, which is the same host machine where you install Operations Console.You must identify the JINI Server Host Name to register Demo Feed Handler with the JINI Server.

Table B-2: Demo Feed Handler dialog 2 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

FAST Feed Handler

FIX Adapted for Streaming (FAST) protocol is a binary encoding method for message oriented data streams. The FAST Feed Handler reads FAST encoded data directly from the wire, decodes the data, and forwards the messages to the Publisher API. See the Developers Guide for detailed information about FAST Feed Handler's architecture.

Note The Sybase RAP FAST Feed Handler supports FAST SCP 1.1

Configuration

If you choose this option, the installer displays several configuration dialogs.

- FAST Feed Handler configuration dialog 1 prompts you for information about the FAST feed.
- FAST Feed Handler configuration dialog 2 prompts you for the data channels that FAST Feed Handler uses to send messages to subscribers.
- FAST Feed Handler configuration dialog 3 prompts you for the JINI Server Host Name. Depending on how you install components, this dialog is optional.

FAST Feed Handler configuration dialog 1

FAST Feed Handler listens for FAST messages sent by a FAST feed. This dialog prompts you for the FAST feed parameters.

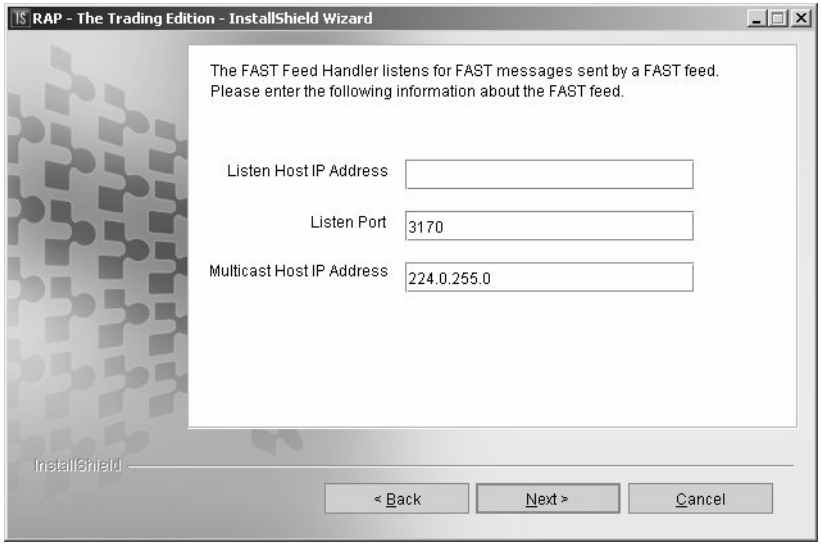


Table B-3: FAST Feed Handler dialog 1 configuration options

Field Name	Default value	Description
Listen Host IP Address	None	Identifies the IP address for a network interface to which the incoming feed's UDP channel will be bound. This allows optimization of network resources, by allowing various channel loads to be spread out among multiple physical network adapters.
Listen Port	3170	Identifies the UDP Local listening port for incoming messages.
Multicast Host IP Address	224.0.255.0	Identifies the class D (multicast) IP address on which this data channel will operate. A publisher can multicast messages to a subscriber on multiple channels, and the combination of the Broadcast IP Address and Broadcast Port must be unique for each channel. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.

FAST Feed Handler configuration dialog 2

FAST Feed Handler multicasts messages to subscribers over one or more data channels. Four channels are recommended for a system that handles 300,000 messages per second. Use this dialog to identify the data channels.

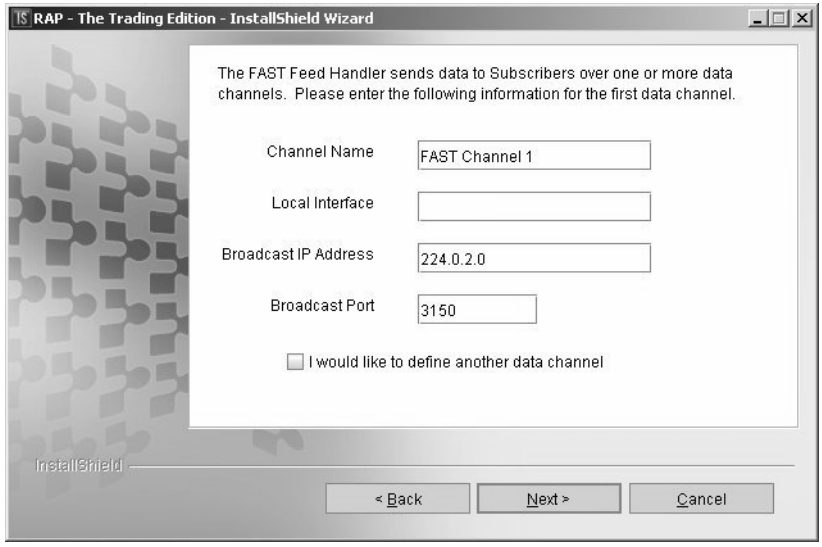


Table B-4: FAST Feed Handler dialog 2 configuration options

Field name	Default value	Description
Channel Name	FAST Channel 1	Identifies the data channel that FAST Feed Handler uses to send messages to a subscriber. Although the installer assigns FAST Channel 1 to the first data channel by default, you can use any value as the channel name. You must supply a Channel Name to any subsequent data channels you define. Assigning unique names to channels can help you track each one more easily in the event that an error is logged.
Local Interface	None	Identifies the IP address of the network interface to which this channel will be bound. This allows a you to bind various data channels to separate network interfaces to enhance performance. • If you plan to install all publishers and subscribers on the same server, you must configure the Local Interface to use the loopback interface (127.0.0.1). • If you install publishers and subscribers on different servers, use the local machine address of the secondary Ethernet network card installed by the system administrator.

Field name	Default value	Description
Broadcast IP Address	224.0.2.0	Identifies the Class D multicast IP address that FAST Feed Handler uses to multicast messages to subscribers. This value should match the multicast address that the subscriber uses to receive messages. Although the installer assigns 224.0.2.0 as a default multicast IP address to all Broadcast IP address fields, your Network Administrator can choose an alternate multicast address. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	3150	Identifies the local port on which this data channel will operate. Publishers can multicast messages to a subscriber on multiple data channels. Data channels can share the same local interface and Broadcast IP Address values, but the Broadcast Port must be unique for each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to define another data channel.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

FAST Feed Handler configuration dialog 3

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register FAST Feed Handler with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

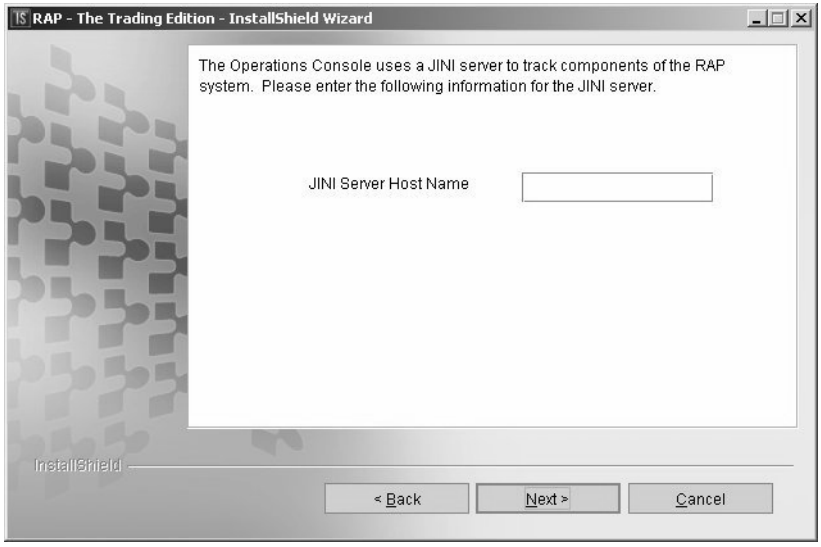


Table B-5: FAST Feed Handler dialog 3 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

Publisher API

The Publisher API formats incoming market data messages, and multicasts those messages to one or more subscribers. This API is used by Sybase feed handlers (Demo Feed Handler and FAST Feed Handler), and developers who want to map proprietary market data into Sybase RAP.

Installation options
An initial installation dialog gives you the option of configuring the Publisher API during the install or later. If you configure later, the installer installs the files, but leaves the configuration settings at their default values. You must manually configure these settings. See Publisher API in the *Sybase RAP - The Trading Edition Developers Guide* for more information.

Architecture

Publishers accept incoming market data messages and send those messages into Sybase RAP. Sybase RAP Data Stream (RDS) templates describing the message formats are used by the Publisher API to format the messages into packets. The Publisher API then sends the packets across data streams to subscribers.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- Publisher API configuration dialog 1 identifies the data channels that the Publisher API uses to send messages to subscribers.
- Publisher API configuration dialog 2 lets you set the number of message buffers.
- Publisher API Configuration dialog 3 prompts you for the JINI Server Host Name. Depending on how you install components, this dialog is optional.

Publisher API configuration dialog 1

The Publisher API multicasts messages to subscribers over one or more data channels. Four channels are recommended for a system that handles 300,000 messages per second. Use this dialog to identify the data channels.

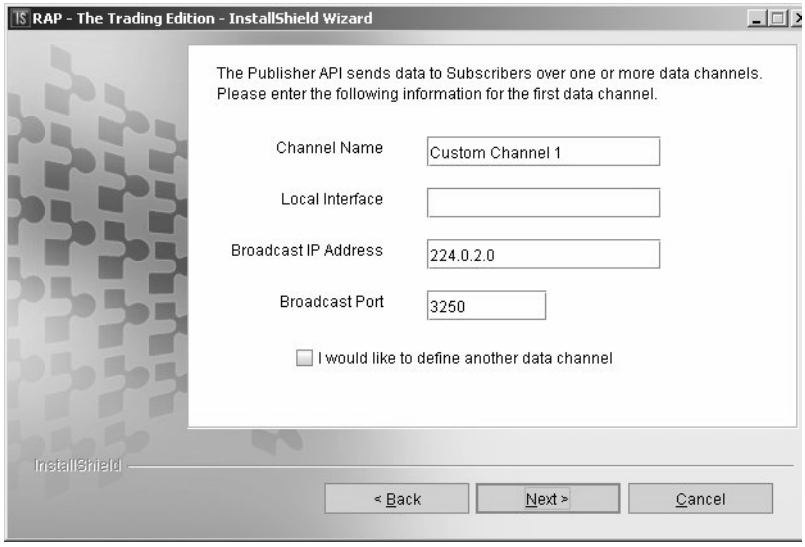


Table B-6: Publisher API dialog 1 configuration options

Field name	Default value	Description
Channel Name	Custom Channel 1	Identifies the data channel that Publisher API uses to send messages to a subscriber. Although the installer assigns Custom Channel 1 to the first data channel by default, you can use any value as the channel name. You must supply a Channel Name to any subsequent data channels you define. Assigning unique names to channels can help you track each one more easily in the event that an error is logged.
Local Interface	None	Identifies the IP address of the network interface to which this channel will be bound. This allows a user to bind various data channels to separate network interfaces to enhance performance. • If you plan to install all publishers and subscribers on the same server, you must configure the Local Interface to use the loopback interface (127.0.0.1). • If you install publishers and subscribers on different servers, use the local machine address of the secondary Ethernet network card installed by the system administrator.

Field name	Default value	Description
Broadcast IP Address	224.0.2.0	Identifies the Class D multicast IP address that the Publisher API uses to multicast messages to subscribers. This value should match the multicast address that the subscribers use to receive messages. Although the installer assigns 224.0.2.0 as a default multicast IP address to all Broadcast IP address fields, your Network Administrator can choose an alternate multicast address. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	3250	Identifies the local port on which this data channel will operate. Publishers can multicast messages to a subscriber on multiple data channels that share the same local interface and Broadcast IP Address, but the Broadcast Port must be unique to that data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to display a blank data channel dialog.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

Publisher API configuration dialog 2

For peak performance, you can tune the number of message buffers. This dialog lets you set the number of message buffers.

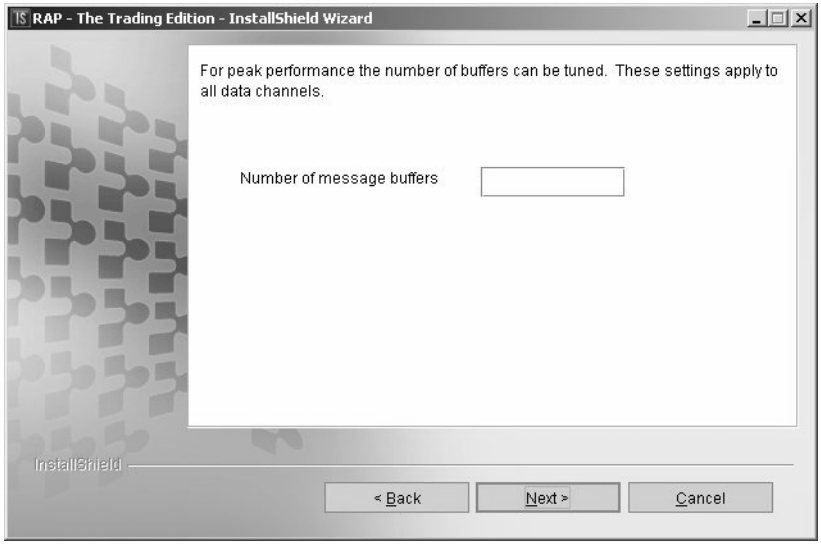


Table B-7: Publisher API dialog 2 configuration options

Field name	Description
Number of message buffers	Identifies the number of packets the subscriber temporarily stores to track data deliveries. Subscribers track the number of packets delivered by a publisher. If a subscriber suspects that a packet may be missing, it issues a resend request to the publisher to retrieve the missing data.

Publisher API Configuration dialog 3

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register the Publisher API with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

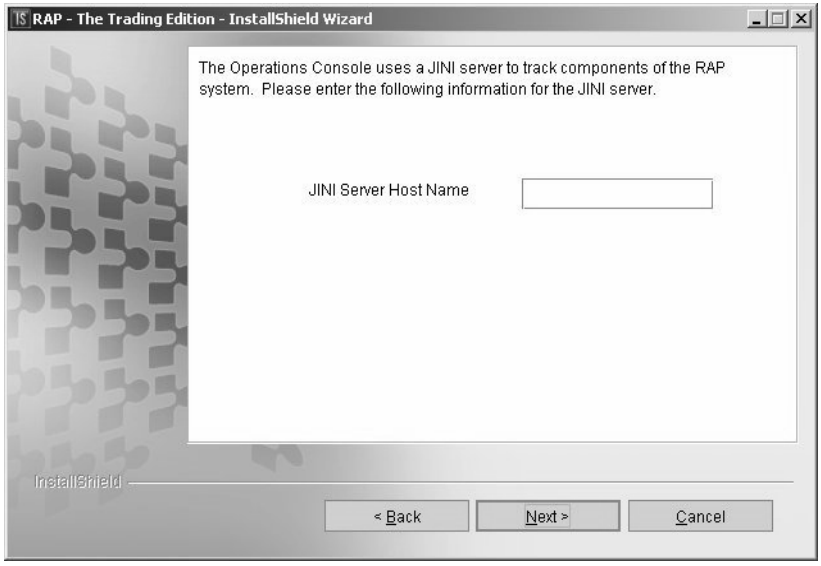


Table B-8: Publisher API dialog 3 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

RAPCache System

The RAPCache System includes the RAPCache Database Server and RAPCache Subscriber.

RAPCache Database Server

If you choose to install the RAPCache Database Server, the installer displays the following dialog. Click Next to install the RAPCache Database Server.

There are no configuration options, but the installer silently installs ASE 15.0.2, which creates the cache database objects, tables and default user accounts for Operations Console. All configuration parameters are set to ASE defaults (see below).

Table B-9: RAPCache Database Server Configuration Settings

Field name	Value	Description
RAPCache Server Name	RAPCache	RAPCache Database Server name.
RAPCache Port	5000	Port that the RAPCache Server uses to service connection requests. Although the default port is 5000, you can specify any free port.
Server Page Size	8K	Page size that Adaptive Server assigns to all database and database objects on the server.
Master Device Size	500MB	Master device size in megabytes (MB).
Master Database Size	100MB	Master database size in megabytes (MB). The master database contains Operations Console database tables and objects.
RAPCache Backup Port	5001	Port number that the backup server uses to handle network connection requests. The default value is 5001

Configuring the JINI Server

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register the RAPCache Database Server with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

RAPCache Subscriber

RAPCache Subscriber processes messages from publishers that arrive over multiple data channels, then loads the RAPCache database.

Architecture

RAPCache Subscriber queues up messages that are consumed by a data stream handler, which routes the messages to their final destination.

Subscribers receive message packets from publishers on multiple data streams. When a packet arrives, the subscriber records that the packet arrived. This record is used to identify any missing packets. If a packet is missing, the subscriber requests a resend from the publisher for that specific packet.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- RAPCache Subscriber configuration dialog 1 prompts you for the data channels that RAPCache Subscriber monitors for message deliveries from publishers.
- RAPCache Subscriber configuration dialog 2 prompts you for the publishers that communicate with RAPCache Subscriber.
- RAPCache Subscriber configuration dialog 3 prompts you for the JINI Server Host Name. Depending on how you install components, this dialog is optional.

RAPCache Subscriber configuration dialog 1

RAPCache Subscriber listens to one or more data channels for data sent by publishers. Use this dialog to identify each channel that you want RAPCache Subscriber to monitor.

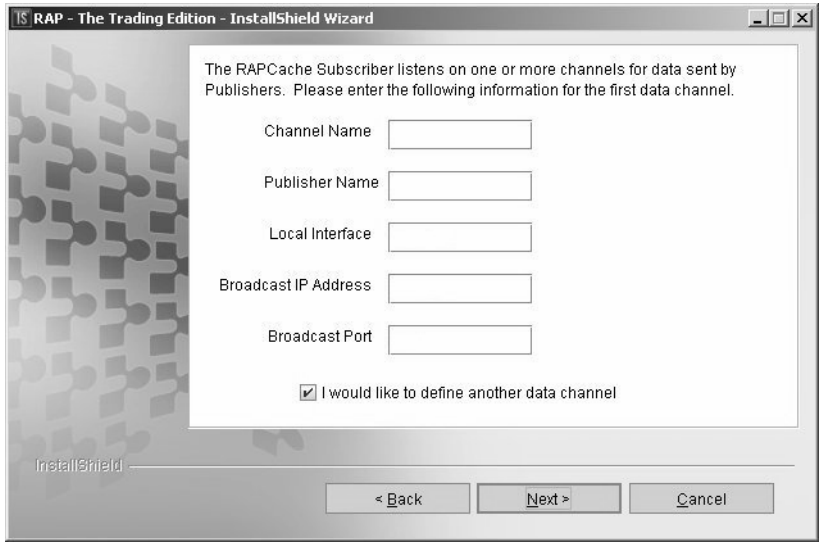


Table B-10: RAPCache Subscriber dialog 1 configuration options

Field name	Description
Channel Name	Identifies a specific data channel that a particular publisher uses to multicast messages to RAPCache Subscriber. The Channel Name depends on the data channel you want RAPCache Subscriber to monitor. If, for example, you want to RAPCache Subscriber to monitor FAST Feed Handler, and kept all the defaults when you defined the FAST Feed data channels, you would use FAST Channel 1 as the first Channel Name. Subsequent channel names would match those of other data channel names.
Publisher Name	Identifies the publisher that sends messages to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.
Local Interface	Identifies the IP address of the network interface that the publisher uses to multicast messages to RAPCache Subscriber. This value must match the local interface value that the publisher uses to multicast messages. If you want RAPCache Subscriber to monitor a particular FAST Feed Handler data channel, for example, the local interface value you use here must match the local interface value for the data channel you defined for FAST Feed Handler.
Broadcast IP Address	Identifies the Class D multicast IP address that the publisher uses to multicast messages to RAPCache Subscriber. This value must match the multicast address that the publisher uses to multicast messages to RAPCache Subscriber. If you want RAPCache Subscriber to monitor a FAST Feed Handler data channel that multicasts messages on 224.0.2.0, for example, you would use that address for this field. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	Identifies the port from which the publisher multicasts messages to RAPCache Subscriber. Publishers can multicast messages to a RAPCache Subscriber on multiple data channels that share the same local interface and Broadcast IP Address, but the Broadcast Port must be unique to each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to define another data channel.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

RAPCache Subscriber configuration dialog 2

RAPCache Subscriber communicates with one or more publishers. Use this dialog identify the publisher(s) you want RAPCache Subscriber to monitor.

Table B-11: RAPCache Subscriber dialog 2 configuration options

Field name	Description
Publisher Name	Identifies the publisher that publishes to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.
IP Address	Identifies the IP address of the machine on which the publisher resides. This is used to connect to the publisher to request resends of packets.

Field name	Description
Resend Listener Port	Identifies the port that the publisher uses to resend cached message packets to the subscriber. Publishers maintain a cache of recently sent message packets to ensure that the subscriber processes all message deliveries.

Note After you define the first publisher, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another publisher** box to define another publisher.
- Remove the checkmark in the **I would like to define another publisher** box, and click **Next** to display the next set of configuration options.

RAPCache Subscriber configuration dialog 3

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register the RAPCache Subscriber with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

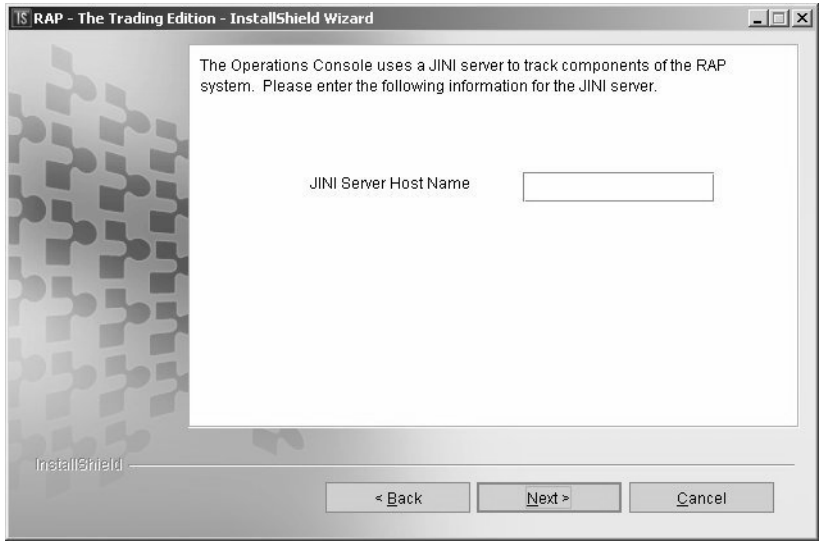


Table B-12: RAPCache Subscriber dialog 3 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

RAPStore System

The RAPStore System includes the RAPStore Database Server and RAPStore Subscriber.

RAPStore Database Server

If you choose to install the RAPCache Database Server, the installer displays the following dialog. Click Next to install the RAPCache Database Server.

There are no configuration options, but the installer silently installs Sybase IQ 12.7 ESD #3, creates the RAPStore database objects. During installation, the installer sets all configuration parameters to IQ defaults.

Configuring the JINI Server

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register the RAPCache Database Server with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

RAPStore Subscriber

RAPStore Subscriber processes message packets that arrive on multiple data streams and loads the RAPStore database.

Architecture

RAPStore Subscriber queues messages that are consumed by a data stream handler, which routes the messages to their final destination.

Subscribers receive message packets from publishers on multiple data streams. If a packet is missing, the subscriber requests a resend from the publisher for that specific packet.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- RAPStore Subscriber configuration dialog 1 lets you decide where to store temporary market data files.
- RAPStore Subscriber configuration dialog 2 identifies the data channels that RAPStore Subscriber monitors for message deliveries from publishers.
- RAPStore Subscriber configuration dialog 3 lets you identify the publishers you want the RAPStore Subscriber to monitor.
- RAPStore Subscriber configuration dialog 4 is an optional dialog that prompts you for the JINI server host name.

RAPStore Subscriber configuration dialog 1

The RAPStore Subscriber temporarily writes market data messages to files. The location of these directories should match the values on your worksheet. You must manually create these directories after you complete the install. See “Create the primary and overflow file directories” on page 45 for more information.

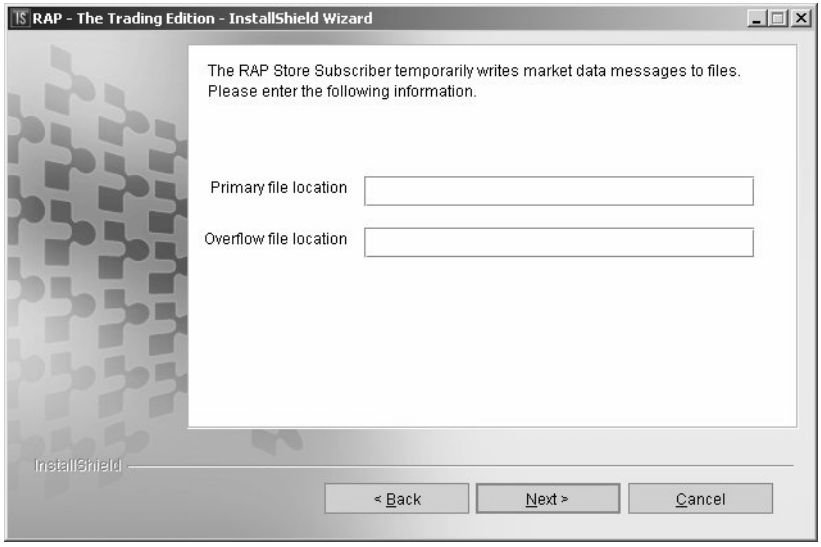


Table B-13: RAPStore Subscriber dialog 1 configuration options

Field name	Description
Primary file location	Identifies the directory where the subscriber temporarily writes market data messages before it loads the database.

Field name	Description
Overflow file location	Identifies the directory where the subscriber stores temporary market data message files when the primary directory is full.

RAPStore Subscriber configuration dialog 2

RAPStore Subscriber listens on one or more data channels for messages sent by publishers. Use this dialog to identify the data channels you want RAPStore Subscriber to monitor.

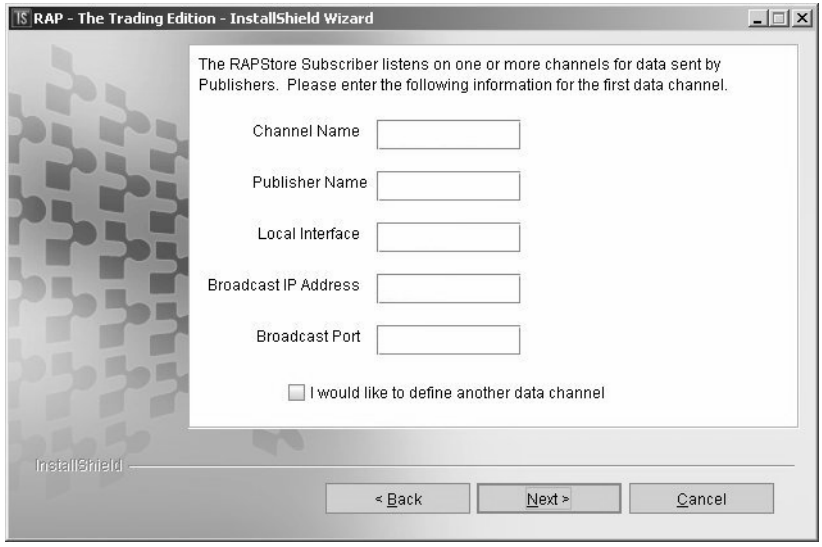


Table B-14: RAPStore Subscriber dialog 2 configuration options

Field name	Description
Channel Name	Identifies a specific data channel that a particular publisher uses to multicast messages to RAPStore Subscriber. The Channel Name depends on the data channel you want RAPStore Subscriber to monitor. If, for example, you want to RAPStore Subscriber to monitor FAST Feed Handler, and kept all the defaults when you defined the FAST Feed data channels, you would use FAST Channel 1 as the first Channel Name. Subsequent channel names would match those of other data channel names.
Publisher Name	Identifies the publisher that sends messages to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.

Field name	Description
Local Interface	Identifies the IP address of the network interface that the publisher uses to multicast messages to RAPStore Subscriber. This value must match the local interface value that the publisher uses to multicast messages. If you want RAPStore Subscriber to monitor a particular FAST Feed Handler data channel, for example, the local interface value you use here must match the local interface value for the data channel you defined for FAST Feed Handler.
Broadcast IP Address	Identifies the Class D multicast IP address that the publisher uses to multicast messages to RAPStore Subscriber. This value must match the multicast address that the publisher uses to multicast messages to RAPStore Subscriber. If you want RAPCache Subscriber to monitor a FAST Feed Handler data channel that multicasts messages on 224.0.2.0, for example, you would use that address for this field. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	Identifies the port from which the publisher multicasts messages to RAPStore Subscriber. Publishers can multicast messages to a RAPStore Subscriber on multiple data channels that share the same local interface and Broadcast IP Address, but the Broadcast Port must be unique to each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to define another data channel.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

RAPStore Subscriber configuration dialog 3

The RAPStore Subscriber communicates with one or more publishers. Use this dialog to identify the publishers you want the RAPStore Subscriber to monitor.

Table B-15: RAPStore Subscriber dialog 3 configuration options

Field Name	Description
Publisher Name	Identifies the publisher that publishes to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.
IP Address	Identifies the IP address of the machine on which the publisher resides. This is used to connect to the publisher to request resends of packets.
Resend Listener Port	Identifies the port that the publisher uses to resend cached message packets to the subscriber. Publishers maintain a cache of recently sent message packets to ensure that the subscriber processes all message deliveries.

RAPStore Subscriber configuration dialog 4

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register RAPStore Subscriber with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

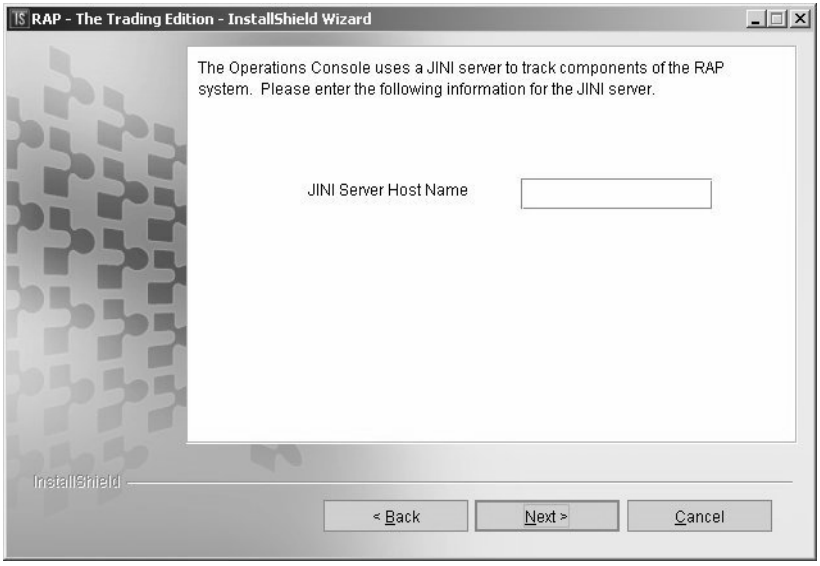


Table B-16: RAPStore Subscriber dialog 4 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

Demo Subscriber

Demo Subscriber processes message packets that arrive over data streams from a publisher. The purpose of the Demo Subscriber is to demonstrate, in conjunction with the Demo Feed Handler, message flow through Sybase RAP.

Architecture

The Demo Subscriber receives messages from publishers on multiple data streams. Each data stream listens for packets on one UDP broadcast channel. When a packet arrives, the Demo Subscriber will either write the messages to a file or discard the message, depending on how the Demo Subscriber is configured.

Configuration

If you choose this option, the installer displays several configuration dialogs.

- Demo Subscriber configuration dialog 1 identifies the data channels that Demo Subscriber monitors for message deliveries from publishers.
- Demo Subscriber configuration dialog 2 prompts you for the publishers that send messages to Demo Subscriber.
- Demo Subscriber configuration dialog 3 prompts you for the JINI Server Host Name.

Demo Subscriber configuration dialog 1

Demo Subscriber listens to one or more data channels for messages sent by publishers. Use this dialog to identify each channel that you want Demo Subscriber to monitor.

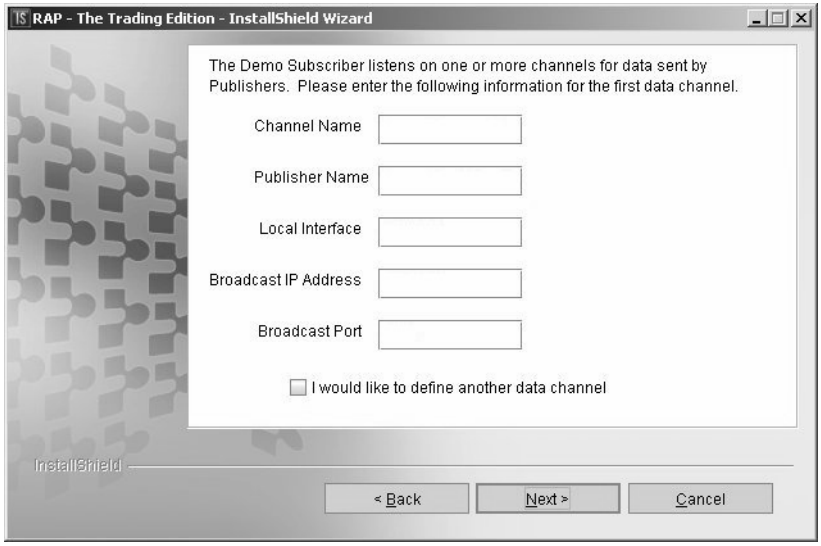


Table B-17: Demo Subscriber dialog 1 configuration options

Field name	Description
Channel Name	Identifies a specific data channel that a particular publisher uses to multicast messages to Demo Subscriber. The Channel Name depends on the data channel you want Demo Subscriber to monitor. If, for example, you want to Demo Subscriber to monitor FAST Feed Handler, and kept all the defaults when you defined the FAST Feed data channels, you would use FAST Channel 1 as the first Channel Name. Subsequent channel names would match those of other data channel names.
Publisher Name	Identifies the publisher that sends messages to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.

Field name	Description
Local Interface	Identifies the IP address of the network interface that the publisher uses to multicast messages to Demo Subscriber. This value must match the local interface value that the publisher uses to multicast messages. If you want Demo Subscriber to monitor a particular FAST Feed Handler data channel, for example, the local interface value you use here must match the local interface value for the data channel you defined for FAST Feed Handler.
Broadcast IP Address	Identifies the Class D multicast IP address that the publisher uses to multicast messages to Demo Subscriber. This value must match the multicast address that the publisher uses to multicast messages to Demo Subscriber. If you want Demo Subscriber to monitor a FAST Feed Handler data channel that multicasts messages on 224.0.2.0, for example, you would use that address for this field. All multicast IP addresses must range from *224.0.2.0* through *224.0.255.0*.
Broadcast Port	Identifies the port from which the publisher multicasts messages to Demo Subscriber. Publishers can multicast messages to Demo Subscriber on multiple data channels that share the same local interface and Broadcast IP Address, but the Broadcast Port must be unique to each data channel.

Note
After you define the first channel, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another data channel** box to display a blank data channel dialog.
- Remove the checkmark in the **I would like to define another data channel** box, and click **Next** to display the next set of configuration options.

Demo Subscriber configuration dialog 2

Demo Subscriber communicates with one or more publishers. Use this dialog to identify the publisher(s) you want Demo Subscriber to monitor.

Table B-18: Demo Subscriber dialog 2 configuration options

Field name	Description
Publisher Name	Identifies the publisher that publishes to this subscriber. The Publisher Name can be any value. It is used to identify the publisher in the event that an error is logged.
IP Address	Identifies the IP address of the machine on which the publisher resides. This is used to connect to the publisher to request resends of packets.
Resend Listener Port	Identifies the port that the publisher uses to resend cached message packets to the subscriber. Publishers maintain a cache of recently sent message packets to ensure that the subscriber processes all message deliveries.

Note After you define the first publisher, do one of the following:

- Click **Next** with a checkmark in the **I would like to define another publisher** box to display a blank publisher dialog.
- Remove the checkmark in the **I would like to define another publisher** box, and click **Next** to display the next set of configuration options.

Demo Subscriber configuration dialog 3

Installing Operations Console also installs a JINI server that tracks RAP components. This dialog only appears if you install components separately. If you install multiple components at the same time, and previously identified the JINI server host name, the installer does not display this dialog.

If this dialog appears, you must identify the JINI server host name to register Demo Subscriber with the JINI Server. The JINI server host name is the same host machine where you install Operations Console.

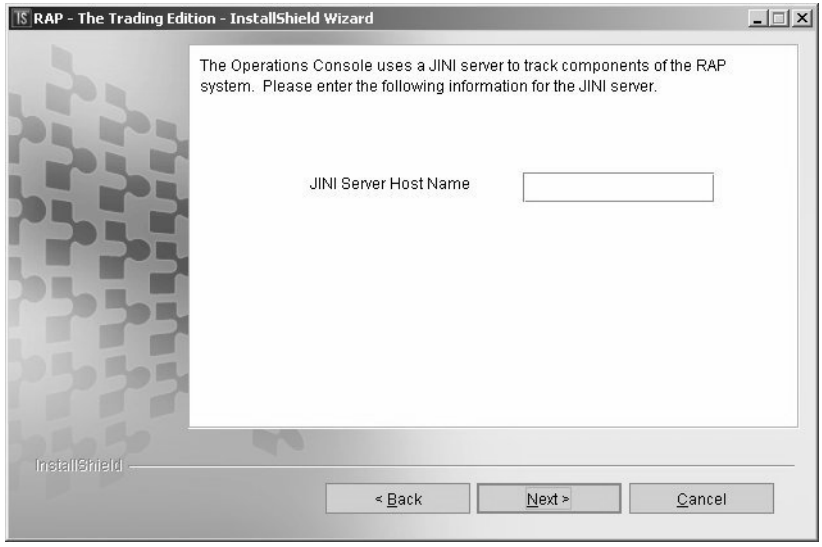


Table B-19: Demo Subscriber dialog 3 configuration options

Field name	Description
JINI Server Host Name	Identifies the host name of the server where the Sybase RAP JINI server resides. This is the machine on which Operations Console is installed.

Operations Console

Operations Console is a web-based interface that provides Sybase RAP administration and monitoring functions.

Architecture

Operations Console interacts with different system components through the Unified Agent Framework (UAF). UAF plug-ins running on each server handle communication between Operations Console and Sybase RAP components. When an agent starts on a server, the agent starts all of its plug-ins and registers with the JINI server. Operations Console queries the JINI server to learn about all agents (and their plug-ins).

Interaction between Operations Console and the plug-ins occurs via RMI (remote method invocation). Publisher and subscriber plug-ins communicate with publishers and subscribers components via TCP over a local socket. Each publisher and subscriber has an administration channel listening on a specific port. When Operations Console issues a command, the agent intercepts the request, and contacts the component. The component sends a response through the agent back to Operations Console.

Configuration

Installation in an Express setup is non-interactive. Configuration settings appear in the tables below.

Table B-20: Operation Console configuration settings

Field name	Value	Description
RMI Port	9092	RMI port that the JINI server uses internally in its operation.
JINI Server Class Server Port	9095	Port that the JINI server uses internally in its operation.
RAPCache Database Server Host Name	Matches host	Machine that hosts the RAPCache Database server.
Port	5000	Port that the RAPCache Database Server listens for connections.
RAP User Name	RAP_USER	Default Sybase RAP user for RAPCache Database Server.
RAPStore Host Name	Matches host	Machine that hosts the RAPStore Database server.
RAPStore Database Port	2638	Port that the RAPCache Database Server uses to listen for connections.
User Name	RAP_USER	Default Sybase RAP user for the RAPStore Database Server.
HTTP Port	8080	Port that Tomcat uses to listen for incoming connections from web browsers.

Field name	Value	Description
Shutdown Port	8085	Identifies the port that Tomcat uses to close all Web applications, and shutdown all services.
AJP Port	8009	AJP (TCP) port that Tomcat uses to create a server socket and wait for incoming connections.

APPENDIX C

Installation and Configuration File Locations

This appendix identifies the location of your installed components. As a convention, this document uses the variable *$RAP* to identify the root installation directory.

RAPCache Database Server

The default RAPCache Database Server installation directory is *$RAP/RAPCache*. See the table below for subdirectory and file descriptions.

Directory	Description
/ASE15.0	RAPCache database (ASE installation)
/datamodel/MarketData	SQL and PDM files to create the market data model
/datamodel/OpsConsole	SQL and PDM files to create the Operations Console data model
/data/marketdata	Sample data to load into market data message tables. These files are in a format that can be loaded into the database.
/data/supportingdata	Sample data to load into supporting (non-message) tables

Directory	Description
/data/scripts	RAPCache sample queries
/datamodel/scripts	Scripts used to load the sample data into the database
/UAF_20	UAF installation

RAPCache Subscriber

The default RAPCache Subscriber installation directory is *$RAP/RAPCacheSubscriber*. See the table below for subdirectory and file descriptions.

Component	Description
/bin/rapcachesubscriber	RAPCache Subscriber executable
/bin/rapcachesubscriber.sh	Script to launch RAPCache Subscriber
/templates	Directory containing RDS templates
/template.xsd	XML Schema file for the RDS templates
/config/opensubscriber.xml	Open Subscriber configuration file
/config/rapcache.xml	RAPCache Data Stream Handler configuration file
/ua	UAF installation
/bin/rapencrypt	A tool to encrypt passwords to be used while configuring Sybase RAP system.

RAPStore Database Server

The default RAPStore Database Server installation directory is *$RAP/RAPStore*. See the table below for subdirectory names and file descriptions.

Component	Description
/ASIQ-12_7	RAPStore database server (IQ installation)
/datamodel/MarketData	SQL and PDM files to create the market data model
/datamodel/Work	SQL and PDM files to create the work tables used by the RAPStore Subscriber
/data/marketdata	Sample data to load into market data message tables. These files are in a format that can be loaded into the database.
/data/supportingdata	Sample data to load into supporting (nonmessage) tables

Component	Description
/data/scripts	RAPStore sample queries
/datamodel/scripts	Scripts used to load the sample data into the database

RAPStore Subscriber

The default RAPStore Subscriber installation directory is *$RAP/RAPStoreSubscriber*. See the table below for subdirectory names and file descriptions.

Component	Description
/bin/rapstoresubscriber	RAPStore Subscriber executable
/bin/rapstoresubscriber.sh	Script to launch RAPStore Subscriber
/templates	Directory containing RDS templates
/template.xsd	XML Schema file for the RDS templates
/config/opensubscriber.xml	Open Subscriber configuration file
/config/rapstore.xml	RAPStore Data Stream Handler configuration file
/ua	UAF installation
/bin/rapencrypt	A tool to encrypt passwords to be used while configuring Sybase RAP system.

Demo Feed Handler

The default installation directory of the Demo Feed Handler is *$RAP/DemoFeedHandler*. See the table below for subdirectory names and file descriptions.

Component	Description
/bin/demofeedhandler	Demo Feed Handler executable
/bin/demofeedhandler.sh	Script to launch Demo Feed Handler
/templates	Directory containing RDS templates
/marketdata/messageformat	Directory containing *STOCK_QUOTE.csv*. This is the Stock Quote demo data file, which contains 57780 messages.
/template.xsd	XML Schema file for the RDS templates
/config/publisher.xml	Publisher configuration file

Component	Description
/src	Source files for the Demo Feed Handler (*demofeedhandler.cpp*)
fasttemplates	Contains specific templates for the FASTFeedHandler.
/ua	UAF installation

FAST Feed Handler

The default FAST Feed Handler installation directory is *$RAP/FASTFeedHandler*. See the table below for subdirectory names and file descriptions.

Component	Description
/bin/fastfeedhandler	FAST Feed Handler executable
/bin/fastfeedhandler.sh	Script to execute FAST Feed Handler
/templates	Directory containing RDS templates
/template.xsd	XML Schema file for the RDS templates
/config/publisher.xml	Publisher configuration file
/ua	The UAF installation
/lib/ffhfixmsghandler.so	Shared library for the FIX message handler
/config/fastfeedhandler.xml	FAST Feed Handler configuration file
/include	Contains all public interface C header files

Publisher API

The default Sybase RAP Publisher API installation directory is *$RAP/PublisherAPI*. See the table below for subdirectory names and file descriptions.

Component	Description
/lib/publisher.a	Publisher library
/include/logger.h	Loggining header file
/include/publisher.h	Public header file
/PublisherAPI/templates	Directory containing RDS templates
/template.xsd	XML Schema file for the RDS templates

Component	Description
/config/publisher.xml	Publisher configuration file
/ua	UAF installation

Demo Subscriber

The default Sybase RAP Demo Subscriber installation directory is *$RAP/DemoSubscriber*. See the table below for subdirectory names and file descriptions.

Component	Description
/bin/demosubscriber	Demo Subscriber executable
/bin/demosubscriber.sh	Script to launch Demo Subscriber
/config/	Contains these files: • *opensubscriber.xml* • *demodsh.xml* • *messagefilter.xml* • *messagefilter.xsd*
/config/opensubscriber.xml	Open Subscriber configuration file
/ua	UAF installation
/marketdata/messageformat/STOCK_QUOTE.csv	Sample file used to run the DemoFeedHandler out of the box.

Operations Console

The default Operations Console installation directory is *$RAP/OpsConsole*. See the table below for subdirectory names and file descriptions.

Component	Description
/WebMonitoringConsole	Operations Console web application and Tomcat
/StartOpsConsole.sh	Script to launch Operations Console
/WebMonitoringConsole/bin	Location of the Operations Console configuration file (*opsconsole.properties)*
/ua	UAF installation, including JINI Server

Component	Description
/bin/rapencrypt	A tool to encrypt passwords to be used while configuring Sybase RAP system.

Samples

The default Samples installation directory is *$RAP/Samples*. See the table below for subdirectory names and file descriptions.

Component	Description
/queries/rapcache	Sample SQL scripts for querying the RAPCache
/queries/rapstore	Sample SQL scripts for querying the RAPStore
/data/marketdata/database format	Sample data to load into market data message tables. These files are in a format that can be loaded through isql.

Index

A

C

D

E

F

H

I

J

L

M

N

O

P

R

S

U